

QGIS Desktop 3.44 User Guide

QGIS Project

сеп 06, 2026

Садржај

1 Početak **3**
- 1.1 Instaliranje QGIS-a . . . 3
  - 1.1.1 Instaliranje iz binarnih datoteka . . . 3
  - 1.1.2 Instaliranje iz izvornog koda . . . 3
  - 1.1.3 Instaliranje na eksterne medije . . . 3
  - 1.1.4 Preuzimanje primernih podataka . . . 3
- 1.2 Pokretanje i zaustavljanje QGIS-a . . . 4
- 1.3 Primer sesije: Učitavanje rasterskih i vektorskih slojeva . . . 4

2 Working with Project Files **13**
- 2.1 Introducing QGIS projects . . . 13
- 2.2 Handling broken file paths . . . 15
- 2.3 Generating output . . . 16

3 QGIS GUI **17**
- 3.1 Traka menija . . . 18
  - 3.1.1 Projekat . . . 18
  - 3.1.2 Uređivanje . . . 19
  - 3.1.3 Prikaz . . . 22
  - 3.1.4 Sloj . . . 25
  - 3.1.5 Podešavanja . . . 28
  - 3.1.6 Dodaci . . . 28
  - 3.1.7 Vektor . . . 28
  - 3.1.8 Raster . . . 30
  - 3.1.9 Baza podataka . . . 32
  - 3.1.10 Veb . . . 32
  - 3.1.11 Mesh . . . 32
  - 3.1.12 Obrada . . . 33
  - 3.1.13 Pomoć . . . 33
  - 3.1.14 QGIS . . . 34
- 3.2 Paneli i trake sa alatkama . . . 34
  - 3.2.1 Trake sa alatkama . . . 34
  - 3.2.2 Paneli . . . 35
- 3.3 Statusna traka . . . 35
  - 3.3.1 Traka lokatora . . . 36
  - 3.3.2 Prikaz radnji . . . 36
  - 3.3.3 Upravljanje platnom karte . . . 36
  - 3.3.4 Poruke . . . 37

4 The Browser panel **39**
- 4.1 Resources that can be opened / run from the Browser . . . 41

4.2 Browser panel top-level entries . . . 41
4.2.1 Favorites . . . 41
4.2.2 Spatial Bookmarks . . . 41
4.2.3 Project Home . . . 42
4.2.4 Drives and file system . . . 42
4.2.5 Data type . . . 43
4.2.6 Database entries . . . 43
4.2.7 Tiles and Web Services . . . 47
4.2.8 Project and supported file types . . . 48

5 QGIS Configuration **49**
5.1 Options . . . 49
5.1.1 General Settings . . . 50
5.1.2 System Settings . . . 52
5.1.3 User Profiles Settings . . . 54
5.1.4 CRS and Transforms Settings . . . 54
5.1.5 Data Sources settings . . . 57
5.1.6 Rendering Settings . . . 61
5.1.7 Canvas and Legend Settings . . . 64
5.1.8 Map tools Settings . . . 66
5.1.9 Elevation settings . . . 70
5.1.10 3D settings . . . 70
5.1.11 Colors settings . . . 71
5.1.12 Fonts Settings . . . 72
5.1.13 Layouts settings . . . 73
5.1.14 Variables settings . . . 73
5.1.15 Authentication settings . . . 74
5.1.16 Network settings . . . 75
5.1.17 GPS settings . . . 77
5.1.18 Locator settings . . . 79
5.1.19 Acceleration settings . . . 81
5.1.20 IDE settings . . . 81
5.1.21 Processing settings . . . 85
5.1.22 Advanced settings . . . 85
5.2 Working with User Profiles . . . 86
5.2.1 The concept . . . 86
5.2.2 Setting user profile . . . 87
5.3 Project Properties . . . 88
5.3.1 General Properties . . . 88
5.3.2 Metadata Properties . . . 91
5.3.3 View Settings . . . 92
5.3.4 CRS Properties . . . 92
5.3.5 Transformations Properties . . . 92
5.3.6 Styles Properties . . . 93
5.3.7 Colors properties . . . 94
5.3.8 Data Sources Properties . . . 95
5.3.9 Relations Properties . . . 97
5.3.10 Variables Properties . . . 98
5.3.11 Macros Properties . . . 98
5.3.12 QGIS Server Properties . . . 98
5.3.13 Temporal Properties . . . 99
5.3.14 Elevation Properties . . . 100
5.3.15 Sensors Properties . . . 103
5.4 Customization . . . 104
5.5 Keyboard shortcuts . . . 106
5.6 Running QGIS with advanced settings . . . 107
5.6.1 Command line and environment variables . . . 107
5.6.2 Deploying QGIS within an organization . . . 112

6 Working with Projections **115**
6.1 Overview of Projection Support . . . 115
6.2 Layer Coordinate Reference Systems . . . 115
6.3 Project Coordinate Reference Systems . . . 117
6.4 Coordinate Reference System Selector . . . 119
6.5 Custom Coordinate Reference System . . . 120
6.5.1 Integrate an NTv2-transformation in QGIS . . . 121
6.6 Datum Transformations . . . 122

7 Vizuelizacija karata **125**
7.1 2D Map View . . . 125
7.1.1 Exploring the map view . . . 126
7.1.2 Controlling map rendering . . . 127
7.1.3 Time-based control on the map canvas . . . 127
7.1.4 The Elevation Controller panel . . . 129
7.1.5 Bookmarking extents on the map . . . 130
7.1.6 Decorating the map . . . 132
7.1.7 Annotation Tools . . . 139
7.1.8 Measuring . . . 144
7.1.9 Setting additional map views . . . 146
7.1.10 Exporting the map view . . . 148
7.2 3D Map View . . . 150
7.2.1 Scene Configuration . . . 152
7.2.2 Navigation options . . . 159
7.2.3 Creating an animation . . . 160
7.2.4 3D vector layers . . . 160
7.3 Elevation Profile View . . . 160
7.3.1 The interface . . . 161
7.3.2 Creating an elevation profile . . . 163
7.3.3 Interacting with the profile Views . . . 164
7.3.4 Exporting from the elevation profile . . . 165

8 General Tools **167**
8.1 Context help . . . 167
8.2 Paneli . . . 167
8.2.1 Layers Panel . . . 167
8.2.2 Layer Styling Panel . . . 174
8.2.3 Layer Order Panel . . . 177
8.2.4 Overview Panel . . . 177
8.2.5 Log Messages Panel . . . 177
8.2.6 Undo/Redo Panel . . . 178
8.2.7 Statistical Summary Panel . . . 178
8.2.8 Debugging/Development Tools Panel . . . 179
8.3 Embedding layers from external projects . . . 182
8.4 Interacting with features . . . 183
8.4.1 Izbor objekata . . . 183
8.4.2 Identifying Features . . . 186
8.5 Save and Share Layer Properties . . . 190
8.5.1 Managing Custom Styles . . . 190
8.5.2 Storing Styles in a File or a Database . . . 192
8.5.3 Layer definition file . . . 194
8.6 Documenting your data . . . 194
8.6.1 Metadata . . . 194
8.6.2 Layer notes . . . 195
8.7 Storing values in Variables . . . 196
8.8 Common widgets . . . 197
8.8.1 Color Selector . . . 197
8.8.2 Symbol Widget . . . 202

8.8.3 Remote or embedded file selector . . . 202
8.8.4 Visibility Scale Selector . . . 202
8.8.5 Spatial Extent Selector . . . 203
8.8.6 Font Selector . . . 203
8.8.7 Unit Selector . . . 204
8.8.8 Number Formatting . . . 205
8.8.9 Blending Modes . . . 207
8.8.10 Data defined override setup . . . 209
8.8.11 Authentication . . . 212

9 Napredujte uz izraze **215**
9.1 Expressions . . . 215
9.1.1 The Expression string builder . . . 215
9.1.2 Function Editor . . . 220
9.2 Lista funkcija . . . 222
9.2.1 Agregatne funkcije . . . 222
9.2.2 Funkcije za nizove . . . 230
9.2.3 Funkcije za boje . . . 238
9.2.4 Uslovne funkcije . . . 244
9.2.5 Funkcije za pretvaranje . . . 247
9.2.6 Funkcije za CRS . . . 251
9.2.7 Prilagođene funkcije . . . 251
9.2.8 Funkcije za datum i vreme . . . 251
9.2.9 Polja i vrednosti . . . 259
9.2.10 Funkcije za datoteke i putanje . . . 259
9.2.11 Funkcije za forme . . . 261
9.2.12 Funkcije za približno podudaranje . . . 262
9.2.13 Opšte funkcije . . . 263
9.2.14 Funkcije za geometriju . . . 264
9.2.15 Funkcije za prikaz . . . 325
9.2.16 Slojevi karte . . . 325
9.2.17 Funkcije za mape . . . 327
9.2.18 Matematičke funkcije . . . 330
9.2.19 Funkcije za mreže . . . 335
9.2.20 Operatori . . . 337
9.2.21 Funkcije za obradu . . . 343
9.2.22 Funkcije za rastere . . . 343
9.2.23 Funkcije za zapise i atribute . . . 344
9.2.24 Relacije . . . 351
9.2.25 Funkcije za senzore . . . 351
9.2.26 Funkcije za niske . . . 352
9.2.27 Korisnički izrazi . . . 360
9.2.28 Promenljive . . . 360
9.2.29 Nedavne funkcije . . . 364

10 Biblioteka stilova **365**
10.1 The Style Manager . . . 365
10.1.1 The Style Manager dialog . . . 365
10.1.2 Setting a Color Ramp . . . 371
10.1.3 Creating a Legend Patch Shape . . . 373
10.2 The Symbol Selector . . . 375
10.2.1 The symbol layer tree . . . 376
10.2.2 Configuring a symbol . . . 377
10.3 Setting a label . . . 396
10.3.1 Formatting the label text . . . 397
10.3.2 Configuring interaction with labels . . . 407
10.4 Creating 3D Symbols . . . 416
10.4.1 Point Layers . . . 417

10.4.2 Line layers . . . 418
10.4.3 Polygon Layers . . . 419
10.4.4 Shading the texture . . . 421
10.4.5 Application example . . . 421

11 Upravljanje izvorom podataka **423**
11.1 Opening Data . . . 423
11.1.1 The Browser Panel . . . 425
11.1.2 The DB Manager . . . 428
11.1.3 Provider-based loading tools . . . 429
11.1.4 The Layer Metadata Search Panel . . . 449
11.1.5 QGIS Custom formats . . . 450
11.1.6 QLR - QGIS Layer Definition File . . . 450
11.1.7 Connecting to web services . . . 451
11.2 Creating Layers . . . 459
11.2.1 Creating new vector layers . . . 459
11.2.2 Creating new layers from an existing layer . . . 467
11.2.3 Creating new DXF files . . . 471
11.2.4 Creating new layers from the clipboard . . . 473
11.2.5 Creating SQL Query Layers . . . 473
11.2.6 Creating virtual layers . . . 476
11.3 Georeferencer . . . 479
11.3.1 Georeferenciranje rasterskog sloja . . . 480
11.3.2 Georeferenciranje vektorskog sloja . . . 485
11.3.3 Prikaz i prilagođavanje svojstava sloja . . . 487
11.3.4 Konfigurisanje georeferencer-a . . . 487
11.4 Exploring Data Formats and Fields . . . 487
11.4.1 Raster data . . . 487
11.4.2 Vector Data . . . 487
11.4.3 Layers crossing 180° longitude . . . 495

12 Rad sa vektorskim podacima **497**
12.1 The Vector Properties Dialog . . . 497
12.1.1 Information Properties . . . 498
12.1.2 Source Properties . . . 498
12.1.3 Symbology Properties . . . 502
12.1.4 Labels Properties . . . 524
12.1.5 Diagrams Properties . . . 536
12.1.6 Masks Properties . . . 541
12.1.7 3D View Properties . . . 542
12.1.8 Fields Properties . . . 544
12.1.9 Attributes Form Properties . . . 545
12.1.10 Joins Properties . . . 557
12.1.11 Auxiliary Storage Properties . . . 557
12.1.12 Actions Properties . . . 565
12.1.13 Display Properties . . . 571
12.1.14 Rendering Properties . . . 574
12.1.15 Temporal Properties . . . 575
12.1.16 Variables Properties . . . 576
12.1.17 Elevation Properties . . . 577
12.1.18 Metadata Properties . . . 579
12.1.19 Dependencies Properties . . . 579
12.1.20 Legend Properties . . . 579
12.1.21 QGIS Server Properties . . . 580
12.1.22 Digitizing Properties . . . 581
12.2 Rad sa atributskom tabelom . . . 583
12.2.1 Uvod: prostorne i neprostorne tabele . . . 583
12.2.2 Predstavljanje interfejsa atributske tabele . . . 583

12.2.3 Interakcija sa objektima u atributskoj tabeli . . . 589
12.2.4 Uređivanje vrednosti atributa . . . 593
12.2.5 Istraživanje atributa objekata pomoću alatke Identifikuj . . . 597
12.3 Uređivanje . . . 599
12.3.1 Setting the snapping tolerance and search radius . . . 600
12.3.2 Snapping and Digitizing Options . . . 600
12.3.3 Topological editing . . . 603
12.3.4 Digitizing an existing layer . . . 604
12.3.5 Advanced digitizing . . . 615
12.3.6 Shape digitizing . . . 624
12.3.7 The Advanced Digitizing panel . . . 627
12.3.8 The Processing in-place layer modifier . . . 635
12.4 Connecting and Editing Data Across Layers . . . 637
12.4.1 Joining features between two layers . . . 637
12.4.2 Setting relations between multiple layers . . . 639
12.4.3 Storing and fetching an external resource . . . 653

13 Rad sa rasterskim podacima **657**
13.1 Raster Properties Dialog . . . 657
13.1.1 Information Properties . . . 658
13.1.2 Source Properties . . . 658
13.1.3 Symbology Properties . . . 659
13.1.4 Transparency Properties . . . 669
13.1.5 Labels Properties . . . 670
13.1.6 Histogram Properties . . . 672
13.1.7 Rendering Properties . . . 673
13.1.8 Temporal Properties . . . 674
13.1.9 Pyramids Properties . . . 675
13.1.10 Elevation Properties . . . 676
13.1.11 Metadata Properties . . . 677
13.1.12 Legend Properties . . . 678
13.1.13 Display Properties . . . 678
13.1.14 QGIS Server Properties . . . 679
13.1.15 Attribute Tables . . . 680
13.1.16 Identify raster cells . . . 684
13.2 Raster Analysis . . . 685
13.2.1 Raster Calculator . . . 685

14 Working with Mesh Data **689**
14.1 What's a mesh? . . . 689
14.2 Supported formats . . . 691
14.3 Mesh Dataset Properties . . . 691
14.3.1 Information Properties . . . 692
14.3.2 Source Properties . . . 693
14.3.3 Symbology Properties . . . 694
14.3.4 Labels Properties . . . 699
14.3.5 3D View Properties . . . 701
14.3.6 Rendering Properties . . . 703
14.3.7 Temporal Properties . . . 703
14.3.8 Elevation Properties . . . 704
14.3.9 Metadata Properties . . . 706
14.4 Exploring mesh datasets . . . 706
14.5 Editing a mesh layer . . . 707
14.5.1 Overview of the mesh digitizing tools . . . 708
14.5.2 Exploring the Z value assignment logic . . . 708
14.5.3 Selecting mesh elements . . . 710
14.5.4 Modifying mesh elements . . . 711
14.5.5 Reindexing meshes . . . 715

14.6 Mesh Calculator . . . 715

15 Working with Vector Tiles **719**
15.1 What are Vector Tiles? . . . 719
15.2 Supported Formats . . . 720
15.3 Vector Tiles Dataset Properties . . . 721
15.3.1 Information Properties . . . 721
15.3.2 Source Properties . . . 721
15.3.3 Symbology and Labels Properties . . . 722
15.3.4 Rendering Properties . . . 723
15.3.5 Metadata Properties . . . 723
15.4 QGIS Server Properties . . . 723

16 Working with Point Clouds **725**
16.1 Introduction to Point Clouds . . . 725
16.2 Point Clouds Properties . . . 725
16.2.1 Information Properties . . . 726
16.2.2 Source Properties . . . 727
16.2.3 Symbology Properties . . . 730
16.2.4 3D View Properties . . . 738
16.2.5 Rendering Properties . . . 740
16.2.6 Elevation Properties . . . 740
16.2.7 Metadata Properties . . . 741
16.2.8 Statistics Properties . . . 742
16.3 Virtual point cloud . . . 742

17 Working with 3D Tiles **745**
17.1 What are 3D Tiles? . . . 745
17.1.1 State of 3D Tiles Support . . . 746
17.2 3D Tiles Properties . . . 746
17.2.1 Information Properties . . . 747
17.2.2 Source Properties . . . 747
17.2.3 Symbology Properties . . . 747
17.2.4 3D View Properties . . . 748
17.2.5 Rendering Properties . . . 749
17.2.6 Elevation Properties . . . 749
17.2.7 Metadata Properties . . . 749

18 Postavljanje mapa **751**
18.1 Pregled rasporeda za štampu . . . 751
18.1.1 Primer sesije za početnike . . . 751
18.1.2 Menadžer rasporeda . . . 752
18.1.3 Meniji, alatke i paneli rasporeda za štampu . . . 753
18.2 Stavke rasporeda . . . 767
18.2.1 Zajedničke opcije stavki rasporeda . . . 767
18.2.2 The Map Item . . . 772
18.2.3 The 3D Map Item . . . 782
18.2.4 The Label Item . . . 783
18.2.5 The Legend Item . . . 787
18.2.6 The Scale Bar Item . . . 795
18.2.7 The Table Items . . . 798
18.2.8 The Marker, Picture and North Arrow Items . . . 807
18.2.9 The Elevation Profile Item . . . 811
18.2.10 The HTML Frame Item . . . 814
18.2.11 The Shape Items . . . 817
18.3 Creating an Output . . . 820
18.3.1 Podešavanja izvoza . . . 821
18.3.2 Export as Image . . . 822
18.3.3 Export as SVG . . . 823

18.3.4 Export as PDF . . . 824
18.3.5 Generate an Atlas . . . 826
18.4 Creating a Report . . . 832
18.4.1 What is it? . . . 832
18.4.2 Get started . . . 832
18.4.3 Layout Report Workspace . . . 833
18.4.4 Podešavanja izvoza . . . 848

19 Working with OGC / ISO protocols **849**
19.1 WMS/WMTS Client . . . 849
19.1.1 Overview of WMS Support . . . 849
19.1.2 Overview of WMTS Support . . . 850
19.1.3 Selecting WMS/WMTS Servers . . . 851
19.1.4 Loading WMS/WMTS Layers . . . 853
19.1.5 Tilesets . . . 855
19.1.6 Using the Identify Tool . . . 855
19.1.7 Viewing Properties . . . 856
19.1.8 Show WMS legend graphic in table of contents and layout . . . 859
19.2 WCS Client . . . 860
19.3 WFS and WFS-T Client . . . 862
19.4 SensorThings . . . 865
19.4.1 Setting connection . . . 865
19.4.2 Loading SensorThings data . . . 866
19.4.3 Working with a vector layer from SensorThings . . . 868
19.5 STAC (SpatioTemporal Asset Catalogs) . . . 868
19.5.1 Setting connection . . . 869
19.5.2 Browsing STAC Catalogs . . . 870
19.5.3 Filtering and Searching STAC Items . . . 870
19.6 ArcGIS REST Server . . . 872

20 Rad sa GPS podacima **873**
20.1 Introducing GNSS/GPS Data . . . 873
20.1.1 What is GPS? . . . 873
20.1.2 Transferring or loading GPS data . . . 873
20.2 Live GPS tracking . . . 876
20.2.1 GPS toolbar . . . 876
20.2.2 GPS Information Panel . . . 877
20.2.3 Connect to a Bluetooth GPS for live tracking . . . 879
20.2.4 Examples of GPS devices connection in QGIS . . . 880

21 Sistem za autentifikaciju **883**
21.1 Authentication System Overview . . . 883
21.1.1 Authentication database . . . 883
21.1.2 Custom authentication databases . . . 884
21.1.3 Master password . . . 885
21.1.4 Authentication Configurations . . . 886
21.1.5 Authentication Methods . . . 888
21.1.6 Master Password and Auth Config Utilities . . . 893
21.1.7 Using authentication configurations . . . 895
21.1.8 Python bindings . . . 895
21.2 User Authentication Workflows . . . 896
21.2.1 HTTP(S) authentication . . . 896
21.2.2 Database authentication . . . 897
21.2.3 PKI authentication . . . 898
21.2.4 Handling bad layers . . . 905
21.2.5 Changing authentication config ID . . . 906
21.2.6 QGIS Server support . . . 907
21.2.7 SSL server exceptions . . . 907

22 GRASS GIS Integration **911**
22.1 Demo dataset . . . 911
22.2 Loading GRASS raster and vector layers . . . 911
22.3 Importing data into a GRASS LOCATION via drag and drop . . . 912
22.4 Managing GRASS data in QGIS Browser . . . 912
22.5 GRASS Options . . . 912
22.6 Starting the GRASS plugin . . . 912
22.7 Opening GRASS mapset . . . 912
22.8 GRASS LOCATION and MAPSET . . . 913
22.9 Importing data into a GRASS LOCATION . . . 913
22.9.1 Creating a new GRASS LOCATION . . . 914
22.9.2 Adding a new MAPSET . . . 915
22.10 The GRASS vector data model . . . 915
22.11 Creating a new GRASS vector layer . . . 916
22.12 Digitizing and editing a GRASS vector layer . . . 916
22.13 The GRASS region tool . . . 919
22.14 The GRASS Toolbox . . . 919
22.14.1 Working with GRASS modules . . . 920
22.14.2 GRASS module examples . . . 922
22.14.3 Customizing the GRASS Toolbox . . . 925

23 QGIS okvir za obradu **927**
23.1 Introduction . . . 927
23.2 Configuring the Processing Framework . . . 930
23.2.1 General . . . 930
23.2.2 Menus . . . 932
23.2.3 Models and Scripts . . . 932
23.2.4 Providers . . . 932
23.3 The Toolbox . . . 932
23.3.1 The algorithm dialog . . . 935
23.3.2 Data objects generated by algorithms . . . 942
23.4 The history manager . . . 943
23.4.1 The processing history . . . 943
23.4.2 The processing log . . . 944
23.5 Dizajner modela . . . 944
23.5.1 Interfejs dizajnera modela . . . 945
23.5.2 Pravljenje modela . . . 948
23.5.3 Čuvanje i učitavanje modela . . . 956
23.5.4 Uređivanje modela . . . 957
23.6 The batch processing interface . . . 958
23.6.1 Introduction . . . 958
23.6.2 The parameters table . . . 959
23.6.3 Filling the parameters table . . . 960
23.6.4 Executing the batch process . . . 962
23.7 Using processing algorithms from the console . . . 962
23.7.1 Calling algorithms from the Python console . . . 962
23.7.2 Creating scripts and running them from the toolbox . . . 967
23.7.3 Pre- and post-execution script hooks . . . 970
23.8 Using processing from the command line . . . 970
23.9 Writing new Processing algorithms as Python scripts . . . 972
23.9.1 Extending QgsProcessingAlgorithm . . . 972
23.9.2 The @alg decorator . . . 976
23.9.3 Handing algorithm output . . . 978
23.9.4 Communicating with the user . . . 978
23.9.5 Documenting your scripts . . . 979
23.9.6 Flags . . . 979
23.9.7 Best practices for writing script algorithms . . . 979
23.10 Ulazni i izlazni tipovi za algoritme Obrade . . . 979

23.10.1 Ulazni tipovi . . . 979
23.10.2 Izlazni tipovi . . . 982
23.11 Configuring external applications . . . 983
23.11.1 A note for Windows users . . . 983
23.11.2 A note on file formats . . . 983
23.11.3 A note on vector layer selections . . . 983
23.11.4 Using third-party Providers . . . 984

24 Provajderi i algoritmi obrade **995**
24.1 Dobavljač QGIS algoritma . . . 995
24.1.1 3D pločice . . . 995
24.1.2 Kartografija . . . 997
24.1.3 Check Geometry . . . 1016
24.1.4 Baza podataka . . . 1061
24.1.5 Alati za datoteke . . . 1067
24.1.6 Fix Geometry . . . 1070
24.1.7 GPS . . . 1094
24.1.8 Interpolacija . . . 1097
24.1.9 Alatke za slojeve . . . 1110
24.1.10 Mesh . . . 1113
24.1.11 Alati za metapodatke . . . 1125
24.1.12 Alati za modelovanje . . . 1129
24.1.13 Analiza mreže . . . 1137
24.1.14 Grafikoni . . . 1151
24.1.15 Pretvaranje oblaka tačaka . . . 1163
24.1.16 Upravljanje podacima oblaka tačaka . . . 1170
24.1.17 Izdvajanje iz oblaka tačaka . . . 1181
24.1.18 Analiza rastera . . . 1187
24.1.19 Pravljenje rastera . . . 1250
24.1.20 Analiza rasterskog terena . . . 1266
24.1.21 Rasterski alati . . . 1279
24.1.22 Vektorska analiza . . . 1289
24.1.23 Vektorska pokrivenost . . . 1314
24.1.24 Kreiranje vektora . . . 1317
24.1.25 Vektor opšte . . . 1345
24.1.26 Vektorska geometrija . . . 1381
24.1.27 Vektorsko preklapanje . . . 1495
24.1.28 Izbor vektora . . . 1514
24.1.29 Vektorska tabela . . . 1529
24.1.30 Vektorske pločice . . . 1546
24.2 Dobavljač GDAL algoritma . . . 1549
24.2.1 Analiza rastera . . . 1549
24.2.2 Pretvaranje rastera . . . 1577
24.2.3 Izdvajanje iz rastera . . . 1585
24.2.4 Raster razno . . . 1593
24.2.5 Rasterske projekcije . . . 1611
24.2.6 Pretvaranje vektora . . . 1616
24.2.7 Vektorska geoobrada . . . 1622
24.2.8 Razni vektorski alati . . . 1630

25 Dodaci **1643**
25.1 QGIS Plugins . . . 1643
25.1.1 Core and External plugins . . . 1643
25.1.2 The Plugins Dialog . . . 1643
25.2 Korišćenje osnovnih QGIS dodataka . . . 1647
25.2.1 DB Manager Plugin . . . 1647
25.2.2 Geometry Checker Plugin . . . 1651
25.2.3 MetaSearch klijent kataloga . . . 1655

25.2.4 Dodatak za oflajn uređivanje . . . 1661
25.2.5 Topology Checker Plugin . . . 1664
25.3 QGIS Python console . . . 1666
25.3.1 The Interactive Console . . . 1667
25.3.2 The Code Editor . . . 1669

26 Dodaci **1673**
26.1 Appendix C: QGIS File Formats . . . 1673
26.1.1 QGS/QGZ - The QGIS Project File Format . . . 1673
26.1.2 QLR - The QGIS Layer Definition file . . . 1675
26.1.3 QML - The QGIS Style File Format . . . 1676
26.2 Appendix D: QGIS R script syntax . . . 1677
26.2.1 Inputs . . . 1678
26.2.2 Izlazi . . . 1678
26.2.3 Syntax Summary for QGIS R scripts . . . 1678
26.2.4 Primeri . . . 1680
26.3 Dodatak E: Mrežne veze QGIS aplikacije . . . 1682

27 Literatura i veb reference **1685**

ГЛАВА 1

Početak

Ovo poglavlje pruža kratak pregled instaliranja QGIS-a, preuzimanja QGIS primernih podataka i pokretanja prve jednostavne sesije vizuelizacije rasterskih i vektorskih podataka.

1.1 Instaliranje QGIS-a

QGIS projekat pruža različite načine za instaliranje QGIS-a u zavisnosti od vaše platforme.

1.1.1 Instaliranje iz binarnih datoteka

Standardni instalateri su dostupni za MS Windows i macOS. Binarni paketi (rpm i deb) ili softverski repozitorijumi su obezbeđjeni za mnoge varijante GNU/Linux.

Za više informacija i uputstava za vaš operativni sistem posetite https://qgis.org/download/.

1.1.2 Instaliranje iz izvornog koda

Ako treba da napravite QGIS iz izvornog koda, pogledajte uputstva za instalaciju. Ona se distribuiraju sa izvornim kodom QGIS-a u datoteci pod nazivom `INSTALL`. Takodje ih možete pronaći na mreži na https://github.com/qgis/QGIS/blob/release-3_44/INSTALL.md.

Ako želite da napravite određjeno izdanje, a ne verziju u razvoju, trebalo bi da zamenite `master` sa granom release (obično u obliku `release-X_Y`) u gore pomenutom linku (uputstva za instalaciju mogu se razlikovati).

1.1.3 Instaliranje na eksterne medije

Moguće je instalirati QGIS (sa svim dodacima i podešavanjima) na fleš disk. To se postiže definisanjem opcije *–profiles-path* koja zamenjuje podrazumevanu putanju *korisnički profil* i primorava **QSettings** da takođe koristi ovaj direktorijum. Pogledajte odeljak *System Settings* za dodatne informacije.

1.1.4 Preuzimanje primernih podataka

Ovaj korisnički vodič sadrži primere zasnovane na QGIS uzorku skupa podataka (takođje nazvanom „Aljaski skup podataka“). Preuzmite uzorke podataka sa https://github.com/qgis/QGIS-Sample-Data/archive/master.zip i raspakujte arhivu na bilo koju pogodnu lokaciju na vašem sistemu.

Skup podataka za Aljasku sadrži sve GIS podatke koji se koriste za primere i snimke ekrana u ovom korisničkom vodiču; takođje uključuje malu GRASS bazu podataka. Projekcija za QGIS primere skupova podataka je Aljaska Albers jednaka površina sa jedinicama stopa. EPSG kod je 2964.

```
PROJCS["Albers Equal Area",
GEOGCS["NAD27",
DATUM["North_American_Datum_1927",
SPHEROID["Clarke 1866",6378206.4,294.978698213898,
AUTHORITY["EPSG","7008"]],
TOWGS84[-3,142,183,0,0,0,0],
AUTHORITY["EPSG","6267"]],
PRIMEM["Greenwich",0,
AUTHORITY["EPSG","8901"]],
UNIT["degree",0.0174532925199433,
AUTHORITY["EPSG","9108"]],
AUTHORITY["EPSG","4267"]],
PROJECTION["Albers_Conic_Equal_Area"],
PARAMETER["standard_parallel_1",55],
PARAMETER["standard_parallel_2",65],
PARAMETER["latitude_of_center",50],
PARAMETER["longitude_of_center",-154],
PARAMETER["false_easting",0],
PARAMETER["false_northing",0],
UNIT["us_survey_feet",0.30480060960121921]]
```

Ako nameravate da koristite QGIS kao grafički interfejs za GRASS, možete pronaći izbor primera lokacija (npr. Spearfish ili Južna Dakota) na zvaničnoj GRASS GIS veb stranici, https://grass.osgeo.org/download/data/.

1.2 Pokretanje i zaustavljanje QGIS-a

QGIS se može pokrenuti kao i bilo koja druga aplikacija na vašem računaru. To znači da QGIS možete pokrenuti na sledeći način:

- korišćenjem menija Aplikacije, menija Start ili Dock-a
- dvostrukim klikom na ikonu u folderu Aplikacije ili prečici na radnoj površini
- dvostrukim klikom na postojeću datoteku QGIS projekta (sa ekstenzijom `.qgz` ili `.qgs`). Imajte na umu da će se ovim takođe otvoriti projekat.
- kucanjem „qgis“ u komandnoj liniji (pod pretpostavkom da je QGIS dodat u vašu putanju ili da ste u njegovoj instalacionoj fascikli)

Da biste zaustavili QGIS, koristite:

- opciju menija *Projekat* ► *Izađji iz QGIS-a* ili koristite prečicu `Ctrl+Q`
- *QGIS* ► *Zatvori QGIS*, ili koristi prečicu `Cmd+Q`
- ili koristite crveni krst u gornjem desnom uglu glavnog interfejsa aplikacije.

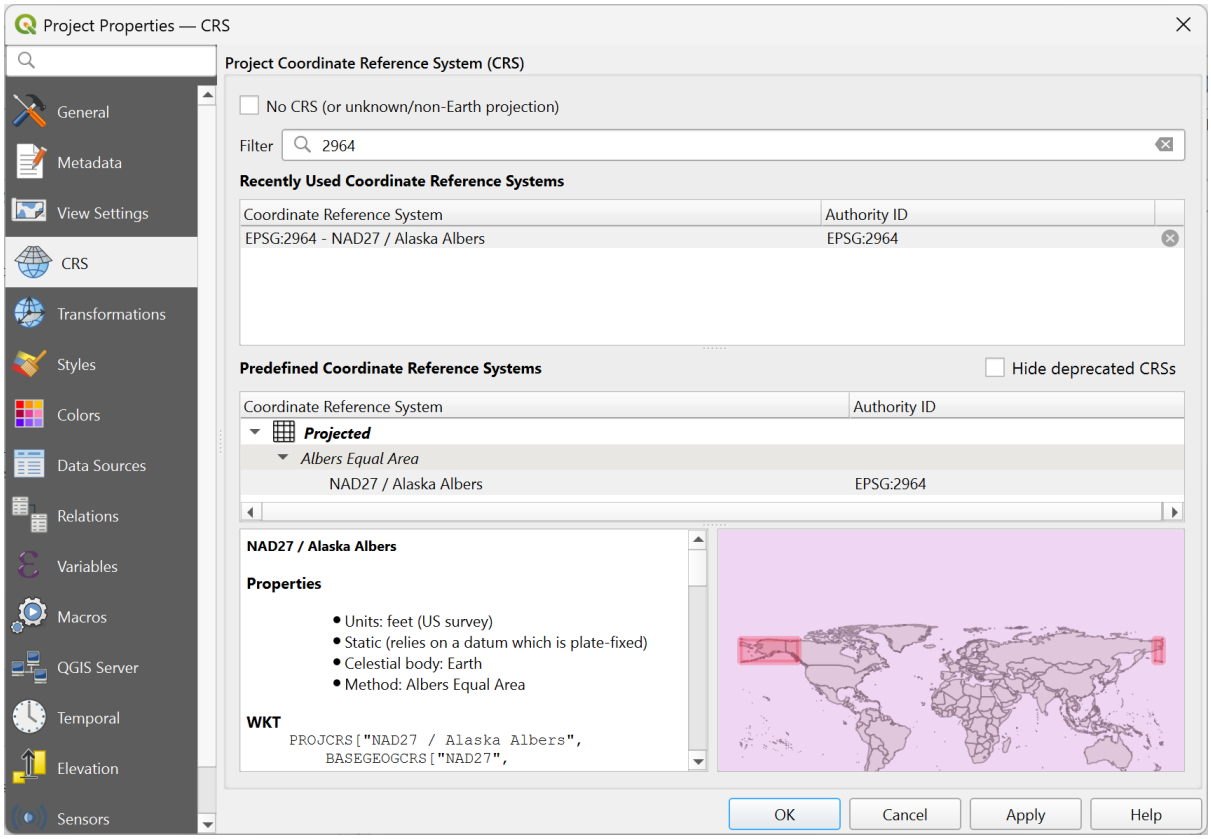
1.3 Primer sesije: Učitavanje rasterskih i vektorskih slojeva

Sada kada imate *QGIS instaliran* i *primer skupa podataka* dostupan, demonstriraćemo prvu primer sesije. U ovom primeru, vizualizovaćemo rasterski i vektorski sloj. Koristićemo:

- rasterski sloj `landcover` (`qgis_sample_data/raster/landcover.img`)
- i vektorski sloj `lakes` (`qgis_sample_data/gml/lakes.gml`)

Gde `qgis_sample_data` predstavlja putanju do raspakovanog skupa podataka.

1. Pokrenite QGIS kao što je prikazano u *Pokretanje i zaustavljanje QGIS-a*.
2. Podaci sa kojima ćemo raditi su u `Albers Equal Area`, pa hajde da u skladu sa tim podesimo CRS projekta:
    1. Kliknite na dugme Izbor projekcije u donjem desnom uglu QGIS interfejsa. Otvara se dijalog svojstava projekta sa aktivnom karticom *CRS*.
    2. Ukucajte `2964` u polje za tekst *Filter*.
    3. Izaberite red sa nazivom CRS-a `NAD27 / Alaska Albers`.

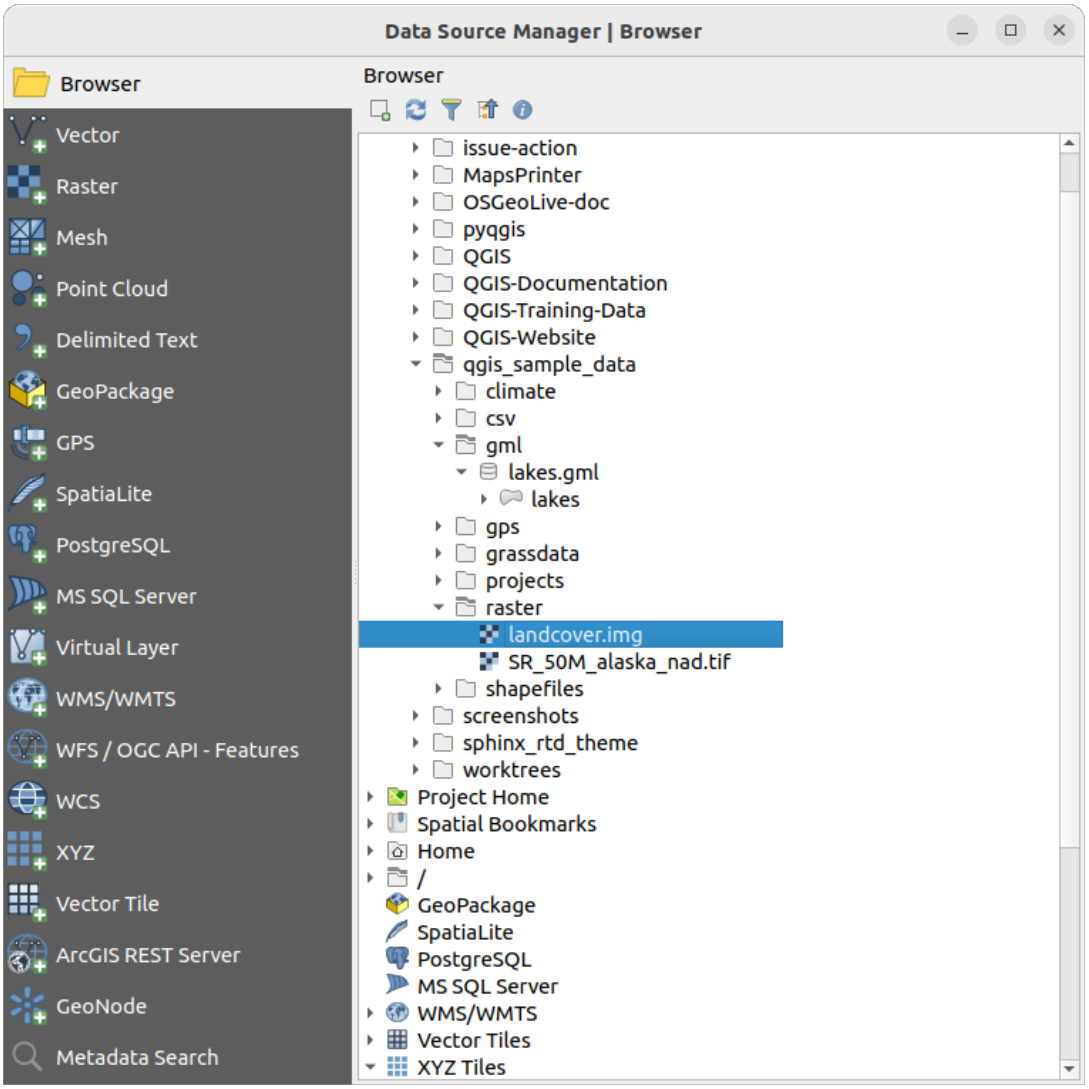


Слика1.1: Izbor koordinatnog referentnog sistema podataka

4. Pritisnite *OK*

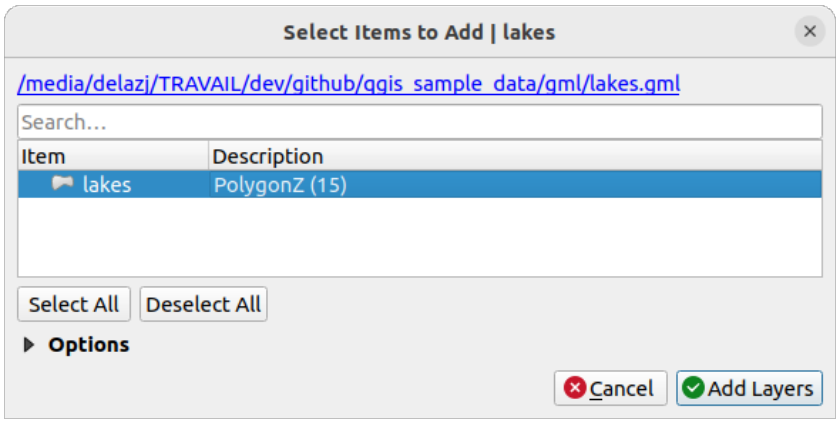
> **Белешка**
>
> Za sada možete ignorisati/zatvoriti poruku „ballpark transform“ koja se može pojaviti.

3. Učitajte datoteke u QGIS:
    1. Kliknite na ikonu Otvori upravljač izvorima podataka. Upravljač izvorima podataka bi trebalo da se otvori u režimu pregledača.
    2. Pređite u folder `qgis_sample_data/raster/`
    3. Izaberite ERDAS IMG datoteku *landcover.img* i dvaput kliknite na nju. Sloj landcover se dodaje u pozadini dok prozor upravljača izvorima podataka ostaje otvoren.



Слика1.2: Dodavanje podataka u novi projekat u QGIS-u

4. Da biste učitali podatke o jezerima, pređite u folder `qgis_sample_data/gml/` i prevucite datoteku *lakes.gml* na glavni prozor QGIS-a. (Ili jednostavno dvaput kliknite kao što je gore pomenuto.)
5. Otvara se dijalog *Izbor stavki za dodavanje* i skenira datoteku. Razlog je to što format datoteke `.gml` može da sadrži više slojeva odjednom.



Слика1.3: Izbor slojeva unutar datoteke

6. U našem slučaju postoji samo jedan sloj *lakes*. Izaberite ga i pritisnite *Dodaj slojeve*.
7. Sloj se dodaje u panel *Slojevi*

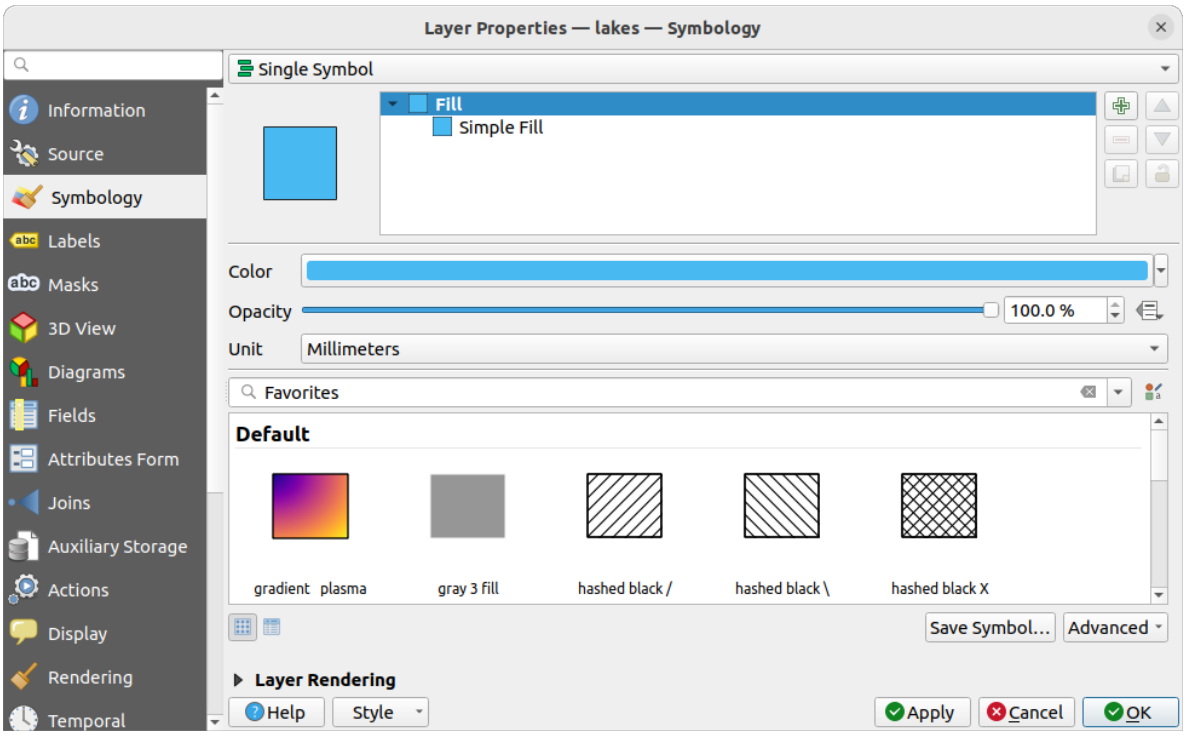
4. Zatvorite prozor upravljača izvorima podataka

U panelu *Slojevi* možete primetiti da sloj *lakes* pored sebe prikazuje Sloju nije podešen koordinatni referentni sistem. Hajde da to podesimo.

1. Kliknite na ikonu . Otvara se dijalog *Izbor koordinatnog referentnog sistema*.
2. Kao i ranije, pronađite i izaberite CRS stavku *NAD27 / Alaska Albers*.
3. Kliknite na *OK*

Sada imate oba sloja dostupna u projektu, u nasumično dodeljenim bojama. Hajde da malo prilagodimo sloj jezera.

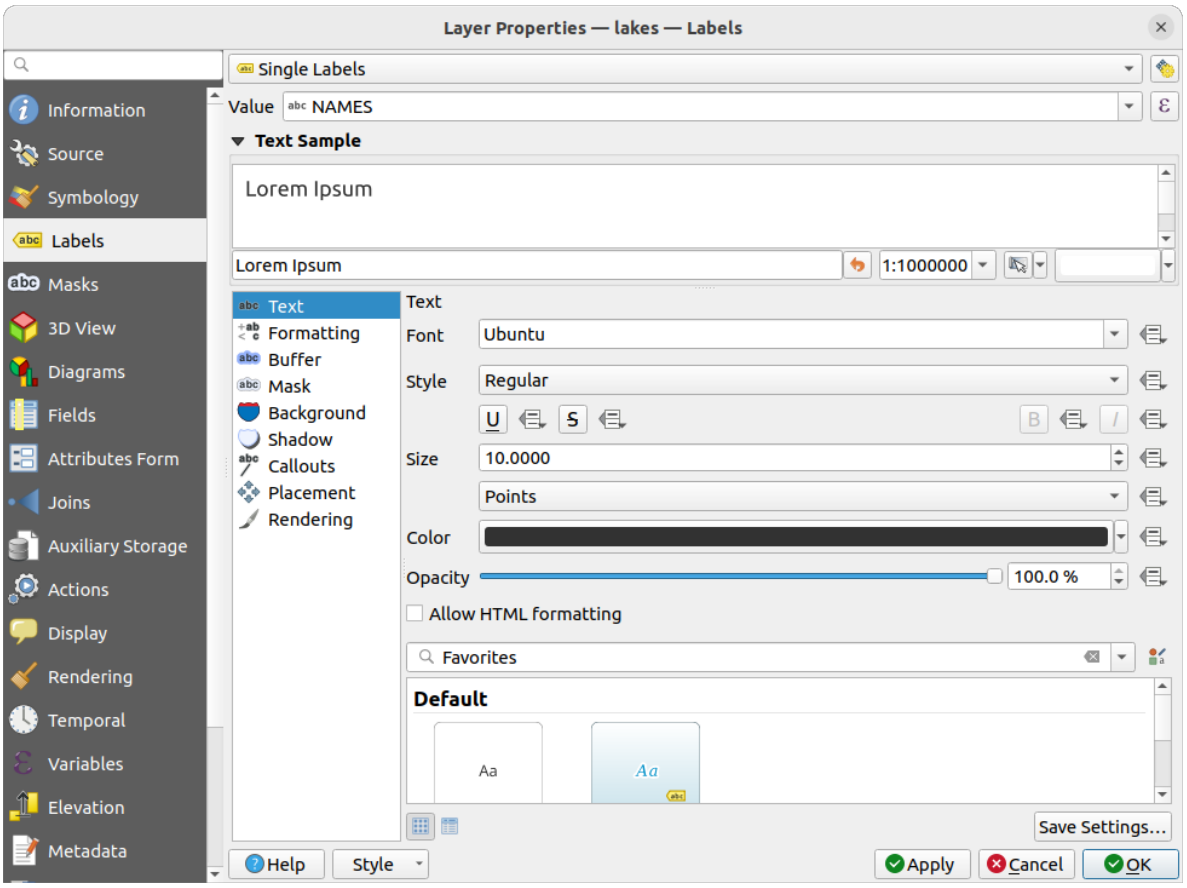
1. Izaberite alatku Uvećaj na traci sa alatkama *Navigacija*
2. Uvećajte prikaz na područje sa nekoliko jezera
3. Dvaput kliknite na sloj `lakes` u legendi karte da biste otvorili dijalog *Svojstva*
4. Da biste promenili boju jezera:
    1. Kliknite na karticu *Simbologija*
    2. Izaberite plavu kao boju popune.



Слика1.4: Izbor boje jezera

3. Pritisnite *OK*. Jezera se sada prikazuju plavom bojom na platnu karte.

5. Da biste prikazali nazive jezera:
   1. Ponovo otvorite dijalog *Svojstva* sloja `lakes`
   2. Kliknite na karticu *Oznake*
   3. Izaberite *Pojedinačne oznake* u padajućem meniju da biste omogućili označavanje.
   4. Iz liste *Označi sa* izaberite polje `NAMES`.

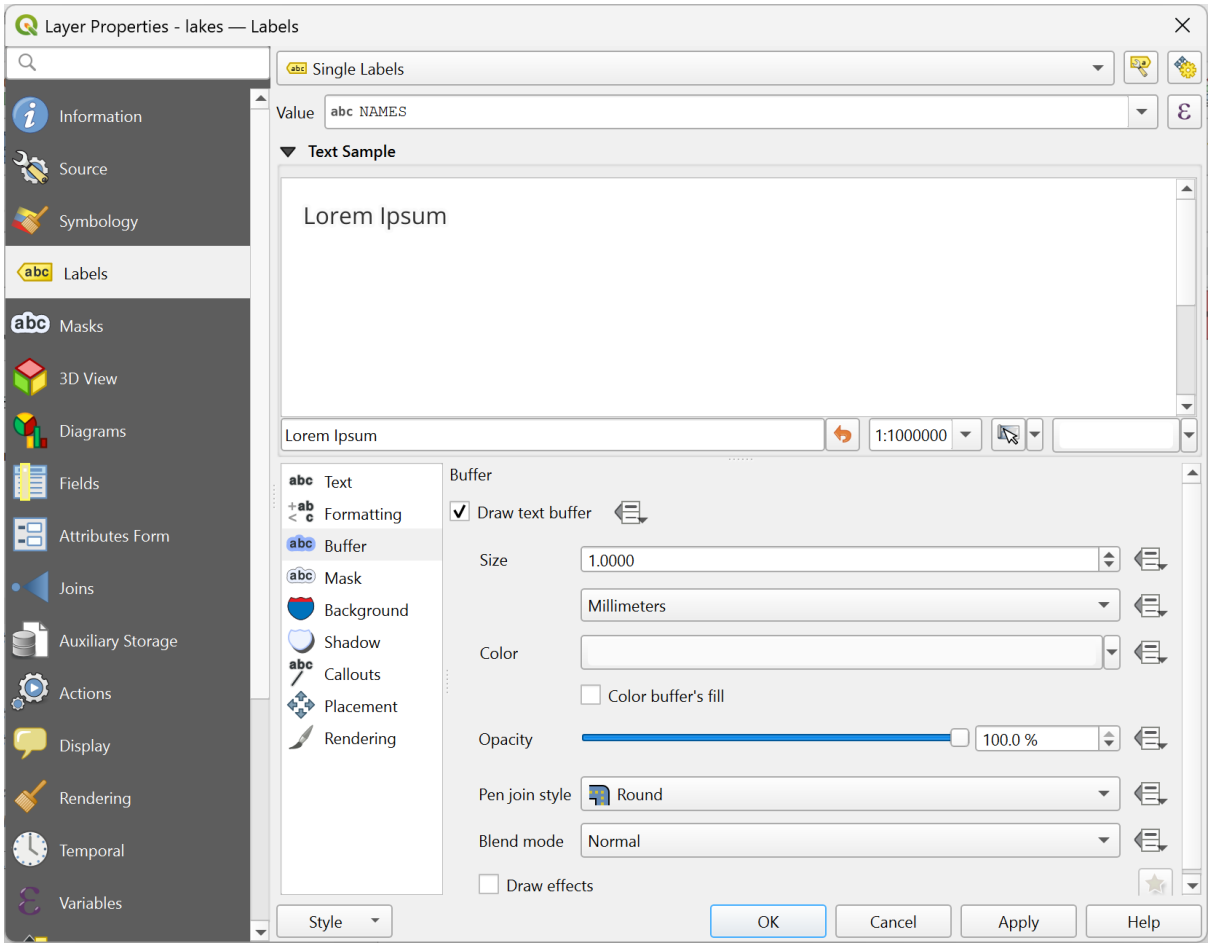


Слика1.5: Prikaz naziva jezera

5. Pritisnite *Primeni*. Nazivi će se sada učitati preko granica.

6. Čitljivost oznaka možete poboljšati dodavanjem belog bafera oko njih:

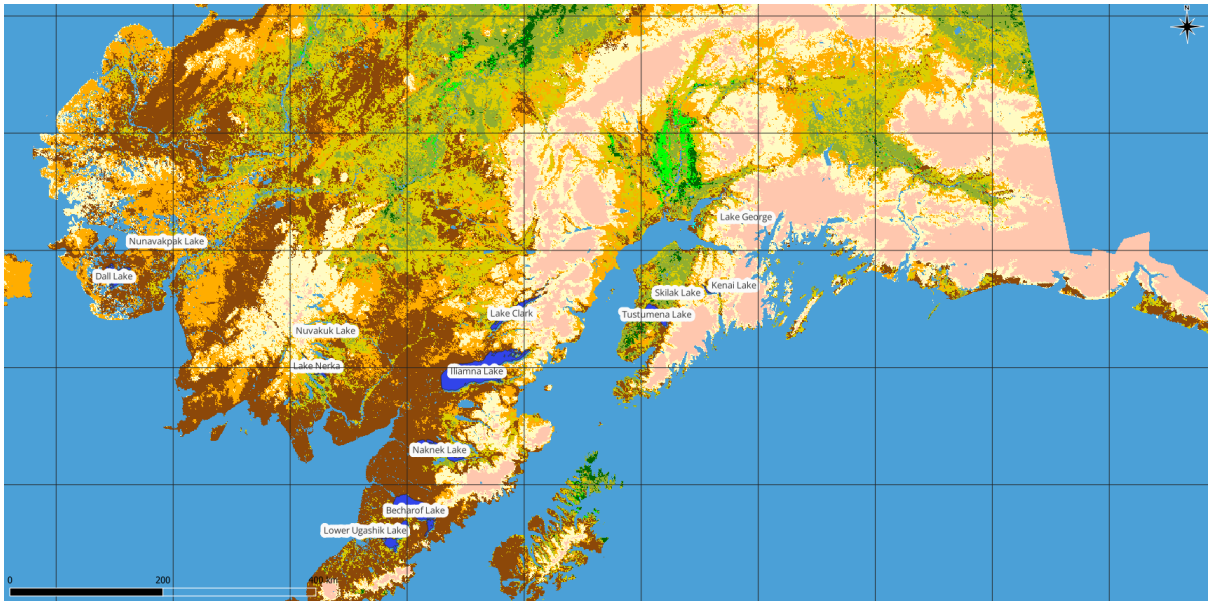
   1. Kliknite na karticu *Bafer* u listi sa leve strane
   2. Označite *Iscrtaj bafer teksta*
   3. Izaberite `3` kao veličinu bafera
   4. Kliknite na *Primeni*
   5. Proverite da li rezultat izgleda dobro i po potrebi izmenite vrednost.
   6. Na kraju kliknite na *OK* da biste zatvorili dijalog *Svojstva sloja* i primenili izmene.



Слика1.6: Prikaz bafera oko oznaka

Hajde sada da dodamo nekoliko dekoracija kako bismo oblikovali kartu i izvezli je iz QGIS-a:

1. Izaberite meni *Prikaz* ► *Dekoracije* ► *Razmernik*
2. U dijalogu koji se otvori označite opciju *Omogući razmernik*
3. Prilagodite opcije dijaloga po želji
4. Pritisnite *Primeni*
5. Na isti način, iz menija dekoracija, dodajte još stavki (strelica severa, autorska prava…) na platno karte sa prilagođenim svojstvima.
6. Kliknite na *Projekat* ► *Uvoz/Izvoz* ► *Izvezi kartu u sliku…*
7. Pritisnite *Sačuvaj* u dijalogu koji se otvori
8. Izaberite lokaciju datoteke i format, pa potvrdite ponovnim pritiskom na *Sačuvaj*.
9. Pritisnite *Projekat* ► *Sačuvaj…* da biste sačuvali izmene kao `.qgz` datoteku projekta.



Слика1.7: Prikaz izvezene karte sa dekoracijama

To je to! Vidite koliko je lako vizuelizovati rasterske i vektorske slojeve u QGIS-u, podesiti ih i napraviti kartu u formatu slike koji možete koristiti u drugim programima. Hajde da nastavimo i saznamo više o dostupnim funkcijama, mogućnostima i podešavanjima, i o tome kako se koriste.

Белешка

Da biste nastavili da učite QGIS kroz vežbe korak po korak, pratite Priručnik za obuku.

ГЛАВА 2

Working with Project Files

2.1 Introducing QGIS projects

The state of your QGIS session is called a project. QGIS works on one project at a time. A setting can be project--specific or an application-wide default for new projects (see section *Options*). QGIS can save the state of your workspace into a *QGIS project file* using the menu options *Project* ► *Save* or *Project* ► *Save As...*.

Белешка

If the project has been modified the * symbol will appear in the title bar and QGIS will, by default, ask you if you would like to save the changes. This behavior is controlled by the *Prompt to save project and data source changes when required* setting under *Settings* ► *Options* ► *General*.

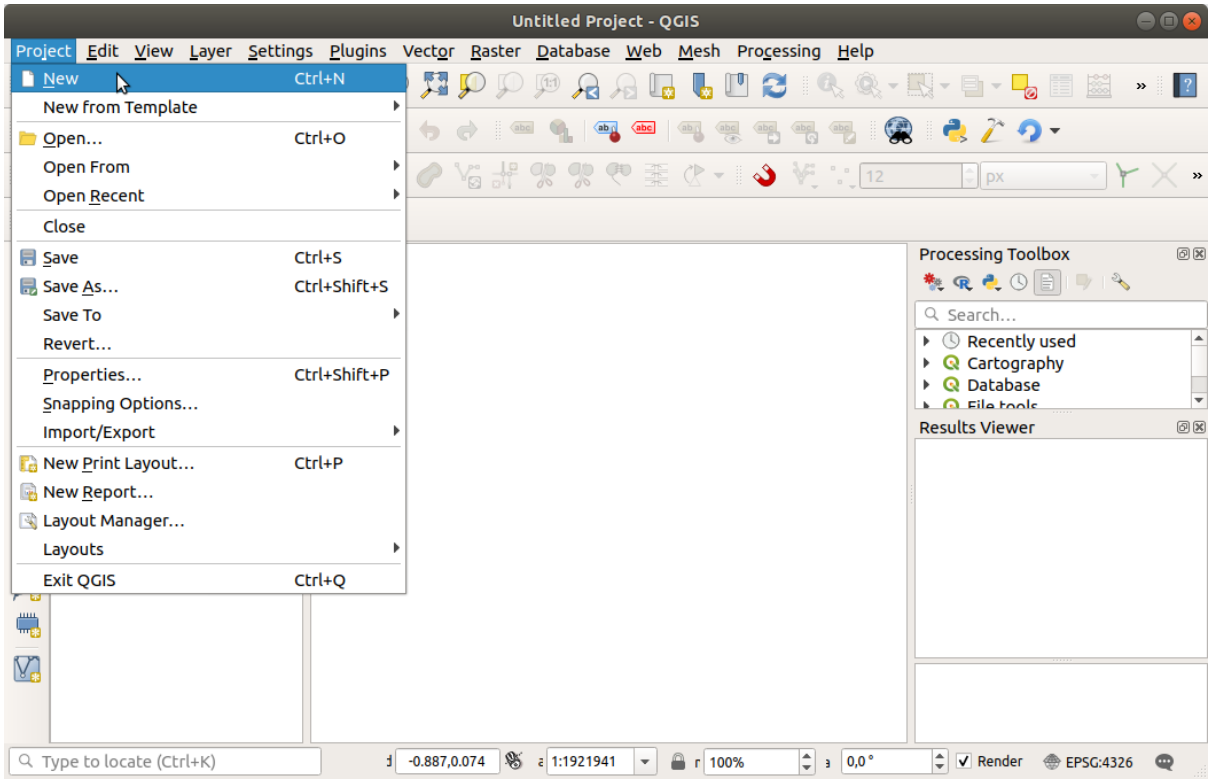
You can load existing projects through *Project* ► *Open...*, *Project* ► *New from template* or *Project* ► *Open Recent* ►.

From the *Browser panel* you can *Open Project*, *Extract Symbols...* – opens the style manager to export symbols to XML, add them to the default style, or export as PNG/SVG. You can also access the *File Properties...* dialog. The project file can be expanded to see its layers. The context menu of a layer offers the same actions as elsewhere in the browser.

At startup, a list of *Project Templates* and *Recent Projects* are displayed, including screenshots, names and file paths (for up to ten projects). The *Recent Projects* list is handy to access recently used projects. Double-click an entry to open the project or project template. Right-click an entry to *Pin to List*, *Open Directory...* or *Remove from List*. If you see your project on the *Recent Projects* list but can't find it in your file manager use the *Open Directory...* option to help you locate projects that may be missing, moved or renamed. You can also go to *Clear List* if you want to remove all projects from the *Recent Projects* list. If you have pinned projects, the *Clear List* action will be followed by message box asking whether the pinned projects should also be removed. You can also add a layer to create a new project automatically. The lists will then disappear, giving way to the map canvas.

If you want to clear your session and start fresh, go to *Project* ► *New*. This will prompt you to save the existing project if changes have been made since it was opened or last saved.

When you open a fresh project, the title bar will show `Untitled Project` until you save it.



Слика2.1: Starting a new project in QGIS

The information saved in a project file includes:

- Layers added
- Which layers can be queried
- Layer properties, including symbolization and styles
- Layer notes
- 2D and 3D map views
- Projection for each map view
- Last viewed extent for each map
- Print layouts
- Print layout elements with settings
- Print layout atlas settings
- Digitizing settings
- Table Relations
- Project Macros
- Project default styles
- Plugins settings
- QGIS Server settings from the OWS settings tab in the Project properties
- Queries stored in the DB Manager

The project file is saved in XML format (see *QGS/QGZ - The QGIS Project File Format*). This means that it is possible to edit the file outside of QGIS if you know what you are doing. The project file format has been updated several times. Project files from older QGIS versions may not work properly any more.

Белешка

By default, QGIS will warn you of version differences. This behavior is controlled in the *General* tab of *Settings* ► *Options* (☑ *Warn when opening a project file saved with an older version of QGIS*).

Whenever you save a `.qgs` project file in QGIS, a backup of the file is created in the same directory as the project file, with the extension `.qgs~`.

The extension for QGIS projects is `.qgs` but when saving from QGIS, the default is to save using a compressed format with the `.qgz` extension. The `.qgs` file is embedded in the `.qgz` file (a zip archive), together with its associated SQLite database (`.qgd`) for *auxiliary data*. You can get to these files by unzipping the `.qgz` file.

Белешка

The *Auxiliary Storage Properties* mechanism makes a zipped project particularly useful, since it embeds auxiliary data.

Projects can also be saved/loaded to/from a PostgreSQL, GeoPackage or Oracle database using the following Project menu items:

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

Both menu items have a sub-menu with a list of extra project storage implementations (PostgreSQL, GeoPackage and Oracle). Clicking the action will open a dialog to pick a GeoPackage connection and project, a PostgreSQL connection, schema and project or Oracle connection, owner and project.

Projects stored in GeoPackage, PostgreSQL or Oracle can also be loaded through the QGIS browser panel, either by double-clicking them or by dragging them to the map canvas.

2.2 Handling broken file paths

When opening a project, QGIS may fail to reach some data sources due to unavailable service/database, or to a renamed or moved file. QGIS then opens the *Handle Unavailable Layers* dialog, referencing the unfound layers. You can:

- Double-click in the *Datasource* field, adjust the path of each layer and click *Apply changes*;
- Select a row, press *Browse* to indicate the correct location and click *Apply changes*;
- Press *Auto-Find* to browse the folders and try to automatically fix all or selected broken path(s). Be aware that the browsing may take some time. Then click *Apply changes*.
- Ignore the message and open your project with the broken path(s) by clicking *Keep Unavailable Layers*.

  Your layer is then displayed in the *Layers* panel, but without any data until you fix the path using the ⚠ Unavailable layer! icon next to it in the *Layers* panel, or *Repair Data Source…* in the layer contextual menu.

  With the *Repair Data Source…* tool, once a layer path has been fixed, QGIS scans through all other broken paths and tries to auto-fix those that have the same broken file path.

- 🗑 *Remove Unavailable Layers* from the project.

Launching QGIS from command line using the *--skipbadlayers* option can help you skip the *Handle Unavailable Layers* dialog at startup.

2.3 Generating output

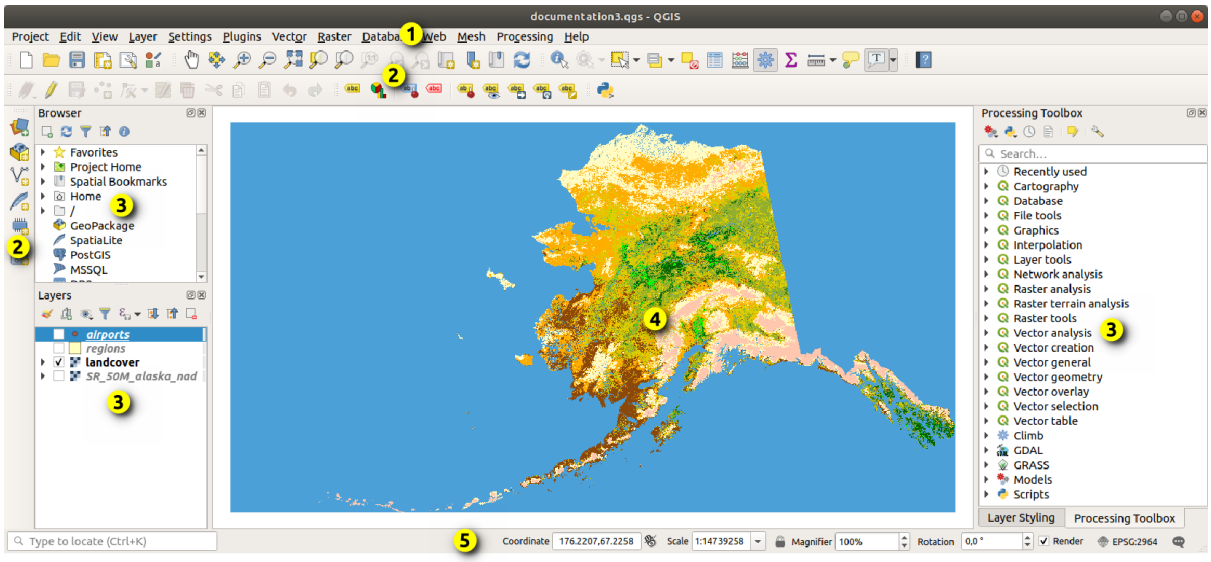
There are several ways to generate output from your QGIS session. We have already discussed saving as a project file in *Introducing QGIS projects*. Other ways to produce output files are:

- Creating images: *Project* ► *Import/Export* ►  *Export Map to Image…* outputs the map canvas rendering to an image format (PNG, JPG, TIFF…) at custom scale, resolution and size. Including georeference information in the exported image is possible, simply enable  *Append georeference information (embedded or via world file)*. See *Exporting the map view* for more details.
- Exporting to PDF files: *Project* ► *Import/Export* ► *Export Map to PDF…* outputs the map canvas rendering to PDF at custom scale, resolution, and with some advanced settings (simplification, georeferencing, …). See *Exporting the map view* for more details.
- Exporting to DXF files: *Project* ► *Import/Export* ► *Export Project to DXF…* opens a dialog where you can define the 'Symbology mode', the 'Symbology scale' and vector layers you want to export to DXF. Through the 'Symbology mode', symbols from the original QGIS Symbology can be exported with high fidelity (see section *Creating new DXF files*).
- Designing maps: *Project* ►  *New Print Layout…* opens a dialog where you can layout and print the current map canvas (see section *Postavljanje mapa*).

ГЛАВА 3

QGIS GUI

Графичко корисничко окруженje (GUI) QGIS-a прикazano je na slici ispod (brojevi od 1 do 5 u žutim krugovima označavaju važne elemente QGIS GUI-ja i objašnjeni su u nastavku).



Слика3.1: QGIS GUI sa uzorkom podataka za Aljasku

Белешка

Ukrasi prozora (naslovna traka i sl.) mogu izgledati drugačije u zavisnosti od vašeg operativnog sistema i menadžera prozora.

Glavni QGIS GUI (Сл.3.1) sastoji se od pet komponenti / tipova komponenti:

1. *Traka menija*
2. *Trake sa alatkama*
3. *Paneli*

4. *Prikaz karte*
5. *Statusna traka*

Skrolujte nadole za detaljna objašnjenja.

3.1 Traka menija

Traka menija omogućava pristup QGIS funkcijama pomoću standardnih hijerarhijskih menija. Meniji, njihove opcije, pripadajuće ikone i prečice na tastaturi opisani su u nastavku. Prečice na tastaturi mogu se ponovo podesiti (*Podešavanja* ► *Prečice na tastaturi*).

Većina opcija menija ima odgovarajuću alatku i obrnuto. Međutim, meniji nisu organizovani potpuno isto kao trake sa alatkama. Položaji opcija menija u trakama sa alatkama naznačeni su u tabeli ispod. Dodaci mogu dodati nove opcije u menije. Više informacija o alatkama i trakama sa alatkama potražite u *Trake sa alatkama*.

Белешка

QGIS je višeplatformska aplikacija. Alatke su uglavnom dostupne na svim platformama, ali mogu biti smeštene u različitim menijima, u zavisnosti od operativnog sistema. Spiskovi ispod prikazuju najčešće položaje, uključujući poznate varijacije.

3.1.1 Projekat

Meni *Projekat* omogućava pristup i izlaz za *projektne datoteke*. Sadrži alatke za:

- Pravljenje *Nove* projektne datoteke od nule ili korišćenje druge projektne datoteke kao šablona (pogledajte *Opcije projektnih datoteka* za podešavanje šablona)
- *Otvaranje…* projekta iz datoteke, GeoPackage-a, PostgreSQL ili Oracle baze podataka
- *Zatvaranje* projekta ili vraćanje na poslednje sačuvano stanje
- *Čuvanje* projekta u `.qgs` ili `.qgz` formatu datoteke, bilo kao datoteku ili unutar GeoPackage-a, PostgreSQL ili Oracle baze podataka
- Izvoz platna karte u različite formate ili korišćenje *štampanog prikaza* za složeniji izlaz
- Podešavanje svojstava projekta i opcija lepljenja za uređivanje geometrije.

Табела3.1: Stavke menija Projekat

Opcija menija	Prečica	Traka sa alatkama	Referenca
Novo	`Ctrl+N`	*Projekat*	*Introducing QGIS projects*
Novo iz šablona ►			*Introducing QGIS projects*
Otvori…	`Ctrl+O`	*Projekat*	*Introducing QGIS projects*
Otvori iz ►			*Introducing QGIS projects*
► *GeoPackage…*			*Introducing QGIS projects*
► *PostgreSQL…*			*Introducing QGIS projects*
► *Oracle…*			*Introducing QGIS projects*
Otvori nedavno ►	`Alt+J + R`		*Introducing QGIS projects*
► *Očisti listu*			*Introducing QGIS projects*
Zatvori			*Introducing QGIS projects*

наставак на следећој страни

Табела 3.1 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
Sačuvaj	`Ctrl+S`	*Projekat*	*Introducing QGIS projects*
Sačuvaj kao…	`Ctrl+Shift+S`	*Projekat*	*Introducing QGIS projects*
Sačuvaj u ►			*Introducing QGIS projects*
► *Šabloni…*			*Introducing QGIS projects*
► *GeoPackage…*			*Introducing QGIS projects*
► *PostgreSQL…*			*Introducing QGIS projects*
Vrati…			
Svojstva…	`Ctrl+Shift+P`		*Project Properties*
Opcije lepljenja…			*Snapping and Digitizing Options*
Uvoz/Izvoz ►			
► *Izvezi kartu u sliku…*			*Exporting the map view*
► *Izvezi kartu u PDF…*			*Exporting the map view*
► *Izvezi projekat u DXF…*			*Creating new DXF files*
► *Uvezi slojeve iz DWG/DXF…*			*Importing a DXF or DWG file*
Novi štampani prikaz…	`Ctrl+P`	*Projekat*	*Postavljanje mapa*
Novi izveštaj…			*Creating a Report*
Menadžer prikaza…		*Projekat*	*Postavljanje mapa*
Prikazi ►			*Postavljanje mapa*
Modeli ►			*Dizajner modela*
Izađi iz QGIS-a	`Ctrl+Q`		

Na macOS-u, komanda *Izađi iz QGIS-a* odgovara *QGIS* ► *Napusti QGIS* (`Cmd+Q`).

3.1.2 Uređivanje

Meni *Uređivanje* sadrži većinu ugrađenih alatki potrebnih za uređivanje atributa ili geometrije sloja. Da biste omogućili opcije menija *Uređivanje*, morate preći u režim uređivanja klikom na Uključi/isključi uređivanje (pogledajte *Uređivanje* za detalje).

Табела3.2: Stavke menija Uređivanje

Opcija menija	Prečica	Traka sa alatkama	Referenca
Poništi	`Ctrl+Z`	*Digitalizacija*	*Undo and Redo*
Ponovi	`Ctrl+Shift+Z`	*Digitalizacija*	*Undo and Redo*
Iseci objekte	`Ctrl+X`	*Digitalizacija*	*Cutting, Copying and Pasting Features*

наставак на следећој страни

Табела 3.2 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
Kopiraj objekte	`Ctrl+C`	*Digitalizacija*	*Cutting, Copying and Pasting Features*
Nalepi objekte	`Ctrl+V`	*Digitalizacija*	*Cutting, Copying and Pasting Features*
Nalepi objekte kao ►			*Rad sa atributskom tabelom*
► *Novi vektorski sloj…*			*Rad sa atributskom tabelom*
► *Privremeni radni sloj…*	`Ctrl+Alt+V`		*Rad sa atributskom tabelom*
Obriši izabrano		*Digitalizacija*	*Deleting Selected Features*
Izbor ►			*Izbor objekata*
► *Izaberi objekat(e)*		*Izbor*	*Izbor objekata*
► *Izaberi objekte poligonom*		*Izbor*	*Izbor objekata*
► *Izaberi objekte slobodnom rukom*		*Izbor*	*Izbor objekata*
► *Izaberi objekte polupречnikom*		*Izbor*	*Izbor objekata*
► *Izaberi objekte po vrednosti…*	`F3`	*Izbor*	*Izbor objekata*
► *Izaberi objekte izrazom…*	`Ctrl+F3`	*Izbor*	*Izbor objekata*
► *Poništi izbor objekata na svim slojevima*	`Ctrl+Alt+A`	*Izbor*	*Izbor objekata*
► *Poništi izbor objekata na trenutno aktivnom sloju*	`Ctrl+Shift+A`	*Izbor*	*Izbor objekata*
► *Ponovo izaberi objekte*			*Izbor objekata*
► *Izaberi sve objekte*	`Ctrl+A`	*Izbor*	*Izbor objekata*
► *Obrni izbor objekata*		*Izbor*	*Izbor objekata*
Dodaj zapis	`Ctrl+.`	*Digitalizacija*	
Dodaj tačkasti objekat	`Ctrl+.`	*Digitalizacija*	*Adding Features*
Dodaj linijski objekat	`Ctrl+.`	*Digitalizacija*	*Adding Features*
Dodaj poligonski objekat	`Ctrl+.`	*Digitalizacija*	*Adding Features*
Dodaj napomenu ►			*Annotation Tools*
► *Napomena u formi*		*Napomene*	*Annotation Tools*
► *HTML napomena*		*Napomene*	*Annotation Tools*
Uredi atribute ►			
► *Izmeni atribute izabranih objekata*		*Digitalizacija*	*Uređivanje vrednosti atributa*
► *Spoji atribute izabranih objekata*		*Napredna digitalizacija*	*Merge attributes of selected features*

наставак на следећој страни

Табела 3.2 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
Uredi geometriju ►			
► *Pomeri objekat(e)*		*Napredna digitalizacija*	*Move Feature(s)*
► *Kopiraj i pomeri objekat(e)*		*Napredna digitalizacija*	*Move Feature(s)*
► *Rotiraj objekat(e)*		*Napredna digitalizacija*	*Rotate Feature(s)*
► *Skaliraj objekat(e)*		*Napredna digitalizacija*	*Scale Feature*
► *Uprosti objekat*		*Napredna digitalizacija*	*Simplify Feature*
► *Dodaj prsten*		*Napredna digitalizacija*	*Add Ring*
► *Dodaj deo*		*Napredna digitalizacija*	*Add Part*
► *Popuni prsten*		*Napredna digitalizacija*	*Fill Ring*
► *Obriši prsten*		*Napredna digitalizacija*	*Delete Ring*
► *Obriši deo*		*Napredna digitalizacija*	*Delete Part*
► *Preoblikuj objekte*		*Napredna digitalizacija*	*Reshape Features*
► *Odmakni krivu*		*Napredna digitalizacija*	*Offset Curves*
► *Podeli objekte*		*Napredna digitalizacija*	*Split Features*
► *Podeli delove*		*Napredna digitalizacija*	*Split parts*

наставак на следећој страни

Табела 3.2 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
► *Spoji izabrane objekte*		*Napredna digitalizacija*	*Merge selected features*
► *Alatka za temena (svi slojevi)*		*Digitalizacija*	*Vertex tool*
► *Alatka za temena (trenutni sloj)*		*Digitalizacija*	*Vertex tool*
► *Obrni liniju*		*Napredna digitalizacija*	*Reverse Line*
► *Skrati/produži objekat*		*Napredna digitalizacija*	*Trim/Extend Feature*
Rotiraj tačkaste simbole		*Napredna digitalizacija*	*Rotate Point Symbols*
Odmakni tačkaste simbole		*Napredna digitalizacija*	*Offset Point Symbols*

Alatke koje zavise od tipa geometrije izabranog sloja, tj. tačka, polilinija ili poligon, aktiviraju se u skladu s tim:

Табела3.3: Ikone alatke „Pomeri objekat“ prema tipu geometrije

Opcija menija	Tačka	Polilinija	Poligon
Pomeri objekat(e)			
Kopiraj i pomeri objekat(e)			

3.1.3 Prikaz

Karta se iscrtava u prikazima karte. Sa ovim prikazima možete raditi pomoću alatki menija *Prikaz*. Na primer, možete:

- Napraviti nove 2D ili 3D prikaze karte pored glavnog platna karte
- *Uvećati ili pomerati* do bilo kog mesta
- Postavljati upite nad atributima ili geometrijom prikazanih objekata
- Obogatiti prikaz karte režimima pregleda, napomenama ili ukrasima
- Pristupiti bilo kom panelu ili traci sa alatkama

Meni takođe omogućava reorganizaciju samog QGIS interfejsa pomoću radnji kao što su:

- *Uključi/isključi režim celog ekrana*: prekriva ceo ekran uz skrivanje naslovne trake
- *Uključi/isključi vidljivost panela*: prikazuje ili skriva omogućene *panele* - korisno pri digitalizaciji objekata (za maksimalnu vidljivost platna), kao i za (projektovane/snimane) prezentacije uz korišćenje glavnog QGIS platna
- *Prikaži samo kartu*: skriva panele, trake sa alatkama, menije i statusnu traku i prikazuje samo platno karte. U kombinaciji sa opcijom celog ekrana, na ekranu se prikazuje samo karta.

Табела3.4: Stavke menija Prikaz

Opcija menija	Prečica	Traka sa alatkama	Referenca
Novi prikaz karte	`Ctrl+M`	*Navigacija kartom*	*2D Map View*
3D prikazi karte ►			*3D Map View*
► *Novi 3D prikaz karte*	`Ctrl+Alt+M`	*Navigacija kartom*	*3D Map View*
► *Upravljaj 3D prikazima karte*			*3D Map View*
Pomeraj kartu		*Navigacija kartom*	*Exploring the map view*
Pomeri kartu na izbor		*Navigacija kartom*	*Exploring the map view*
Uvećaj	`Ctrl+Alt++`	*Navigacija kartom*	*Exploring the map view*
Umanji	`Ctrl+Alt+-`	*Navigacija kartom*	*Exploring the map view*
Filtriranje podataka ►			*2D Map View*
► *Vremenski kontroler*		*Navigacija kartom*	*The temporal controller panel*
► *Visinski kontroler*			*The Elevation Controller panel*
Identifikuj objekte	`Ctrl+Shift+I`	*Atributi*	*Identifying Features*
Merenje ►		*Atributi*	*Measuring*
► *Izmeri liniju*	`Ctrl+Shift+M`	*Atributi*	*Measuring*
► *Izmeri površinu*	`Ctrl+Shift+J`	*Atributi*	*Measuring*
► *Izmeri azimut*		*Atributi*	*Measuring*
► *Izmeri ugao*		*Atributi*	*Measuring*
Statistički pregled		*Atributi*	*Statistical Summary Panel*
Visinski profil			*Elevation Profile View*
Uvećaj na pun obuhvat	`Ctrl+Shift+F`	*Navigacija kartom*	*Exploring the map view*
Uvećaj na izbor	`Ctrl+J`	*Navigacija kartom*	*Exploring the map view*

наставак на следећој страни

Табела 3.4 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
Uvećaj na sloj(eve)		*Navigacija kartom*	*Exploring the map view*
Uvećaj na izvornu rezoluciju (100%)		*Navigacija kartom*	*Exploring the map view*
Prethodno uvećanje		*Navigacija kartom*	*Exploring the map view*
Sledeće uvećanje		*Navigacija kartom*	*Exploring the map view*
Ukrasi ►	`Alt+V + D`		*Decorating the map*
► *Mreža…*			*Grid*
► *Razmernik…*			*Scale Bar*
► *Slika…*			*Image Decoration*
► *Strelica severa…*			*North Arrow*
► *Natpis naslova…*			*Title Label*
► *Natpis o autorskim pravima…*			*Copyright Label*
► *Obuhvati prikaza…*			*Layout Extents*
Režim pregleda ►			
► *Normalno*			
► *Simuliraj monohromatski prikaz*			
► *Simuliraj ahromatopsiju (sivi tonovi)*			
► *Simuliraj protanopiju (bez crvene)*			
► *Simuliraj deuteranopiju (bez zelene)*			
► *Simuliraj tritanopiju (bez plave)*			
Prikaži savete na karti		*Atributi*	*Display Properties*
Novi prostorni obeleživač…	`Ctrl+B`	*Navigacija kartom*	*Bookmarking extents on the map*
Prikaži prostorne obeleživače	`Ctrl+Shift+B`	*Navigacija kartom*	*Bookmarking extents on the map*
Prikaži menadžer prostornih obeleživača			*Bookmarking extents on the map*
Osveži	`F5`	*Navigacija kartom*	
Vidljivost slojeva ►			*Layers Panel*
► *Prikaži sve slojeve*	`Ctrl+Shift+U`		*Layers Panel*
► *Sakrij sve slojeve*	`Ctrl+Shift+H`		*Layers Panel*
► *Prikaži izabrane slojeve*			*Layers Panel*

наставак на следећој страни

Табела 3.4 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
► *Sakrij izabrane slojeve*			*Layers Panel*
► *Uključi/isključi izabrane slojeve*			*Layers Panel*
► *Uključi/isključi izabrane slojeve nezavisno*			*Layers Panel*
► *Sakrij neizabrane slojeve*			*Layers Panel*
Paneli ►			*Paneli i trake sa alatkama*
Trake sa alatkama ►			*Paneli i trake sa alatkama*
Uključi/isključi režim celog ekrana	`F11`		
Uključi/isključi vidljivost panela	`Ctrl+Tab`		
Prikaži samo kartu	`Ctrl+Shift+Tab`		

Na Linux KDE-u, *Paneli* ►, *Trake sa alatkama* ► i *Uključi/isključi režim celog ekrana* nalaze se u meniju *Podešavanja*.

3.1.4 Sloj

Meni *Sloj* sadrži veliki skup alatki za *pravljenje* novih izvora podataka, *dodavanje* u projekat ili *čuvanje izmena* nad njima. Koristeći iste izvore podataka, možete i:

- *Duplirati* sloj i tako napraviti kopiju u kojoj možete promeniti naziv, stil (simbologiju, natpise, …), spojeve, … Kopija koristi isti izvor podataka kao original.
- *Kopirati* i *Nalepiti* slojeve ili grupe iz jednog projekta u drugi kao novu instancu čija se svojstva mogu menjati nezavisno. Kao i kod *Dupliranja*, slojevi se i dalje zasnivaju na istom izvoru podataka.
- ili *Ugraditi slojeve i grupe…* iz drugog projekta, kao kopije samo za čitanje koje ne možete menjati (pogledajte *Embedding layers from external projects*)

Meni *Sloj* takođe sadrži alatke za podešavanje, kopiranje ili lepljenje svojstava sloja (stil, razmera, CRS…).

Табела3.5: Stavke menija Sloj

Opcija menija	Prečica	Traka sa alatkama	Referenca
Menadžer izvora podataka	`Ctrl+L`	*Menadžer izvora podataka*	*Otvaranje podataka*
Napravi sloj ►			*Creating new vector layers*
► *Novi GeoPackage sloj…*	`Ctrl+Shift+N`	*Menadžer izvora podataka*	*Creating a new GeoPackage layer*
► *Novi Shapefile sloj…*		*Menadžer izvora podataka*	*Creating a new Shapefile layer*
► *Novi SpatiaLite sloj…*		*Menadžer izvora podataka*	*Creating a new SpatiaLite layer*
► *Novi privremeni radni sloj…*		*Menadžer izvora podataka*	*Creating a new Temporary Scratch Layer*
► *Novi mrežni sloj…*		*Menadžer izvora podataka*	*Creating a new Mesh layer*

наставак на следећој страни

Табела 3.5 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatka-ma	Referenca
► *Novi GPX sloj…*		*Menadžer izvora podataka*	*Creating a new GPX layer*
► *Novi virtuelni sloj…*		*Menadžer izvora podataka*	*Creating virtual layers*
Dodaj sloj ►			*Opening Data*
► *Dodaj vektorski sloj……*	`Ctrl+Shift+V`	*Upravljanje slojevima*	*Loading a layer from a file*
► *Dodaj rasterski sloj…*	`Ctrl+Shift+R`	*Upravljanje slojevima*	*Loading a layer from a file*
► *Dodaj mrežni sloj…*		*Upravljanje slojevima*	*Loading a mesh layer*
► *Dodaj sloj iz razgraničenog teksta…*	`Ctrl+Shift+T`	*Upravljanje slojevima*	*Importing a delimited text file*
► *Dodaj PostgreSQL sloj…*	`Ctrl+Shift+D`	*Upravljanje slojevima*	*Database related tools*
► *Dodaj SpatiaLite sloj…*	`Ctrl+Shift+L`	*Upravljanje slojevima*	*SpatiaLite Layers*
► *Dodaj MS SQL Server sloj…*		*Upravljanje slojevima*	*Database related tools*
► *Dodaj Oracle Spatial sloj…*		*Upravljanje slojevima*	*Database related tools*
► *Dodaj SAP HANA Spatial sloj…*		*Upravljanje slojevima*	*Database related tools*
► *Dodaj/uredi virtuelni sloj…*		*Upravljanje slojevima*	*Creating virtual layers*
► *Dodaj WMS/WMTS sloj…*	`Ctrl+Shift+W`	*Upravljanje slojevima*	*Loading WMS/WMTS Layers*
► *Dodaj XYZ sloj…*			*Using XYZ Tile services*
► *Dodaj WCS sloj…*		*Upravljanje slojevima*	*WCS Client*
► *Dodaj WFS / OGC API - Features sloj…*		*Upravljanje slojevima*	*WFS and WFS-T Client*
► *Dodaj ArcGIS REST Server sloj…*		*Upravljanje slojevima*	*Using ArcGIS REST Servers*
► *Dodaj sloj vektorskih pločica…*			*Using Vector Tiles services*
► *Dodaj sloj oblaka tačaka…*			*Working with Point Clouds*
► *Dodaj GPX sloj…*			*Introducing GNSS/GPS Data*
Ugradi slojeve i grupe…			*Embedding layers from external projects*
Dodaj iz datoteke definicije sloja…			*Layer definition file*
Georeferencer…			*Georeferencer*

наставак на следећој страни

Табела 3.5 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatka-ma	Referenca
Kopiraj stil			*Save and Share Layer Properties*
Nalepi stil			*Save and Share Layer Properties*
Kopiraj sloj			
Nalepi sloj/grupu			
Otvori atributsku tabelu	`F6`	*Atributi*	*Rad sa atributskom tabelom*
Filtriraj atributsku tabelu ►			*Rad sa atributskom tabelom*
► *Otvori atributsku tabelu (izabrani objekti)*	`Shift+F6`	*Atributi*	*Rad sa atributskom tabelom*
► *Otvori atributsku tabelu (vidljivi objekti)*	`Ctrl+F6`	*Atributi*	*Rad sa atributskom tabelom*
► *Otvori atributsku tabelu (uređeni i novi objekti)*		*Atributi*	*Rad sa atributskom tabelom*
Uključi/isključi uređivanje		*Digitalizacija*	*Digitizing an existing layer*
Sačuvaj izmene sloja		*Digitalizacija*	*Saving Edited Layers*
Trenutne izmene ►		*Digitalizacija*	*Saving Edited Layers*
► *Sačuvaj za izabrani sloj(eve)*		*Digitalizacija*	*Saving Edited Layers*
► *Poništi za izabrani sloj(eve)*		*Digitalizacija*	*Saving Edited Layers*
► *Otkaži za izabrani sloj(eve)*		*Digitalizacija*	*Saving Edited Layers*
► *Sačuvaj za sve slojeve*		*Digitalizacija*	*Saving Edited Layers*
► *Poništi za sve slojeve*		*Digitalizacija*	*Saving Edited Layers*
► *Otkaži za sve slojeve*		*Digitalizacija*	*Saving Edited Layers*
Sačuvaj kao…			*Creating new layers from an existing layer*
Sačuvaj kao datoteku definicije sloja…			*Layer definition file*
Ukloni sloj/grupu	`Ctrl+D`		
Dupliraj sloj(eve)			
Podesi vidljivost sloja(eva) po razmeri			*Visibility Scale Selector*
Podesi CRS sloja(eva)	`Ctrl+Shift+C`		*Layer Coordinate Reference Systems*
Podesi CRS projekta prema sloju			*Project Coordinate Reference Systems*
Svojstva sloja…			• *vector* • *raster* • *mreža* • *oblak tačaka* • *vektorske pločice*
Filter…	`Ctrl+F`		*Query Builder*
Natpisi			*Labels Properties*
Prikaži u pregledu			*Overview Panel*
Prikaži sve u pregledu			*Overview Panel*
Sakrij sve iz pregleda			*Overview Panel*

3.1.5 Podešavanja

Табела3.6: Stavke menija Podešavanja

Opcija menija	Referenca
Korisnički profili ►	*Working with User Profiles*
► *podrazumevani*	*Working with User Profiles*
► *Otvori folder aktivnog profila*	*Working with User Profiles*
► *Novi profil…*	*Working with User Profiles*
Menadžer stilova…	*The Style Manager*
Prilagođene projekcije…	*Custom Coordinate Reference System*
Prečice na tastaturi…	*Keyboard shortcuts*
Prilagođavanje interfejsa…	*Customization*
Opcije…	*Options*

Na Linux KDE-u pronaći ćete više alatki u meniju *Podešavanja*, kao što su *Paneli* ►, *Trake sa alatkama* ► i *Uključi/isključi režim celog ekrana*.

3.1.6 Dodaci

Табела3.7: Stavke menija Dodaci

Opcija menija	Prečica	Traka sa alatkama	Referenca
Upravljaj i instaliraj dodatke…			*The Plugins Dialog*
Python konzola	`Ctrl+Alt+P`	*Dodaci*	*QGIS Python console*

Pri prvom pokretanju QGIS-a nisu učitani svi osnovni dodaci.

3.1.7 Vektor

Meni *Vektor* sadrži alatke za analizu, upravljanje i manipulaciju vektorskim skupovima podataka. Podrazumevano je popunjen skupom unapred definisanih algoritama *Obrade*, grupisanih po podmenijima. Time se dobijaju prečice za mnoge uobičajene GIS zadatke nad vektorima od osnovnih provajdera. Ako nisu svi ovi podmeniji dostupni, omogućite dodatak *Obrada* u *Dodaci* ► *Upravljaj i instaliraj dodatke…*.

Белешка

Lista alatki i njihov meni mogu se izmeniti/proširiti bilo kojim algoritmima Obrade (pročitajte *Configuring the Processing Framework*) ili nekim *dodacima* trećih strana.

Ovako izgleda meni *Vektor* ako su svi osnovni dodaci omogućeni.

Табела3.8: Stavke menija Vektor

Opcija menija	Prečica	Traka sa alatkama	Referenca
Proveri geometrije…			*Geometry Checker Plugin*

наставак на следећој страни

Табела 3.8 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alat-kama	Referenca
Provera topologije		*Vektor*	*Topology Checker Plugin*
Alatke za geoobradu ►	`Alt+O + G`		
► *Bafer…*			*Bafer*
► *Iseci…*			*Isecanje*
► *Konveksni omotač…*			*Konveksni omotač*
► *Razlika…*			*Razlika*
► *Stapanje…*			*Stapanje*
► *Presek…*			*Presek*
► *Simetrična razlika…*			*Simetrična razlika*
► *Unija…*			*Unija*
► *Eliminiši izabrane poligone…*			*Eliminiši izabrane poligone*
Alatke za geometriju ►	`Alt+O + E`		
► *Centroidi…*			*Centroidi*
► *Sakupi geometrije…*			*Prikupi geometrije*
► *Izdvoj temena…*			*Izdvoj temena*
► *Višedelno u jednodelno…*			*Višedelno u jednodelno*
► *Poligoni u linije…*			*Poligoni u linije*
► *Uprosti…*			*Uprosti*
► *Proveri ispravnost…*			*Proveri validnost*
► *Delonijeva triangulacija…*			*Delonijeva triangulacija*
► *Zgusni po broju…*			*Zgusni po broju*
► *Dodaj atribute geometrije…*			*Dodaj atribute geometrije*
► *Linije u poligone…*			*Linije u poligone*
► *Voronoji poligoni…*			*Voronoi poligoni*
Alatke za analizu ►	`Alt+O + A`		
► *Presek linija…*			*Presek linija*
► *Srednja koordinata(e)…*			*Srednje koordinate*
► *Osnovna statistika za polja…*			*Osnovna statistika za polja*
► *Prebroj tačke u poligonu…*			*Brojanje tačaka u poligonu*
► *Matrica rastojanja…*			*Matrica rastojanja*
► *Izlistaj jedinstvene vrednosti…*			*Izlistavanje jedinstvenih vrednosti*
► *Analiza najbližeg suseda…*			*Analiza najbližeg suseda*
► *Zbir dužina linija…*			*Zbir dužina linija*
Alatke za upravljanje podacima ►	`Alt+O + D`		
► *Spoji vektorske slojeve…*			*Spajanje vektorskih slojeva*

наставак на следећој страни

Табела 3.8 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alat-kama	Referenca
► *Reprojektuj sloj…*			*Reprojektovanje sloja*
► *Napravi prostorni indeks…*			*Kreiranje prostornog indeksa*
► *Spoji atribute po lokaciji…*			*Spajanje atributa po lokaciji*
► *Podeli vektorski sloj…*			*Deljenje vektorskog sloja*
Istraživačke alatke ►	Alt+O + R		
► *Napravi mrežu…*			*Kreiranje mreže*
► *Izdvoj obuhvat sloja…*			*Izdvajanje obuhvata sloja*
► *Slučajne tačke u obuhvatu…*			*Slučajne tačke u obuhvatu*
► *Slučajne tačke u poligonima…*			*Slučajne tačke u poligonima*
► *Slučajne tačke na linijama…*			*Slučajne tačke na linijama*
► *Izaberi po lokaciji…*			*Izbor po lokaciji*
► *Izaberi unutar rastojanja…*			*Izbor unutar rastojanja*
► *Slučajne tačke u granicama sloja…*			*Slučajne tačke u granicama sloja*
► *Slučajne tačke unutar poligona…*			*Slučajne tačke unutar poligona*
► *Slučajan izbor…*			*Slučajan izbor*
► *Slučajan izbor unutar podskupova…*			*Slučajan izbor unutar podskupova*
► *Pravilne tačke…*			*Pravilne tačke*

3.1.8 Raster

Meni *Raster* sadrži alatke za analizu, upravljanje i manipulaciju rasterskim skupovima podataka. Podrazumevano je popunjen skupom unapred definisanih algoritama *Obrade*, grupisanih po podmenijima. Time se dobijaju prečice za mnoge uobičajene GIS zadatke nad rasterima od osnovnih provajdera. Ako nisu svi ovi podmeniji dostupni, omogućite dodatak *Obrada* u *Dodaci* ► *Upravljaj i instaliraj dodatke…*.

Белешка

Lista alatki i njihov meni mogu se izmeniti/proširiti bilo kojim algoritmima Obrade (pročitajte *Configuring the Processing Framework*) ili nekim *dodacima* trećih strana.

Ovako izgleda meni *Raster* ako su svi osnovni dodaci omogućeni.

Табела3.9: Stavke menija Raster

Opcija menija	Prečica	Traka sa alat-ka-ma	Referenca
Rasterski kalkulator…			*Raster Calculator*
Poravnaj rastere…			*Poravnaj rastere*
Analiza ►			

наставак на следећој страни

Табела 3.9 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
► *Ekspozicija…*			*Ekspozicija*
► *Popuni nodata…*			*Popuni NoData*
► *Mreža (pokretni prosek)…*			*Mreža (pokretni prosek)*
► *Mreža (metrike podataka)…*			*Mreža (metrike podataka)*
► *Mreža (inverzno rastojanje na stepen)…*			*Mreža (inverzno rastojanje na stepen)*
► *Mreža (najbliži sused)…*			*Mreža (IDW sa pretragom najbližeg suseda)*
► *Skoro crno…*			*Skoro crno*
► *Osenčeni reljef…*			*Osenčeni reljef*
► *Blizina (rastojanje u rasteru)…*			*Blizina (rastojanje u rasteru)*
► *Hrapavost…*			*Hrapavost*
► *Prosej…*			*Prosej*
► *Nagib…*			*Nagib*
► *Indeks topografskog položaja (TPI)…*			*Indeks topografskog položaja (TPI)*
► *Indeks razuđenosti terena (TRI)…*			*Indeks razuđenosti terena (TRI)*
Projekcije ►			
► *Dodeli projekciju…*			*Dodela projekcije*
► *Izdvoj projekciju…*			*Izdvajanje projekcije*
► *Warp (reprojektovanje)…*			*Warp (reprojektovanje)*
Razno ►			
► *Napravi virtuelni raster…*			*Kreiranje virtuelnog rastera*
► *Informacije o rasteru…*			*Informacije o rasteru*
► *Spoji…*			*Spajanje*
► *Napravi preglede (piramide)…*			*Kreiranje pregleda (piramida)*
► *Indeks pločica…*			*Indeks pločica*
Izdvajanje ►			
► *Iseci raster po obuhvatu…*			*Iseci raster po obuhvatu*
► *Iseci raster maskirnim slojem…*			*Iseci raster maskirnim slojem*
► *Izolinije…*			*Izolinije*
Pretvaranje ►			
► *PCT u RGB…*			*PCT u RGB*
► *Poligonizuj (raster u vektor)…*			*Poligonizuj (raster u vektor)*
► *Rasterizuj (vektor u raster)…*			*Rasterizacija (vektor u raster)*

наставак на следећој страни

Табела 3.9 – настављено са претходне стране

Opcija menija	Prečica	Traka sa alatkama	Referenca
► *RGB u PCT…*			*RGB u PCT*
► *Translate (pretvori format)…*			*Translate (pretvori format)*

3.1.9 Baza podataka

Meni *Baza podataka* sadrži alatke za rad sa bazama podataka. Ako nije omogućen nijedan dodatak vezan za baze podataka, meni *Baza podataka* neće postojati. Ovako izgleda meni *Baza podataka* ako su svi osnovni dodaci omogućeni.

Табела3.10: Stavke menija Baza podataka

Opcija menija	Prečica	Traka sa alatkama	Referenca
Uređivanje van mreže…	`Alt+D` + `O`		*Dodatak za oflajn uređivanje*
► *Pretvori u projekat van mreže…*		*Baza podataka*	*Dodatak za oflajn uređivanje*
► *Sinhronizuj*		*Baza podataka*	*Dodatak za oflajn uređivanje*
Istorija upita…		*Baza podataka*	*Creating SQL Query Layers*
DB menadžer…		*Baza podataka*	*DB Manager Plugin*

Pri prvom pokretanju QGIS-a nisu učitani svi osnovni dodaci.

3.1.10 Veb

Ovako izgleda meni *Veb* ako su svi osnovni dodaci omogućeni. Ako nije omogućen nijedan veb dodatak, meni *Veb* neće postojati.

Табела3.11: Stavke menija Veb

Opcija menija	Prečica	Traka sa alatkama	Referenca
MetaSearch ►	`Alt+W` + `M`		*MetaSearch klijent kataloga*
► *Metasearch*		*Veb*	*MetaSearch klijent kataloga*
► *Pomoć*			*MetaSearch klijent kataloga*

Pri prvom pokretanju QGIS-a nisu učitani svi osnovni dodaci.

3.1.11 Mesh

Meni *Mreža* sadrži alatke potrebne za rad sa *mrežnim slojevima*. Dodaci trećih strana mogu dodati stavke u ovaj meni.

Таблела3.12: Stavke menija Mreža

Opcija menija	Prečica	Traka sa alatkama	Referenca
Kalkulator mreže…			*Mesh Calculator*
Ponovo indeksiraj lica i temena			*Reindexing meshes*

3.1.12 Obrada

Таблела3.13: Stavke menija Obrada

Opcija menija	Prečica	Traka sa alat-kama	Referenca
Kutija sa alatkama	`Ctrl+Alt+T`		*The Toolbox*
Dizajner modela…	`Ctrl+Alt+G`		*Dizajner modela*
Istorija…	`Ctrl+Alt+H`		*The history manager*
Pregledač rezultata	`Ctrl+Alt+R`		*Configuring external applications*
Uredi objekte na mestu			*The Processing in-place layer modifier*

Pri prvom pokretanju QGIS-a nisu učitani svi osnovni dodaci.

3.1.13 Pomoć

Таблела3.14: Stavke menija Pomoć

Opcija menija	Prečica	Traka sa alatkama	Referenca
Sadržaj pomoći	`F1`	*Pomoć*	
API dokumentacija			
Dodaci ►			
Prijavi problem			
Potrebna vam je komercijalna podrška?			
QGIS početna strana	`Ctrl+H`		
Proveri verziju QGIS-a			
O programu			
Podržavaoci QGIS-a			

3.1.14 QGIS

Ovaj meni je dostupan samo na macOS-u i sadrži neke komande vezane za operativni sistem.

Табела3.15: Stavke menija QGIS

Opcija menija	Prečica	Referenca
O QGIS-u		
Postavke	`Cmd+,`	*Options*
Sakrij QGIS	`Cmd+H`	
Prikaži sve		
Sakrij ostale	`Shift+Cmd+H`	
Napusti QGIS	`Cmd+Q`	

Postavke odgovara *Podešavanja* ► *Opcije*, *O QGIS-u* odgovara *Pomoć* ► *O programu*, a *Napusti QGIS* odgovara *Projekat* ► *Izađi iz QGIS-a* na drugim platformama.

3.2 Paneli i trake sa alatkama

Iz menija *Prikaz* (ili *Podešavanja*) možete uključivati i isključivati QGIS vidžete (*Paneli* ►) i trake sa alatkama (*Trake sa alatkama* ►). Da biste ih (de)aktivirali, kliknite desnim tasterom na traku menija ili traku sa alatkama i izaberite željenu stavku. Paneli i trake sa alatkama mogu se pomerati i postavljati gde god želite unutar QGIS interfejsa. Lista se može proširiti i aktiviranjem *osnovnih ili spoljnih dodataka*.

3.2.1 Trake sa alatkama

Trake sa alatkama omogućavaju pristup većini funkcija iz menija, uz dodatne alatke za rad sa kartom. Svaka stavka trake sa alatkama ima iskačuću pomoć. Zadržite miš iznad stavke i prikazaće se kratak opis namene alatke.

Dostupne trake sa alatkama su:

Табела3.16: QGIS trake sa alatkama

Naziv trake sa alatkama	Glavna referenca za alatke
Napredna digitalizacija	*Advanced digitizing*
Napomene	*Annotation Tools*
Atributi	*Rad sa atributskom tabelom*, *General Tools*
Menadžer izvora podataka	*Upravljanje izvorom podataka*
Baza podataka	*DB Manager Plugin*
Digitalizacija	*Digitizing an existing layer*
GPS	*Live GPS tracking*
GRASS	*GRASS GIS Integration*
Pomoć	
Natpis	*The Label Toolbar*
Upravljanje slojevima	*Opening Data*
Navigacija kartom	*Exploring the map view*
Digitalizacija mreže	*Editing a mesh layer*
Dodaci	*Dodaci*
Projekat	*Working with Project Files*, *Postavljanje mapa*, *Biblioteka stilova*
Algoritmi obrade	*Configuring the Processing Framework*
Raster	*Dodaci*
Izbor	*Izbor objekata*
Digitalizacija oblika	*Shape digitizing*
Lepljenje	*Setting the snapping tolerance and search radius*
Vektor	*Dodaci*
Veb	*Dodaci*, *MetaSearch klijent kataloga*

Белешка

Dodaci trećih strana mogu proširiti podrazumevanu traku sa alatkama sopstvenim alatkama ili obezbediti sopstvenu traku sa alatkama.

Савет

Vraćanje traka sa alatkama

Ako ste slučajno sakrili traku sa alatkama, možete je vratiti pomoću *Prikaz* ► *Trake sa alatkama* ► (ili *Podešavanja* ► *Trake sa alatkama* ►). Ako iz nekog razloga traka sa alatkama (ili neki drugi vidžet) potpuno nestane iz interfejsa, savete kako da je vratite naći ćete u *vraćanje početnog GUI-ja*.

3.2.2 Paneli

QGIS nudi mnoge panele. Paneli su posebni vidžeti sa kojima možete raditi (birati opcije, označavati polja, unositi vrednosti…) radi izvršavanja složenijih zadataka.

Ispod je lista podrazumevanih panela koje nudi QGIS:

Табела3.17: QGIS paneli

Naziv panela	Prečica	Referenca
Napredna digitalizacija	`Ctrl+4`	*The Advanced Digitizing panel*
Pregledač	`Ctrl+2`	*The Browser Panel*
Pregledač (2)		*The Browser Panel*
Alatke za otklanjanje grešaka/razvoj	`F12`	*Debugging/Development Tools Panel*
Visinski profil		*Elevation Profile View*
Provera geometrije		*Digitizing Properties*
GPS informacije	`Ctrl+0`	*Live GPS tracking*
GRASS alatke		*GRASS GIS Integration*
Redosled slojeva	`Ctrl+9`	*Layer Order Panel*
Stilizovanje sloja	`Ctrl+3`	*Layer Styling Panel*
Slojevi	`Ctrl+1`	*Layers Panel*
Poruke dnevnika		*Log Messages Panel*
Pregled	`Ctrl+8`	*Overview Panel*
Kutija sa alatkama za obradu		*The Toolbox*
Pregledač rezultata		*The Toolbox*
Opcije lepljenja i digitalizacije		*Setting the snapping tolerance and search radius*
*Menadžer prostornih obeleživa*č	`Ctrl+7`	*Bookmarking extents on the map*
Statistika	`Ctrl+6`	*Statistical Summary Panel*
Vremenski kontroler		*The temporal controller panel*
Razmera pločica		*Tilesets*
Poništi/ponovi	`Ctrl+5`	*Undo/Redo Panel*
Uređivač temena		*The Vertex Editor Panel*

3.3 Statusna traka

Statusna traka pruža opšte informacije o prikazu karte i o obrađenim ili dostupnim radnjama, a nudi i alatke za upravljanje prikazom karte.

3.3.1 Traka lokatora

Na levoj strani statusne trake, traka lokatora — vidžet za brzu pretragu — pomaže vam da pronađete i pokrenete bilo koji objekat ili opciju u QGIS-u:

1. Kliknite u tekstualni vidžet da biste aktivirali traku pretrage lokatora ili pritisnite `Ctrl+K`.
2. Ukucajte tekst povezan sa stavkom koju tražite (naziv, oznaka, ključna reč, …). Podrazumevano se rezultati vraćaju za omogućene filtere lokatora, ali pretragu možete ograničiti na određeni opseg dodavanjem prefiksa *filtera lokatora* ispred teksta, tj. kucanje `l cad` vratiće samo slojeve čiji naziv sadrži `cad`.

   Filter se može izabrati i dvoklikom u meniju koji se prikazuje pri pristupu vidžetu lokatora.
3. Kliknite na rezultat da biste izvršili odgovarajuću radnju, u zavisnosti od tipa stavke.

Савет

Ograničite pretragu na određena polja aktivnog sloja

Podrazumevano, pretraga sa filterom „objekti aktivnog sloja“ (`f`) prolazi kroz celu atributsku tabelu sloja. Pretragu možete ograničiti na određeno polje pomoću prefiksa `@`. Npr. `f @name sal` ili `@name sal` vraća samo objekte čiji atribut „name“ sadrži 'sal'. Automatsko dopunjavanje teksta je aktivno tokom pisanja, a predlog se može primeniti tasterom `Tab`.

Napredniju kontrolu nad poljima nad kojima se vrši upit možete postići iz kartice *Polja* sloja. Pročitajte *Fields Properties* za detalje.

Pretraga se obavlja pomoću niti, tako da rezultati uvek postaju dostupni što je brže moguće, čak i ako su instalirani spori filteri pretrage. Oni se takođe pojavljuju čim ih filter pronađe, što znači da će npr. filter za pretragu datoteka prikazivati rezultate jedan po jedan dok se skenira stablo datoteka. Time se obezbeđuje da korisnički interfejs uvek reaguje, čak i ako je prisutan veoma spor filter pretrage (npr. onaj koji koristi onlajn servis).

Белешка

Alatka lokatora Nominatim može se ponašati drugačije (bez pretrage sa automatskim dopunjavanjem, kašnjenje u preuzimanju rezultata, …) u skladu sa pravilima korišćenja OpenStreetMap Nominatim servisa.

Савет

Brz pristup podešavanjima lokatora

Kliknite na ikonu  unutar vidžeta lokatora na statusnoj traci da biste prikazali listu filtera koje možete koristiti i stavku *Konfiguriši* koja otvara karticu *Lokator* u meniju *Podešavanja* ► *Opcije…*.

3.3.2 Prikaz radnji

U delu pored trake lokatora, po potrebi se prikazuje sažetak radnji koje ste izvršili (kao što su izbor objekata na sloju, uklanjanje sloja, rastojanje i pravac pomeranja) ili duži opis alatke iznad koje se nalazi pokazivač (nije dostupno za sve alatke).

U slučaju dugotrajnih operacija, kao što su sakupljanje statistike na rasterskim slojevima, izvršavanje algoritama Obrade ili iscrtavanje više slojeva u prikazu karte, na statusnoj traci se prikazuje traka napretka.

3.3.3 Upravljanje platnom karte

Opcija  *Koordinata* prikazuje trenutni položaj miša, prateći ga dok se kreće preko prikaza karte. Jedinice (i preciznost) možete podesiti u kartici *Projekat* ► *Svojstva…* ► *Opšte*. Kliknite na malo dugme levo od tekstualnog

polja da biste prebacivali između opcije Koordinata i opcije  *Obuhvati*, koja prikazuje koordinate trenutnog donjeg levog i gornjeg desnog ugla prikaza karte u jedinicama karte.

Pored prikaza koordinata naći ćete prikaz *Razmera*. On prikazuje razmeru prikaza karte. Postoji birač razmere koji vam omogućava izbor između *unapred definisanih i prilagođenih razmera*.

Sa desne strane prikaza razmere pritisnite dugme  da biste zaključali razmeru radi uvećavanja ili umanjivanja pomoću lupe. Lupa vam omogućava da uvećate kartu bez menjanja razmere karte, što olakšava precizno podešavanje položaja natpisa i simbola. Nivo uvećanja izražava se u procentima. Ako *Lupa* ima nivo od 100%, trenutna karta nije uvećana, tj. iscrtava se u tačnoj razmeri u odnosu na rezoluciju monitora (DPI). Podrazumevana vrednost uvećanja može se definisati u *Podešavanja* ► *Opcije* ► *Iscrtavanje* ► *Ponašanje pri iscrtavanju*, što je veoma korisno za ekrane visoke rezolucije radi uvećanja malih simbola. Pored toga, podešavanje u *Podešavanja* ► *Opcije* ► *Platno i legenda* ► *DPI* određuje da li QGIS poštuje fizički DPI svakog monitora ili koristi ukupni logički DPI sistema.

Desno od alatke lupe možete definisati trenutnu rotaciju prikaza karte u smeru kazaljke na satu, u stepenima.

Na desnoj strani statusne trake, polje  *Iscrtaj* može se koristiti za privremeno obustavljanje iscrtavanja prikaza karte (pogledajte odeljak *Controlling map rendering*).

Desno od funkcije  *Iscrtaj* nalazi se dugme  *EPSG:kod* koje prikazuje trenutni CRS projekta. Klikom na njega otvara se dijalog *Svojstva projekta* i omogućava vam da reprojektujete prikaz karte ili podesite bilo koje drugo svojstvo projekta.

Савет

Računanje ispravne razmere platna karte

Kada pokrenete QGIS, podrazumevani CRS je `WGS 84 (EPSG 4326)`, a jedinice su stepeni. To znači da će QGIS svaku koordinatu u vašem sloju tumačiti kao zadatu u stepenima. Da biste dobili ispravne vrednosti razmere, ovo podešavanje možete ručno promeniti u kartici *Opšte* pod *Projekat* ► *Svojstva…* (npr. u metre) ili možete koristiti ikonu  EPSG:kod prikazanu iznad. U drugom slučaju, jedinice se postavljaju na ono što projekcija projekta zadaje (npr. `+units=us-ft`).

Imajte u vidu da se izbor CRS-a pri pokretanju može podesiti u *Podešavanja* ► *Opcije* ► *Rukovanje CRS-om*.

3.3.4 Poruke

Dugme  Poruke pored njega otvara *Panel poruka dnevnika* koji sadrži informacije o pozadinskim procesima (pokretanje QGIS-a, učitavanje dodataka, alatke za obradu…)

ГЛАВА 4

The Browser panel

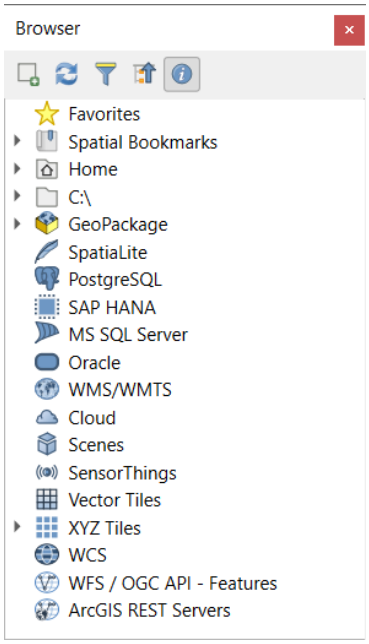
- *Resources that can be opened / run from the Browser*
- *Browser panel top-level entries*
  - *Favorites*
  - *Spatial Bookmarks*
  - *Project Home*
  - *Drives and file system*
  - *Data type*
  - *Database entries*
  - *Tiles and Web Services*
  - *Project and supported file types*

The QGIS Browser panel is a great tool for browsing, searching, inspecting, copying and loading QGIS resources. Only resources that QGIS knows how to handle are shown in the browser.

Using the Browser panel you can locate, inspect and add data, as described in *The Browser Panel*. In addition, the Browser panel supports drag and drop of many QGIS resources, such as project files, Python scripts, Processing scripts and Processing models.

Python scripts, Processing scripts and Processing models can also be opened for editing in an external editor and the graphical modeller.

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



Слика4.1: The Browser panel

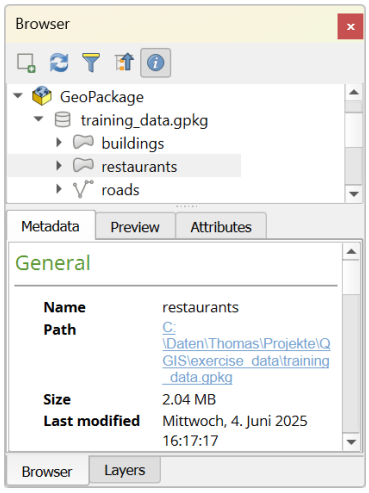
The browser panel (Сл.4.1) is organised as an expandable hierarchy with some fixed top-level entries that organise the resources handled by the browser. Node entries are expanded by clicking on to the left of the entry name. A branch is collapsed by clicking on . The Collapse All button collapses all top-level entries.

In *Settings* ► *Interface Customization* it is possible to disable resources. If you, for instance, would not like to show Python scripts in the browser, you can uncheck the *Browser* ► *py* entry, and if you want to get rid of your home folder in the browser, you can uncheck the *Browser* ► *special:Home* entry.

A filter (Filter Browser) can be used for searching based on entry names (both leaf entries and node entries in the hierarchy). Using the Options pull-down menu next to the filter text field, you can

- toggle *Case Sensitive* search
- set the *Filter pattern syntax* to one of
  - *Normal*
  - *Wildcard(s)*
  - *Regular Expressions*

The *Properties widget*, showing useful information about some entries / resources, can be enabled / disabled using the Enable/disable properties widget button. When enabled, it opens at the bottom of the browser panel, as shown in Сл.4.2.



Слика4.2: The properties widget

A second browser panel can be opened by activating the *Browser (2)* panel in *View* ► *Panels*. Having two browser panels can be useful when copying layers between resources that are locationed deep down in different branches of the browser hierarchy.

4.1 Resources that can be opened / run from the Browser

A lot can be accomplished in the Browser panel

- Add vector, raster and mesh layers to your map by double-clicking, dragging onto the map canvas or clicking the Add Selected Layers button (after selecting layers)
- Run Python scripts (including Processing algorithms) by double-clicking or dragging onto the map canvas
- Run models by double-clicking or dragging onto the map canvas
- *Extract Symbols…* from QGIS Project files using the context menu
- Open files with their default applications (*Open <file type> Externally…* in the context menu). Examples: HTML files, spreadsheets, images, PDFs, text files, …
- Copy entries
- Rename and delete (multiple) layers (context menu: *Manage* ►)
- Open a file explorer window and directly select the file *Show in Files*

Resource specific actions are listed for the different resource groups sorted under the top-level entries listed below.

4.2 Browser panel top-level entries

4.2.1 Favorites

Often used file system locations can be tagged as favorites. The ones you have tagged will appear here.

In addition to the operations described under *Home*, the context menu allows you to *Rename Favorite…* and *Remove Favorite*.

4.2.2 Spatial Bookmarks

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

Табела4.1: available actions for spatial bookmarks

Level	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 Project Home

Available once the project file has been saved, the *Project home* entry is a folder containing data and other contents (scripts, models, text, …) that may be used within the current project. Displayed in the *Browser* panel, it allows you to quickly access data and other files of the project.

It defaults to the project file folder but can be changed through the *Project* ► *Properties…* ► *General* ► *Project home* option, or by right-clicking on the *Project Home* item of the Browser panel and selecting *Set project home…*. Customizing that folder is especially useful in contexts where QGIS projects are not stored in the root folder of an organisational 'project', along with datasets.

4.2.4 Drives and file system

The next items of the *Browser* panel depend on the OS in use and concern the top level entries of its file system.

They are mainly:

- The *Home* folder, pointing to the current user home folder
- on Unix-based machines, the root `/` folder
- the connected drives, either local or network. Depending on the OS, they are directly listed (eg, `C:\`, `D:\`) or through the `/Volumes` entry.

From the contextual menu of each of these folders or drives, you can:

- refresh the contents
- create a *New* ► subitem that is a *Directory*, *GeoPackage* or ESRI *Shapefile* format dataset
- hide the directory (*Hide from Browser*)
- *Set color*: customize the folder icon color, aiding in rapid browser navigation of complex folder structures
- enable *Scanning*:
  - *Monitor for changes*: allows to manually control whether a particular directory should be monitored and automatically updated. This setting applies to the selected directory and all subdirectories. This means that you can manually opt-in to monitoring of network drives if you know there's no issue, or manually opt-out of monitoring of large directories which you don't want monitored for other reasons. By default, remote or network drives are not automatically monitored.
  - *Fast scan this directory*
- open the directory in your file manager (*Open Directory…*)
- open the directory in a terminal window (*Open in Terminal…*)
- inspect the *Properties…* or the parent *Directory Properties…*

4.2.5 Data type

Табела4.2: Available actions for plain layer files

Level and Context menu	Types of layers Raster (*Reference*)	Vector (*Reference*)	Point clouds (*Reference*)	Mesh (*Reference*)
Datoteka				
Delete File <dataset name>	☑	☑		
Show in Files	☑	☑	☑	☑
Manage	☑	☑	☑	☑
File Properties…	☑	☑	☑	☑
Open <file type> Externally…	☑			
Sloj				
Add Layer to Project	☑	☑	☑	☑
Export Layer	☑	☑		
Layer Properties…	☑	☑	☑	☑
Open with Data Source Manager…	☑	☑		
Execute SQL…		☑		

4.2.6 Database entries

Depending on your OS and installed drivers, you might have access to different database types to use in QGIS. Below are listed the different entries of contextual menu at each level of the dataset tree.

Табела4.3: Available actions for database entries

Level and Context menu	Type of database GeoPackage ([1])	SpatiaLite	PostgreSQL	SAP HANA	MS SQL Server	Oracle
Top menu						
Create a *New Connection…* to an existing database	☑	☑	☑	☑	☑	☑
Create Database…	☑	☑				
Create Database and Layer…	☑					
Save Connections… details to a file			☑	☑	☑	
Load Connections…			☑	☑	☑	
Connection / Database						
Refresh a connection			☑	☑	☑	☑
Edit Connection… settings			☑	☑	☑	☑
Duplicate Connection			☑	☑	☑	☑
Remove Connection…	☑	☑	☑	☑	☑	☑

наставак на следећој страни

Табела 4.3 – настављено са претходне стране

Import Vector Layer…	✓	✓	✓	✓	✓
New Field Domain ► *New Range Domain*	✓				
New Field Domain ► *New Coded Values Domain*	✓				
New Field Domain ► *New Glob Domain*	✓				
New Relationship…	✓				
Delete <database_name>	✓	✓			
Compact Database (VACUUM)	✓				
Create a *New Schema…*			✓	✓	✓
Create a *New Table…*	✓	✓	✓	✓	
Execute SQL… query	✓	✓	✓	✓	
Schema					
Refresh a schema			✓	✓	✓
Schema Operations ► *Rename Schema…*			✓	✓	✓
Schema Operations ► *Delete Schema…*			✓	✓	✓
Create a *New Table…*			✓	✓	
Execute SQL… query			✓	✓	
Table / Layer					
Table Operations ► *Rename Table…*			✓	✓	✓
Table Operations ► *Truncate Table…*			✓		✓
Execute SQL… query	✓	✓	✓		
Export Layer ► *To file…*	✓	✓	✓	✓	✓
Manage ► *Rename Layer <layer_name>…*	✓	✓			✓
Manage ► *Delete Layer <layer_name>…*	✓	✓	✓	✓	✓
Manage ► *Delete Selected Layers*	✓	✓	✓	✓	✓
Manage ► *Add Layer to Project*	✓	✓	✓	✓	✓
Manage ► *Add Selected Layers to Project*	✓	✓	✓	✓	✓
Manage ► *Move to Another Schema*			✓		✓
Open *Layer Properties…* dialog	✓	✓	✓	✓	✓
Open *File Properties…* dialog	✓				

наставак на следећој страни

Табела 4.3 – настављено са претходне стране

Open with Data Source Manager	☑	☑			
Fields					
Add New Field…	☑	☑	☑	☑	
Polje					
Set Field Domain	☑				
Rename Field ([2])	☑		☑		☑
Set Alias…	☑				
Set Comment…	☑		☑		
Delete Field…	☑	☑	☑	☑	

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 Manager Plugin*).

You can start the import in two ways:

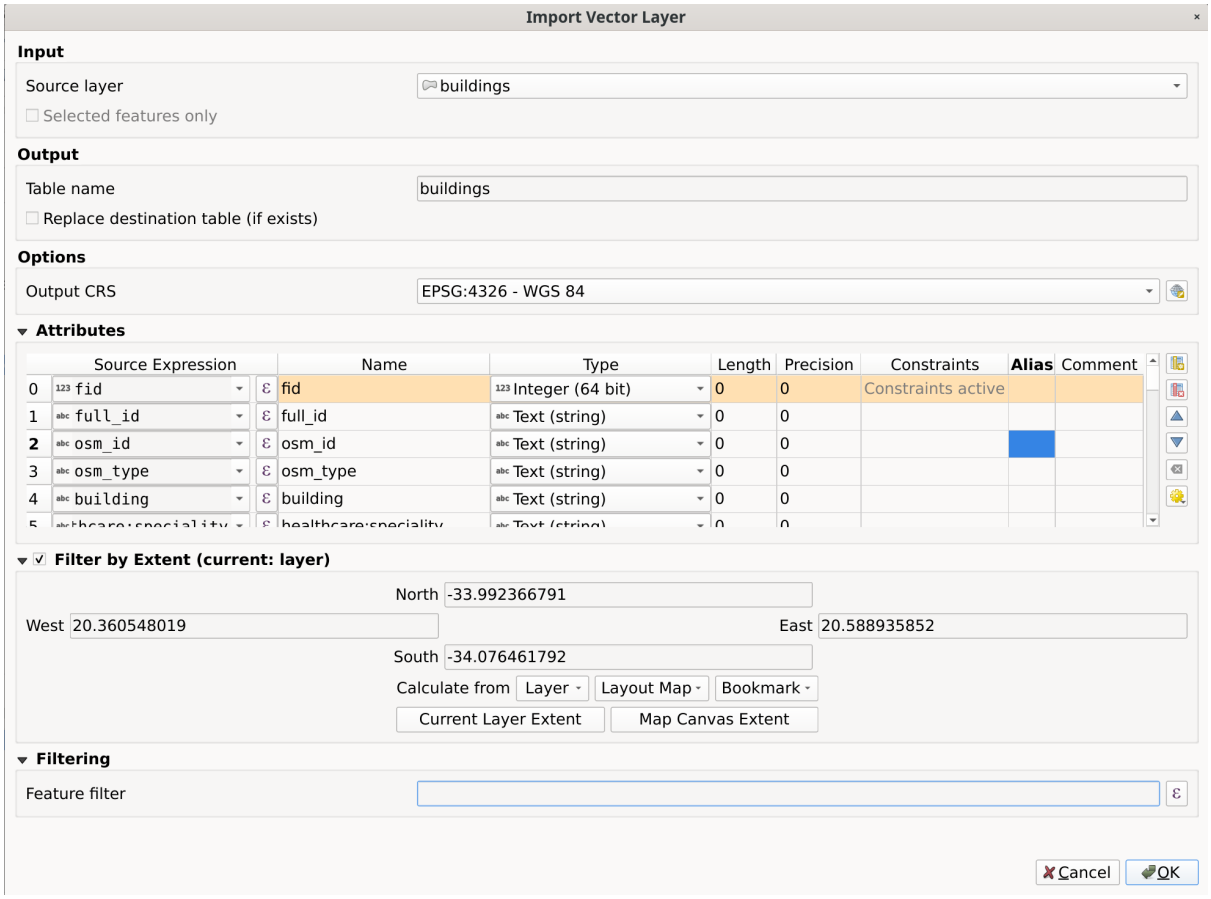
- Drag and drop a single vector layer from the Browser panel or from the Project into a database connection or schema item.
- Right-click on a database connection or schema item and choose *Import Vector Layer…* from the context menu.

Белешка

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.

[1] The different entries may also be available for some GDAL supported vector file formats such as ESRI File Geodatabase, FlatGeobuf, GeoParquet, NetCDF, … when compatible.

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



Слика4.3: 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 Tiles and Web Services

Табела4.4: Available actions for web services (1)

Level and Context menu	Type of services WMS / WMTS (*Reference*)	XYZ Tiles (*Reference*)	WCS (*Reference*)	WFS / OGC API - Features (*Reference*)	ArcGIS REST Servers (*Reference*)
Top menu					
Create a *New Connection…*	☑	☑	☑	☑	☑
Save Connections… details to a file	☑	☑	☑	☑	☑
Load Connections…	☑	☑	☑	☑	☑
Connection					
Refresh connection	☑	☑	☑	☑	☑
Edit Connection… settings	☑	☑	☑	☑	☑
Duplicate Connection	☑	☑	☑	☑	☑
Remove Connection…	☑	☑	☑	☑	☑
View Service Info in Web browser					☑
Table / Layer					
Export Layer ► To File…	☑	☑	☑	☑	☑
Add Layer to Project	☑	☑	☑	☑	☑
Add Filtered Layer to Project…					☑
Open *Layer properties…* dialog	☑	☑	☑	☑	☑
View Service Info in Web browser					☑

Табела4.5: Available actions for web services (2)

Level and Context menu	Type of services Vektorske pločice (*Reference*)	Scene (*Reference*)	Cloud (*Reference*)	SensorThings (*Reference*)	STAC (*Reference*)
Top menu					
Create a *New Connection…*		☑	☑	☑	☑
Create a *New Generic Connection…*	☑				
Create a *New ArcGIS Vector Tile Service Connection…*	☑				
Save Connections… details to a file	☑	☑	☑	☑	☑
Load Connections…	☑	☑	☑	☑	☑
Connection					
Edit Connection… settings	☑	☑	☑	☑	☑
Duplicate Connection	☑	☑	☑	☑	
Delete Connection…	☑	☑	☑	☑	☑
Refresh					☑
Details…					☑
Table / Layer					
Export Layer ► To File…				☑	
Add Layer to Project	☑	☑		☑	
Open *Layer properties…* dialog	☑	☑		☑	
Refresh					☑
Download assets…					☑
Details…					☑

4.2.8 Project and supported file types

Табела4.6: Available actions for various supported files :class: longtable

File type	Context menu
QGIS Project file (.qgz, .qgs)	*Open Project*
	Extract Symbols… (opens Style Manager to export symbols to XML, add to default style, or export as PNG/SVG)
	File Properties…
QGIS Layer Definition (.qlr)	*Export Layer ► To File…*
	Add Layer to Project
	Layer Properties…
Processing Model (.model3)	*Run Model…*
	Edit Model…
Print Composer Template (.qpt)	*New Layout from Template*
Python script (.py)	*Run Script…*
	Open in External Editor

ГЛАВА 5

QGIS Configuration

QGIS is highly configurable. Through the *Settings* menu, it provides different tools to:

- *Style Manager…*: create and manage *symbols, styles and color ramps*.
- *Custom Projections…*: create your own *coordinate reference systems*.
- *Keyboard Shortcuts…*: define your own set of *keyboard shortcuts*. Also, they can be overridden during each QGIS session by the *project properties* (accessible under *Project* menu).
- *Interface Customization…*: configure the *application interface*, hiding dialogs or tools you may not need.
- *Options…*: set global *options* to apply in different areas of the software. These preferences are saved in the active *User profile* settings and applied by default whenever you open a new project with this profile.

5.1 Options

Some basic options for QGIS can be selected using the *Options* dialog. Select the menu option *Settings* ► *Options*. You can modify the options according to your needs. Some of the changes may require a restart of QGIS before they will be effective.

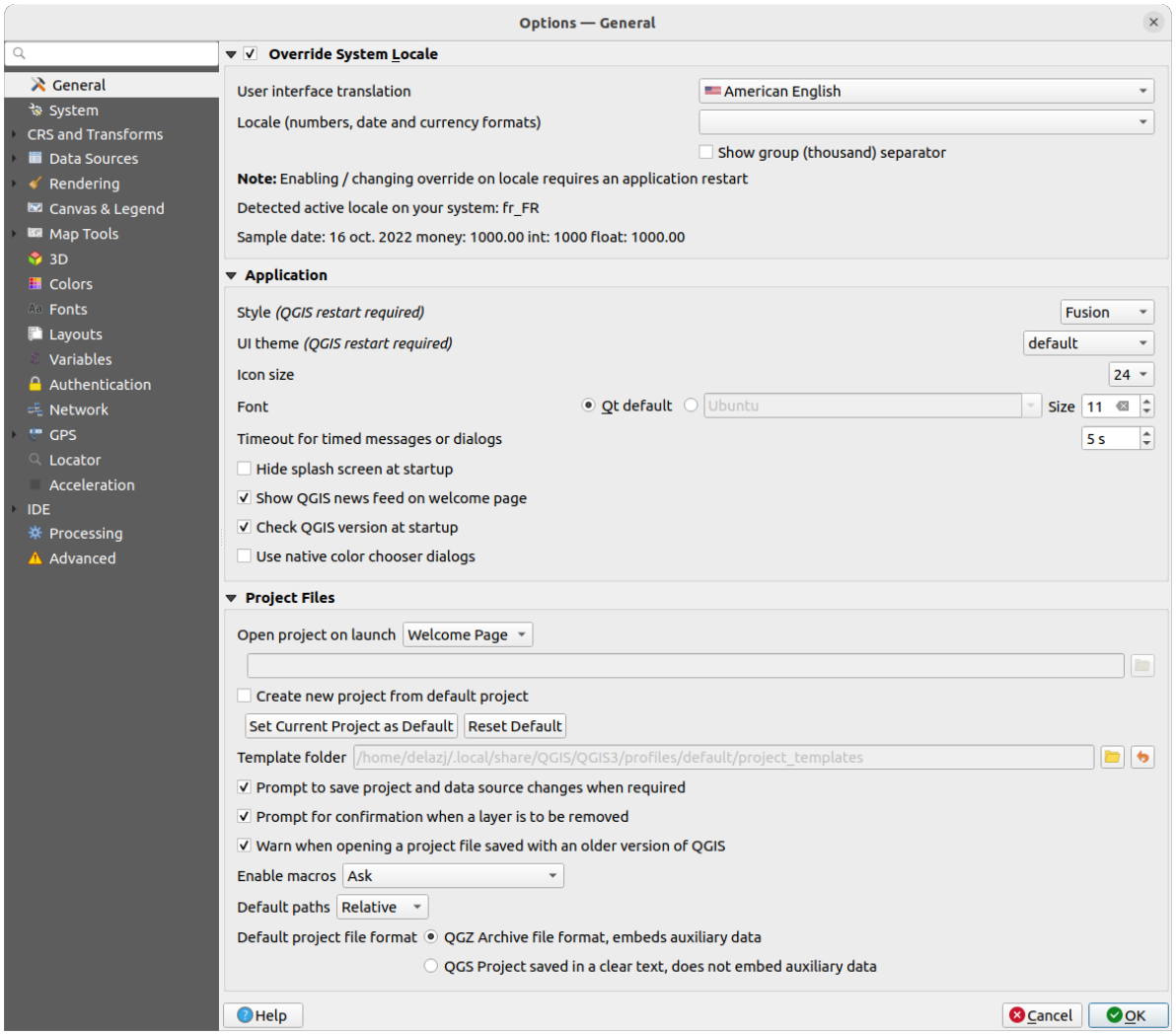
The tabs where you can customize your options are described below.

Белешка

Plugins can embed their settings within the Options dialog

While only Core settings are presented below, note that this list can be extended by *installed plugins* implementing their own options into the standard Options dialog. This avoids each plugin having their own config dialog with extra menu items just for them…

5.1.1 General Settings



Слика5.1: General Settings

Override System Locale

By default, QGIS relies on your Operating System configuration to set language and manipulate numerical values. Enabling this group allows you to customize the behavior.

- Select from *User interface translation* the language to apply to the GUI
- Select in *Locale (number, date and currency formats)* the system on which date and numeric values should be input and rendered
- *Show group (thousand) separator*

A summary of the selected settings and how they would be interpreted is displayed at the bottom of the frame.

Application

- Select the *Style (QGIS restart required)* ie, the widgets look and placement in dialogs. Possible values depend on your Operating System.
- Define the *UI theme (QGIS restart required)* . It can be 'default', 'Night Mapping', or 'Blend of Gray'
- Define the *Icon size*
- Define the *Font* and its *Size*. The font can be *Qt default* or a user-defined one
- Change the *Timeout for timed messages or dialogs*
- *Hide splash screen at startup*
- *Show QGIS news feed on welcome page*: displays a curated QGIS news feed on the welcome page, giving you a direct way to be aware of project news (user/developer meetings date and summary, community surveys, releases announcements, various tips…)
- *Check QGIS version at startup* to keep you informed if a newer version is released
- *Use native color chooser dialogs* (see *Color Selector*)

Project files

- *Open project on launch*
  - 'Welcome Page' (default): can display the „News" feed, the project template(s) and the most recent projects (with thumbnails) of the *user profile*. No project is opened by default.
  - 'New': opens a new project, based on the default template
  - 'Most recent': reopens the last saved project
  - and 'Specific': opens a particular project. Use the … button to define the project to use by default.
- *Create new project from default project*. You have the possibility to press on *Set current project as default* or on *Reset default*. You can browse through your files and define a directory where you find your user-defined project templates. This will be added to *Project* ► *New From Template*. If you first activate *Create new project from default project* and then save a project in the project templates folder.
- *Prompt to save project and data source changes when required* to avoid losing changes you made.
- *Prompt for confirmation when a layer is to be removed*
- *Warn when opening a project file saved with an older version of QGIS*. You can always open projects created with older version of QGIS but once the project is saved, trying to open with older release may fail because of features not available in that version.
- *Enable project's embedded Python code* . This option handles execution of macros that are written to perform an action on project events, as well as custom Python functions to be used as expressions. You can choose between 'Never', 'Ask', 'For this session only' and 'Always (not recommended)'.
- *Default paths*: defines whether paths to files and layers used in new projects are stored as 'Absolute' or 'Relative' to the project file. This setting can be overwritten at the project level.
- *Default project file format*
  - *QGZ Archive file format, embeds auxiliary data* (see *auxiliary data*)
  - *QGS Project saved in a clear text, does not embed auxiliary data*: the auxiliary data is stored in a separate `.qgd` file along with the project file.

5.1.2 System Settings

SVG paths

Add or Remove *Path(s) to search for Scalable Vector Graphic (SVG) symbols*. These SVG files are then available to symbolize or label the features or decorate your map composition.

Also read *Remote or embedded file selector* for different ways to refer to svg files in a QGIS path.

Plugin paths

Add or Remove *Path(s) to search for additional C++ plugin libraries*.

Documentation paths

Add or Remove *Documentation Path(s)* to use for QGIS help. By default, a link to the official online User Manual corresponding to the version being used is added (i.e., `https://docs.qgis.org/$qgis_short_version/$qgis_locale/docs/user_manual/`). You can however add other links and prioritize them from top to bottom: each time you click on a *Help* button in a dialog, the topmost link is checked and if no corresponding page is found, the next one is tried, and so on.

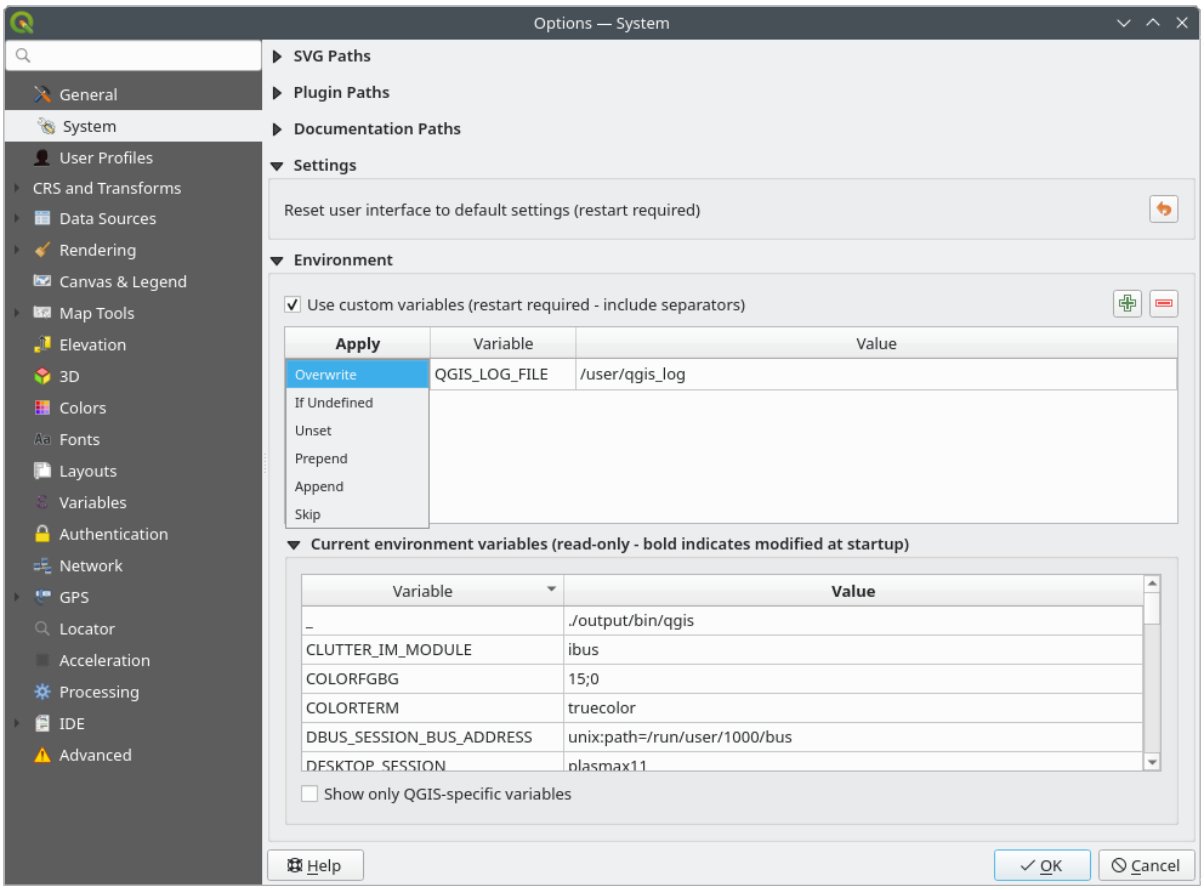
Белешка

Documentation is versioned and translated only for QGIS Long Term Releases (LTR), meaning that if you are running a regular release (eg, QGIS 3.0), the help button will by default open the next LTR manual page (ie. 3.4 LTR), which may contain description of features in newer releases (3.2 and 3.4). If no LTR documentation is available then the *testing* doc, with features from newer and development versions, is used.

Settings

It helps you *Reset user interface to default settings (restart required)* if you made any *customization*.

Environment



Слика5.2: System environment variables

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.

Check *Use custom variables (restart required - include separators)* and you can Add and Remove environment variables. For each new item, you can configure a *Variable* name, its *Value* and the *Apply* method to use, among which:

- *Overwrite*: replace any preexisting value of the variable
- *If undefined*: use this value for the variable if not already defined at a higher level (e.g., OS or application levels)
- *Unset*: remove the variable from the environment (the *Value* parameter is not used)
- *Prepend*: prepend the value to the preexisting value of the variable
- *Append*: append the value to the preexisting value of the variable
- *Skip*: the item is kept in the list for future reference but unused

Already defined environment variables are displayed in *Current environment variables*, and it's possible to filter them by activating *Show only QGIS-specific variables*.

5.1.3 User Profiles Settings

Белешка

For more information on how to manage user profiles, please read the dedicated section at *Working with User Profiles*.

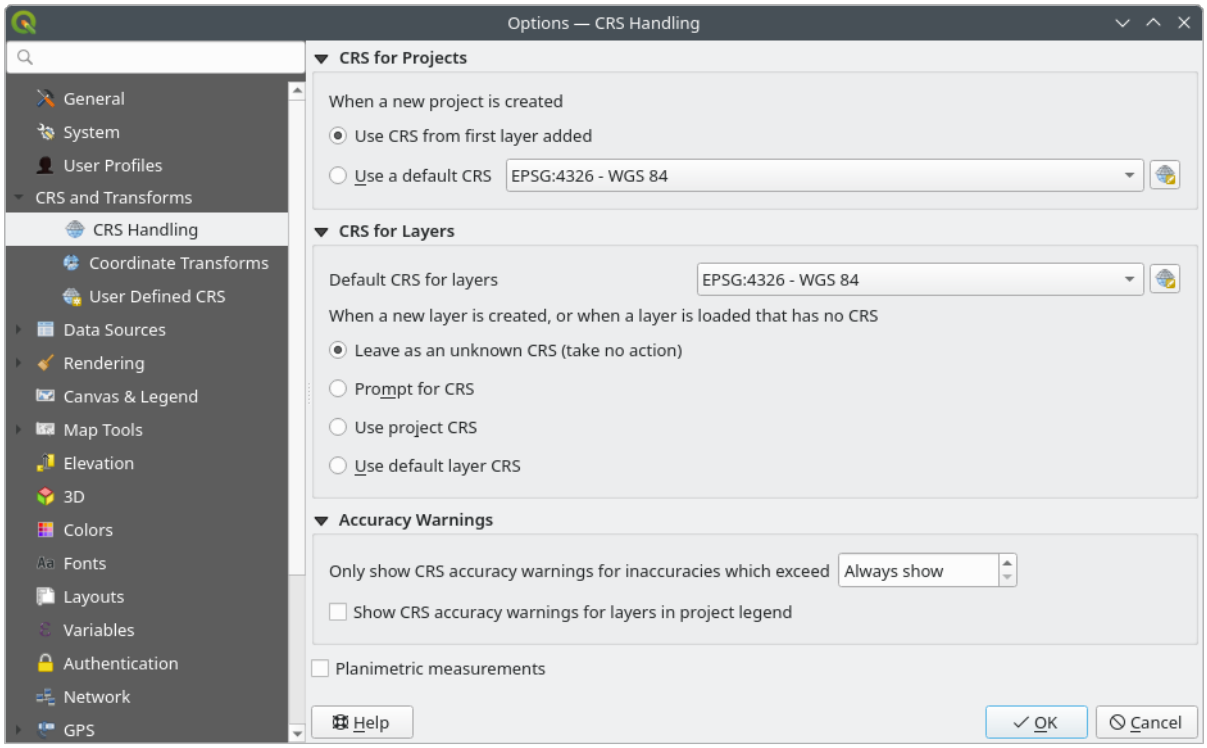
5.1.4 CRS and Transforms Settings

Белешка

For more information on how QGIS handles layer projection, please read the dedicated section at *Working with Projections*.

CRS Handling

In the *CRS Handling* tab you can configure which CRS will be used for a new project or layer.



Слика5.3: CRS Settings

CRS for Projects

There is an option to automatically set new project's CRS:

- *Use CRS from first layer added*: the CRS of the project will be set to the CRS of the first layer loaded into it
- *Use a default CRS*: a preselected CRS is applied by default to any new project and is left unchanged when adding layers to the project.

The choice will be saved for use in subsequent QGIS sessions. The Coordinate Reference System of the project can still be overridden from the *Project* ► *Properties…* ► *CRS* tab.

CRS for Layers

Default CRS for layers: select a default CRS to use when you create a layer

You can also define the action to take when a new layer is created, or when a layer without a CRS is loaded.

- *Leave as unknown CRS (take no action)*
- *Prompt for CRS*
- *Use project CRS*
- *Use default layer CRS*

Accuracy Warnings

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.

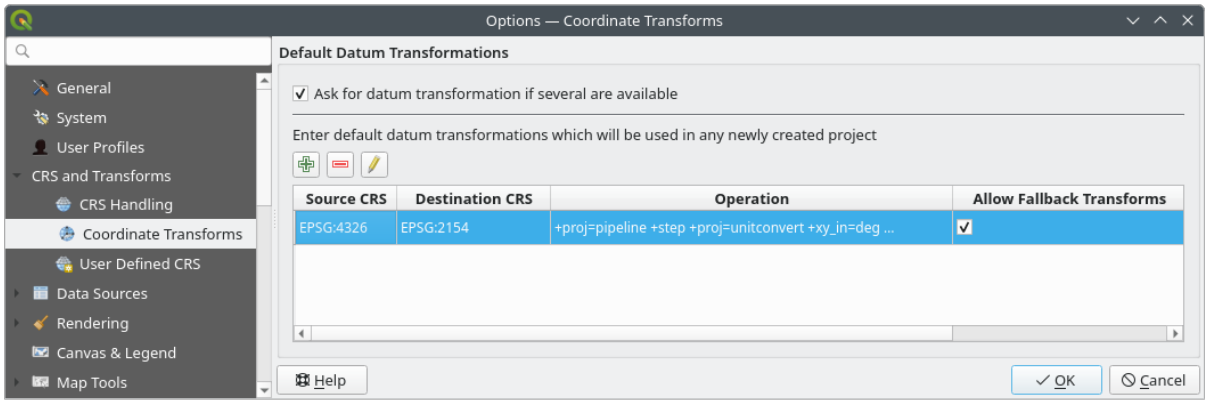
Show CRS accuracy warning for layers in project legend: If checked, any layer with a CRS with accuracy issues (i.e., a dynamic crs with no coordinate epoch available, or a CRS based on a datum ensemble with inherent inaccuracy exceeding the user-set limit) will have the warning icon in the *Layers* panel reflecting that it is a low-accuracy layer.

This is designed for use in engineering, BIM, asset management, and other fields where inaccuracies of meter/submeter level are potentially very dangerous or expensive!

Planimetric measurements: sets the default for the *planimetric measurements* property for newly created projects.

Coordinate Transforms

The *Coordinate Transforms* tab helps you set coordinate transformations and operations to apply when loading a layer to a project or reprojecting a layer.



Слика5.4: Transformations settings

Default Datum Transformations

Here you can control whether reprojecting layers to another CRS should be:

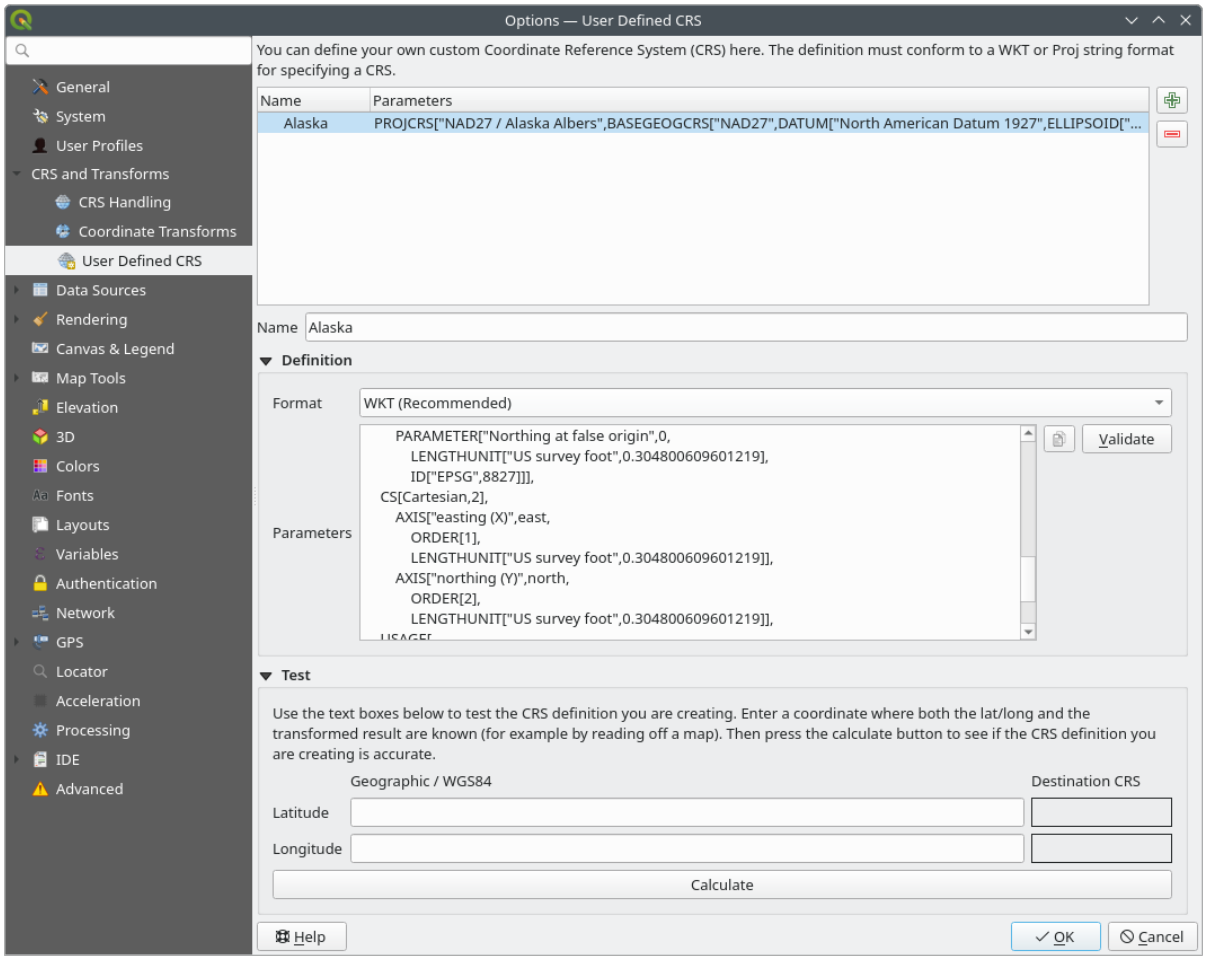
- automatically processed using QGIS default transformations settings;
- and/or more controlled by you with custom preferences such as:
  - *Ask for datum transformation if several are available*

- a predefined list of datum transformations to apply by default. See *Datum Transformations* for more details.

You can Add, Remove or Edit transformations, which will be used in any newly created project.

User Defined CRS

The *User Defined CRS* tab helps you to define a custom CRS which must conform to a WKT or Proj string format.



Слика5.5: User Defined CRS

Set a *Name* and use Add new CRS. If you want to delete an existing one you can use Remove CRS.

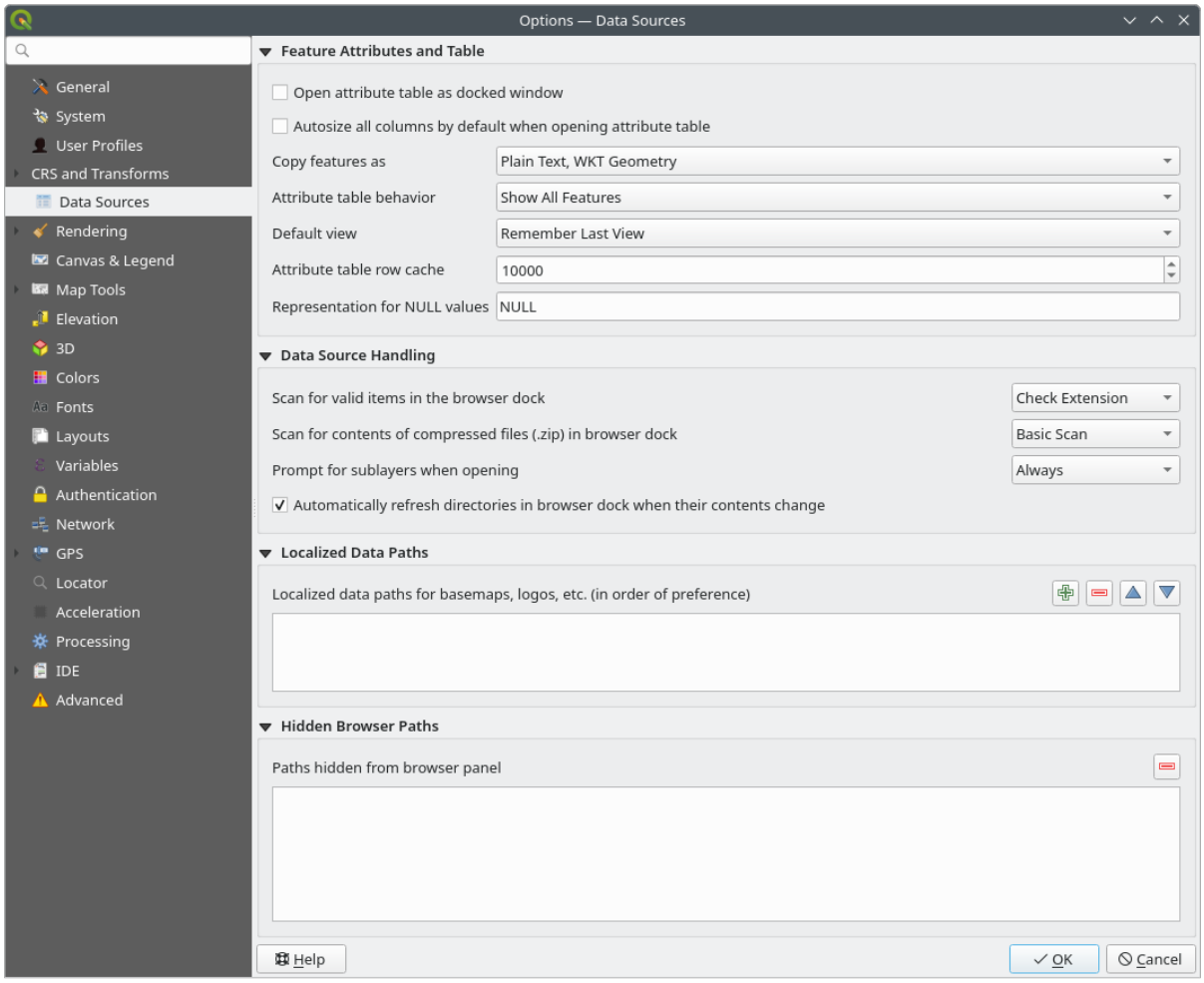
Definition

- ***Format***
  - WKT (Recommended)
  - Proj String (Legacy - Not Recommended)
- ***Parameters***
  - Copy parameters from an existing CRS.
  - *Validate* tests if your expression is correct.

Test

Here you can test your created CRS definition by Latitude and Longitude. Use a known coordinate to control if your definition is accurate.

5.1.5 Data Sources settings



Слика5.6: Data Sources settings

Feature attributes and table

- *Open attribute table as docked window*
- *Autosize all columns by default when opening attribute table*
- *Copy features as*: defines the format to use for data when pasting features in other applications.
  - *Plain text, no geometry*: attributes are copied as text but the geometry is skipped
  - *Plain text, WKT geometry*: attributes are copied as text and the geometry is returned in WKT
  - *Plain text, WKB geometry*: attributes are copied as text and the geometry is returned in WKB (hex format)
  - *GeoJSON*: attributes and geometry are formatted as GeoJSON data
- *Attribute table behavior* : set filter on the attribute table at the opening. There are three possibilities: 'Show all features', 'Show selected features' and 'Show features visible on map'.
- *Default view*: define the view mode of the attribute table at every opening. It can be 'Remember last view', 'Table view' or 'Form view'.

- *Attribute table row cache*  . This row cache makes it possible to save the last loaded N attribute rows so that working with the attribute table will be quicker. The cache will be deleted when closing the attribute table.
- *Representation for NULL values*. Here, you can define a value for data fields containing a NULL value.

Савет

Improve opening of big data attribute table

When working with layers with big amount of records, opening the attribute table may be slow as the dialog request all the rows in the layer. Setting the *Attribute table behavior* to **Show features visible on map** will make QGIS request only the features in the current map canvas when opening the table, allowing a quick data loading.

Note that data in this attribute table instance will be always tied to the canvas extent it was opened with, meaning that selecting **Show All Features** within such a table will not display new features. You can however update the set of displayed features by changing the canvas extent and selecting **Show Features Visible On Map** option in the attribute table.

Data source handling

- *Scan for valid items in the browser dock*  . You can choose between 'Check extension' and 'Check file contents'.
- *Scan for contents of compressed files (.zip) in browser dock*  defines how detailed is the widget information at the bottom of the Browser panel when querying such files. 'No', 'Basic scan' and 'Full scan' are possible options.
- *Prompt for sublayers when opening*. Some rasters support sublayers — they are called subdatasets in GDAL. An example is netCDF files — if there are many netCDF variables, GDAL sees every variable as a subdataset. The option allows you to control how to deal with sublayers when a file with sublayers is opened. You have the following choices:
  - 'Always': Always ask (if there are existing sublayers)
  - 'If needed': Ask if layer has no bands, but has sublayers
  - 'Never': Never prompt, will not load anything
  - 'Load all': Never prompt, but load all sublayers
-  *Automatically refresh directories in browser dock when their contents change*: Allows you to manually opt-out of monitoring directories in the *Browser* panel by default (eg, to avoid potential slow down due to network latency).

Localized data paths

It is possible to use localized paths for any kind of file based data source. They are a list of paths which are used to abstract the data source location. For instance, if `C:\my_maps` is listed in the localized paths, a layer having `C:\my_maps\my_country\ortho.tif` as data source will be saved in the project using `localized:my_country\ortho.tif`.

The paths are listed by order of preference, in other words QGIS will first look for the file in the first path, then in the second one, etc.

Hidden browser paths

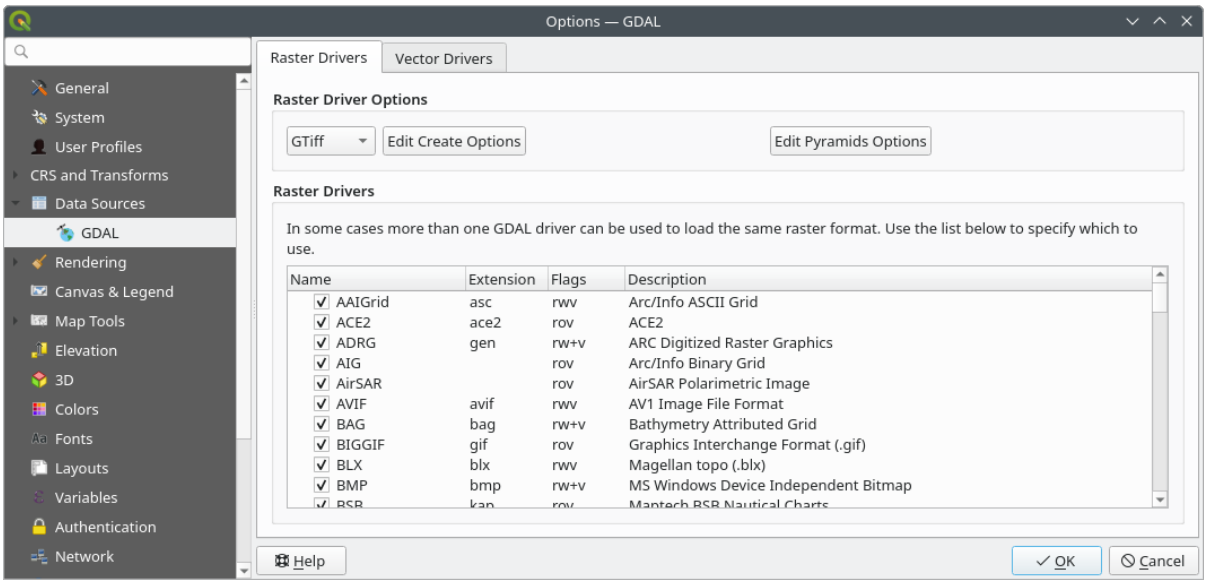
This widget lists all the folders you chose to hide from the *Browser panel*. Removing a folder from the list will make it available in the *Browser* panel.

GDAL Settings

GDAL is a data exchange library for geospatial data that supports a large number of vector and raster formats. It provides drivers to read and (often) write data in these formats. The *GDAL* tab exposes the drivers for raster and vector formats with their capabilities.

GDAL raster and vector drivers

The *Raster Drivers* and *Vector Drivers* tabs allow you to define which GDAL driver is enabled to read and/or write files, as in some cases more than one GDAL driver is available.



Слика5.7: GDAL Settings - Raster drivers

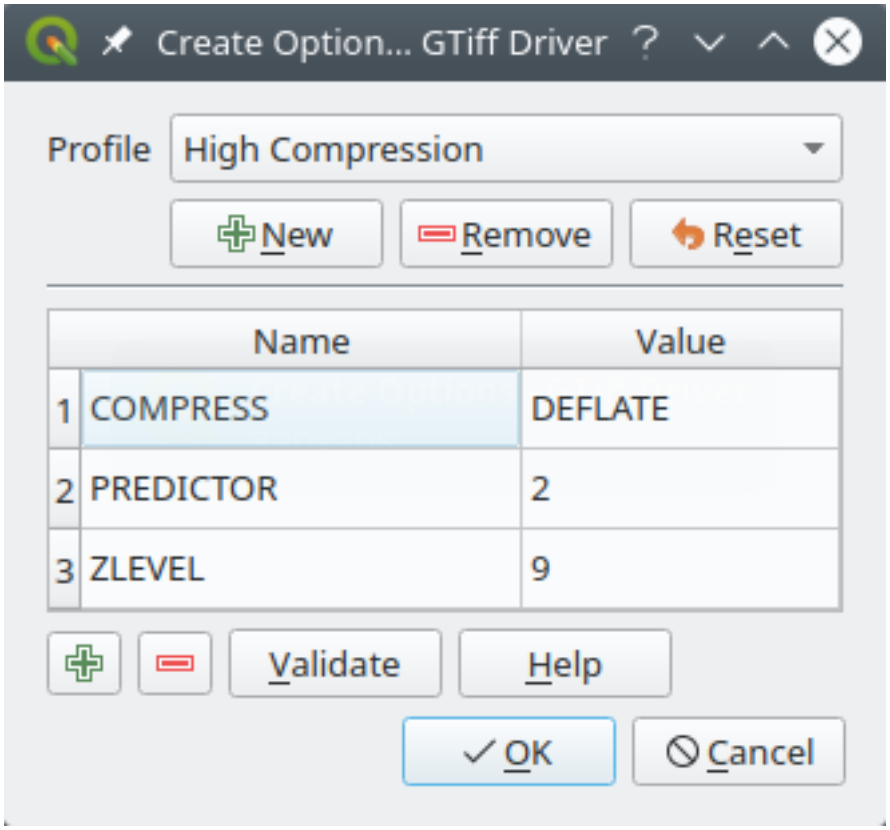
Савет

Double-click a raster driver that allows read and write access (`rw+(v)`) opens the *Edit Create options* dialog for customization.

Raster driver options

This frame provides ways to customize the behavior of raster drivers that support read and write access:

- *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.



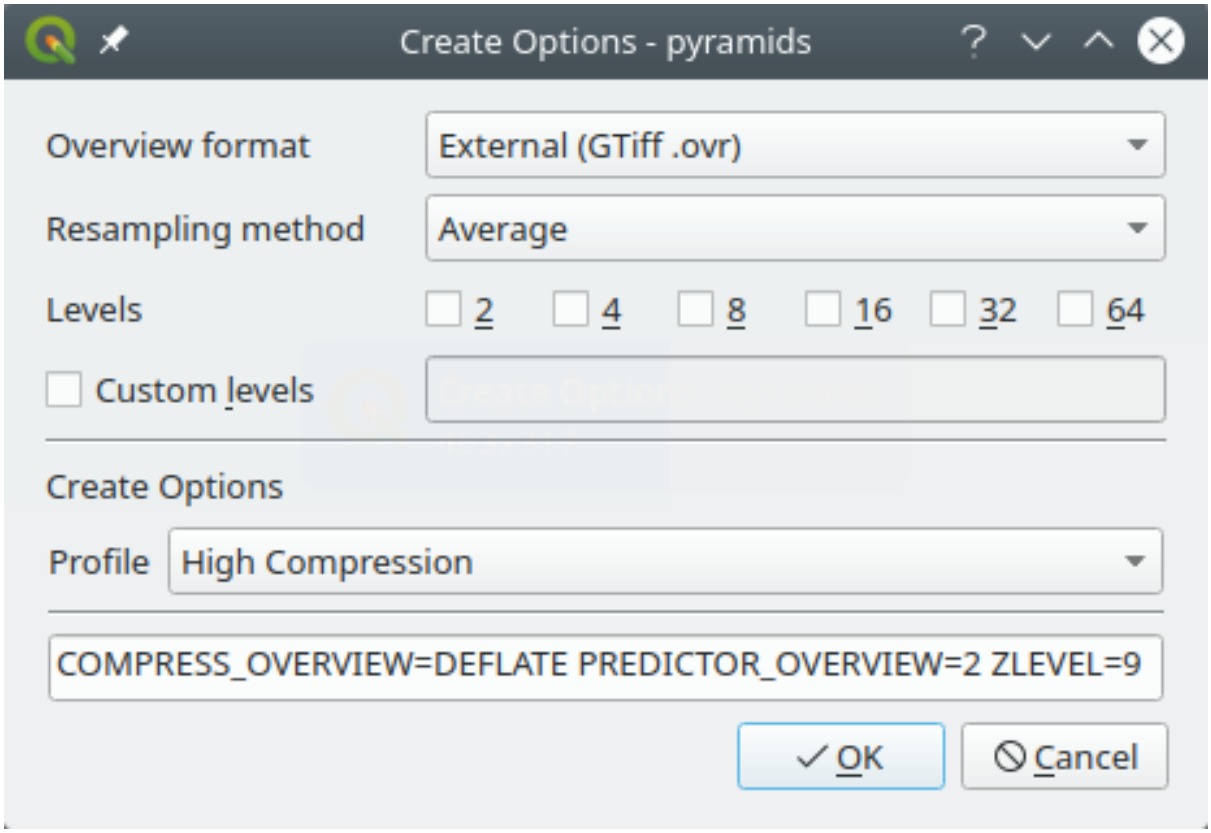
Слика5.8: Sample of create options profile (for GeoTiff)

The upper part of the dialog lists the current profile(s) and allows you to add new ones or remove any of them. You can also reset the profile to its default parameters if you have changed them. Some drivers (eg, GeoTiff) have some sample of profiles you can work with.

At the bottom of the dialog:

- The button lets you add rows to fill with the parameter name and value
- The button deletes the selected parameter
- Click the *Validate* button to check that the creation options entered for the given format are valid
- Use the *Help* button to find the parameters to use, or refer to the GDAL raster drivers documentation.

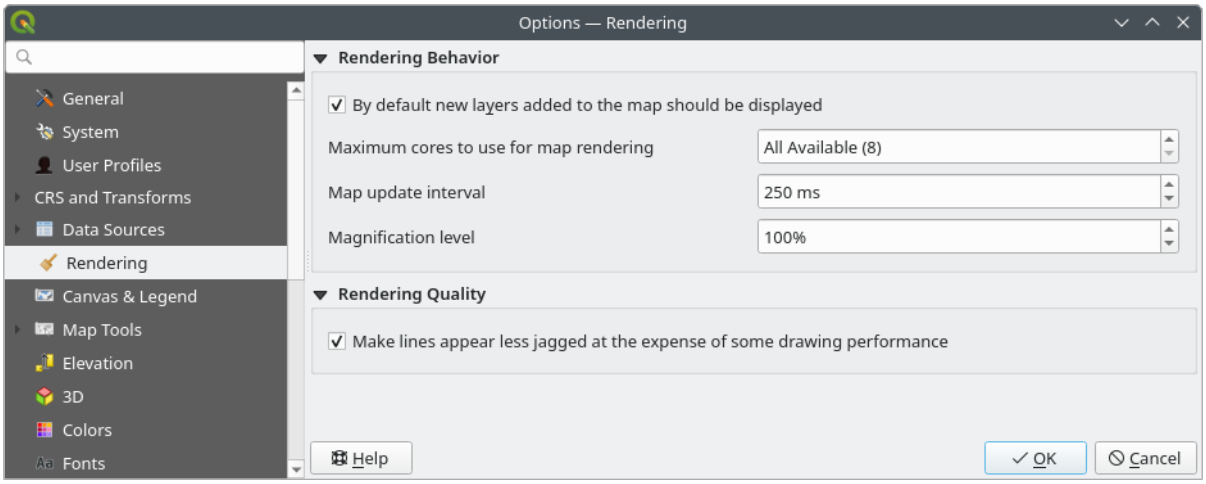
- *Edit Pyramids Options*



Слика5.9: Sample of pyramids profile

5.1.6 Rendering Settings

The *Rendering* tab provides settings for controlling layers rendering in the map canvas.



Слика5.10: Rendering settings

Rendering Behavior

- *By default new layers added to the map should be displayed*: unchecking this option can be handy when loading multiple layers to avoid each new layer being rendered in the canvas and slow down the process

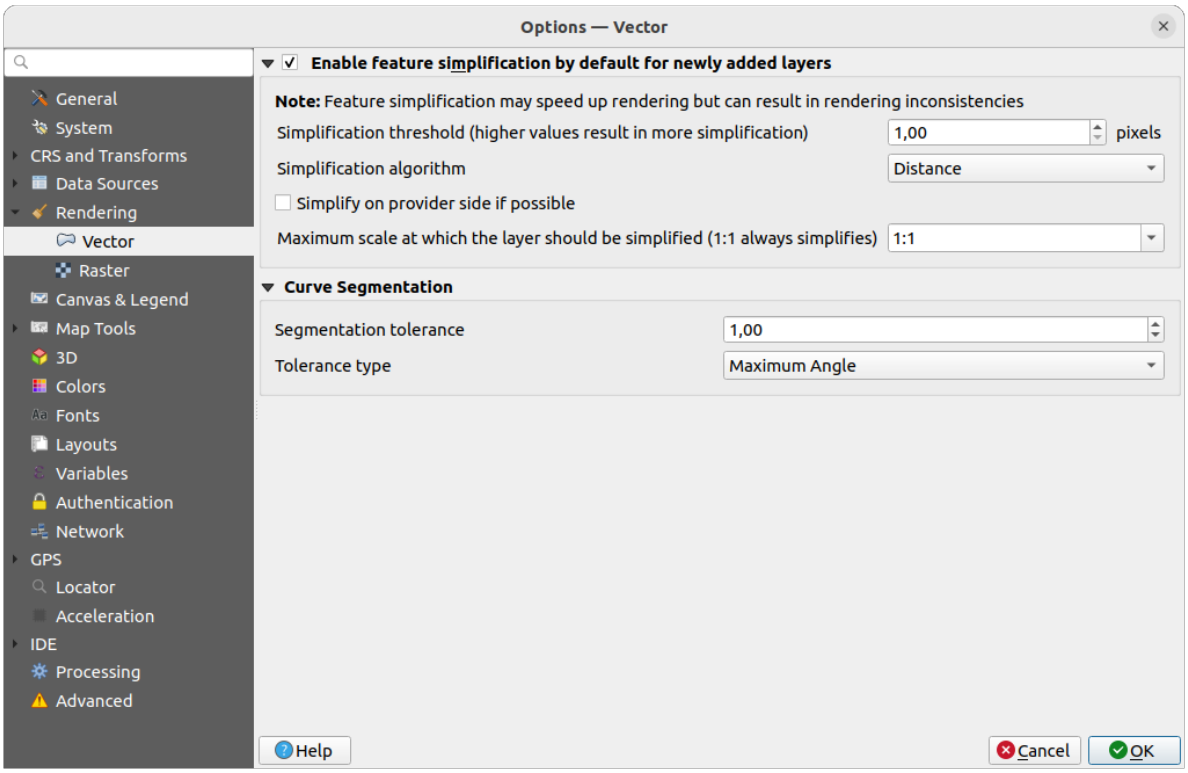
- Set the *Maximum cores to use for map rendering*
- The map canvas renders in the background onto a separate image and at each *Map update interval* (defaults to 250 ms), the content from this (off-screen) image will be taken to update the visible screen representation. However, if rendering finishes faster than this duration, it will be shown instantaneously.
- *Magnification level* (see the *magnifier*)

Rendering Quality

- *Make lines appear less jagged at the expense of some drawing performance*

Vector rendering settings

The *Vector* tab contains specific settings for rendering vector layers.



Слика5.11: Vector rendering settings

- *Enable Feature Simplification by Default for Newly Added Layers*: you simplify features' geometry (fewer nodes) and as a result, they display more quickly. Be aware that this can cause rendering inconsistencies. Available settings are:
  - *Simplification threshold (higher values result in more simplification)*
  - *Simplification algorithm*: This option performs a local „on-the-fly" simplification on feature's and speeds up geometry rendering. It doesn't change the geometry fetched from the data providers. This is important when you have expressions that use the feature geometry (e.g., calculation of area) - it ensures that these calculations are done on the original geometry, not on the simplified one. For this purpose, QGIS provides three algorithms: 'Distance' (default), 'SnapToGrid' and '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
  - *Maximum scale at which the layer should be simplified (1:1 always simplifies)*

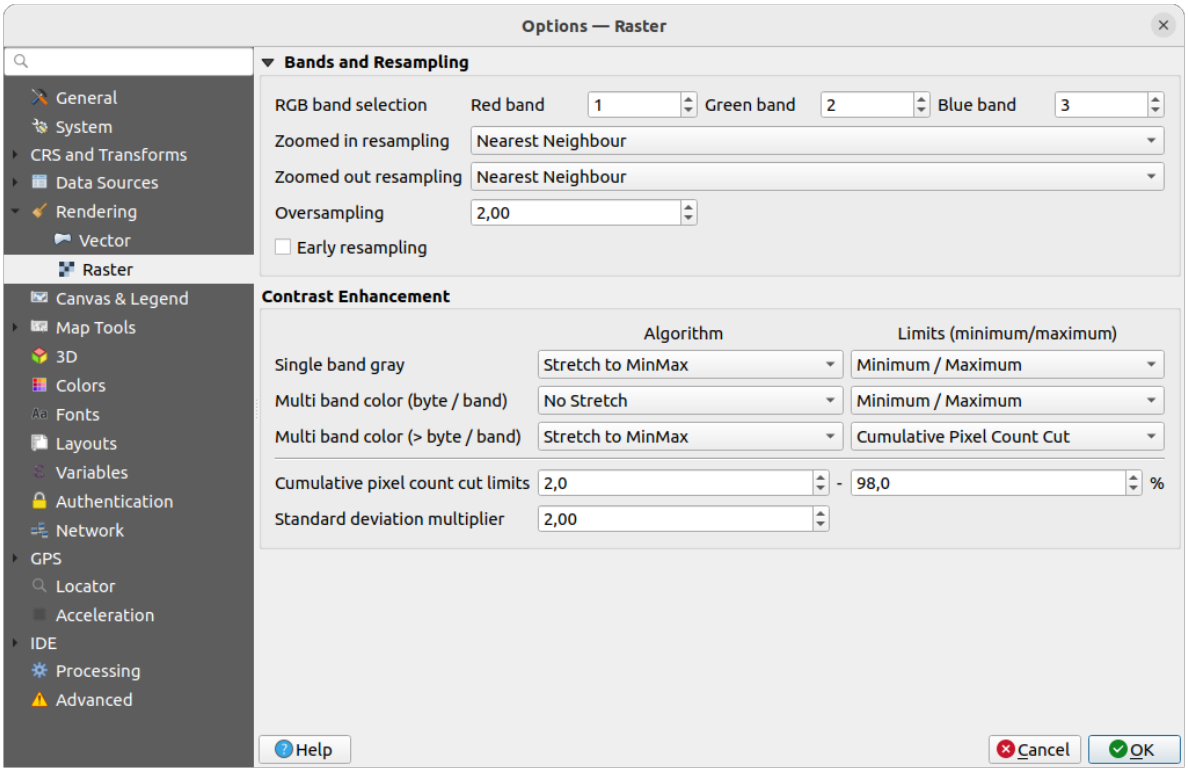
Белешка

Besides the global setting, feature simplification can be set for any specific layer from its *Layer properties* ► *Rendering* menu.

- *Curve Segmentation*
  - *Segmentation tolerance*: this setting controls the way circular arcs are rendered. **The smaller** maximum angle (between the two consecutive vertices and the curve center, in degrees) or maximum difference (distance between the segment of the two vertices and the curve line, in map units), the **more straight line** segments will be used during rendering.
  - *Tolerance type*: it can be *Maximum angle* or *Maximum difference* between approximation and curve.

Raster rendering settings

The *Raster* tab contains specific settings for rendering raster layers.



Слика5.12: Raster rendering settings

Under *Bands and Resampling*:

- With *RGB band selection*, you can define the number for the Red, Green and Blue band.
- The *Zoomed in resampling* and the *Zoomed out resampling* methods can be defined. For *Zoomed in resampling* you can choose between three resampling methods: 'Nearest neighbour', 'Bilinear (2x2 kernel)' and 'Cubic (4x4 kernel)'. For *Zoomed out resampling* you can choose between 'Nearest Neighbour' and 'Average'. You can also set the *Oversampling* value (between 0.0 and 99.99 - a large value means more work for QGIS - the default value is 2.0).
- *Early resampling*: allows to calculate the raster rendering at the provider level where the resolution of the source is known, and ensures a better zoom in rendering with QGIS custom styling. Really convenient

for tile rasters loaded using an *interpretation method*. The option can also be set at the layer level (*Symbology* properties)

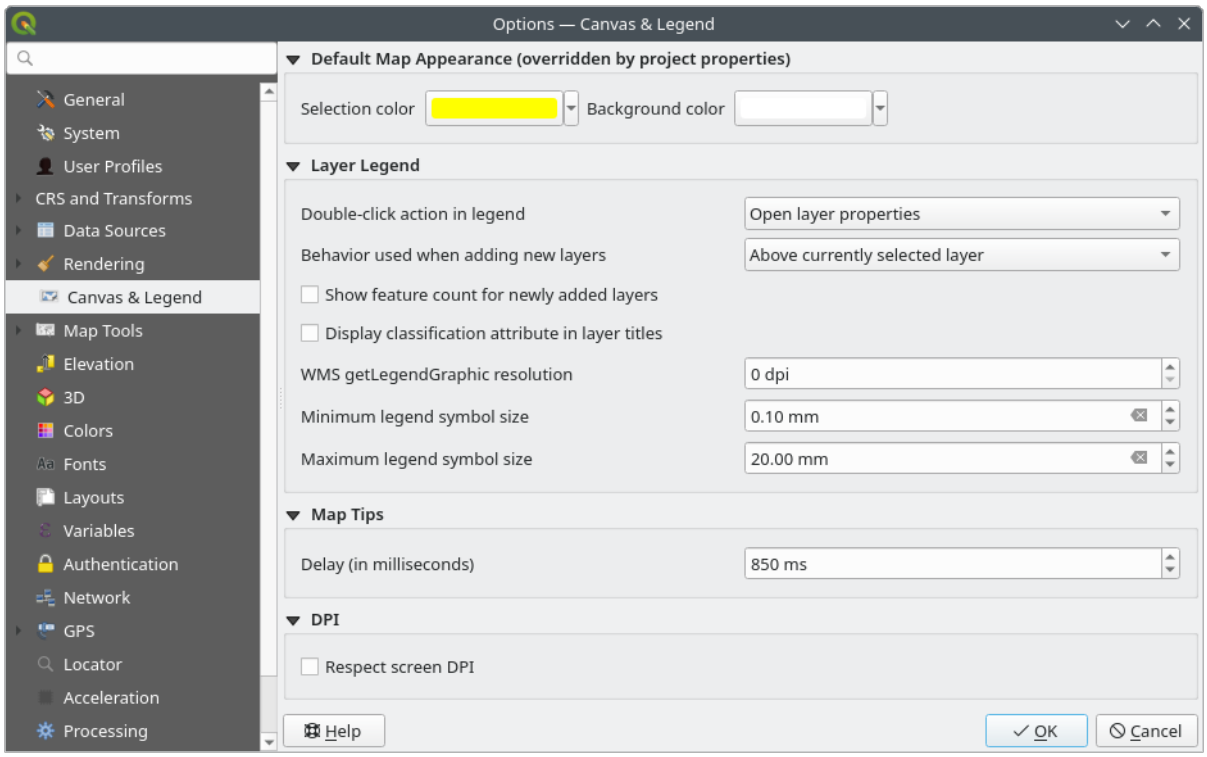
Contrast Enhancement options can be applied to *Single band gray*, *Multi band color (byte/band)* or *Multi band color (>byte/band)*. For each, you can set:

- the *Algorithm* to use, whose values can be 'No stretch', 'Stretch to MinMax', 'Stretch and Clip to MinMax' or 'Clip to MinMax'
- the *Limits (minimum/maximum)* to apply, with values such as 'Cumulative pixel count cut', 'Minimum/Maximum', 'Mean +/- standard deviation'.

The *Contrast Enhancement* options also include:

- *Cumulative pixel count cut limits*
- *Standard deviation multiplier*

5.1.7 Canvas and Legend Settings



Слика5.13: Canvas & Legend settings

These properties let you set:

- the **Default map appearance (overridden by project properties)**: the *Selection color* and *Background color*.
- **Layer legend** interaction:
  - *Double click action in legend*: whether a double-click on a layer should either *Open layer properties* (default), *Open attribute table* or *Open layer styling dock*.
  - *Behavior used when adding new layers*: determines where layers are placed in the *Layers* panel when loaded into QGIS. It can be:
    - *Above currently selected layer*
    - *Always on top of the layer tree*

        * *Optimal index within current layer tree group*: Unlike the other options that sort the new layers among them and place them as a stack at the desired location, this option extents the sorting logic to the whole layer tree (or the active group) and inserts new layers in an „optimal" fashion by insuring that point layers sit on top of point layers, followed by line layers on top of lines, followed by polygon layers, etc.

    - *Show feature count for newly added layers*: displays in the *Layers* panel the number of features next to the layer name. Feature count of classes, if any, is as well displayed. You can right-click on a layer to turn on/off its feature count.
    - *Display classification attribute names* in the Layers panel, e.g., when applying a categorized or rule--based renderer (see *Symbology Properties* for more information).
    - the *WMS getLegendGraphic Resolution*
    - *Minimum* and *Maximum legend symbol size* to control symbol size display in the *Layers* panel

- the *Delay* in milliseconds of layers *map tips* display

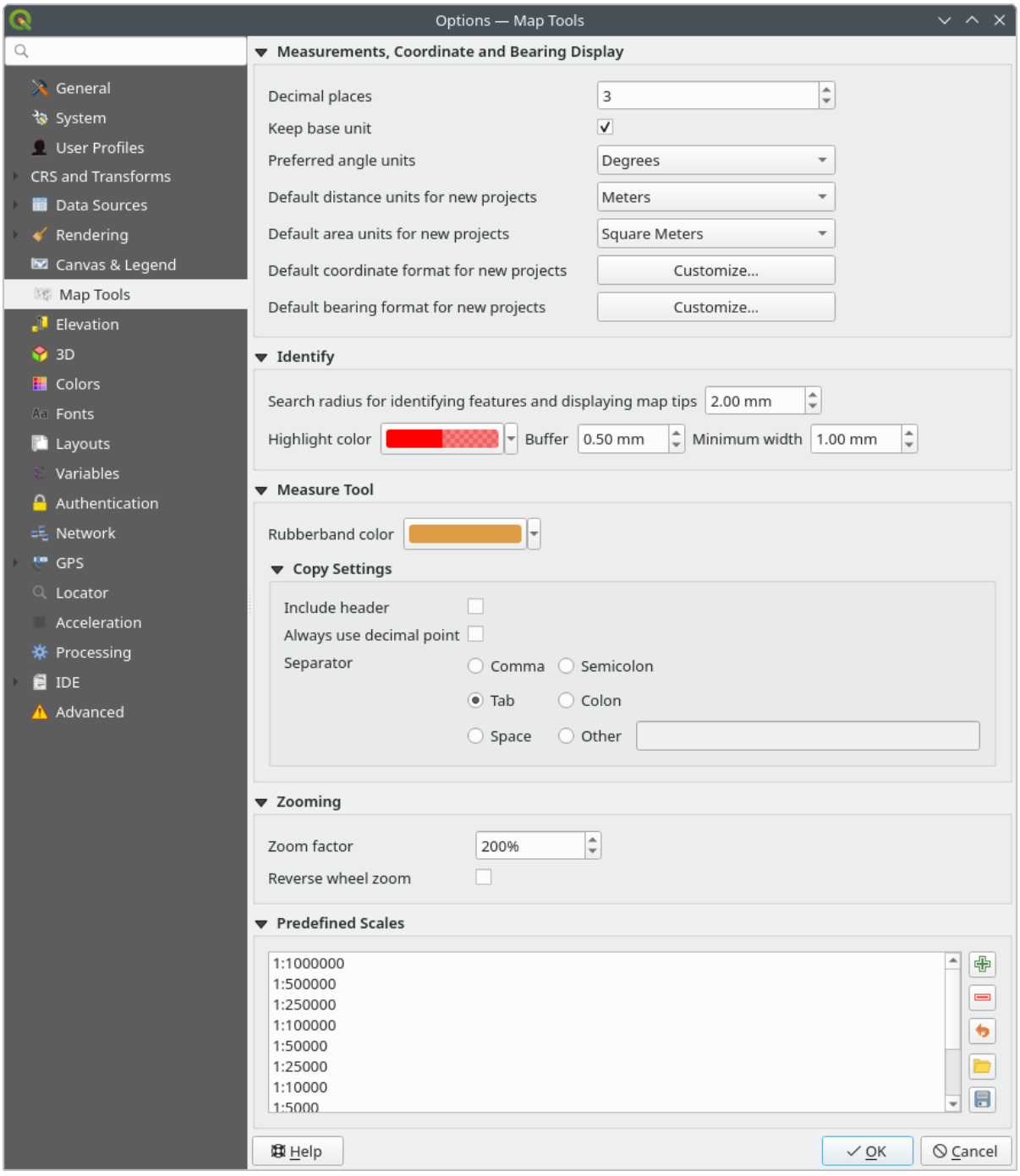
- Whether QGIS should *Respect screen DPI*: If enabled, QGIS will attempt to display the canvas with physically accurate scale on screen, depending on the monitor's physical DPI. Symbology with specified display size will also be rendered accurately, e.g., a 10mm symbol will show as 10mm on screen. However, label font sizes on canvas may differ from those in QGIS' UI or other applications. If this setting is turned off, QGIS will use the operating system's logical DPI, which will be consistent with other applications on the system. However, canvas scale and symbology size may be physically inaccurate on screen. In particular, on high-dpi screens, symbology is likely to appear too small.

  For best experience, it is recommended to enable *Respect screen DPI*, especially when using multiple or different monitors and preparing visually high-quality maps. Disabling *Respect screen DPI* will generate output that may be more suitable for mapping intended for on-screen use only, especially where font sizes should match other applications.

Белешка

Rendering in layouts is not affected by the *Respect screen DPI* setting; it always respects the specified DPI for the target output device. Also note that this setting uses the physical screen DPI as reported by the operating system, which may not be accurate for all displays.

5.1.8 Map tools Settings



Слика5.14: Map tools settings

Measurements, Coordinate and Bearing Display

This section provides ways to configure default units parameters:

- Define *Decimal places*
- *Keep base unit* to not automatically convert large numbers (e.g., meters to kilometers)
- *Preferred angle units*: options are 'Degrees', 'Radians', 'Gon/gradians', 'Minutes of arc', 'Seconds of arc', 'Turns/revolutions', milliradians (SI definition) or mil (NATO/military 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)
- *Default coordinate format for new projects*, as displayed in the *Coordinates* box on QGIS status bar and in the *Derived* section of the  Identify features tool's results
- *Default bearing format for new projects*, as displayed in the status bar for the map canvas panning direction and by the  Measure bearing tool.

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*.

- *Search radius for identifying features and displaying map tips* is a tolerance distance within which the identify tool will depict results as long as you click within this tolerance.
- *Highlight color* allows you to choose with which color features being identified should be highlighted.
- *Buffer* determines a buffer distance to be rendered from the outline of the identify highlight.
- *Minimum width* determines how thick should the outline of a highlighted object be.

Measure tool

- Define *Rubberband color* for measure tools

Measure Tool Copy Settings

These settings allow you to control the behavior of the *Copy* button in the  Measure Line window. You can choose to  *Include header* to keep column names, and you can also select *Separator* of your choice. You can also choose  *Always use decimal point* to keep your coordinates copied to the clipboard with a dot as a decimal separator, even if your language settings in QGIS options are set up to use a comma. In this case, if you don't have the *Always use decimal point* option checked, you will be unable to select comma as a field separator.

Panning and zooming

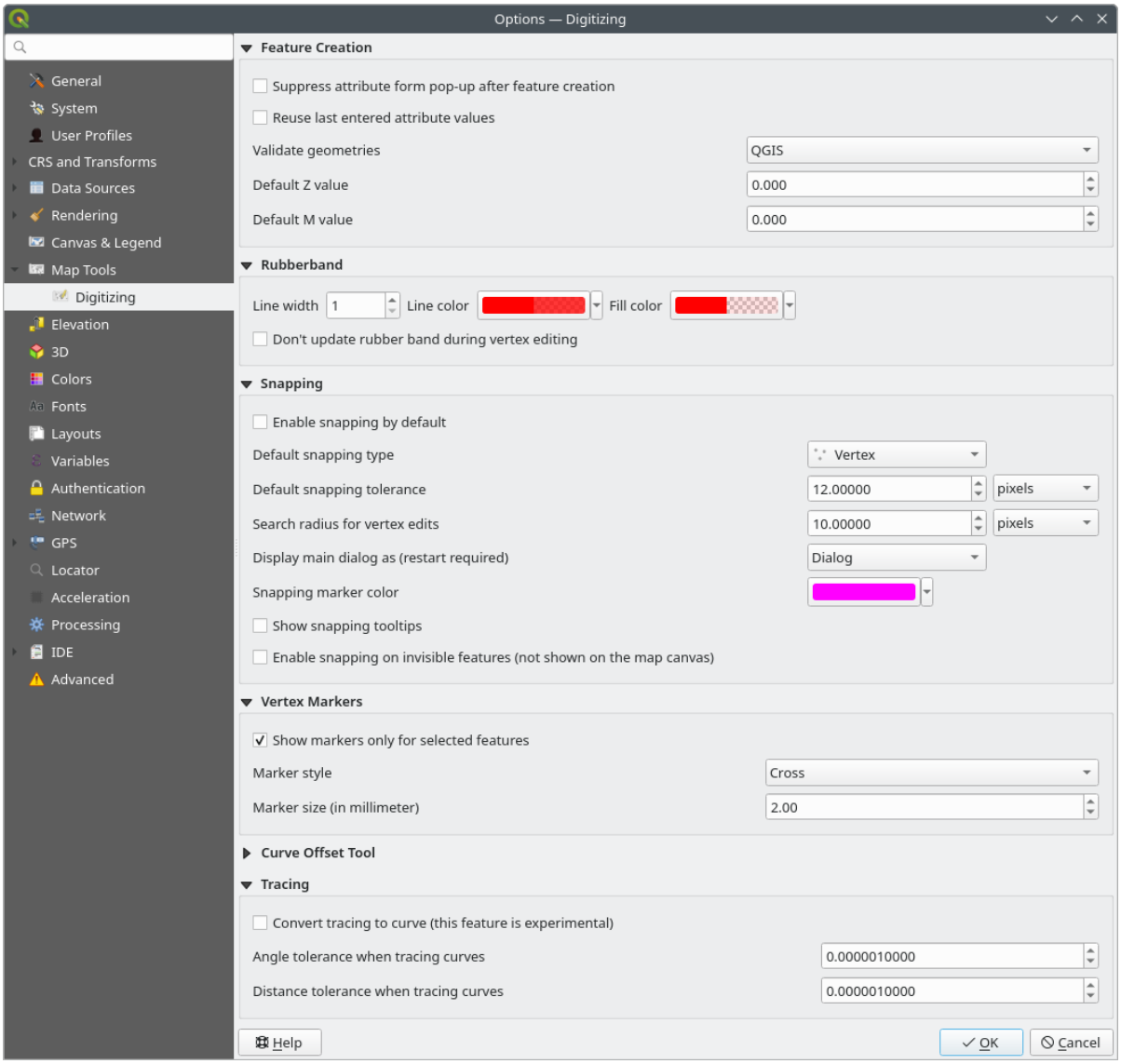
- Define a *Zoom factor* for zoom tools or wheel mouse
-  *Reverse wheel zoom* allows you to adjust mouse wheel scrolling direction.

Predefined scales

Here, you find a list of predefined scales to display by default in the scale-related drop-down widgets, such as the status bar *Scale*, the visibility scales selector or secondary 2D map view settings,… With the  and  buttons you can add or remove your personal scales. You can also import or export scales from/to a `.XML` file. Note that you still have the possibility to remove your changes and reset to the predefined list.

From the project properties dialog, you can also set your own list of scales, overriding this global one in the widgets.

Digitizing settings



Слика5.15: Digitizing settings

This tab helps you configure general settings when *editing vector layer* (attributes and geometry).

Feature creation

- *Suppress attribute form pop-up after feature creation*: this choice can be overridden in each layer properties dialog.
- *Reuse last entered attribute values*: remember the last used value of every attribute and use it as default for the next feature being digitized. Works per layer. This behavior can also be controlled on a per-field basis (see *Configure the field behavior*).
- *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.

  Note that depending on the selection, reports of geometry errors may differ (see *Tipovi poruka o grešci i njihova značenja*)

- *Default Z value* to use when creating new 3D features.

Rubberband

- Define Rubberband *Line width*, *Line color* and *Fill color*.
- *Don't update rubberband during vertex editing*.

Snapping

- *Enable snapping by default* activates snapping by default on all new projects
- Define *Default snap mode* ('Vertex', 'Segment', 'Centroid', 'Middle of segments', 'Line endpoints', 'Area')
- Define *Default snapping tolerance* in map units or pixels
- Define the *Search radius for vertex edits* in map units or pixels
- *Display main dialog as (restart required)*: set whether the Advanced Snapping dialog should be shown as 'Dialog' or 'Dock'.
- *Snapping marker color*
- *Show snapping tooltips* such as name of the layer whose feature you are about to snap. Helpful when multiple features overlap.
- *Enable snapping on invisible features (not shown on the map canvas)*

Vertex markers

- *Show markers only for selected features*
- Define vertex *Marker style* ('Cross' (default), 'Semi transparent circle' or 'None')
- Define vertex *Marker size (in millimeter)*

Curve offset tool

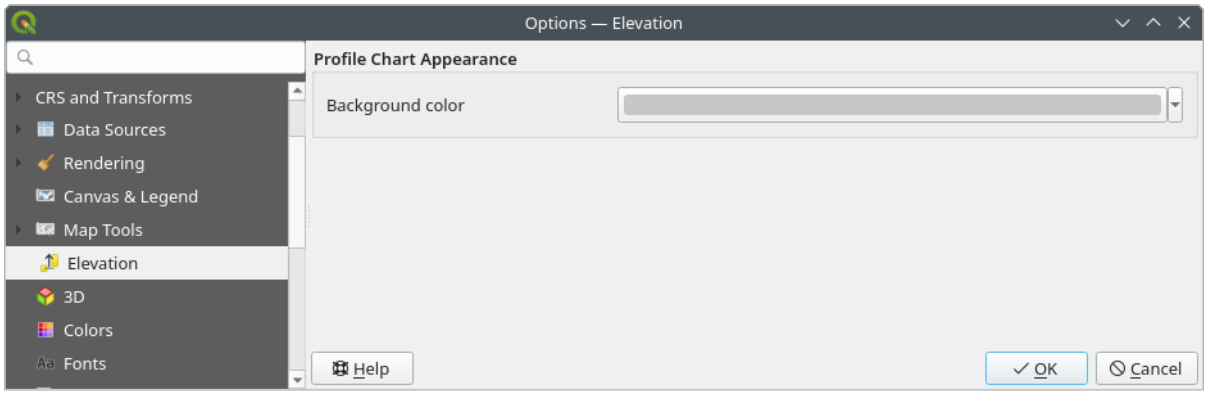
The next options refer to the Offset Curve tool in *Advanced digitizing*. Through the various settings, it is possible to influence the shape of the line offset.

- *Join style*: specifies whether 'Round', 'Miter' or 'Bevel' joins should be used when offsetting corners
- *Quadrant segments*: controls the number of line segments to use to approximate a quarter circle when creating rounded offsets
- *Miter limit*: 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 style)
- *End cap style*: controls how line endings are handled; they can be 'Round', 'Flat' or 'Square'.

Tracing

By activating the *Convert tracing to curve* you can create curve segments while digitizing. Keep in mind that your data provider must support this feature.

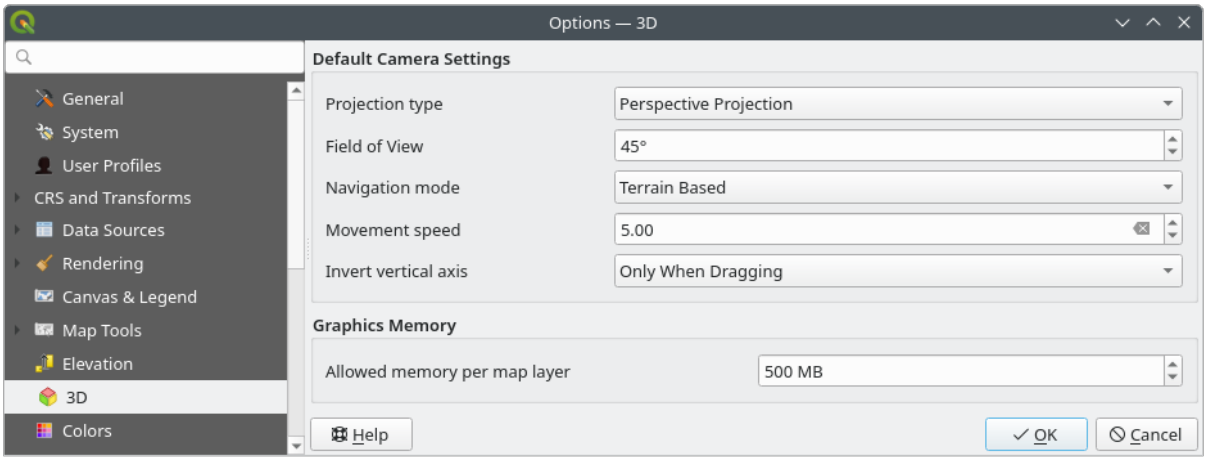
5.1.9 Elevation settings



Слика5.16: Elevation settings

In *Elevation* menu, you can set a specific color to use as the *Background color* for *elevation profiles*. This can make the chart more readable for certain datasets, e.g., point clouds with RGB coloring, where the default background color is too similar to point colors to be easily discernible. If unchanged, the elevation profiles will continue to display using the standard system background color.

5.1.10 3D settings



Слика5.17: 3D settings

The *3D* menu helps you configure some default settings to use for any *3D Map view*. These can refer to:

- *Default Camera Settings*:
  - *Projection type*: allowing to view the 3D scene in a:
    - *Perspective projection* (default): Parallel lines appear to meet in the distance. Objects appear to shrink the farther they are from the camera.
    - or an *Orthogonal projection*: Parallel lines appear parallel. Objects appear the same size regardless of distance.
  - Camera's *Field of view*: only relevant in perspective projection, specifies the current vertical field of view in degrees and determines how much of the scene is visible to the camera. Default value is 45°.
  - *Navigation mode*: provides different means to interact with the 3D scene. Available modes are:

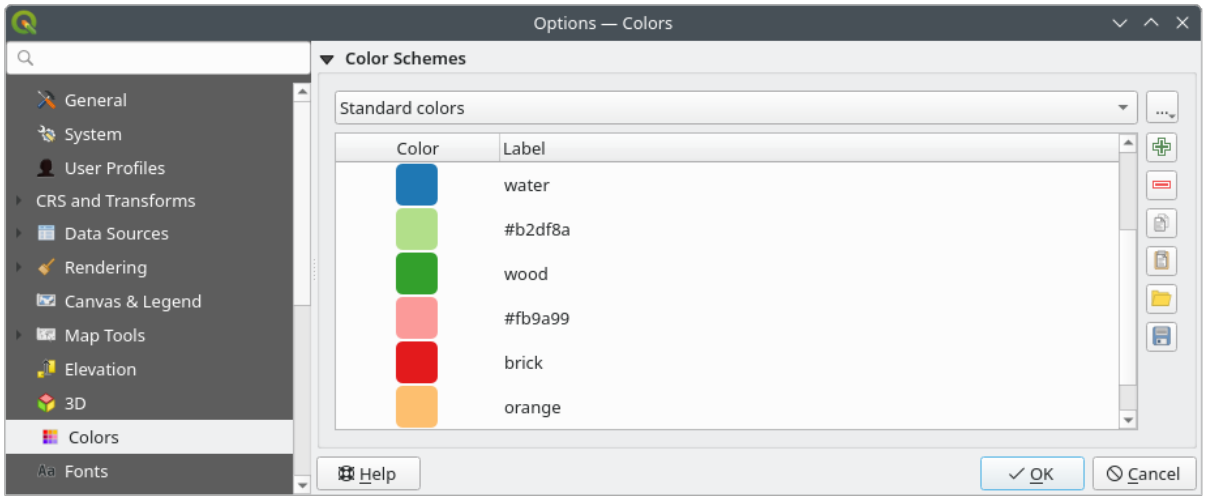
    - *Terrain based*: the camera follows around a fixed position on the surface of the terrain as the scene is navigated.
    - *Walk mode (first person)*

    Depending on the selected mode, *navigation commands* differ.

  - *Movement speed*
  - *Invert vertical axis*: Controls whether vertical axis movements should be inverted from their normal behaviour. Only affects movement in the *Walk mode*. It can be set to:
    - *Never*
    - *Only when dragging*: causes the vertical motion to inverted only when performing a click-and-drag camera rotation
    - and *Always*: causes the motions to be inverted when both click-and-dragging and when the camera movement is locked to the cursor (via a ~ key press)

- Under *Graphics memory*, the *Allowed memory per layer* option lets you set the GPU memory limit configuration on each layer. This is useful for users utilizing large 3D scenes which exhaust the available GPU memory resources. When a limit is hit, a warning is also displayed, which should assist in troubleshooting large scenes.

5.1.11 Colors settings



Слика5.18: Colors settings

This menu allows you to create or update palettes of colors used throughout the application in the *color selector widget*. You can choose from:

- *Recent colors* showing recently used colors
- *Standard colors*, the default palette of colors
- *Project colors*, a set of colors specific to the current project (see *Styles Properties* for more details)
- *New layer colors*, a set of colors to use by default when new layers are added to QGIS
- or custom palette(s) you can create or import using the … button next to the palette combobox.

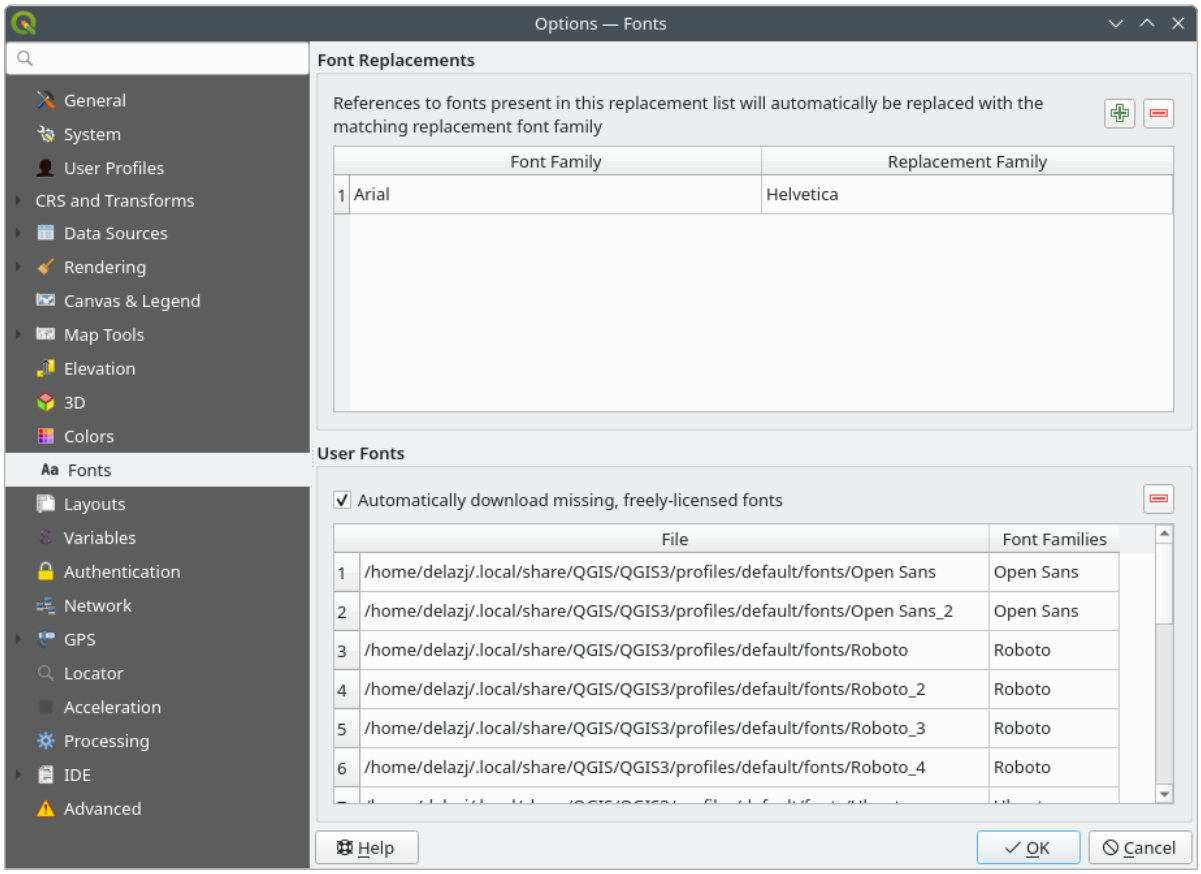
By default, *Recent colors*, *Standard colors* and *Project colors* palettes can not be removed and are set to appear in the color button drop-down. Custom palettes can also be added to this widget thanks to the *Show in Color Buttons* option.

For any of the palettes, you can manage the list of colors using the set of tools next to the frame, ie:

- *Add* or *Remove* color
- *Copy* or *Paste* color
- *Import* or *Export* the set of colors from/to `.gpl` file.

Double-click a color in the list to tweak or replace it in the *Color Selector* dialog. You can also rename it by double--clicking in the *Label* column.

5.1.12 Fonts Settings



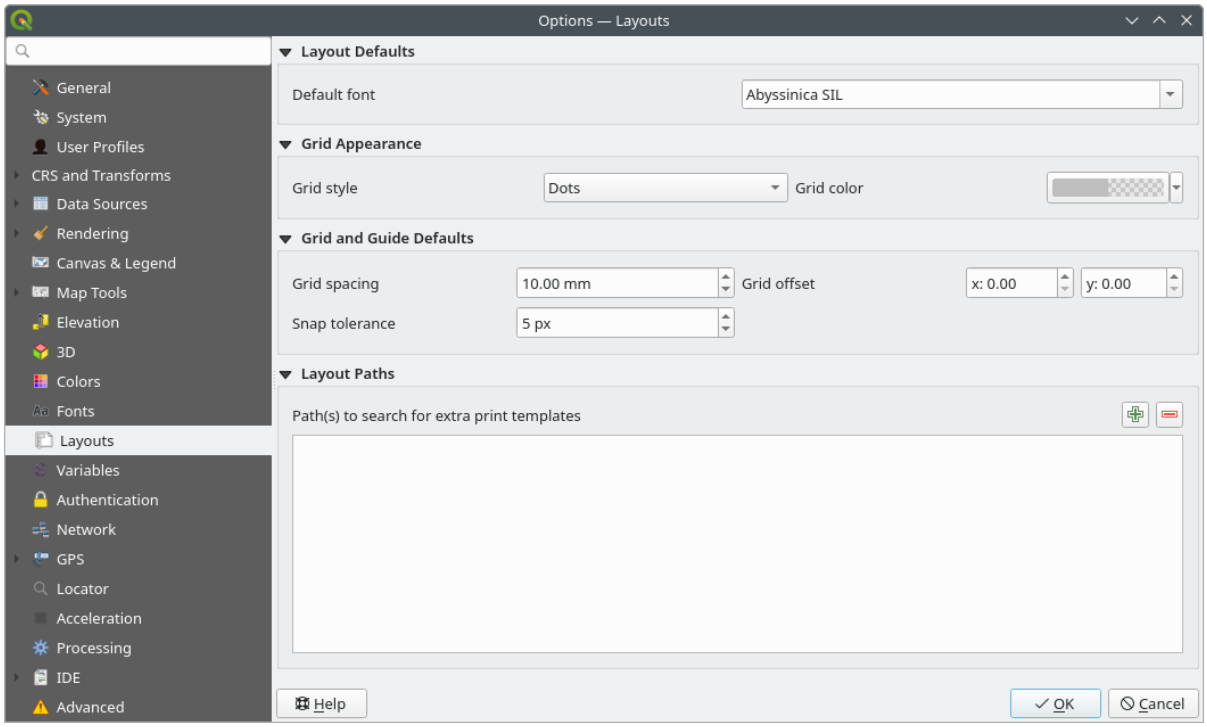
Слика5.19: Fonts settings

The *Fonts* tab provides support to manage fonts used across the projects:

- *Font Replacements*: gives ability to populate a list of automatic font replacements to apply when loading projects or styles, allowing better support for projects and styles to be used across different operating systems (e.g., replace „Arial” with „Helvetica”).
- *User Fonts*: Allows you to place TTF or OTF fonts in the `fonts` sub-folder of the *user profile*. These fonts can be automatically loaded at QGIS startup time. This provides a way to use fonts without requiring them to be installed on an operating system level, which is often blocked in enterprise environments. The panel lists all installed user fonts and allows you to manage (i.e., remove) previously installed user fonts.

  It is also possible to *Automatically download missing, freely-licensed fonts*: E.g., if you open a project or style, or try to load a vector tile layer that references fonts that aren't currently available, then a hard-coded list of freely licensed fonts to download via URL is consulted to determine whether it's possible to automatically download the font to the user profile font directory (with notification of the font license).

5.1.13 Layouts settings



Слика5.20: Layouts settings

Composition defaults

You can define the *Default font* used within the *print layout*.

Grid appearance

- Define the *Grid style* ('Solid', 'Dots', 'Crosses')
- Define the *Grid color*

Grid and guide defaults

- Define the *Grid spacing*
- Define the *Grid offset* for X and Y
- Define the *Snap tolerance*

Layout Paths

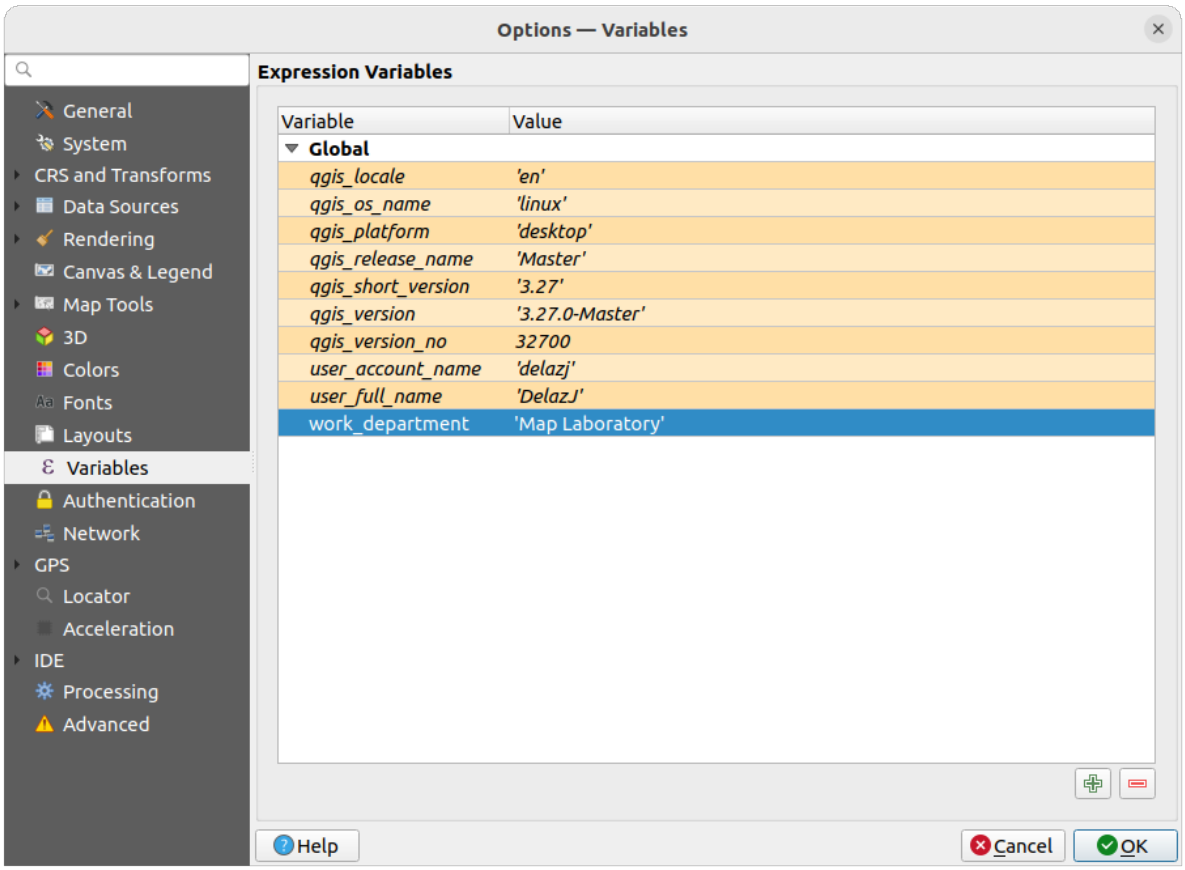
- Define *Path(s) to search for extra print templates*: a list of folders with custom layout templates to use while creating new one.

5.1.14 Variables settings

The *Variables* tab lists all the variables available at the global-level.

It also allows the user to manage global-level variables. Click the button to add a new custom global-level variable. Likewise, select a custom global-level variable from the list and click the button to remove it.

More information about variables in the *Storing values in Variables* section.



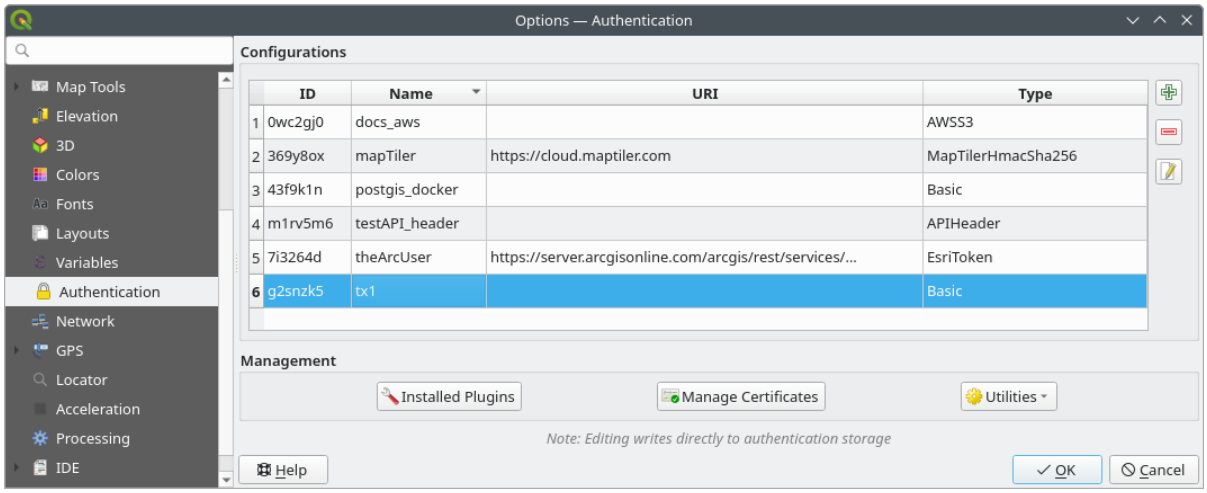
Слика5.21: Variables settings

5.1.15 Authentication settings

In the *Authentication* tab you can set authentication configurations and manage PKI certificates. See *Sistem za autentifikaciju* for more details.

To manage authentications, you can use the list of tools next to the frame, ie:

- Add new authentication configuration
- Remove selected authentication configuration
- Edit selected authentication configuration

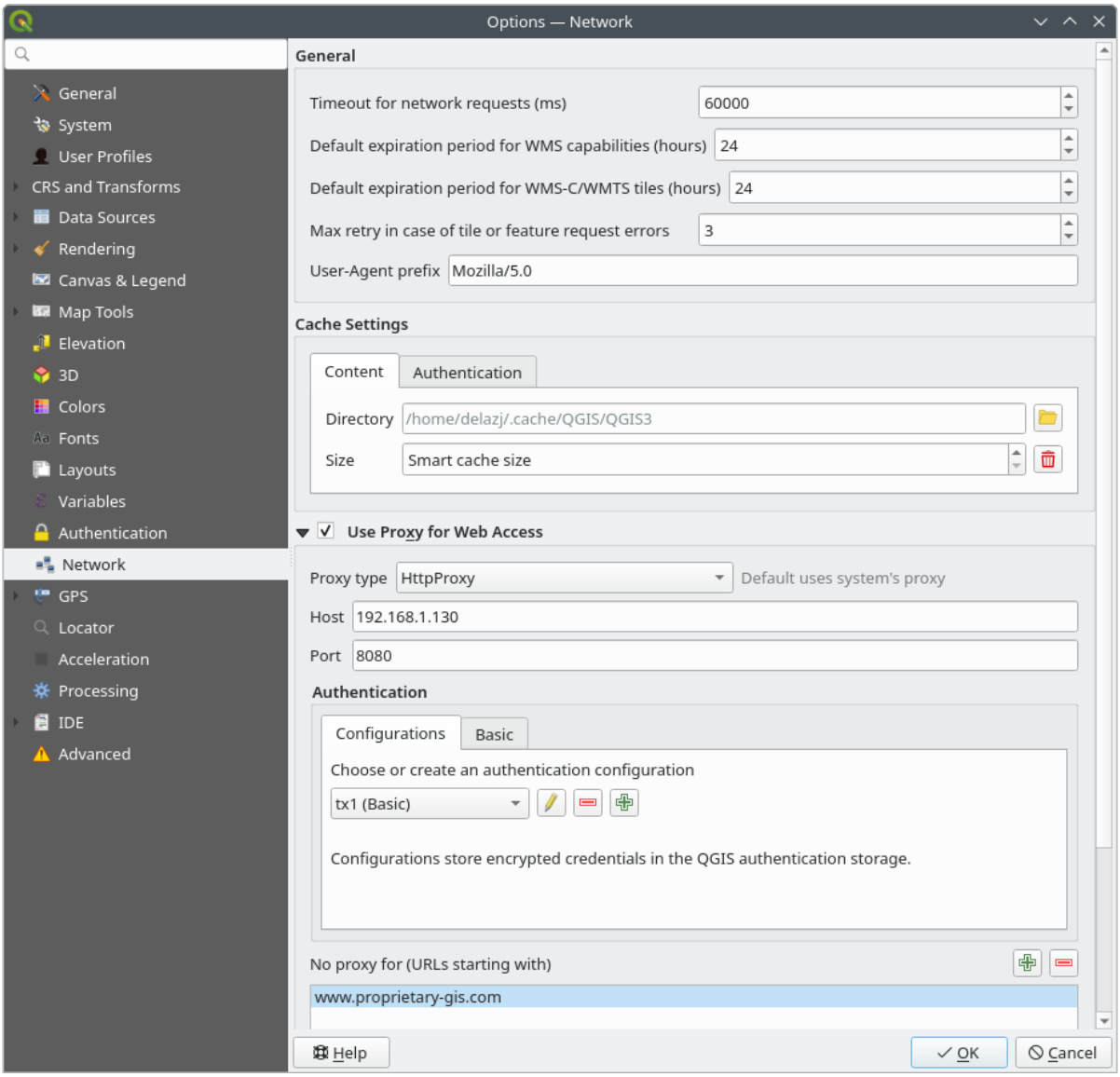


Слика5.22: Authentication settings

5.1.16 Network settings

General

- Define *Timeout for network requests (ms)* - default is 60000
- Define *Default expiration period for WMS Capabilities (hours)* - default is 24
- Define *Default expiration period for WMS-C/WMTS tiles (hours)* - default is 24
- Define *Max retry in case of tile or feature request errors*
- Define *User-Agent prefix* which defaults to `Mozilla/5.0`. This value will be prepended to both QGIS and OS version e.g., `Mozilla/5.0 QGIS/33801/Ubuntu 22.04.4 LTS` to shape the user-agent.



Слика5.23: Network and proxy settings

Cache settings

Defines the *Directory* and a *Size* for the cache specified in megabytes. You can clear the value to enable the *Smart cache size*, which sets the maximum cache size based on available disk space. Also offers tools to *automatically clear the connection authentication cache on SSL errors (recommended)*.

Proxy for web access

- *Use proxy for web access*
- Set the *Proxy type* according to your needs and define 'Host' and 'Port'. Available proxy types are:
  - *Default Proxy*: Proxy is determined based on system's proxy
  - *Socks5Proxy*: Generic proxy for any kind of connection. Supports TCP, UDP, binding to a port (incoming connections) and authentication.
  - *HttpProxy*: Implemented using the „CONNECT" command, supports only outgoing TCP connections; supports authentication.

- *HttpCachingProxy*: Implemented using normal HTTP commands, it is useful only in the context of HTTP requests.
- *FtpCachingProxy*: Implemented using an FTP proxy, it is useful only in the context of FTP requests.

Credentials of proxy are set using the *authentication widget*.

Excluding some URLs can be added to the text box below the proxy settings (see Сл.5.23). No proxy will be used if the target url starts with one of the string listed in this text box.

If you need more detailed information about the different proxy settings, please refer to the manual of the underlying QT library documentation at https://doc.qt.io/archives/qt-5.9/qnetworkproxy.html#ProxyType-enum

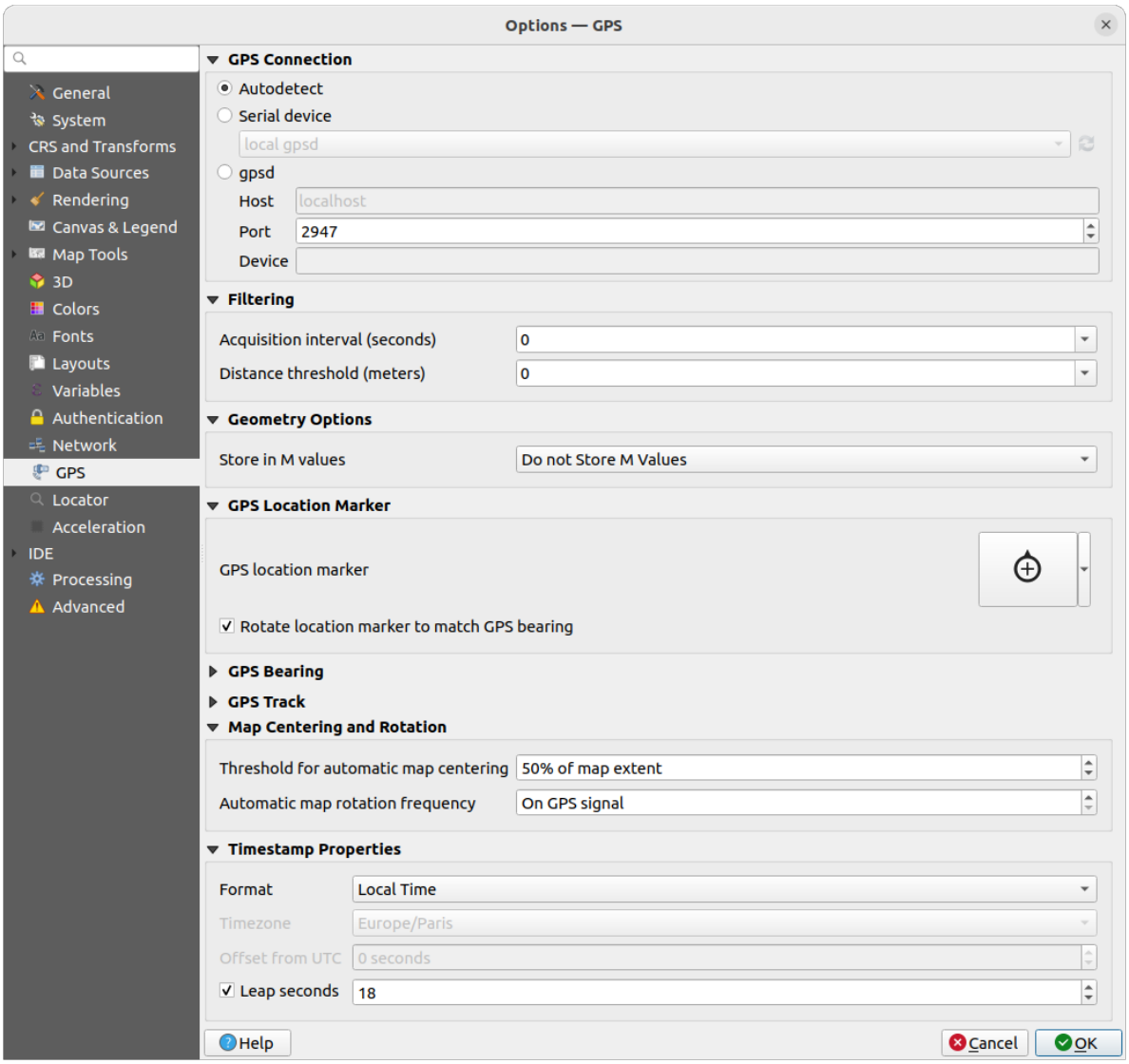
Савет

Using Proxies

Using proxies can sometimes be tricky. It is useful to proceed by 'trial and error' with the above proxy types, to check if they succeed in your case.

5.1.17 GPS settings

The *GPS* dialog helps you configure GPS devices connections and properties in QGIS. It also provides settings for GPS tracking and data digitizing.



Слика5.24: GPS settings

You can specify:

- *GPS Connection*: provides different means to connect the device to QGIS
  - *Autodetect*
  - *Serial device* (reload required if a new GPS Device is connected)
  - *gpsd* (selecting the *Host* and *Port* your GPS is connected to, and providing a *Device* name)

  In case of connection problems, you can try to switch from one to another.
- *Filtering*: You can set an *Acquisition interval (seconds)* and/or a *Distance threshold (meters)* parameters to keep the cursor still active when the receiver is in static conditions.
- Under *Geometry Options*, the *Store in M values* option allows to create geometries with M values from the inbuilt GPS logging tools. This applies to both features digitized from GPS logs and from the *Log to GeoPackage/SpatiaLite* functionality… Options include storing timestamps (as ms since epoch), ground speed, altitudes, bearings, and accuracy components as m values.
- *GPS Location Marker*
  - Configure a symbol for the *GPS location marker*, indicating the current GPS position

  - *Rotate location marker to match GPS bearing*: whether the marker symbol should be rotated to match the GPS direction

- *GPS Bearing*:
  - configure a *Bearing line style* using QGIS line symbol properties
  - set whether to *Calculate Bearings from travel direction*: If checked, the bearing reported by the GPS device will be ignored and the bearing will instead be calculated by the angle between the previous two GPS locations.
- *GPS Track*: set symbol to use for the *Track line style*
- *Map Centering and Rotation*: defines when the map canvas is updated according to the user displacement on the field:
  - *Threshold for automatic map centering*: defines the minimal offset of the GPS position from the map canvas center to trigger an automatic *map recentering* if enabled.
  - *Automatic map rotation frequency*: defines how often the map rotation to match the GPS bearing could happen; it can be on a custom duration or *On GPS signal*.
- *Timestamp Properties* to configure how time values are displayed and stored in the data. Parameters include the *Format* which can be:
  - *Local time*
  - *UTC*
  - *UTC with offset*, to account for daylight savings offsets or other complex time zone issues
  - a specific *Time zone*

  Moreover, *Leap seconds* correction can be applied, by adding the seconds to GPS timestamp.

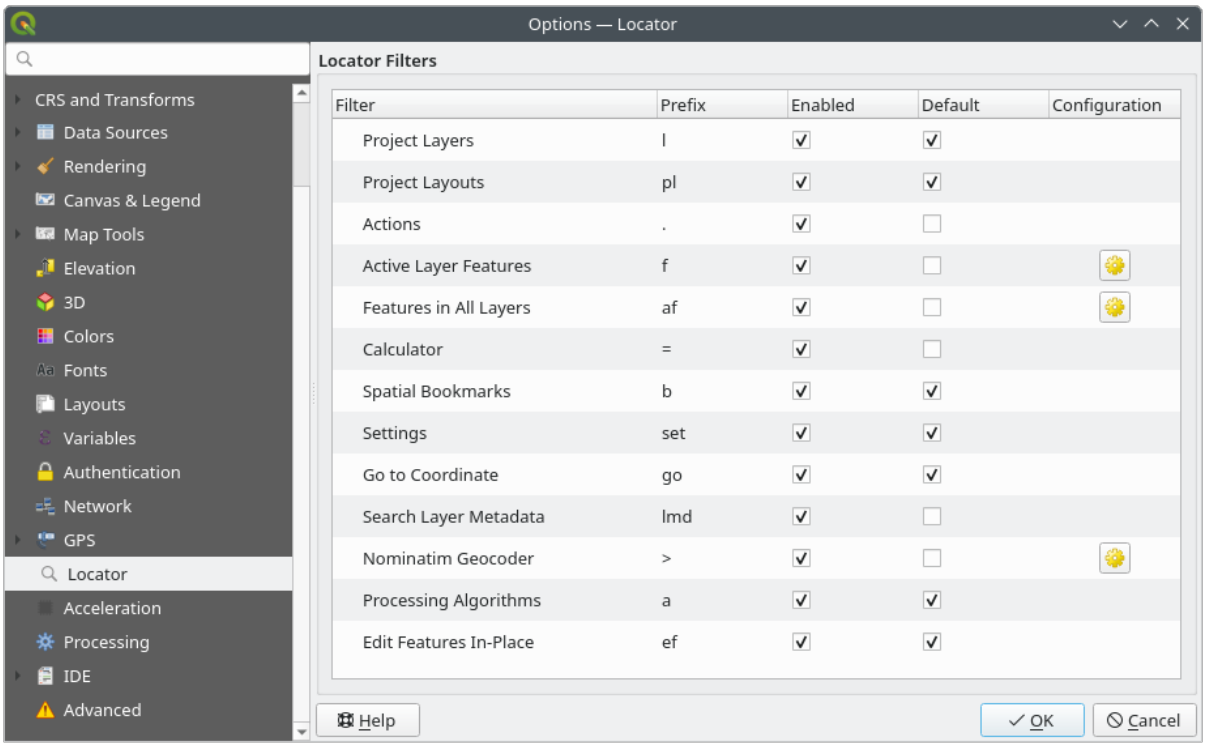
GPSBabel

GPSBabel converts waypoints, tracks, and routes between popular GPS receivers such as Garmin or Magellan and mapping programs like Google Earth or Basecamp. Literally hundreds of GPS receivers and programs are supported. QGIS relies on GPSBabel to interact with these devices and *manipulate their data*.

For details on how-to, please refer to *Loading to or from a device*.

5.1.18 Locator settings

The *Locator* tab lets you configure the *Locator bar*, a quick search widget available on the status bar to help you perform searches in the application. It provides some default filters (with prefix) to use:



Слика5.25: Locator settings

- *Project Layers* (`l`): finds and selects a layer in the *Layers* panel.
- *Project Layouts* (`pl`): finds and opens a print layout.
- Actions (`.`): finds and executes a QGIS action; actions can be any tool or menu in QGIS, opening a panel…
- *Active Layer Features* (`f`): searches for matching attributes in any field from the current active layer and zooms to the selected feature. Press to configure the maximum number of results.
- *Features in All Layers* (`af`): searches for matching attributes in the *display name* of each *searchable layers* and zooms to the selected feature. Press to configure the maximum number of results and the maximum number of results per layer.
- *Calculator* (`=`): allows evaluation of any QGIS expression and, if valid, gives an option to copy the result to the clipboard.
- *Spatial Bookmarks* (`b`): finds and zooms to the bookmark extent.
- *Settings* (`set`): browses and opens project and application-wide properties dialogs.
- *Go to Coordinate* (`go`): pans the map canvas to a location defined by a comma or space separated pair of x and y coordinates or a formatted URL (e.g., OpenStreetMap, Leaflet, OpenLayer, Google Maps, …). The coordinate is expected in WGS 84 (`epsg:4326`) and/or map canvas CRS.
- *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.
- Processing Algorithms (`a`): searches and opens a Processing algorithm dialog.
- *Edit Selected Features* (`ef`): gives quick access and runs a compatible *modify-in-place* Processing algorithm on the active layer.

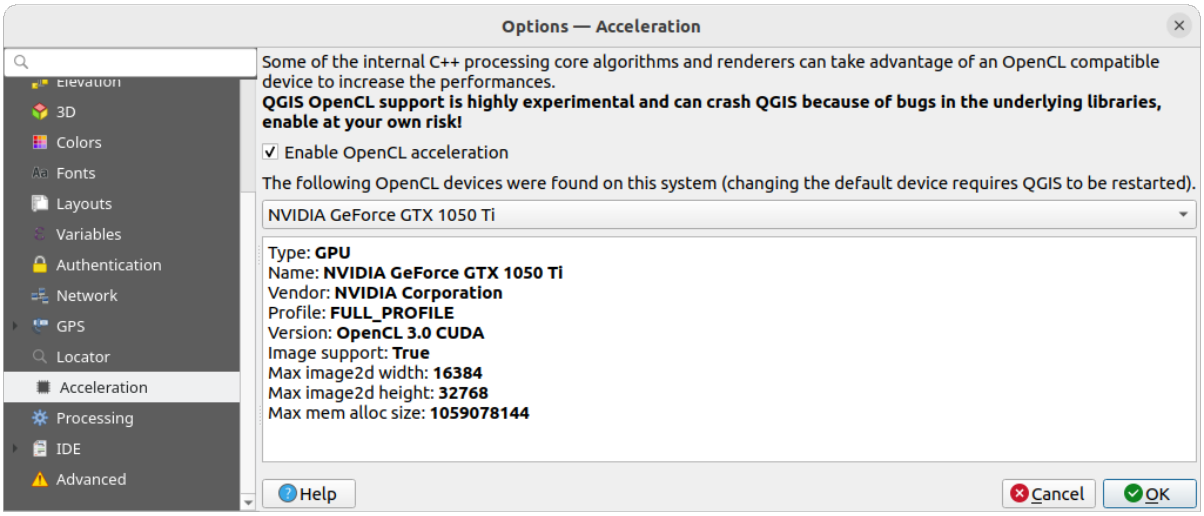
In the dialog, you can:

- customize the filter *Prefix*, i.e., the keyword to use to trigger the filter
- set whether the filter is *Enabled*: the filter can be used in the searches and a shortcut is available in the locator bar menu
- set whether the filter is *Default*: a search not using a filter returns results from only the default filters categories.
- Some filters provide a way to configure the number of results in a search.

The set of default locator filters can be extended by plugins, eg for OSM nominatim searches, direct database searching, layer catalog searches, …

5.1.19 Acceleration settings

OpenCL acceleration settings.



Слика5.26: Acceleration settings

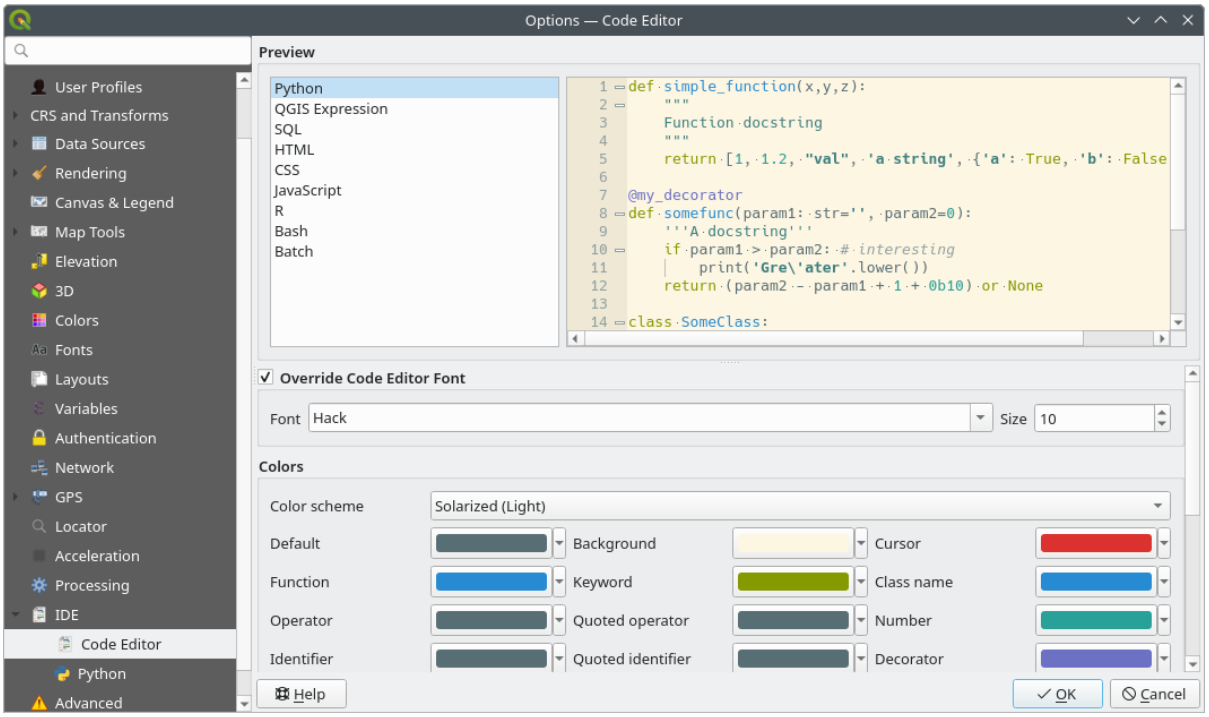
Depending on your hardware and software, you may have to install additional libraries to enable OpenCL acceleration.

5.1.20 IDE settings

Under *GitHub access token*, you can generate a personal token allowing you to share code snippets from within the Python code editor. More details on GitHub authentication

Code Editor settings

In the *Code Editor* tab, you can control the appearance and behaviour of code editor widgets (Python interactive console and editor, expression widget and function editor, …).



Слика5.27: Code Editor settings

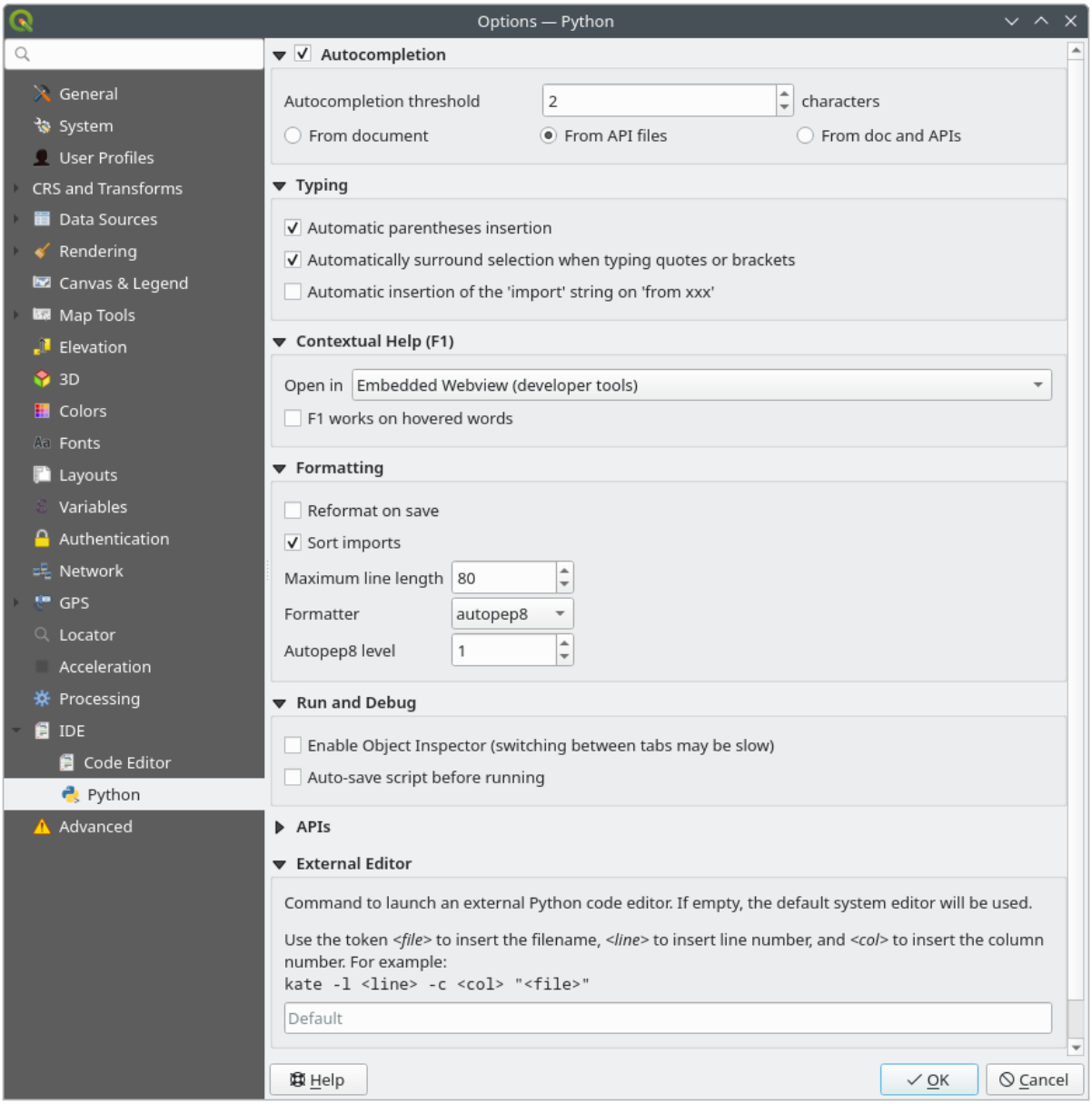
At the top of the dialog, a widget provides a live preview of the current settings, in various coding languages (Python, QGIS expression, HTML, SQL, JavaScript). A convenient way to adjust settings.

- Check *Override Code Editor Font* to modify the default *Font* family and *Size*.
- Under the *Colors* group, you can:
  - select a *Color scheme*: predefined settings are `Default`, `Solarized Dark` and `Solarized Light`. A `Custom` scheme is triggered as soon as you modify a color and can be reset with selecting a predefined scheme.
  - change the *color* of each element in code writing, such as the colors to use for comments, quotes, functions, background, …

Python settings

The *Python* settings help you manage and control the behavior of the Python editors (*interactive console*, *code editor*, *project macros*, *custom expressions*, …). It can also be accessed using the Options… button from:

- the *Python console* toolbar
- the contextual menu of the *Python console* widget
- and the contextual menu of the code editor.



Слика5.28: Python settings

You can specify:

- *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)
- under *Typing*
  - *Automatic parentheses insertion*: When no text is selected, if an opening character (parentheses, quotes, brackets, …) is entered, inserts the matching closing character just after the cursor. Note that this behavior is disabled if the current cursor is inside a string or comment.
  - *Automatically surround selection when typing quotes or brackets*: When an opening character is entered, the selected text is enclosed in the opening/closing pair. Selection remains the same, so it is possible to quote a selected word and enclose it in parentheses just by typing `"` then `(`.

Special case for multiline selection with quotes and double quotes: selection is enclosed in triple single/double quotes.

  - *Automatic insertion of the 'import' string on 'from xxx'*: Enables insertion of 'import' when specifying imports

- *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*.

- under *Formatting*, you can add automated tools to reformat the code you are writing:
  - *Reformat on save*: formatting is applied just before saving the script
  - *Sort imports*: sorts 'import' statements using the isort library
  - *Maximum line length*: controls how the formatter will wrap the lines, and controls the editor ruler
  - *Formatter* - supported tools are *autopep8* and *black*, with dedicated option:
    * *Autopep8 level* - more details at autopep8 advanced usage
    * *Normalize quotes*: replaces all single quotes with double quotes if possible
- under *Run and Debug*
  - *Enable Object Inspector (switching between tabs may be slow)*
  - *Auto-save script before running*: Saves the script automatically when executed. This action will store a temporary file (in the temporary system directory) that will be deleted automatically after running.

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

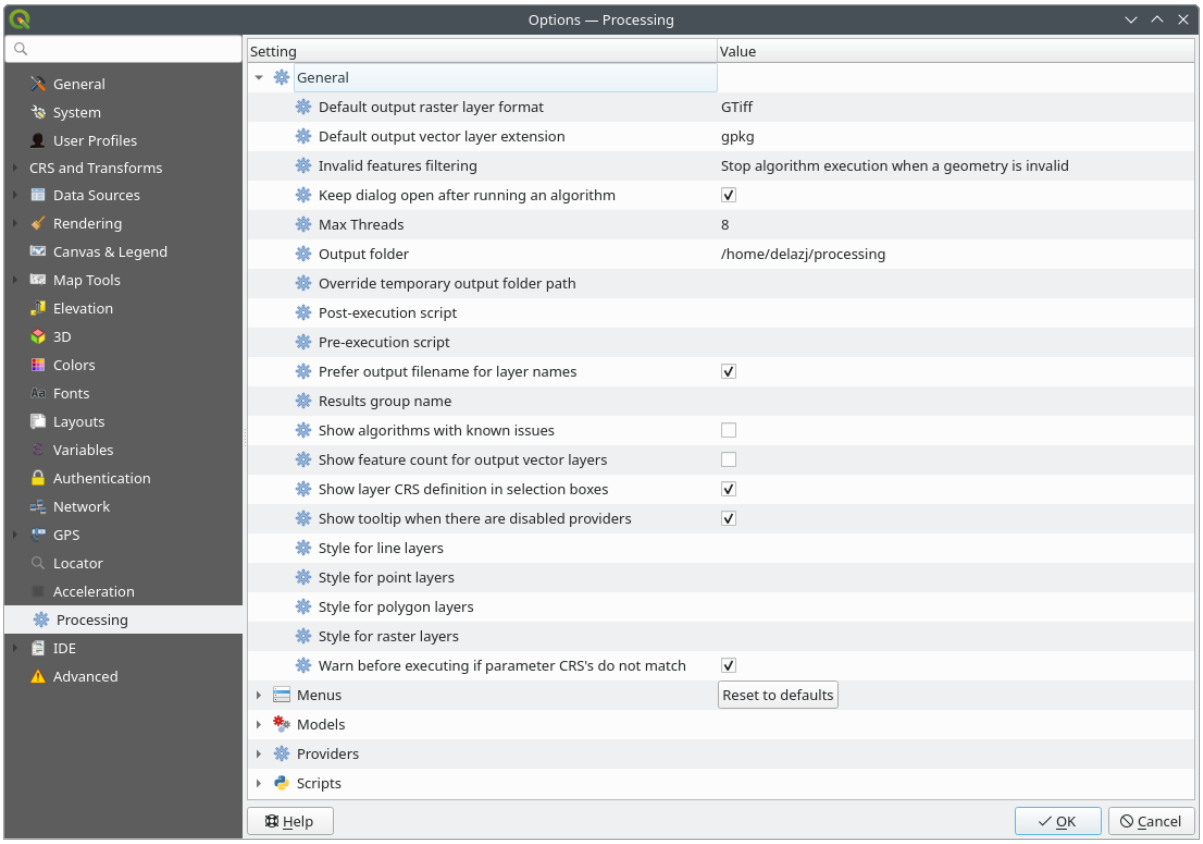
  You can also choose to add specific API files.

- *Using prepared APIs file*: If checked, the chosen `*.pap` file will be used for code completion. To generate a prepared API file you have to load at least one `*.api` file and then compile it by clicking the *Compile APIs...* button.

The *External Editor* group allows you to provide command line instructions to launch an external Python code editor, given a file name, a line and column number. If empty, the default system editor will be used.

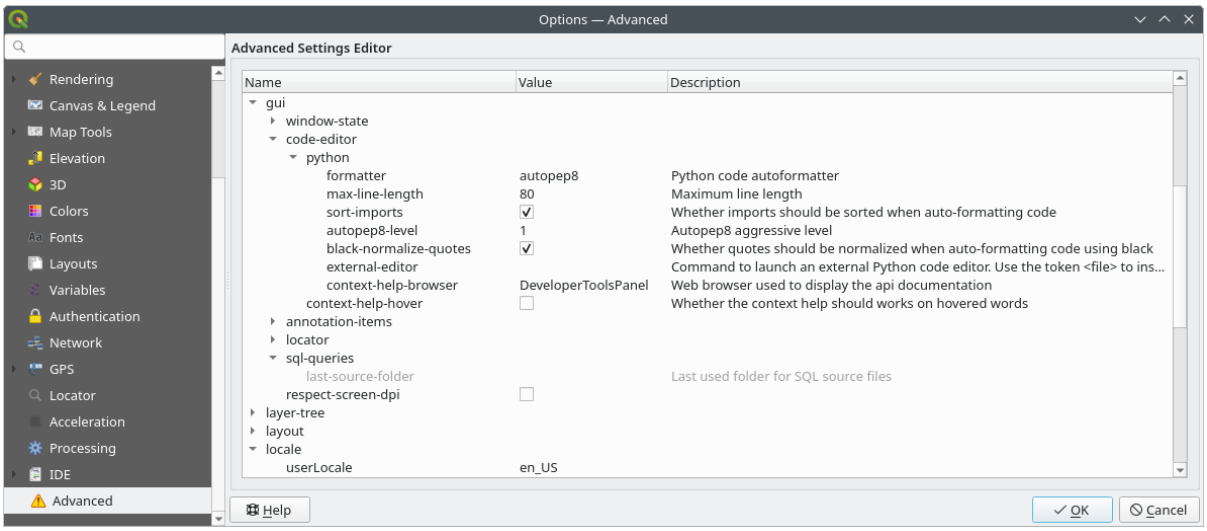
5.1.21 Processing settings

The *Processing* tab provides you with general settings of tools and data providers that are used in the QGIS Processing framework. More information at *Configuring the Processing Framework*.



Слика5.29: Processing settings

5.1.22 Advanced settings



Слика5.30: Advanced settings

All the settings related to QGIS (UI, tools, data providers, Processing configurations, default values and paths, plugins options, expressions, geometry checks…) are saved in a `QGIS/QGIS3.ini` file under the active *user profile* directory. Configurations can be shared by copying this file to other installations.

From within QGIS, the *Advanced* tab offers a way to manage these settings through the *Advanced Settings Editor*. After you promise to be careful, the widget is populated with a tree of all the existing settings, and you can edit their value. Right-click over a setting or a group and you can delete it (to add a setting or group, you have to edit the `QGIS3.ini` file). Changes are automatically saved in the `QGIS3.ini` file.

Упозорење

Avoid using the Advanced tab settings blindly

Be careful while modifying items in this dialog given that changes are automatically applied. Doing changes without knowledge can break your QGIS installation in various ways.

5.2 Working with User Profiles

5.2.1 The concept

The *Settings* ► *User Profiles* menu provides functions to set and access user profiles. A user profile is a unified application configuration that allows to store in a single folder:

- all the *global settings*, including locale, projections, authentication settings, color palettes, shortcuts…
- GUI configurations and *customization*
- grid files and other proj helper files installed for datum transformation
- installed *plugins* and their configurations
- project templates and history of saved project with their image preview
- *processing settings*, logs, scripts, models.

By default, a QGIS installation contains a single user profile named `default`. But you can create as many user profiles as you want:

1. Click the *Settings* ► *User Profiles* ► *New profile…* entry.
2. You'll be prompted to provide a profile name, creating a folder of the same name under `~/<UserProfiles>/` where:
   - `~` represents the **HOME** directory, which on Windows is usually something like `C:\Users\<username>`.
   - and `<UserProfiles>` represents the main profiles folder, i.e.:
     - `.local/share/QGIS/QGIS3/profiles/`
     - `AppData\Roaming\QGIS\QGIS3\profiles\`
     - `Library/Application Support/QGIS/QGIS3/profiles/`

   The user profile folder can be opened from within QGIS using the *Open Active Profile Folder*.
3. A new instance of QGIS is started, using a clean configuration. You can then set your custom configurations.

If you have more than one profile in your QGIS installation, the name of the active profile is shown in the application title bar between square brackets.

As each user profile contains isolated settings, plugins and history they can be great for different workflows, demos, users of the same machine, or testing settings, etc. And you can switch from one to the other by selecting them in the *Settings* ► *User Profiles* menu. You can also run QGIS with a specific user profile from the *command line*.

Савет

Run QGIS under a new user profile to check for bug persistence

Bugs you may encounter with functions in QGIS can be related to leftovers in the current user profile. Running QGIS under another user profile can help you solve them or check the origin of the issue. The general advice is to launch QGIS under a new user profile, thus a cleaner configuration, and run the commands again.

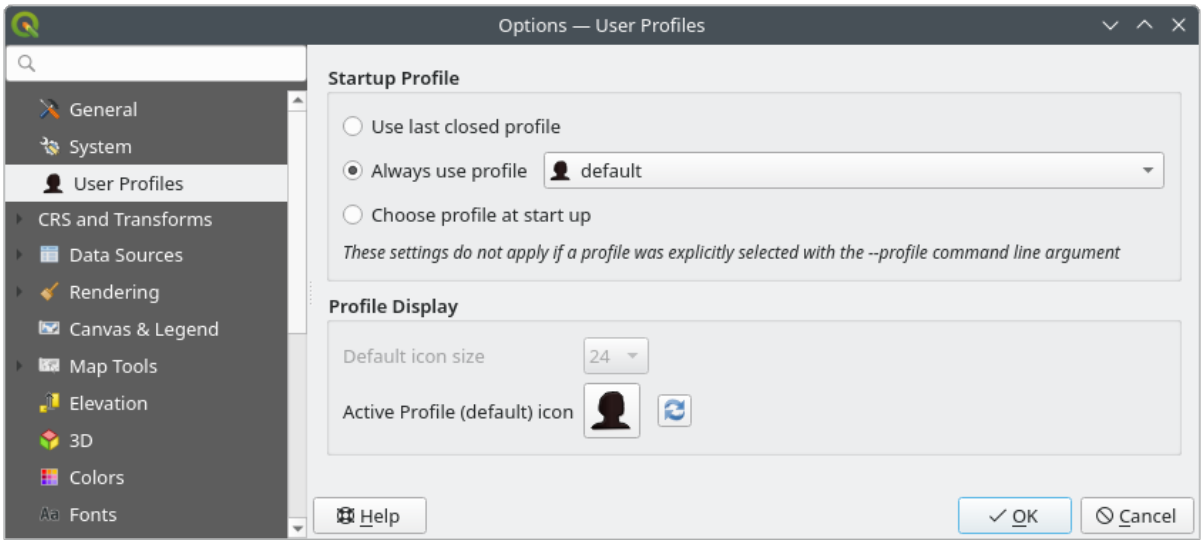
If the bug prevents you to create a new user profile from within the *Settings* ► *User Profiles* menu, you can either:

- Rename in the file explorer, the „broken" user profile folder in the `QGIS3/profiles` folder and restart QGIS. A new `default` user profile will be created and executed.
- Start QGIS from the command line, using the new *profile name* argument:

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

5.2.2 Setting user profile

By default, QGIS opens a new session with the profile of the last closed session. This, among other settings, can be customized in *Settings* ► *Options* ► *User Profiles* tab:



Слика5.31: User profiles settings

- *Startup profile*: indicates the user profile to use when starting a QGIS session. It can be:
  - *Use last closed profile*
  - a specific user profile to select from *Always use profile* drop-down menu
  - *Choose profile at start up*: Opens the *User Profile Selector* dialog listing available user profiles. Double--click an entry or select a profile and press *OK* to use that user profile for the starting session. It is also possible to *Add new profile* to the list: a profile is created from scratch and automatically run with the opening session.
- Under *Profile display*, you can set:
  - The size of icons to use when selecting a profile from the *User Profile Selector* dialog.
  - A specific icon to display next to the current profile in the *Settings* ► *User profiles* menu or the *User Profile Selector* dialog. Press Reset profile icon to remove custom modifications.

5.3 Project Properties

In the properties window for the project under *Project* ► *Project Properties*, you can set project-specific options. The project-specific options overwrite their equivalent in the *Options* dialog described above.

5.3.1 General Properties

In the  *General* tab, the *General settings* let you:

- see the location of the project file
- set the folder for the project home (available in the *Project home* item of the *Browser* panel). The path can be relative to the folder of the project file (type it in) or absolute. The project home can be used for storing data and other content that is useful for the project. Convenient when dataset and project files are not stored at the same place. If not filled, the *Project home* defaults to the project file folder.
- give a title to the project beside the project file path
- choose the color to use for features when they are selected
- choose the background color: the color to use for the map canvas
- set whether the path to layers in the project should be saved as absolute (full) or as relative to the project file location. You may prefer relative path when both layers and project files can be moved or shared or if the project is accessed from computers on different platforms.
- choose to avoid artifacts when project is rendered as map tiles. Note that checking this option can lead to performance degradation.
- *Remember attribute tables windows and docks between sessions*: If checked for a project, then any opened attribute tables will be saved into the project and immediately restored when loading that project. This can improve workflows when you constructed a project with a particular set of attribute table configurations for your requirements, and re-setting up these attribute tables is a hassle.

Calculating areas and distances is a common need in GIS. However, these values are really tied to the underlying projection settings. The *Measurements* frame lets you control these parameters. You can indeed choose:

- the *Ellipsoid*, on which distance, area and angle calculations are entirely based; it can be:
  - **None/Planimetric**: returned values are in this case cartesian measurements. This option can be set as default for new projects from the *Settings* ► *Options* ►  *CRS Handling* menu
  - a **Custom** one: you'll need to set values of the semi-major and semi-minor axes.
  - or an existing one from a predefined list (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:
  - Attribute table field update bar
  - Field calculator calculations
  - *Identify tool* derived length, perimeter and area values
  - *measure dialog*
  - *scale bar decoration*

Белешка

To ensure measurement units (for distance and area) always reflect the units of the current project's CRS, select *Map Units* for these options.

- 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.

    Белешка

    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

Белешка

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

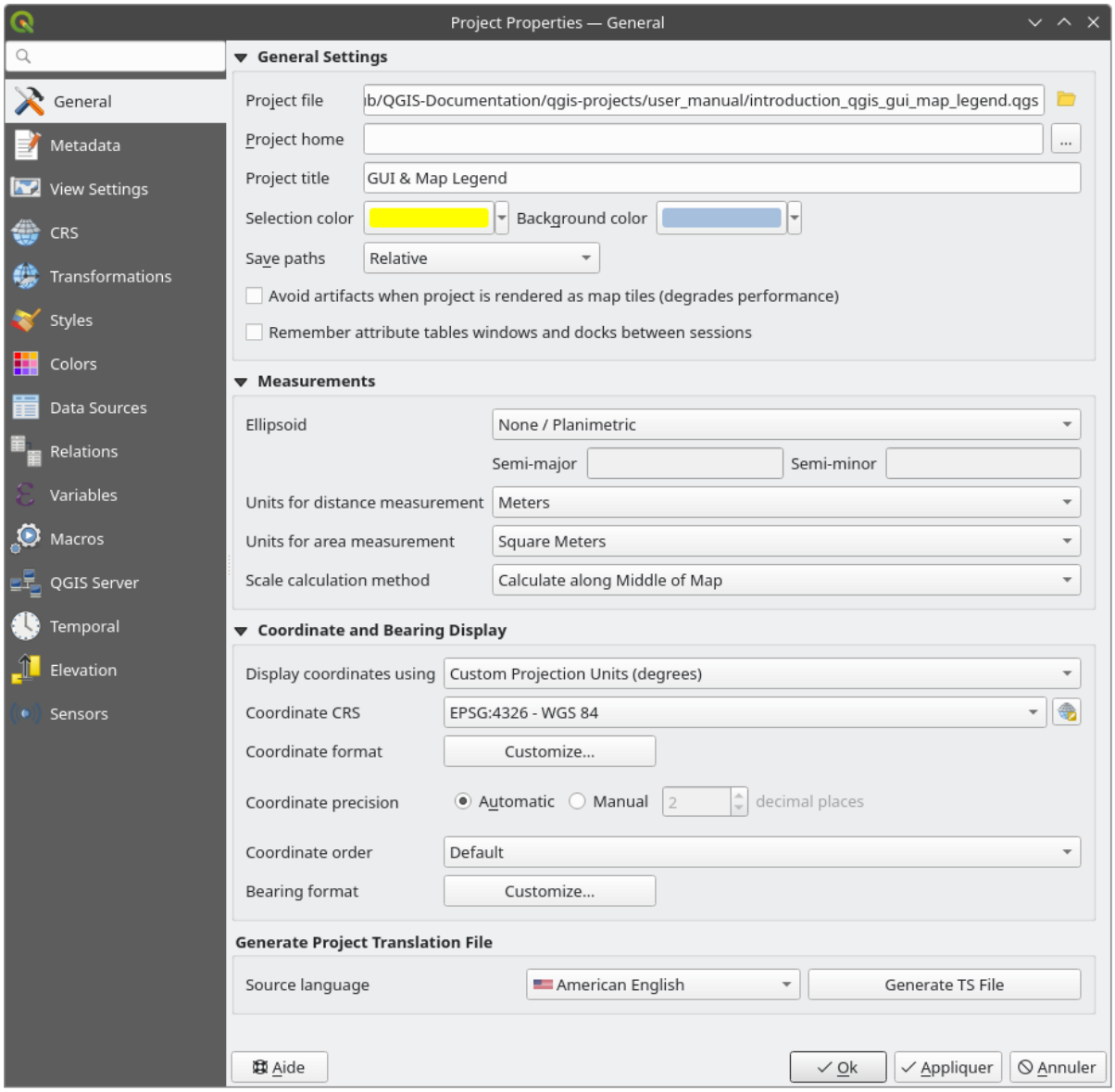
The *Coordinate and Bearing display* allows you to customize the display of:

- the coordinates shown in the *Coordinates* box on QGIS status bar and in the *Derived* section of the  Identify features tool's results
- the bearing value displayed in the status bar for the map canvas panning direction and by the  Measure bearing tool.

Available parameters are:

- *Display coordinates using* either:
  - `Map Units`, based on the project CRS
  - `Map Geographic (degrees)`: based on the project CRS if it is of geographic type, otherwise uses its associated geographic CRS. This is helpful e.g., for non-earth celestial bodies.

  - or `Custom Projection Units`: allows to rely on any CRS you desire for coordinates display
- In the *Coordinate CRS* option, you can view or define the CRS to use depending on your display mode.
- *Coordinate format*: you can configure it as `Decimal Degrees`, `Degrees, Minutes` or `Degrees, Minutes, Seconds`, and whether it should display:
  - ☐ *Show directional suffix*
  - ☐ *Show leading zeros for minutes and seconds*
  - ☐ *Show leading zeros for degrees*
  - ☐ *Show trailing zeros*
- *Coordinate precision*: the number of decimal places can be automatic (derived from the type of CRS) or set manually
- *Coordinate order*: you can opt to display the coordinates in the native order of the CRS (`Default`) or switch it to either `Easting, Northing (Longitude, Latitude)` or `Northing, Easting (Latitude, Longitude)` order
- *Bearing format* possible values are `0 to 180°, with E/W suffix`, `-180 to +180°` or `0 to 360°`. The number of *Decimal places* as well as whether to *Show trailing zeros* can be set.

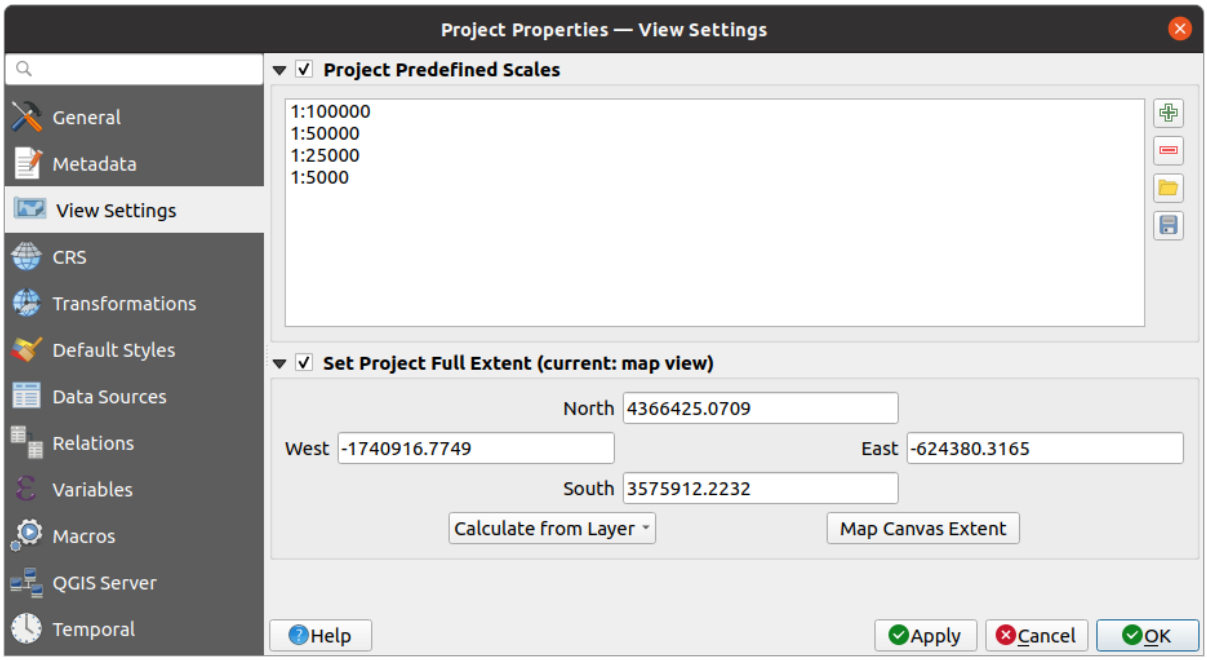


Слика5.32: General tab of the Project Properties dialog

5.3.2 Metadata Properties

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

5.3.3 View Settings



Слика5.33: View Settings tab of the Project Properties dialog

The *View Settings* tab provides means to control the project map canvas. You can:

- set *Project predefined scales*: the list of scales to display in scale-related drop-down widgets, such as the status bar *Scale*, the visibility scales selector or secondary 2D map view settings,… in replacement of the global *predefined scales*.
- *Set Project full Extent*: this extent will be used instead of the extent of all layers when zooming to full map extent ( ). It's useful when a project contains web layers/national layers/global layers yet the actual area of interest for the project is a smaller geographic area. The project full extent coordinates can be set with the *extent selector* widget.

5.3.4 CRS Properties

Белешка

For more information on how QGIS handles project projection, please read the dedicated section at *Working with Projections*.

The *CRS* tab helps you set the coordinate reference system to use in this project. It can be:

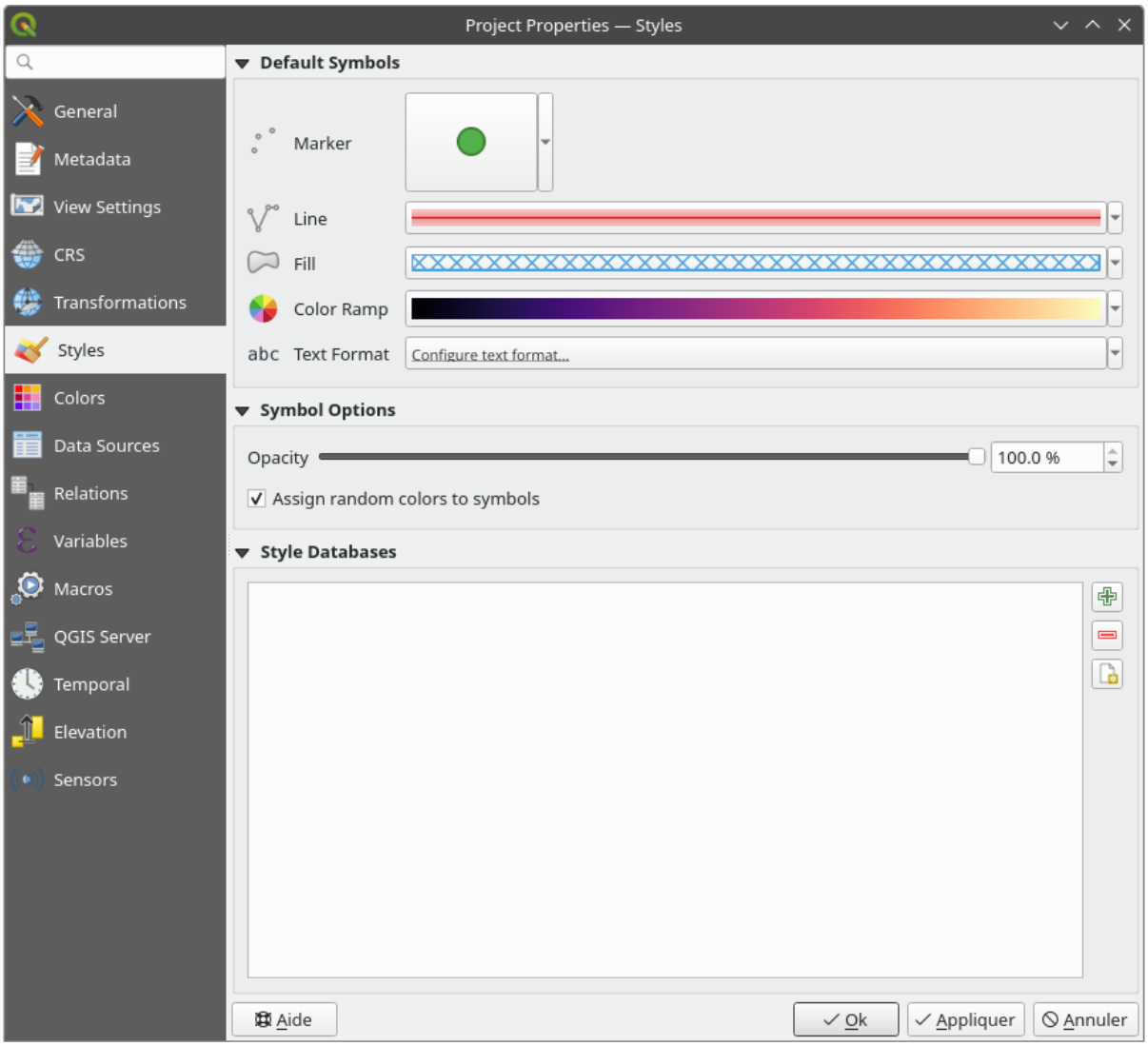
- *No CRS (or unknown/non-Earth projection)*: layers are drawn based on their raw coordinates
- or an existing coordinate reference system that can be *geographic*, *projected* or *user-defined*. Layers added to the project are translated on-the-fly to this CRS in order to overlay them regardless their original CRS.

5.3.5 Transformations Properties

The *Transformations* tab helps you control the layers reprojection settings by configuring the datum transformation preferences to apply in the current project. As usual, these override any corresponding global settings. See *Datum Transformations* for more details.

5.3.6 Styles Properties

Under  *Styles* tab, you can configure symbols and colors inherent to the project, allowing to safely share the project among different machines.



Слика5.34: Styles tab

Default symbols

The *Default Symbols* group lets you control how new layers will be drawn in the project when they do not have an existing `.qml` style defined. You can set *Marker*, *Line*, *Fill* to apply depending on the layer geometry type as well as default *Color Ramp* and *Text Format* (e.g., when enabling labeling). Any of these items can be reset using the *Clear* entry from the corresponding drop-down widget.

Symbol options

In the *Symbol options* group, you can:

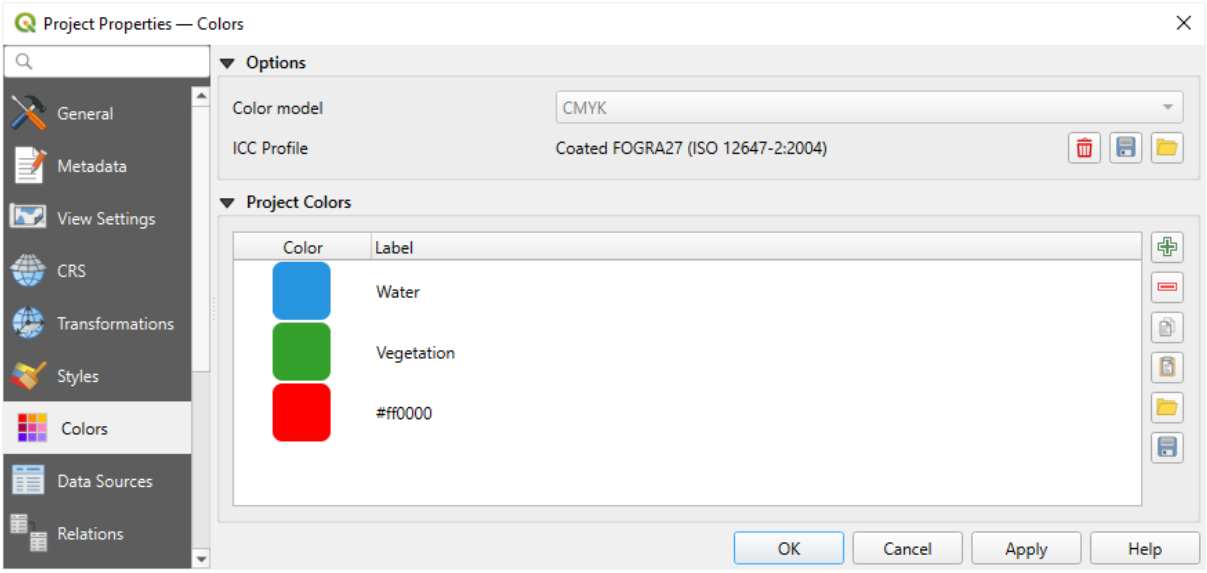
- Apply a default *Opacity* to new layers
-  *Assign random colors to symbols*, modifying the symbols fill colors, hence avoiding same rendering for all layers.

Style Database

A style database in QGIS is a structured repository designed to store symbols, text formats, and other styling elements. It serves as a centralized location where you can organize and manage your symbology resources efficiently. You can create a dedicated style database for a specific client, housing symbols tailored to that client's need. This ensures a clean and organized approach, without cluttering the default style database. In multi-user environments, it's possible to store project-specific styles in a shared location. By linking a project to these styles, all users within the project gain access to common symbology, eliminating the need for manual import or updates to individual local style database. Storing symbols in a project file offers a practical solution, preventing users from overcrowding their global style database with project-specific symbology. This approach guarantees that other users working on the same project immediately have access to all the necessary symbology upon loading the project.

In the *Style Database* section you can choose to *Add* or *Remove* style database or you can *Create new style database*. When you add or remove a style database in this section, the changes will be automatically reflected in the *Symbology Properties*.

5.3.7 Colors properties



Слика5.35: Colors tab

In the *Colors* tab, you can select the *Color model*, i.e., whether selecting a color should by default be defined as `RGB` or `CMYK`. Any color defined in a different color model than the one specified here will be converted to this color model when exporting a layout.

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.

Белешка

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

There is also an additional section where you can define specific colors for the running project. Like the *global colors*, you can:

- *Add* or *Remove* color
- *Copy* or *Paste* color
- *Import* or *Export* the set of colors from/to `.gpl` file.

Double-click a color in the list to tweak or replace it in the *Color Selector* dialog. You can also rename it by double--clicking in the *Label* column.

These colors are identified as *Project colors* and listed as part of *color widgets*.

Савет

Use project colors to quickly assign and update color widgets

Project colors can be referred to using their label and the color widgets they are used in are bound to them. This means that instead of repeatedly setting the same color for many properties and, to avoid a cumbersome update you can:

1. Define the color as a project color
2. Click the *data defined override widget* next to the color property you want to set
3. Hover over the *Color* menu and select the project color. The property is then assigned the expression `project_color('color_label')` and the color widget reflects that color.
4. Repeat steps 2 and 3 as much as needed
5. Update the project color once and the change is reflected EVERYWHERE it's in use.

5.3.8 Data Sources Properties

In the *Data Sources* tab, you can:

- *Transaction mode*, defines how edits are sent to the data provider:
  - *Local Edit Buffer*: edits are buffered locally and sent to the provider when toggling layer editing mode or clicking *Save edits*.
  - *Automatic Transaction Groups*: on supported datasources (postgres and geopackage databases) the edit state of all tables that originate from the same database are synchronized and executed in a server side transaction. Also, instead of buffering edit changes locally, they are directly sent to a transaction in the database which gets committed when toggling layer editing mode or clicking *Save edits*.
  - *Buffered Transaction Groups*: all editable layers, disregarding from which provider, are toggled synchronously and all edits are saved in a local edit buffer. Saving changes is executed within a single transaction on all layers (per provider).

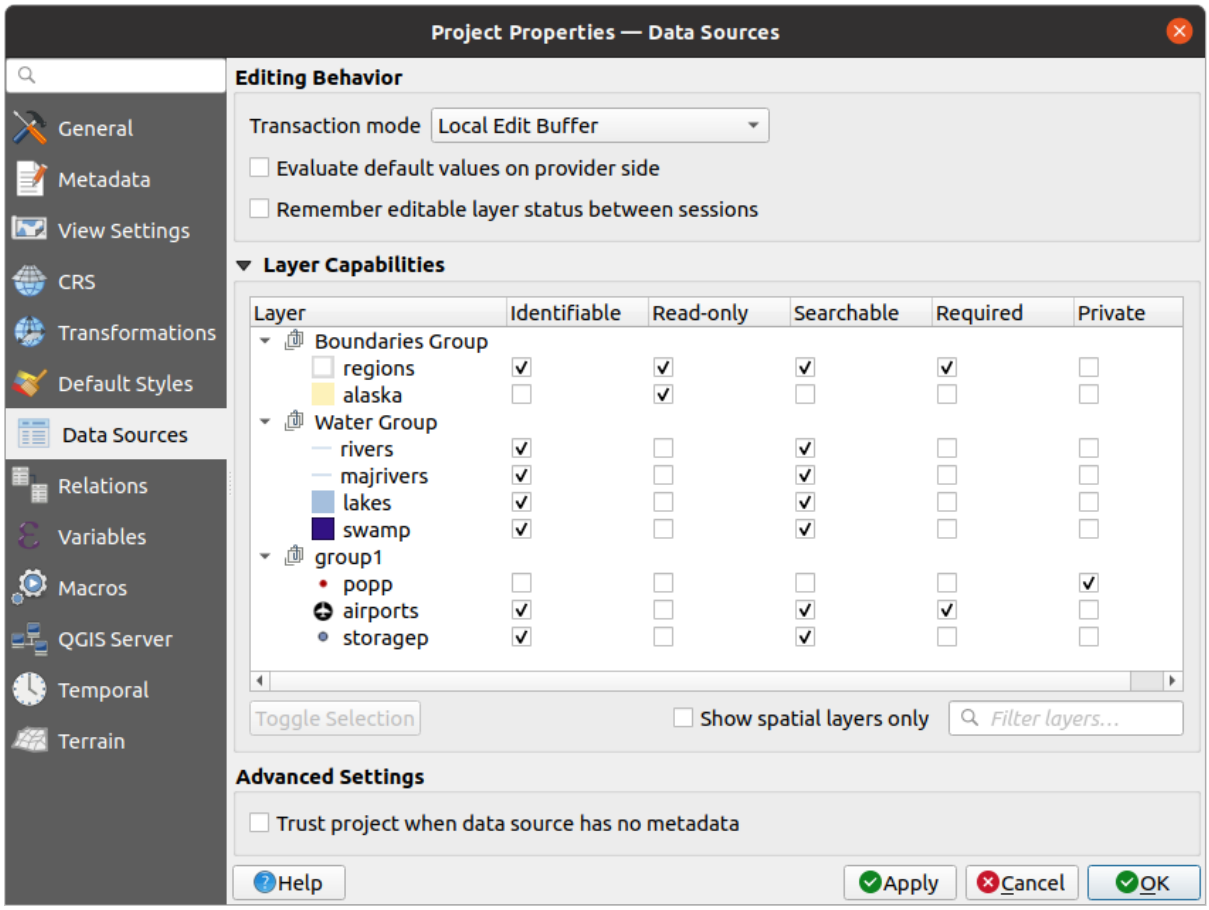
  Note that you can change this option only if no layer is being edited in the project.
- *Evaluate default values on provider side*: When adding new features in a PostgreSQL table, fields with default value constraint are evaluated and populated at the form opening, and not at the commit moment. This means that instead of an expression like `nextval('serial')`, the field in the *Add Feature* form will display expected value (e.g., `25`).
- *Remember editable layer status between sessions*: makes sure that all layers that are editable in a project will be remembered as such when saving the project, as well as making sure that those layers are immediately made editable whenever the project is restored.
- Configure the *Layers Capabilities*, i.e.:
  - Set (or disable) which layers are `identifiable`, i.e., will respond to the *identify tool*. By default, layers are set queryable.

  - Set whether a layer should appear as `read-only`, meaning that it can not be edited by the user, regardless of the data provider's capabilities. Although this is a weak protection, it remains a quick and handy configuration to avoid end-users modifying data when working with file-based layers.
  - Define which layers are `searchable`, i.e., could be queried using the *locator widget*. By default, layers are set searchable.
  - Define which layers are defined as `required`. Checked layers in this list are protected from inadvertent removal from the project.
  - Define which layers are `private`, i.e., hidden from the *Layers* panel. This is meant for accessory layers (basemap, join, lookups for value-relations, most probably aspatial layers, …) that you still need in a project but you don't want them to pollute the legend tree and other layer selection tools. If set visible, they are still displayed in the map canvas and rendered in the print layout legend. Use the  *Filter legend* ► *Show private layers* option in the *Layers* panel top toolbar to temporarily turned them on for any interaction.

  The *Layers Capabilities* table provides some convenient tools to:

  - Select multiple cells and press *Toggle Selection* to have them change their checkbox state;
  -  *Show spatial layers only*, filtering out non-spatial layers from the layers list;
  -  *Filter layers…* and quickly find a particular layer to configure.

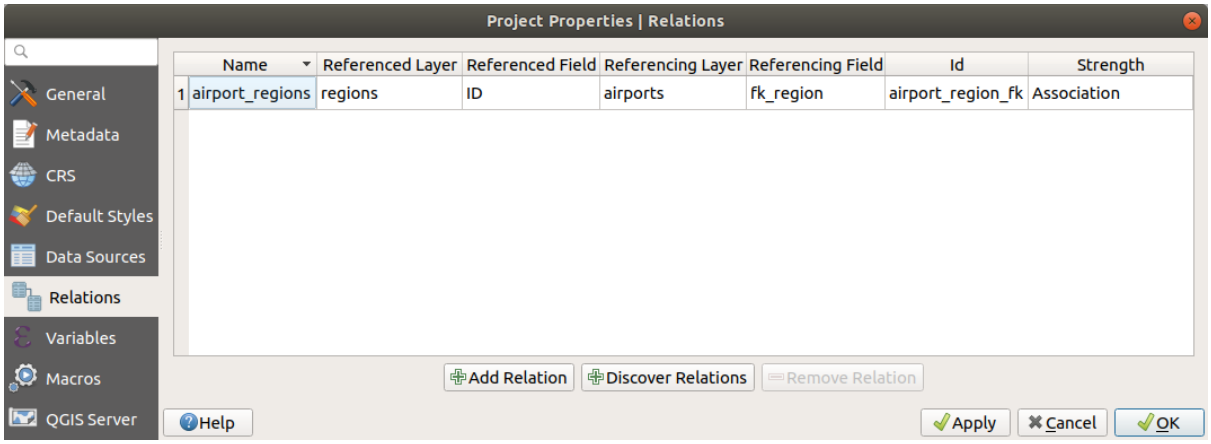
- Under the *Advanced Settings* group, you can select  *Trust project when data source has no metadata*: To speed up project loading by skipping data checks. Useful in QGIS Server context or in projects with huge database views/materialized views. The extent of layers will be read from the QGIS project file (instead of data sources) and when using the PostgreSQL provider the primary key unicity will not be checked for views and materialized views.



Слика5.36: Data Sources tab

5.3.9 Relations Properties

The *Relations* tab is used to define relations between layers. The relations can be of one to one, many to many or polymorphic type. They are defined in the project properties dialog. Once relations exist for a layer, a new user interface element in the form view (e.g., when identifying a feature and opening its form) will list the related entities. This provides a powerful way to express e.g., the inspection history on a length of pipeline or road segment. You can find out more about relations support in section *Setting relations between multiple layers*.



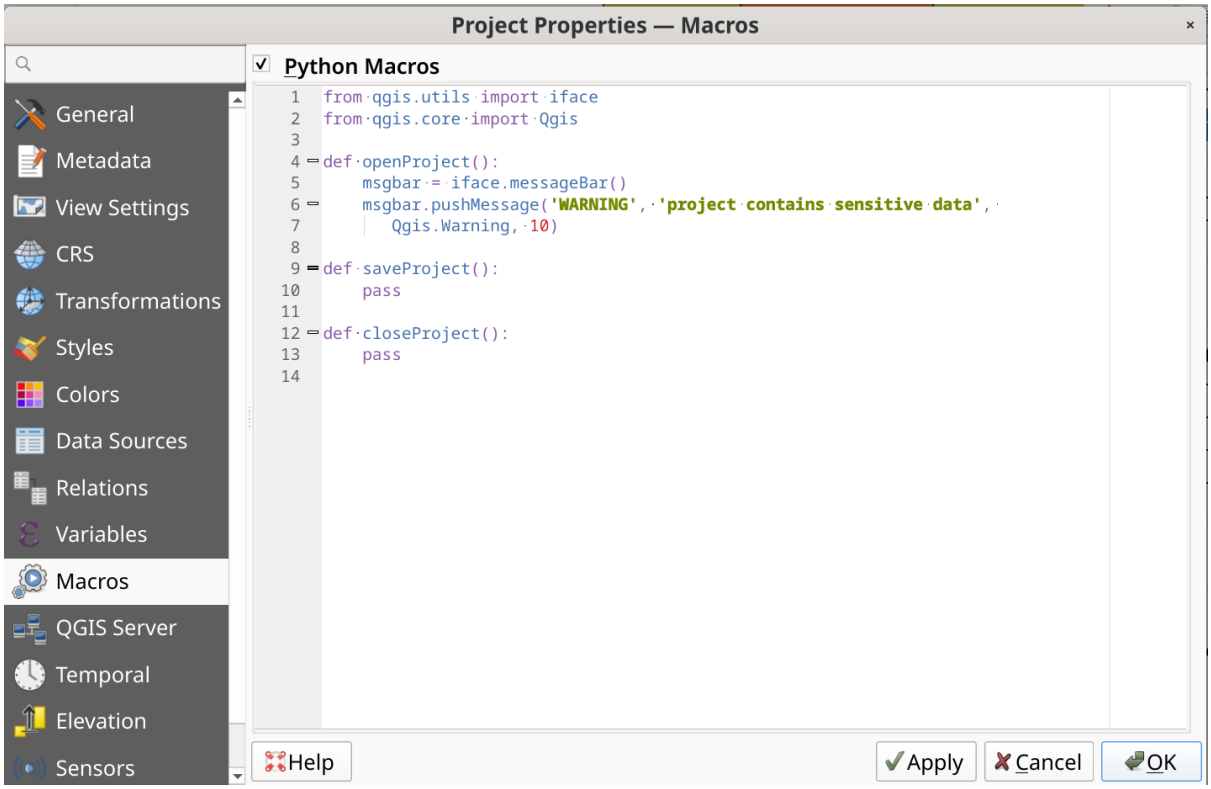
Слика5.37: Relations tab

5.3.10 Variables Properties

The *Variables* tab lists all the variables available at the project's level (which includes all global variables). Besides, it also allows the user to manage project-level variables. Click the button to add a new custom project--level variable. Likewise, select a custom project-level variable from the list and click the button to remove it. More information on variables usage in the General Tools *Storing values in Variables* section.

5.3.11 Macros Properties

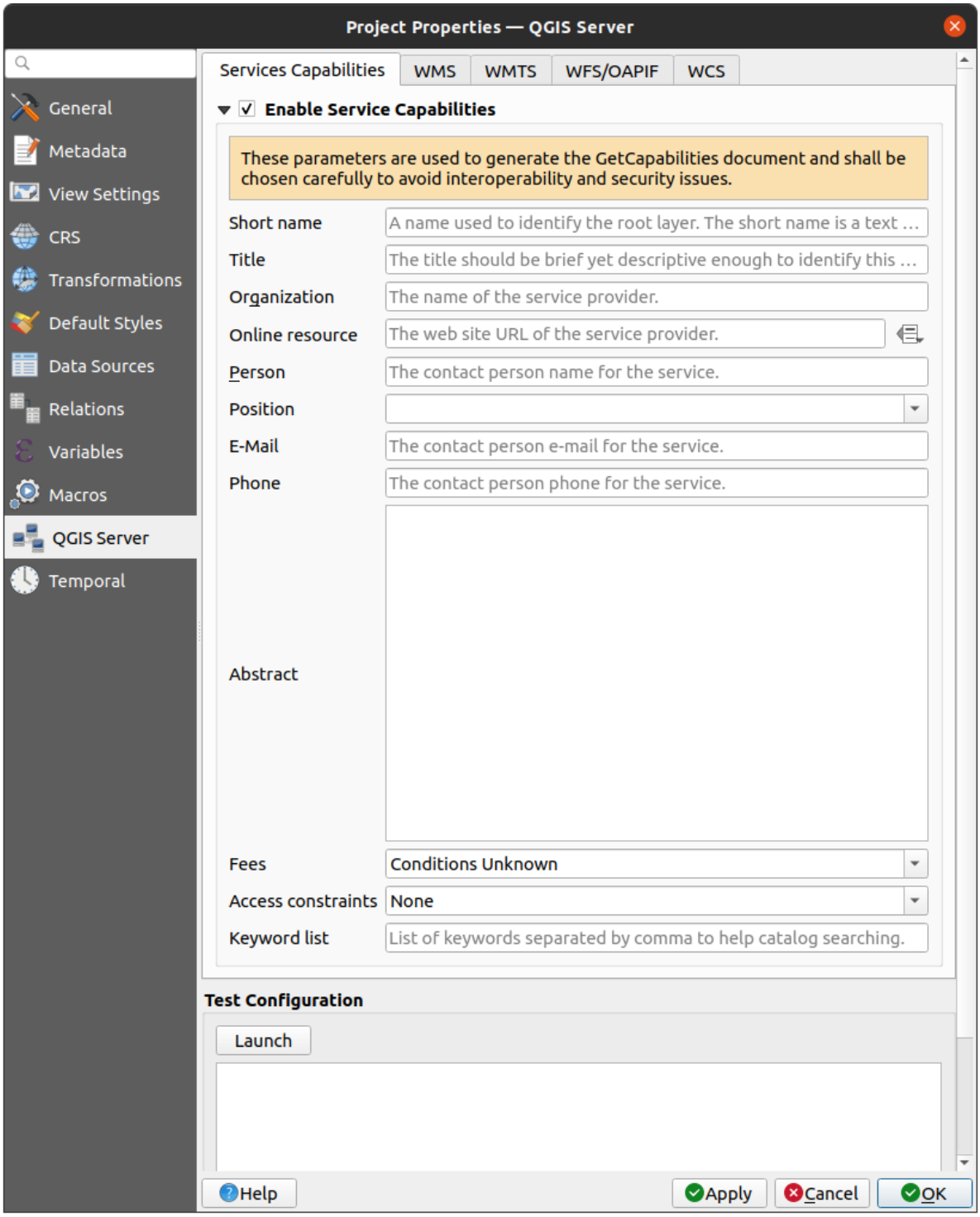
The *Macros* tab is used to edit Python macros for projects. Currently, only three macros are available: `openProject()`, `saveProject()` and `closeProject()`.



Слика5.38: Macro settings

5.3.12 QGIS Server Properties

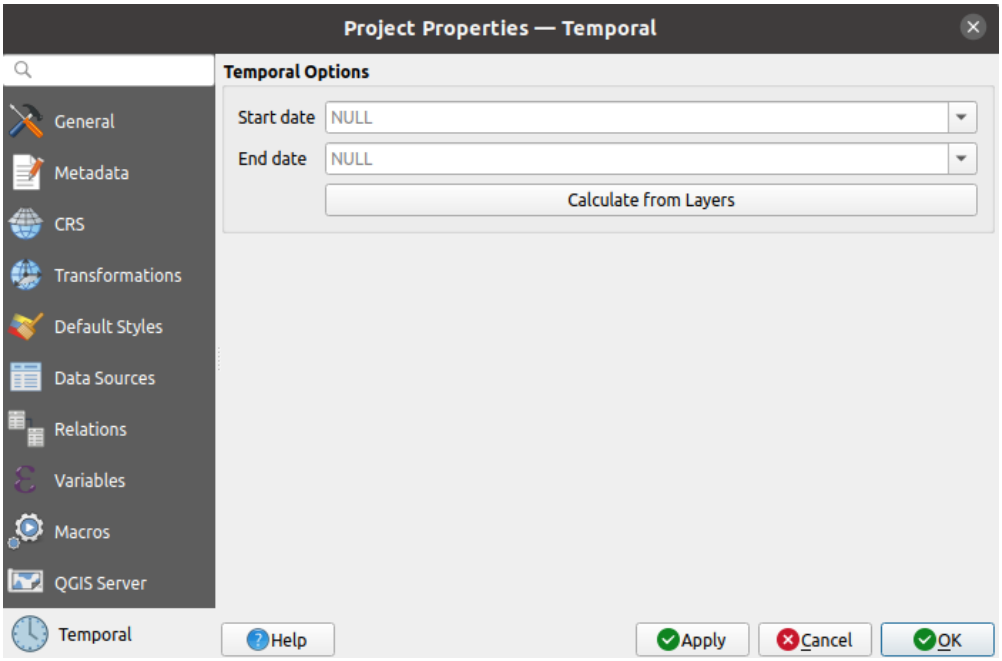
The *QGIS Server* tab allows you to configure your project in order to publish it online. Here you can define information about the QGIS Server WMS and WFS capabilities, extent and CRS restrictions. More information available in section Creatingwmsfromproject and subsequent.



Слика5.39: QGIS Server settings

5.3.13 Temporal Properties

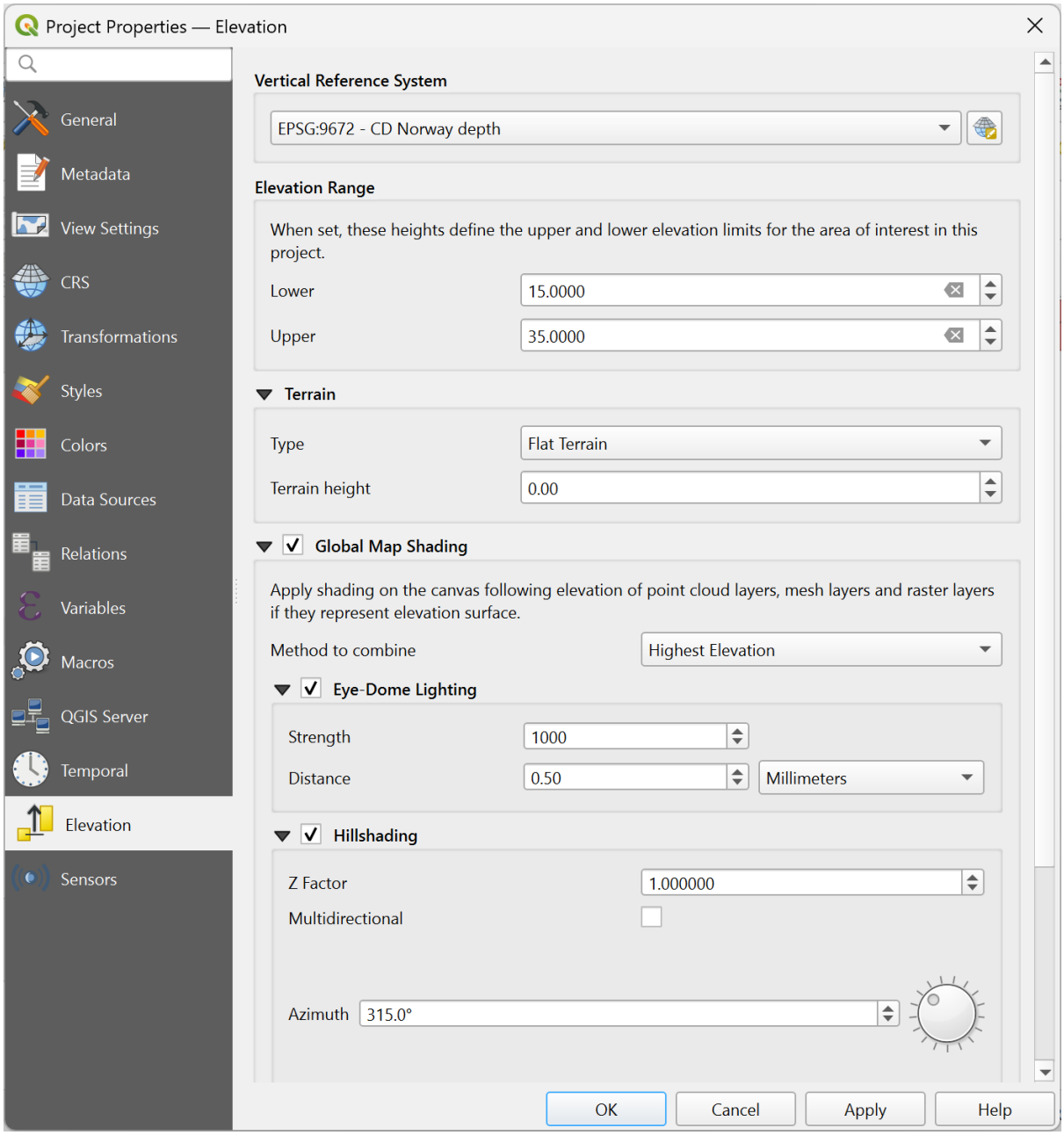
The *Temporal* tab is used to set the temporal range of your project, either by using manual *Start date* and *End date* inputs or by calculating it from the current project temporal layers. The project time range can then be used in the *Temporal controller panel* to manage the map canvas *temporal navigation*.



Слика5.40: Project Temporal tab

5.3.14 Elevation Properties

The *Elevation* tab helps you configure default settings for the terrain and elevation. When any new *3d map* is created in the project, the map will default to using the same terrain settings as are defined for the project. The project elevation settings will also be respected by the *elevation profile* tool.



Слика5.41: Project Elevation tab

- *Vertical Reference System*: If the CRS of your project is compound one (including a Z dimension), then the vertical CRS used for the project will be the vertical component of the project CRS. If your project CRS is horizontal, then you can select a specific vertical CRS by clicking on the Select CRS.
- *Elevation Range*: helps you define the *Lower* and *Upper* limits for the *elevation controller* in the project.
- *Terrain* and elevation options are available for:
  - *Flat terrain* with *Terrain height* setting
  - *DEM (Raster Layer)*: with setting for defining the *Raster layer*, a *Vertical scale* factor to apply to band values and a vertical *Offset*
  - *Mesh*: with setting for defining the *Mesh layer*, a *Vertical scale* factor to apply to vertices Z value and a vertical *Offset*

  These settings can be overwritten from the 3D map *configuration dialog*.

- With *Global map shading* settings, you apply a global shading effect to the map, based on the elevation of all the active layers that have elevation properties enabled, including:
  - *raster layers*
  - *mesh layers*
  - *point cloud layers*

  Method to combine: the elevation value at any particular position for all the relevant layers are combined, and the considered elevation is chosen depending on one of these methods:
  - *Highest elevation* value
  - *Based on layer's order*: the elevation on the topmost layer in the layer tree is considered.

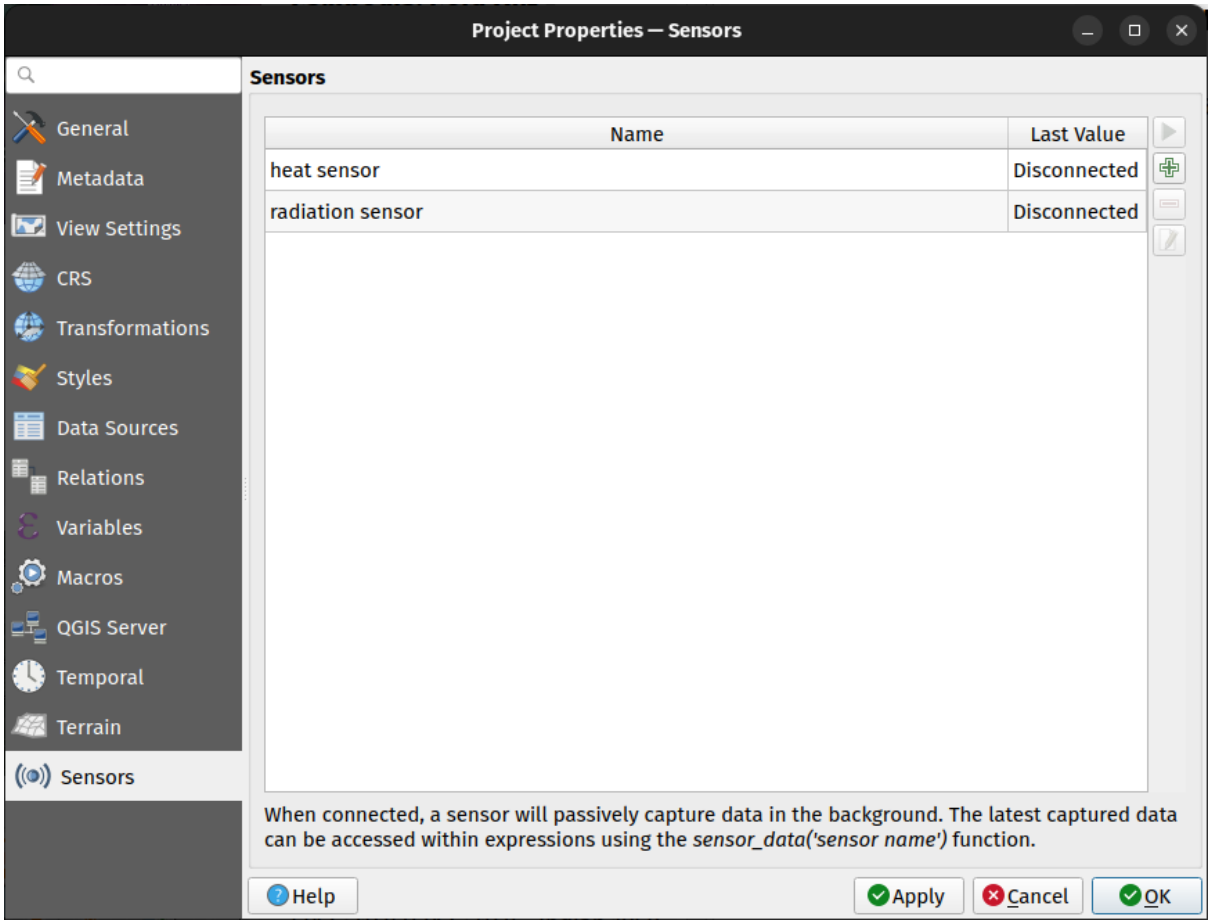
  Supported shading options are:
  - *Eye-dome lighting*: applies shading effects to the map canvas for a better depth rendering. Following parameters can be controlled:
    - *Strength*: increases the contrast, allowing for better depth perception
    - *Distance*: represents the distance of the used pixels off the center pixel and has the effect of making edges thicker.
  - *Hillshading*, shaping some reliefs on the map using shading (levels of gray):
    - *Z Factor*: Scaling factor for the elevation value
    - *Multidirectional*: Specify if multidirectional hillshading is to be used
    - *Azimuth*: The azimuth of the light source
    - *Altitude*: The elevation angle of the light source

Белешка

A shortcut to *Global map shading* properties is available through the *Layer Styling* panel.

5.3.15 Sensors Properties

The *Sensors* tab is used to configure sensors and toggle their connection status. When active, sensors will passively collect data in the background and make available their latest data to expressions and python scripts.

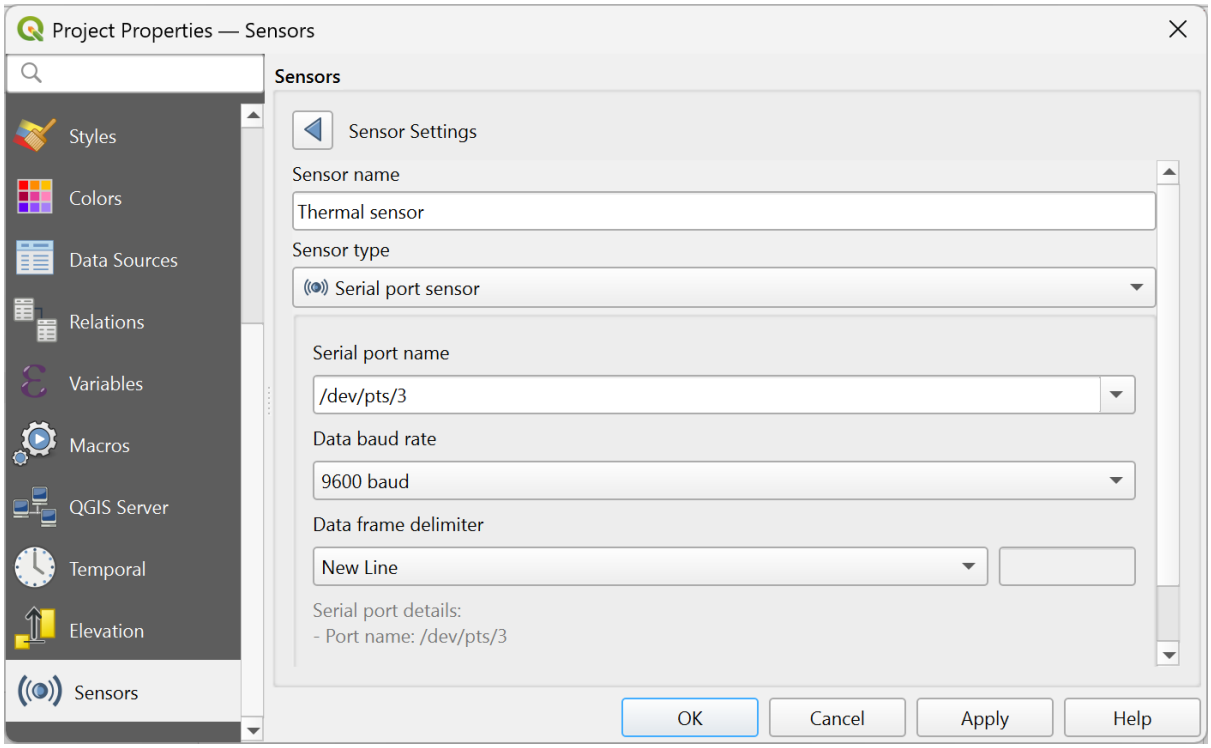


Слика5.42: Project Sensors tab

To add a new sensor, click on the button. A settings sub-panel will open and allow you to configure:

- the *Sensor name*: used to retrieve sensor values in expressions and python scripts;
- the *Sensor type*: TCP, UDP, serial port, etc.; and
- additional type-specific details (e.g., host name and port).

  For serial port sensors, comboboxes with the system's available serial ports and connection baud rate are available for their selection. A *Data frame delimiter* can be used to parse returned data. In case multiple data frames are found in the last chunk of data pushed by the serial port, all the data frames are returned. You can then use an expression to further split the returned sensor last data value into individual frames (e.g., using *string_to_array* function).



Слика5.43: Sensor Settings sub-panel

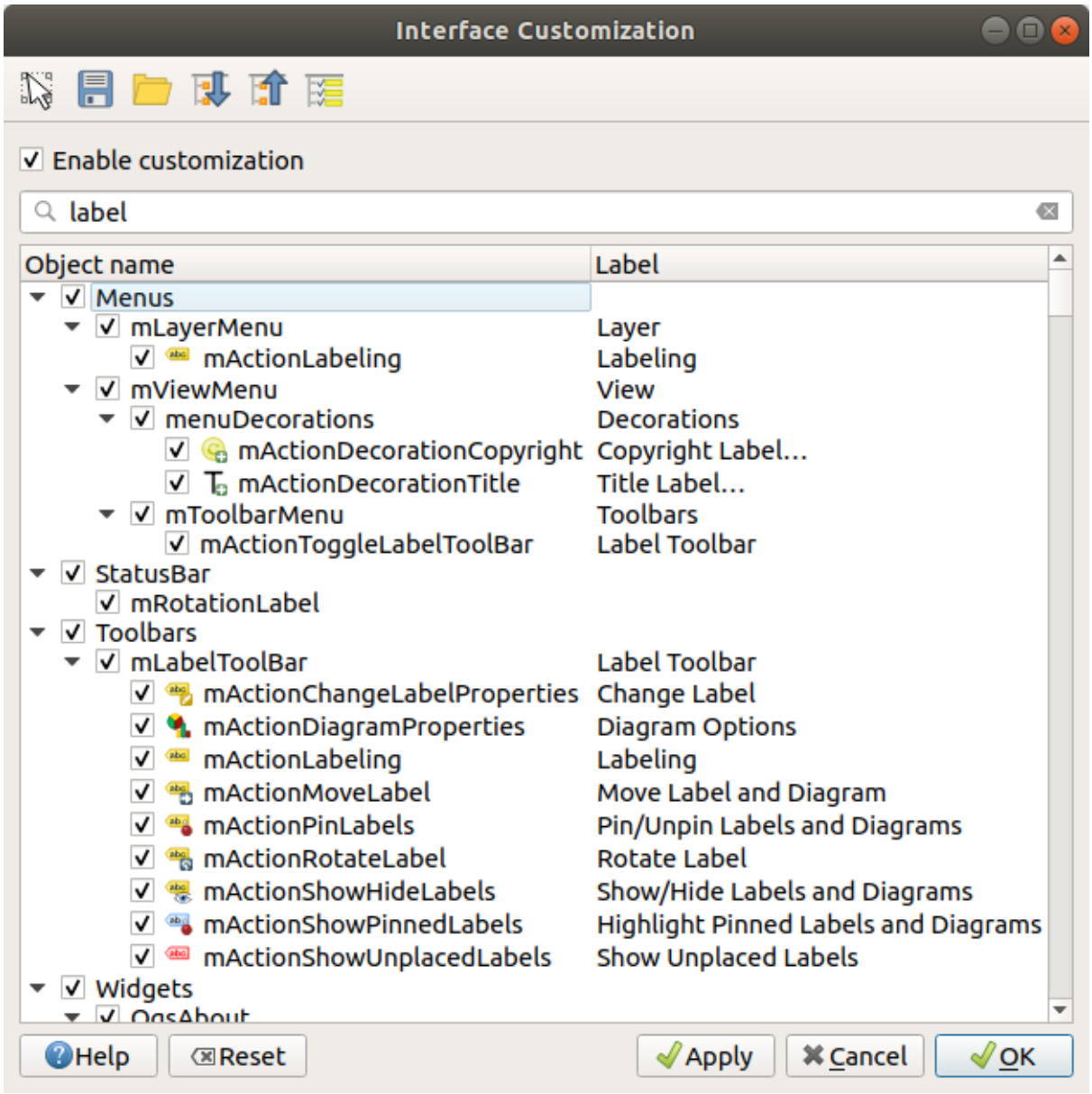
Once a sensor is configured, you can use the Start button to connect the sensor. Once active, the last collected data will be displayed in the sensors table's *Last value* column.

5.4 Customization

The *Customization* dialog lets you (de)activate almost every element in the QGIS user interface. This can be very useful if you want to provide your end-users with a 'light' version of QGIS, containing only the icons, menus or panels they need.

Белешка

Before your changes are applied, you need to restart QGIS.



Слика5.44: The Customization dialog

Ticking the *Enable customization* checkbox is the first step on the way to QGIS customization. This enables the toolbar and the widget panel from which you can uncheck and thus disable some GUI items.

The configurable item can be:

- a **Menu** or some of its sub-menus from the *Traka menija*
- a whole **Panel** (see *Paneli i trake sa alatkama*)
- the **Status bar** described in *Statusna traka* or some of its items
- a **Toolbar**: the whole bar or some of its icons
- or any **widget** from any dialog in QGIS: label, button, combobox…

With  Switch to catching widgets in main application, you can click on an item in QGIS interface that you want to be hidden and QGIS automatically unchecks the corresponding entry in the Customization dialog. You can also use the *Search* box to find items by their name or label.

Once you setup your configuration, click *Apply* or *OK* to validate your changes. This configuration becomes the one used by default by QGIS at the next startup.

The modifications can also be saved in a `.ini` file using  Save To File button. This is a handy way to share a common QGIS interface among multiple users. Just click on  Load from File from the destination computer in order to import the `.ini` file. You can also run *command line tools* and save various setups for different use cases as well.

Савет

Easily restore predefined QGIS

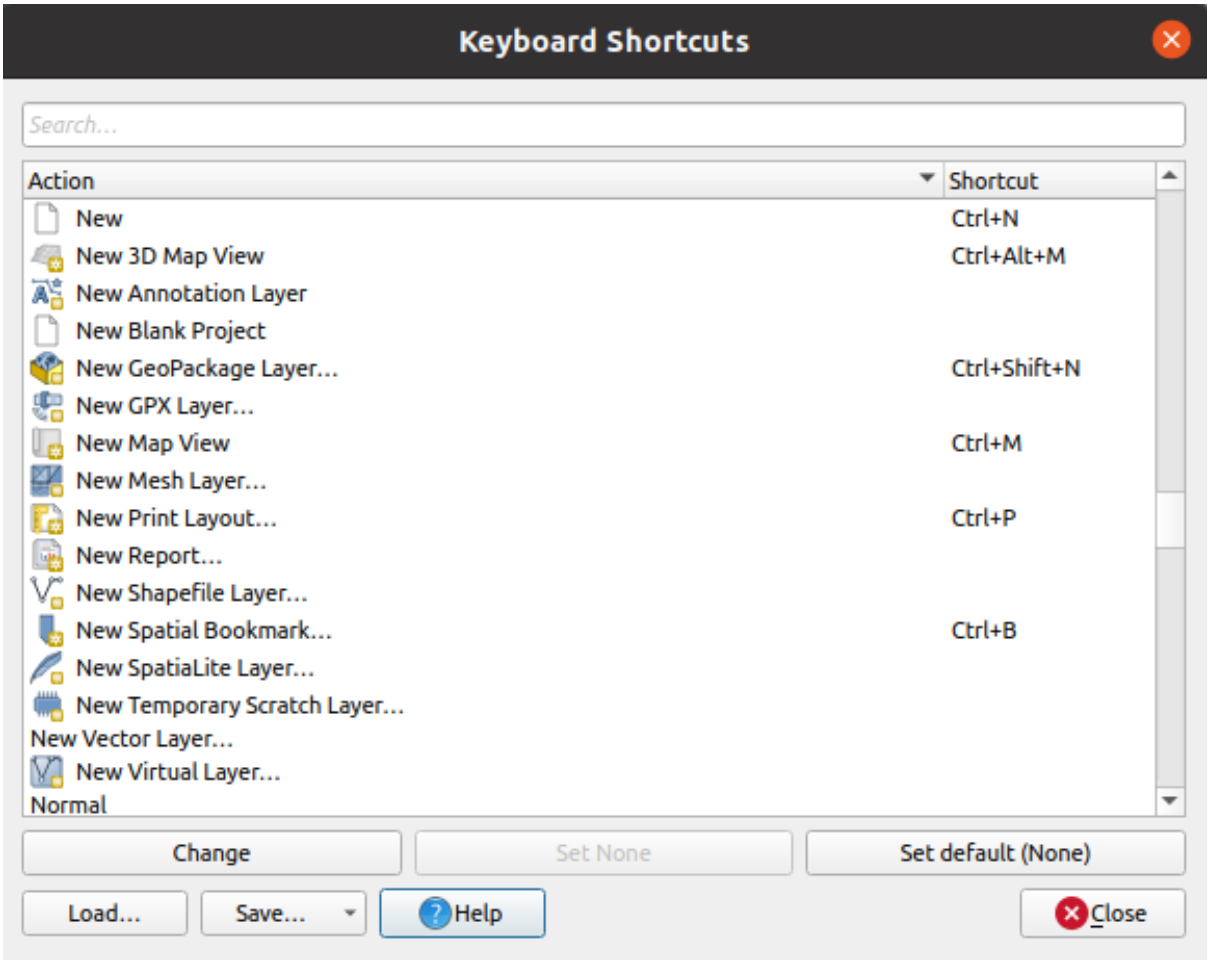
The initial QGIS GUI configuration can be restored by one of the methods below:

- unchecking  *Enable customization* option in the Customization dialog or click the  Check All button
- pressing the *Reset* button in the **Settings** frame under *Settings* ► *Options* menu, *System* tab
- launching QGIS at a command prompt with the following command line `qgis --nocustomization`
- setting to `false` the value of *UI* ► *Customization* ► *Enabled* variable under *Settings* ► *Options* menu, *Advanced* tab (see the *warning*).

In most cases, you need to restart QGIS in order to have the change applied.

5.5 Keyboard shortcuts

QGIS provides default keyboard shortcuts for many features. You can find them in section *Traka menija*. Additionally, the menu option *Settings* ►  *Keyboard Shortcuts...* allows you to change the default keyboard shortcuts and add new ones to QGIS features.



Слика5.45: Define shortcut options

Configuration is very simple. Use the search box at the top of the dialog to find a particular action, select it from the list and click on :

- *Change* and press the new combination you want to assign as new shortcut
- *Set None* to clear any assigned shortcut
- or *Set Default* to backup the shortcut to its original and default value.

Proceed as above for any other tools you wish to customize. Once you have finished your configuration, simply *Close* the dialog to have your changes applied. You can also *Save* the changes either as an `.XML` file with only the User Shortcuts or with all Shortcuts or as an `.PDF` file with all Shortcuts and *Load* them into another QGIS installation.

5.6 Running QGIS with advanced settings

5.6.1 Command line and environment variables

We've seen that *launching QGIS* is done as for any application on your OS. QGIS provides command line options for more advanced use cases (in some cases you can use an environment variable instead of the command line option). To get a list of the options, enter `qgis --help` on the command line, which returns:

```
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
```

(наставак на следећој страни)

(настављено са претходне стране)

```
       [-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
```

Савет

Example Using command line arguments

You can start QGIS by specifying one or more data files on the command line. For example, assuming you are in the `qgis_sample_data` directory, you could start QGIS with a vector layer and a raster file set to load on startup using the following command: `qgis ./raster/landcover.img ./gml/lakes.gml`

`--version`

This option returns QGIS version information.

`--snapshot`

This option allows you to create a snapshot in PNG format from the current view. This comes in handy when you have many projects and want to generate snapshots from your data, or when you need to create snapshots of the same project with updated data.

Currently, it generates a PNG file with 800x600 pixels. The size can be adjusted using the `--width` and `--height` arguments. The filename can be added after `--snapshot`. For example:

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

`--width`

This option returns the width of the snapshot to be emitted (used with `--snapshot`).

`--height`

This option returns the height of the snapshot to be emitted (used with `--snapshot`).

`--lang`

Based on your locale, QGIS selects the correct localization. If you would like to change your language, you can specify a language code. For example, `qgis --lang it` starts QGIS in Italian localization.

`--project`

Starting QGIS with an existing project file is also possible. Just add the command line option `--project` followed by your project name and QGIS will open with all layers in the given file loaded.

`--extent`

To start with a specific map extent use this option. You need to add the bounding box of your extent in the following order separated by a comma:

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

This option probably makes more sense when paired with the `--project` option to open a specific project at the desired extent.

`--nologo`

This option hides the splash screen when you start QGIS.

`--noversioncheck`

Skip searching for a new version of QGIS at startup.

--noplugins

If you have trouble at start-up with plugins, you can avoid loading them at start-up with this option. They will still be available from the Plugins Manager afterwards.

--nocustomization

Using this option, any existing *GUI customization* will not be applied at startup. This means that any hidden buttons, menu items, toolbars, and so on, will show up on QGIS start up. This is not a permanent change. The customization will be applied again if QGIS is launched without this option.

This option is useful for temporarily allowing access to tools that have been removed by customization.

--skipbadlayers

Using this option, you can avoid QGIS prompting the *Handle Unavailable Layers* dialog at startup. The project file is loaded, with missing layers kept as unavailable. More details on the topic at *Handling broken file paths*.

--customizationfile

Using this option, you can define a UI customization file, that will be used at startup.

--globalsettingsfile

The equivalent environment variable is `QGIS_GLOBAL_SETTINGS_FILE`.

Using this option, you can specify the path for a Global Settings file (`.ini`), also known as the Default Settings. The settings in the specified file replace the original inline default ones, but the user profiles' settings will be set on top of those.

QGIS looks for the default global settings file in the following order and only the first found file will be used:

- path specified by the commandline parameter
- path defined by the environment variable
- the AppDataLocation folder, where persistent application data can be stored; it is managed by the user or system administrator and is not touched by installer and does not require any additional setup like passing commandline parameters or settings environment variable. Depending on the OS, it is:
  - `$HOME/.local/share/QGIS/QGIS3/`
  - `C:\Users\<username>\AppData\Roaming\QGIS\QGIS3\`
  - `$HOME/Library/Application Support/QGIS/QGIS3/`
- the installation directory, i.e., `your_QGIS_package_path/resources/qgis_global_settings.ini`.

Presently, there's no way to specify a file to write settings to; therefore, you can create a copy of an original settings file, rename, and adapt it.

Setting the `qgis_global_setting.ini` file path to a network shared folder, allows a system administrator to change global settings and defaults in several machines by only editing one file.

--authdbdirectory

This option is similar to `--globalsettingsfile`, but defines the path to the directory where the authentication database will be stored and loaded.

--code

This option can be used to run a given python file directly after QGIS has started.

For example, when you have a python file named `load_alaska.py` with following content:

```
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)
```

Assuming you are in the directory where the file `load_alaska.py` is located, you can start QGIS, load the raster file `landcover.img` and give the layer the name 'Alaska' using the following command:

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

--py-args

This option allows to pass arguments to the script provided via the *–code* parameter and more generally to each python execution. After `--py-args`, it is not possible to pass any other parameter than – to QGIS. All arguments after `--py-args` till `--` are passed over to the python interpreter and ignored by QGIS.

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

In the above code, `test.py` will have this content in `sys.argv`: ['/home/user/test.py', '–specialScriptArguments', 'a text arg', 'and another arg']. `layer1` and `layer2` will be normally handled by QGIS as layers to load.

--defaultui

On load, **permanently resets** the user interface (UI) to the default settings. This option will restore the panels and toolbars visibility, position, and size. Unless it's changed again, the default UI settings will be used in the following sessions.

Notice that this option doesn't have any effect on *GUI customization*. Items hidden by GUI customization (e.g., the status bar) will remain hidden even using the `--defaultui` option. See also the `--nocustomization` option.

--hide-browser

On load, hides the *Browser* panel from the user interface. The panel can be enabled by right-clicking a space in the toolbars or using the *View* ► *Panels* (*Settings* ► *Panels* in Linux KDE).

Unless it's enabled again, the Browser panel will remain hidden in the following sessions.

--dxf-*

These options can be used to export a QGIS project into a DXF file. Several options are available:

- *–dxf-export*: the DXF filename into which to export the layers;
- *–dxf-extent*: the extent of the final DXF file;
- *–dxf-symbology-mode*: several values can be used here: `none` (no symbology), `symbollayer` (Symbol layer symbology), `feature` (feature symbology);
- *–dxf-scale-denom*: the scale denominator of the symbology;
- *–dxf-encoding*: the file encoding;
- *–dxf-map-theme*: choose a *map theme* from the layer tree configuration.

--take-screenshots

Takes screenshots for the user documentation. Can be used together with `--screenshots-categories` to filter which categories/sections of the documentation screenshots should be created (see QgsAppScreenShots::Categories).

--profile

Loads QGIS using a specific profile from the user's profile folder. If the named profile doesn't exist, it will be created. This option takes precedence over *user profile startup setting*.

--profiles-path

With this option, you can choose a path to load and save the profiles (user settings). It creates profiles inside a `{path}\profiles` folder, which includes settings, installed plugins, processing models and scripts, and so on.

This option allows you to, for instance, carry all your plugins and settings in a flash drive, or, for example, share the settings between different computers using a file sharing service.

The equivalent environment variable is `QGIS_CUSTOM_CONFIG_PATH`.

--version-migration

If settings from an older version are found (*e.g.*, the `.qgis2` folder from QGIS 2.18), this option will import them into the default QGIS profile.

--openclprogramfolder

Using this option, you can specify an alternative path for your OpenCL programs. This is useful for developers while testing new versions of the programs without needing to replace the existing ones.

The equivalent environment variable is `QGIS_OPENCL_PROGRAM_FOLDER`.

--

This option allows to pass following arguments as files to open in QGIS. They can be supported rasters or vectors files, QGIS layer definition files or project files.

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

The above code will start QGIS, open the `project` file and load `layer1` in it.

5.6.2 Deploying QGIS within an organization

If you need to deploy QGIS within an organization with a custom configuration file, first you need to copy/paste the content of the default settings file located in `your_QGIS_package_path/resources/qgis_global_settings.ini`. This file already contains some default sections identified by a block starting with `[]`. We recommend that you keep these defaults values and add your own sections at the bottom of the file. If a section is duplicated in the file, QGIS will take the last one from top to bottom.

You can change `allowVersionCheck=false` to disable the QGIS version check.

If you do not want to display the migration window after a fresh install, you need the following section:

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

If you want to add a custom variable in the global scope:

```
[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
```

Not all settings can be prepared from QGIS GUI, but to discover the possibilities of the settings `.INI` file, we suggest that you set the config you would like in QGIS Desktop and then search for it in your `.INI` file located in your profile using a text editor.

Finally, you need to set the environment variable `QGIS_GLOBAL_SETTINGS_FILE` to the path of your customized file.

In addition, you can also deploy files such as Python macros, color palettes, layout templates, project templates… either in the QGIS system directory or in the QGIS user profile.

- Layout templates must be deployed in the `composer_templates` directory.
- Project templates must be deployed in the `project_templates` directory.
- Custom Python macros must be deployed in the `python` directory.

ГЛАВА 6

Working with Projections

A Coordinate Reference System, or CRS, is a method of associating numerical coordinates with a position on the surface of the Earth. QGIS has support for approximately 7,000 standard CRSs, each with different use cases, pros and cons! Choosing an appropriate reference system for your QGIS projects and data can be a complex task, but fortunately QGIS helps guide you through this choice, and makes working with different CRSs as transparent and accurate as possible.

6.1 Overview of Projection Support

QGIS has support for approximately 7,000 known CRSs. These standard CRSs are based on those defined by the European Petroleum Search Group (EPSG) and the Institut Geographique National de France (IGNF), and are made available in QGIS through the underlying „Proj" projection library. Commonly, these standard projections are identified through use of an authority:code combination, where the authority is an organisation name such as „EPSG" or „IGNF", and the code is a unique number associated with a specific CRS. For instance, the common WGS 84 latitude/longitude CRS is known by the identifier `EPSG:4326`, and the web mapping standard CRS is `EPSG:3857`.

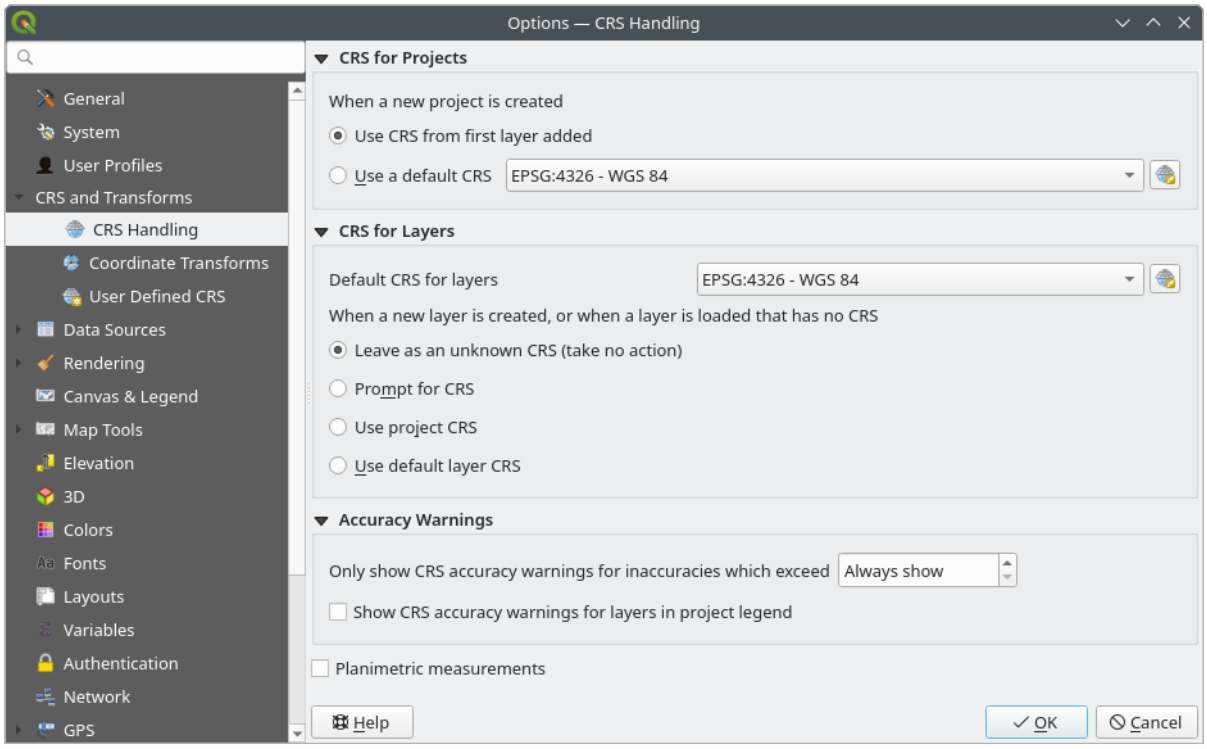
Custom, user-created CRSs are stored in a user CRS database. See section *Custom Coordinate Reference System* for information on managing your custom coordinate reference systems.

6.2 Layer Coordinate Reference Systems

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`.

Whenever a layer is loaded into QGIS, QGIS attempts to automatically determine the correct CRS for that layer. In some cases this is not possible, e.g. when a layer has been provided without retaining this information. You can configure QGIS behavior whenever it cannot automatically determine the correct CRS for a layer:

1. Open *Settings* ► *Options…* ► *CRS*



Слика6.1: The CRS tab in the QGIS Options Dialog

2. Under the *CRS for layers* group, set the action to do *when a new layer is created, or when a layer is loaded that has no CRS*. One of:

   - *Leave as unknown CRS (take no action)*: there will be no prompt to select a CRS when a layer without CRS is loaded, deferring CRS choice to a later time. Convenient when loading a lot of layers at once. Such layers will be identifiable in the *Layers* panel by the icon next to them. They'll also be un-referenced, with coordinates from the layer treated as purely numerical, non-earth values, i.e. the same behavior as all layers get when *a project is set to have no CRS*.
   - *Prompt for CRS*: it will prompt you to manually select the CRS. Selecting the correct choice is crucial, as a wrong choice will place your layer in the wrong position on the Earth's surface! Sometimes, accompanying metadata will describe the correct CRS for a layer, in other cases you will need to contact the original author of the data to determine the correct CRS to use.
   - *Use project CRS*
   - *Use default layer CRS*, as set in the *Default CRS for layers* combobox above.

Савет

To assign the same CRS to multiple layers that have no crs or have a wrong one in one operation:

1. Select the layers in the *Layers* panel
2. Press `Ctrl+Shift+C`. You could also right-click over one of the selected layers or go to *Layer* ► *Set CRS of layer(s)*
3. Find and select the right CRS to use
4. And press *OK*. You can confirm that it has been set correctly in the *Source* tab of the layers' properties dialog.

Note that changing the CRS in this setting does not alter the underlying data source in any way, rather it just changes how QGIS interprets the raw coordinates from the layer in the current QGIS project.

6.3 Project Coordinate Reference Systems

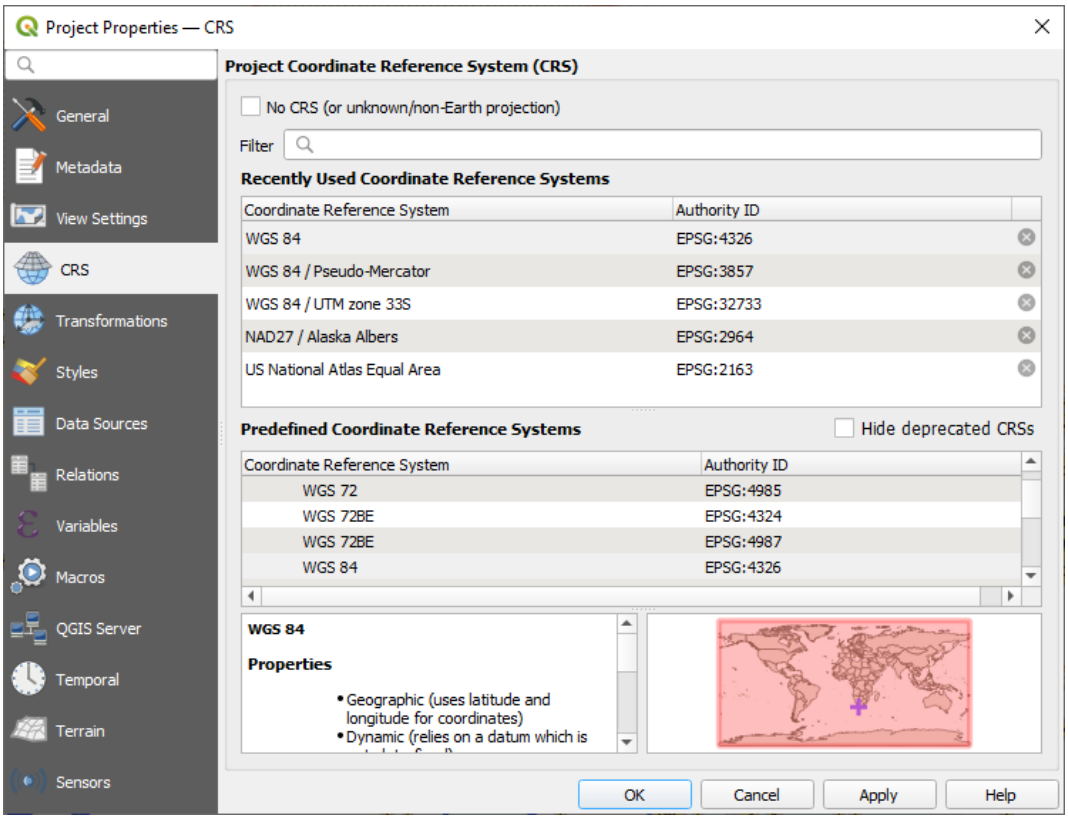
Every project in QGIS also has an associated Coordinate Reference System. The project CRS determines how data is projected from its underlying raw coordinates to the flat map rendered within your QGIS map canvas.

QGIS supports „on the fly" CRS transformation for both raster and vector data. This means that regardless of the underlying CRS of particular map layers in your project, they will always be automatically transformed into the common CRS defined for your project. Behind the scenes, QGIS transparently reprojects all layers contained within your project into the project's CRS, so that they will all be rendered in the correct position with respect to each other!

It is important to make an appropriate choice of CRS for your QGIS projects. Choosing an inappropriate CRS can cause your maps to look distorted, and poorly reflect the real-world relative sizes and positions of features. Usually, while working in smaller geographic areas, there will be a number of standard CRSs used within a particular country or administrative area. It's important to research which CRSs are appropriate or standard choices for the area you are mapping, and ensure that your QGIS project follows these standards.

By default, QGIS starts each new project using a global default projection. This default CRS is `EPSG:4326` (also known as „WGS 84"), and it is a global latitude/longitude based reference system. This default CRS can be changed via the *CRS for New Projects* setting in the *CRS* tab under *Settings* ► *Options...* (see Сл.6.1). There is an option to automatically set the project's CRS to match the CRS of the first layer loaded into a new project, or alternatively you can select a different default CRS to use for all newly created projects. This choice will be saved for use in subsequent QGIS sessions.

The project CRS can also be set through the *CRS* tab of the *Project* ► *Properties...* dialog. It will also be shown in the lower-right of the QGIS status bar.



Слика6.2: Project Properties Dialog

Available options are:

- *No CRS (or unknown/non-Earth projection)*: Checking this setting will disable ALL projection handling within the QGIS project, causing all layers and map coordinates to be treated as simple 2D Cartesian coordinates, with no relation to positions on the Earth's surface. It can be used to guess a layer CRS (based on its raw coordinates or when using QGIS for non earth uses like role-playing game maps, building mapping or microscopic stuff. In this case:
  - No reprojection is done while rendering the layers: features are just drawn using their raw coordinates.
  - The ellipsoid is locked out and forced to `None/Planimetric`.
  - The distance and area units, and the coordinate display are locked out and forced to „unknown units"; all measurements are done in unknown map units, and no conversion is possible.
- or an existing coordinate reference system that can be *geographic*, *projected* or *user-defined*. A preview of the CRS extent on earth is displayed to help you select the appropriate one. Layers added to the project are translated on-the-fly to this CRS in order to overlay them regardless of their original CRS. Use of units and ellipsoid setting are available and make sense and you can perform calculations accordingly.

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.

Савет

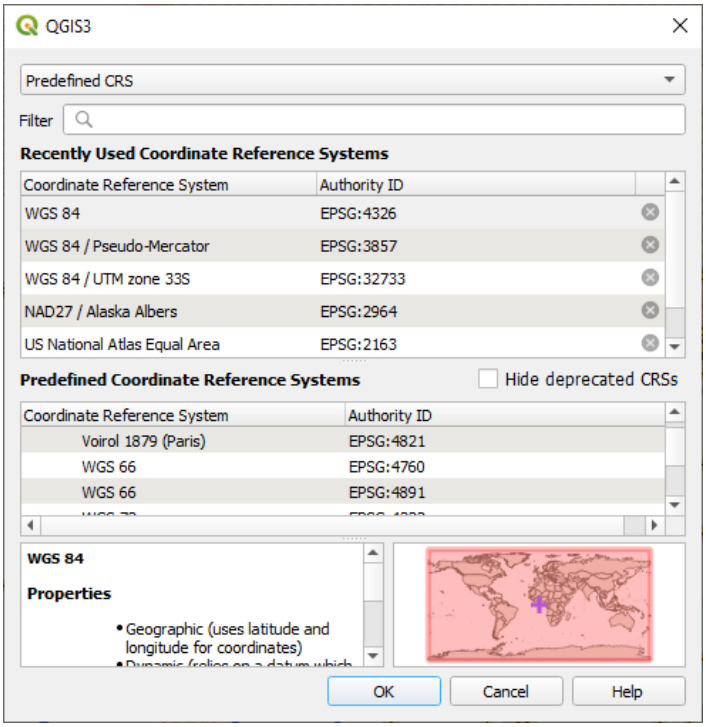
Setting the project CRS from a layer

You can assign a CRS to the project using a layer CRS:

1. In the *Layers* panel, right-click on the layer you want to pick the CRS
2. Select *Set project CRS from Layer*.

The project's CRS is redefined using the layer's CRS. Map canvas extent, coordinates display are updated accordingly and all the layers in the project are on-the-fly translated to the new project CRS.

6.4 Coordinate Reference System Selector



Слика6.3: CRS Selector

This dialog helps you assign a Coordinate Reference System to a project or a layer, provided a set of projection databases. Items in the dialog are:

- **Filter**: If you know the EPSG code, the identifier, or the name for a Coordinate Reference System, you can use the search feature to find it. Enter the EPSG code, the identifier or the name.
- **Recently used coordinate reference systems**: If you have certain CRSs that you frequently use in your everyday GIS work, these will be displayed in this list. Click on one of these items to select the associated CRS.
- **Coordinate reference systems of the world**: This is a list of all CRSs supported by QGIS, including Geographic, Projected and Custom coordinate reference systems. To define a CRS, select it from the list by expanding the appropriate node and selecting the CRS. The active CRS is preselected.
- **PROJ text**: This is the CRS string used by the PROJ projection engine. This text is read-only and provided for informational purposes.

If you want to remove a recently used CRS from the list that can be done in several ways:

- Press Remove from recently used CRS button on the right side.
- Press `Del` after selecting CRS that you want to remove.
- Right-click on used CRS and choose *Remove selected CRS from recently used CRS* to remove selected CRS or *Clear all recently used CRS* to remove all used CRS.

The CRS selector also shows a rough preview of the geographic area for which a selected CRS is valid for use. Many CRSs are designed only for use in small geographic areas, and you should not use these outside of the area they were designed for. The preview map shades an approximate area of use whenever a CRS is selected from the list. In addition, this preview map also shows an indicator of the current main canvas map extent.

6.5 Custom Coordinate Reference System

If QGIS does not provide the coordinate reference system you need, you can define a custom CRS. To define a CRS, select *Custom CRS...* from the *Settings* menu. Custom CRSs are stored in your QGIS user database. In addition to your custom CRSs, this database also contains your spatial bookmarks and other custom data.

Defining a custom CRS in QGIS requires a good understanding of the PROJ projection library. To begin, refer to „Cartographic Projection Procedures for the UNIX Environment - A User's Manual" by Gerald I. Evenden, U.S. Geological Survey Open-File Report 90-284, 1990 (available at https://pubs.usgs.gov/of/1990/of90-284/ofr90-284.pdf).

This manual describes the use of `proj` and related command line utilities. The cartographic parameters used with `proj` are described in the user manual and are the same as those used by QGIS.

The *Custom Coordinate Reference System Definition* dialog requires only two parameters to define a user CRS:

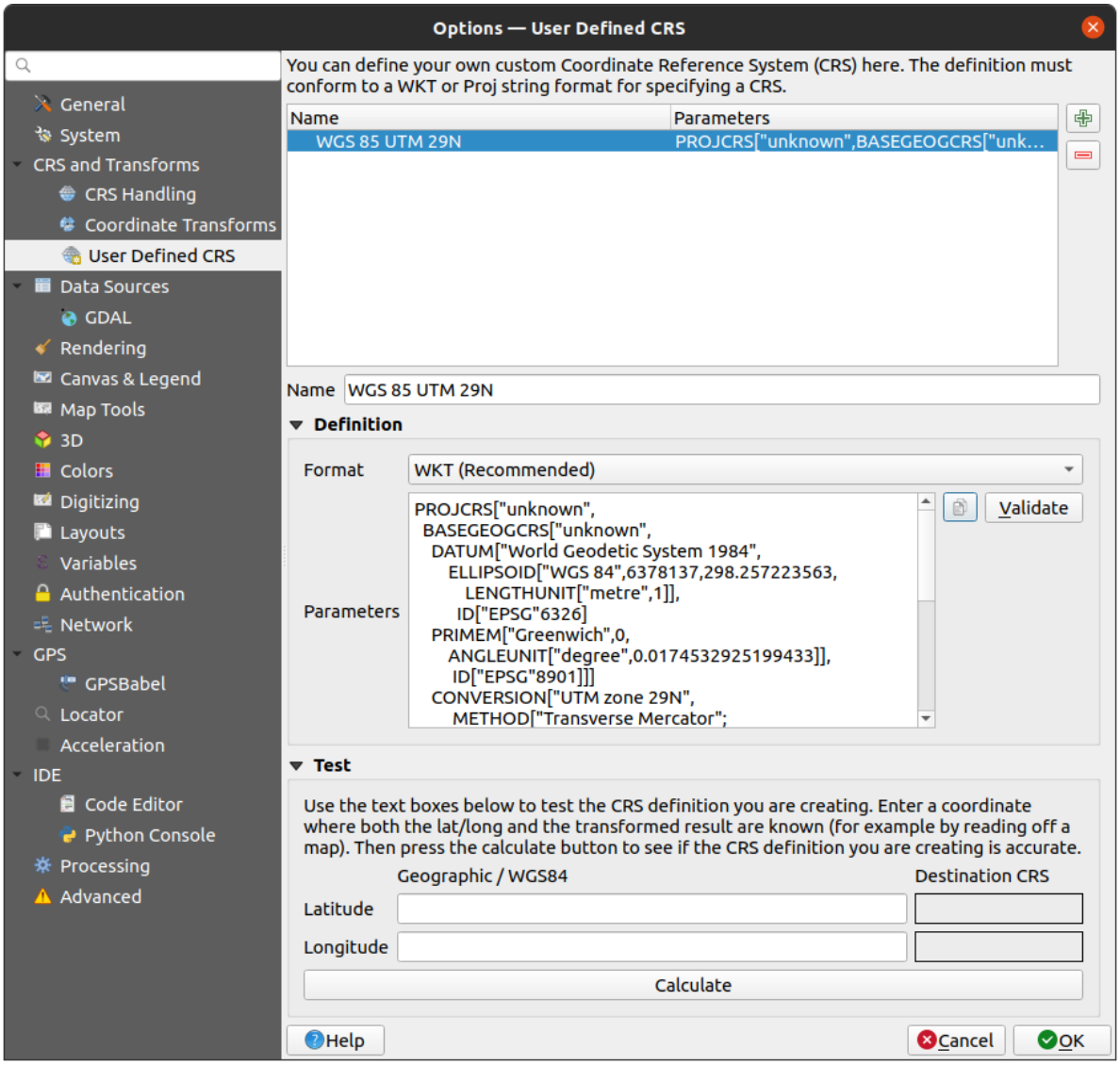
1. A descriptive name
2. The cartographic parameters in PROJ or WKT format

To create a new CRS:

1. Click the Add new CRS button
2. Enter a descriptive name
3. Select the format: it can be *Proj String* or *WKT*
4. Add the CRS *Parameters*.

> **Белешка**
>
> **Prefer storing the CRS definition in WKT format**
>
> Although both `Proj String` and `WKT` formats are supported, it's highly recommended to store projection definitions in the WKT format. Therefore, if the available definition is in the proj format, select that format, enter the parameters and then switch to WKT format. QGIS will convert the definition to the WKT format that you can later save.

5. Click *Validate* to test whether the CRS definition is an acceptable projection definition.



Слика6.4: Custom CRS Dialog

You can test your CRS parameters to see if they give sane results. To do this, enter known WGS 84 latitude and longitude values in *North* and *East* fields, respectively. Click on *Calculate*, and compare the results with the known values in your coordinate reference system.

6.5.1 Integrate an NTv2-transformation in QGIS

To integrate an NTv2 transformation file in QGIS you need one more step:

1. Place the NTv2 file (.gsb) in the CRS/Proj folder that QGIS uses (e.g. `C:\OSGeo4W64\share\proj` for windows users)
2. Add **nadgrids** (`+nadgrids=nameofthefile.gsb`) to the Proj definition in the *Parameters* field of the *Custom Coordinate Reference System Definition* (*Settings* ► *Custom Projections…*).

Слика6.5: Setting an NTv2 transformation

6.6 Datum Transformations

In QGIS, 'on-the-fly' CRS transformation is enabled by default, meaning that whenever you use layers with different coordinate systems QGIS transparently reprojects them to the project CRS. For some CRS, there are a number of possible transforms available to reproject to the project's CRS!

By default, QGIS will attempt to use the most accurate transformation available. However, in some cases this may not be possible, e.g. whenever additional support files are required to use a transformation. Whenever a more accurate transformation is available, but is not currently usable, QGIS will show an informative warning message advising you of the more accurate transformation and how to enable it on your system. Usually, this requires download of an external package of transformation support files, and extracting these to the `proj` folder under your QGIS *user profile* folder.

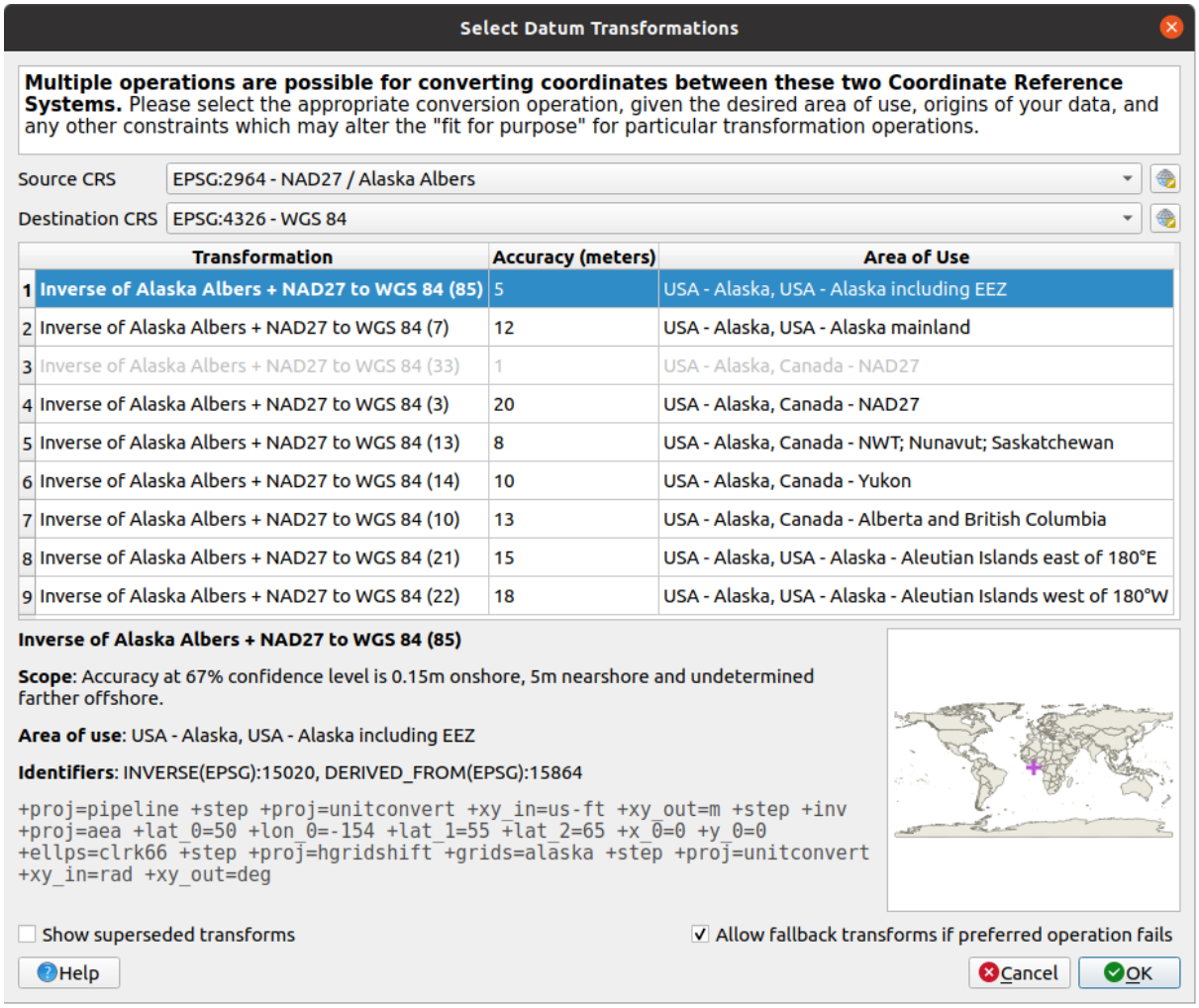
If desired, QGIS can also prompt you whenever multiple possible transformations can be made between two CRSs, and allow you to make an informed selection of which is the most appropriate transformation to use for your data.

This customization is done in the *Settings* ► *Options* ► *Transformations* tab menu under the *Default datum transformations* group:

- using *Ask for datum transformation if several are available*: when more than one appropriate datum transformation exist for a source/destination CRS combination, a dialog will automatically be opened prompting users to choose which of these datum transformations to use for the project. If the *Make default* checkbox is ticked when selecting a transformation from this dialog, then the choice is remembered and automatically applied to any newly created QGIS projects.
- or defining a list of appropriate datum transformations to use as defaults when loading a layer to a project or reprojecting a layer.

  Use the button to open the *Select Datum Transformations* dialog. Then:

  1. Choose the *Source CRS* of the layer, using the drop-down menu or the Select CRS widget.
  2. Provide the *Destination CRS* in the same way.
  3. A list of available transformations from source to destination will be shown in the table. Clicking a row shows details on the settings applied and the corresponding accuracy and area of use of the transformation.



Слика6.6: Selecting a preferred default datum transformation

In some cases a transformation may not be available for use on your system. In this case, the transformation will still be shown (greyed) in this list but can not be picked until you install the required package of transformation support. Usually, a button is provided to download and install the corresponding grid, which is then stored under the `proj` folder in the active *user profile* directory.

4. Find your preferred transformation and select it
5. Set whether you *Allow fallback transforms if preferred operation fails*
6. Click *OK*.

   A row is added to the table under *Default Datum Transformations* with information about the *Source CRS*, the *Destination CRS*, the *Operation* applied for the transformation and whether *Allow fallback Transforms* is enabled.

From now, QGIS automatically uses the selected datum transformations for further transformation between these two CRSs until you remove it ( ) from the list or change the entry ( ) in the list.

Datum transformations set in the *Settings* ► *Options* ► *Transformations* tab will be inherited by all new QGIS projects created on the system. Additionally, a particular project may have its own specific set of transformations specified via the *CRS* tab of the *Project properties* dialog (*Project* ► *Properties…*). These settings apply to the current project only.

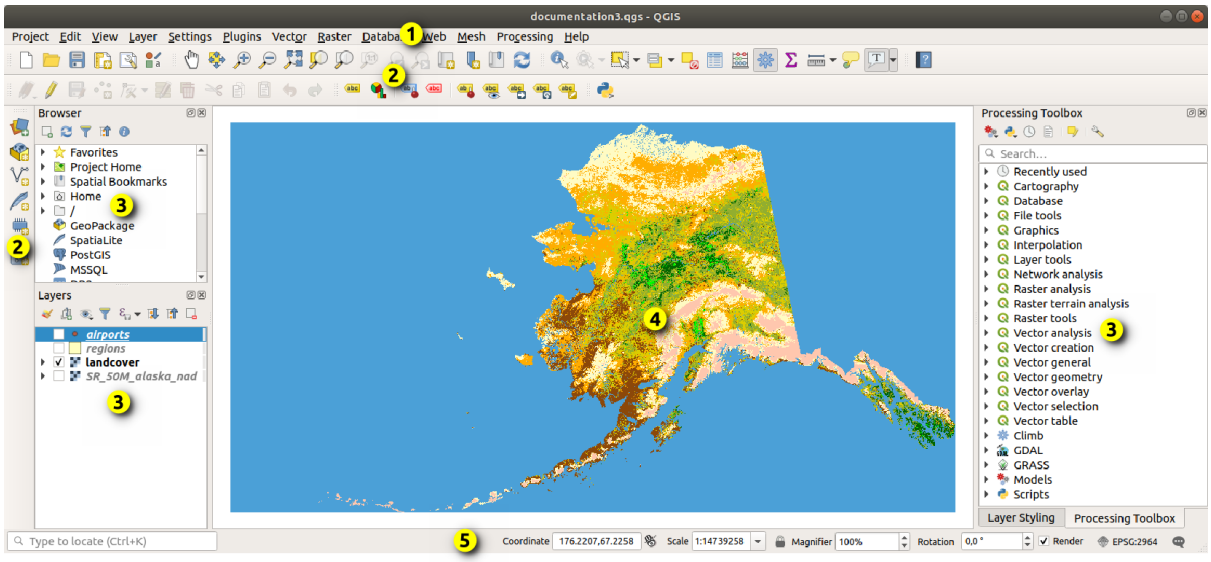
ГЛАВА 7

Vizuelizacija karata

Pravljenje karata je jedna od „praktičnih svrha“ QGIS-a. Prikazi karte su platna za vizuelizaciju prostornih slojeva učitanih u projektu, sa primenjenim raznim podešavanjima iscrtavanja. Mogu biti *2D* ili *3D* tipa ili iscrtani duž *profilne linije*, prikazivati različitu razmeru ili obuhvat, ili prikazivati različit skup učitanih slojeva zahvaljujući *temama karte*.

7.1 2D Map View

The 2D map view (also called **Map canvas**) is the central place where maps are displayed. QGIS opens by default with a single map view (called `main map`), showing layers in 2D, and tightly bound to the *Layers* panel. That window reflects the rendering (symbology, labeling, visibilities…) you applied to the loaded layers.



Слика7.1: QGIS GUI

7.1.1 Exploring the map view

When you add a layer (see e.g. *Opening Data*), QGIS automatically looks for its CRS. If a different CRS is set by default for the project (see *Project Coordinate Reference Systems*) then the layer extent is „on-the-fly" translated to that CRS, and the map view is zoomed to that extent if you start with a blank QGIS project. If there are already layers in the project, no map canvas resize is performed, so only features falling within the current map canvas extent will be visible.

Click on the map view and you should be able to interact with it, panning or zooming to different areas of the map. Dedicated tools are provided in the *Navigation Toolbar* and in the *View* menu, with handful shortcuts from the keyboard or the mouse buttons.

Табела7.1: Map canvas navigation tools

Alatka	Upotreba
Pan Map	• Single left click: the map is centered on the clicked point, at the same scale • Hold down the left mouse button and drag the map canvas.
Zoom In	• Single left click: the map is centered on the clicked point, while the scale gets doubled • Drag a rectangle on the map canvas with the left mouse button to zoom in to an area. • Hold the `Alt` key to switch to the Zoom Out tool.
Zoom Out	• Single left click: the map is centered on the clicked point, while the scale gets halved • Drag a rectangle on the map canvas with the left mouse button to zoom out from an area. • Hold the `Alt` key to switch to the Zoom In tool.
Pan Map to Selection	Pan the map to the selected features of all the selected layers in the *Layers* panel.
Zoom To Selection	Zoom to the selected features of all the selected layers in the *Layers* panel. *Also available in the layer contextual menu*
Zoom To Layer(s)	Zoom to the extent of all the selected layers in the *Layers* panel. *Also available in the layer contextual menu*
Zoom Full	Zoom to the extent of all the layers in the project or to the *project full extent*.
Zoom Last	Zoom the map to the previous extent in history.
Zoom Next	Zoom the map to the next extent in history.
Zoom to Native Resolution	Zoom the map to a level where one pixel of the active raster layer covers one screen pixel. *Also available in the layer contextual menu*
Mouse wheel	• Pan map: Hold and drag the mouse wheel. • Zoom: Roll the mouse wheel to zoom in or zoom out. With `Ctrl` key pressed while rolling the mouse wheel results in a finer zoom. • Press the back or forward button to browse the map canvas zoom history.
Keyboard	• Pan map: Hold down the `Space` key and move the mouse. Press the arrow keys to pan up, down, left and right. • Zoom in: Press `PgUp` or `Ctrl`+`+` • Zoom out: Press `PgDown` or `Ctrl`+`-` • Zoom to area: When certain map tools are active (Identify, Measure…), hold down `Shift` and drag a rectangle on the map to zoom to that area. Not compatible with active selection or edit tools.

Right-click over the map and you should be able to *Copy coordinates* of the clicked point in the map CRS, in WGS84 or in a custom CRS. The copied information can then be pasted in an expression, a script, text editor or

spreadsheet…

7.1.2 Controlling map rendering

By default, QGIS renders all visible layers whenever the map canvas is refreshed. The events that trigger a refresh of the map canvas include:

- changing the visibility of a layer
- modifying symbology of a visible layer
- adding a layer
- panning or zooming
- resizing the QGIS window

QGIS allows you to control the rendering process in a number of ways.

- at the *global level*
- per layer, using e.g. the *scale dependent rendering*
- or with dedicated tools in the GUI.

To stop the map drawing, press the `Esc` key. This will halt the refresh of the map canvas and leave the map partially drawn. It may however take a bit of time after pressing `Esc` for the map drawing to halt.

To suspend rendering, click the *Render* checkbox in the bottom-right corner of the status bar. When *Render* is unchecked, QGIS does not redraw the canvas in response to any of the usual triggers mentioned earlier. Examples of when you might want to suspend rendering include:

- adding many layers and symbolizing them prior to drawing
- adding one or more large layers and setting scale dependency before drawing
- adding one or more large layers and zooming to a specific view before drawing
- any combination of the above

Checking the *Render* checkbox enables rendering and causes an immediate refresh of the map canvas.

7.1.3 Time-based control on the map canvas

QGIS can handle temporal control on loaded layers, i.e. modify the map canvas rendering based on a time variation. To achieve this, you need:

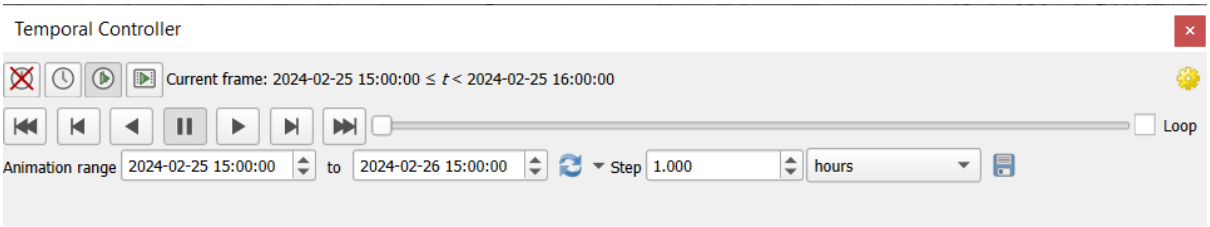
1. Layers that have dynamic temporal properties set. QGIS supports temporal control for different data providers, with custom settings. It's mainly about setting the time range in which the layer would display:
   - *raster layers*: controls whether to display or not the layer.
   - *WMTS layers*: controls whether the data should be rendered based on a static time range or following a dynamic temporal range
   - *vector layers*: features are filtered based on time values associated to their attributes
   - *mesh layers*: displays dynamically the active dataset groups values

   When dynamic temporal options are enabled for a layer, an icon is displayed next to the layer in the *Layers* panel to remind you that the layer is temporally controlled. Click the icon to update the temporal settings.
2. Enable the temporal navigation of the map canvas using the *Temporal controller panel*. The panel is activated:
   - using the Temporal controller panel icon located in the *Map Navigation* toolbar
   - or from the *View* ► *Panels* ► *Temporal controller panel* menu

- or from the *View* ► *Data Filtering* ► *Temporal controller panel* menu

The temporal controller panel

The *Temporal controller* panel has the following modes:



Слика7.2: Temporal Controller Panel in navigation mode

- Turn off temporal navigation: all the temporal settings are disabled and visible layers are rendered as usual
- Fixed range temporal navigation: a time range is set and only layers (or features) whose temporal range overlaps with this range are displayed on the map.
- Animated temporal navigation: a time range is set, split into steps, and only layers (or features) whose temporal range overlaps with each frame are displayed on the map
- Animated movie: a total number of frames is set for the animation, with the map settings advancing at each step, displaying the progression frame by frame without any time-based filtering of data.
- Settings for general control of the animation
  - *Frames rate*: number of steps that are shown per second
  - *Cumulative range*: all animation frames will have the same start date-time but different end dates and times. This is useful if you wish to accumulate data in your temporal visualization instead of showing a 'moving time window' across your data.

Animating a temporal navigation

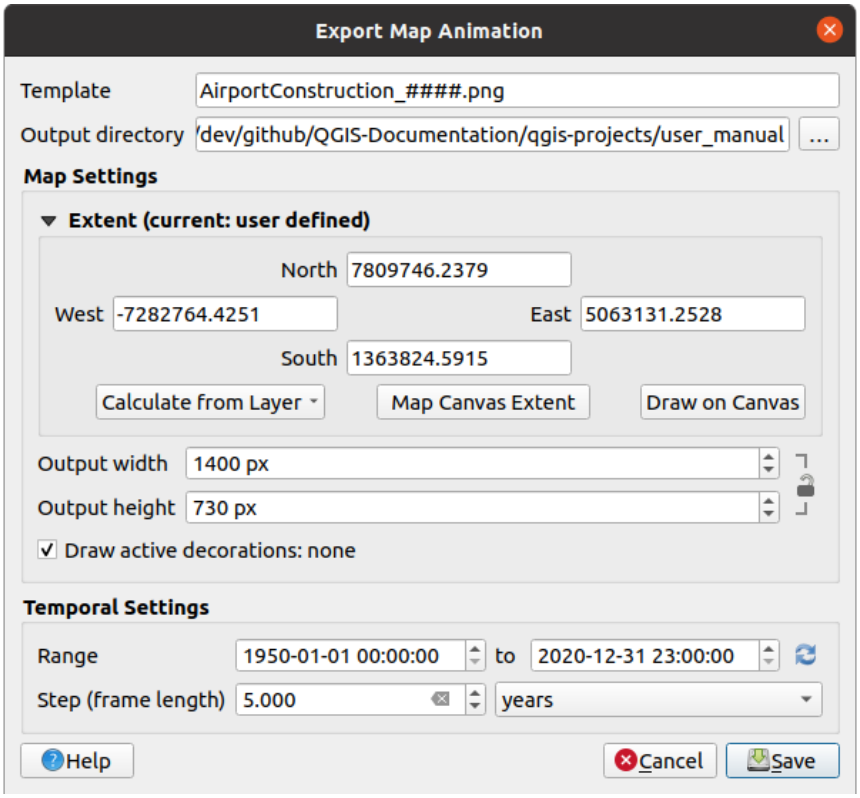
An animation is based on a varying set of visible layers at particular times within a time range. To create a temporal animation:

1. Toggle on the Animated temporal navigation, displaying the animation player widget
2. Enter the *Time range* to consider. Using the button, this can be defined as:
   - *Set to full range* of all the time enabled layers
   - *Set to preset project range* as defined in the *project properties*
   - *Set to single layer's range* taken from a time-enabled layer
3. Fill in the time *Step* to split the time range. Different units are supported, from `seconds` to `centuries`. A `source timestamps` option is also available as step: when selected, this causes the temporal navigation to step between all available time ranges from layers in the project. It's useful when a project contains layers with non-contiguous available times, such as a WMS-T service which provides images that are available at irregular dates. This option will allow you to only step between time ranges where the next available image is shown.
4. Click the button to preview the animation. QGIS will generate scenes using the layers rendering at the set times. Layers display depends on whether they overlap any individual time frame.

The animation can also be previewed by moving the time slider. Checking the *Loop* checkbox will repeatedly run the animation while clicking stops a running animation. A full set of video player buttons is available.

Horizontal scrolling using the mouse wheel (where supported) with the cursor on the map canvas will also allow you to navigate, or "scrub", the temporal navigation slider backwards and forwards.

5. Click the Export animation button if you want to generate a series of images representing the scene. They can be later combined in a video editor software:



Слика7.3: Exporting map canvas animation scenes to images

- The filename *Template*: the `####` are replaced with frame sequence number
- The *Output directory*
- Under *Map settings*, you can:
  - redefine the *spatial extent* to use
  - control the *Resolution* of the image (*Output width* and *Output height*)
  - *Draw active decorations*: whether active *decorations* should be kept in the output
- Under *Temporal settings*, you can redefine:
  - the time *Range* for the animation
  - the *Step (frame length)* in the unit of your choice

7.1.4 The Elevation Controller panel

Elevation Controller allows you to handle elevation z-range data in 2D maps. This option currently supports point cloud layers and raster DEMs. Use the panel to visualize and interact with elevation data in the map canvas.

Activate the *Elevation Controller* panel in the map canvas by clicking on the *Data Filtering* in the *View* menu. The *Elevation Controller* appears as a range slider on the left side of the map canvas. At the top of the slider, there is a Settings menu where you can:

- Set the *Fixed Range Size* value to lock the elevation range to a specific size. This setting is saved within the project and retained across sessions, ensuring consistency for your elevation filter. Use the *Invert Direction* option to reverse the elevation filter slider, so that it moves from high values at the bottom to low values at the top.
- *Set Elevation Range* to open the project elevation tab, where you can set dedicated lower and upper bounds for the elevation range
- *Disable Elevation Filter* to remove the elevation filter from the map canvas

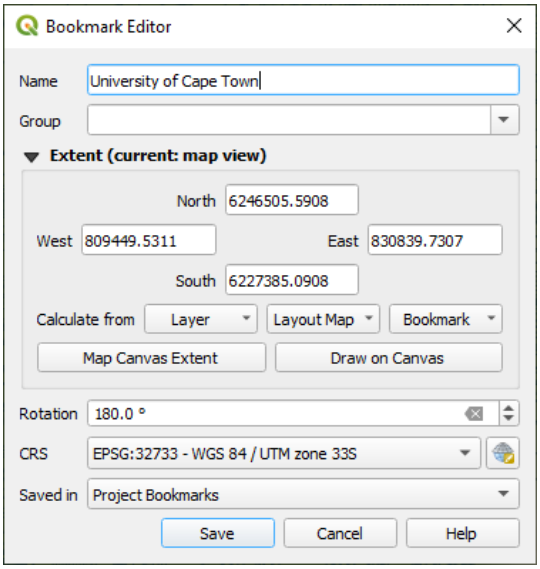
7.1.5 Bookmarking extents on the map

Spatial Bookmarks allow you to „bookmark" a geographic location and return to it later. By default, bookmarks are saved in the user's profile (as *User Bookmarks*), meaning that they are available from any project the user opens. They can also be saved for a single project (named *Project Bookmarks*) and stored within the project file, which can be helpful if the project is to be shared with other users.

Creating a Bookmark

To create a bookmark:

1. Zoom and pan to the area of interest.
2. Select the menu option *View* ► *New Spatial Bookmark...*, press `Ctrl+B` or right-click the *Spatial Bookmarks* entry in the *Browser* panel and select *New Spatial Bookmark*. The *Bookmark Editor* dialog opens.



Слика7.4: The Bookmark Editor Dialog

3. Enter a descriptive name for the bookmark
4. Enter or select a group name in which to store related bookmarks
5. Select the extent of the area you wish to save, using the *extent selector* widget
6. Change the map *Rotation*
7. Indicate the *CRS* to use for the extent

8. Select whether the bookmark will be *Saved in User Bookmarks* or *Project Bookmarks* (by default, this drop-down list is set to *User Bookmarks*)
9. Press *Save* to add the bookmark to the list

Note that you can have multiple bookmarks with the same name.

Working with Bookmarks

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.

You can perform the following tasks:

Табела7.2: Managing bookmark actions

Task	Spatial Bookmark Manager	Browser
Zoom to a Bookmark	Double-click on it, or select the bookmark and press the *Zoom to bookmark* button.	Double-click on it, drag and drop it to the map canvas, or right-click the bookmark and select *Zoom to Bookmark*.
Delete a bookmark	Select the bookmark and click the *Delete bookmark* button. Confirm your choice.	Right-click the bookmark and select *Delete Spatial Bookmark*. Confirm your choice.
Export bookmarks to XML	Click the *Import/Export Bookmarks* button and select *Export*. All the bookmarks (user or project) are saved in an xml file.	Select one or more folders (user or project) or subfolders (groups), then right-click and select *Export Spatial Bookmarks…*. The selected bookmark subset is saved.
Import bookmarks from XML	Click the *Import/Export Bookmarks* button and select *Import*. All bookmarks in the XML file are imported as user bookmarks.	Right-click the *Spatial Bookmarks* entry or one of its folders (user or project) or subfolders (groups) to determine where to import the bookmarks, then select *Import Spatial Bookmarks*. If performed on the *Spatial Bookmarks* entry, the bookmarks are added to *User Bookmarks*.
Edit bookmark	You can change a bookmark by changing the values in the table. You can edit the name, the group, the extent and if it is stored in the project or not.	Right-click the desired bookmark and select *Edit Spatial Bookmark…*. The *Bookmark Editor* will open, allowing you to redefine every aspect of the bookmark as if you were creating it for the first time. You can also drag and drop the bookmark between folders (user and project) and subfolders (groups).

You can manage bookmark actions by right-click on the desired bookmark in the *Spatial Bookmarks Manager*. You can also zoom to bookmarks by typing the bookmark name in the *locator*.

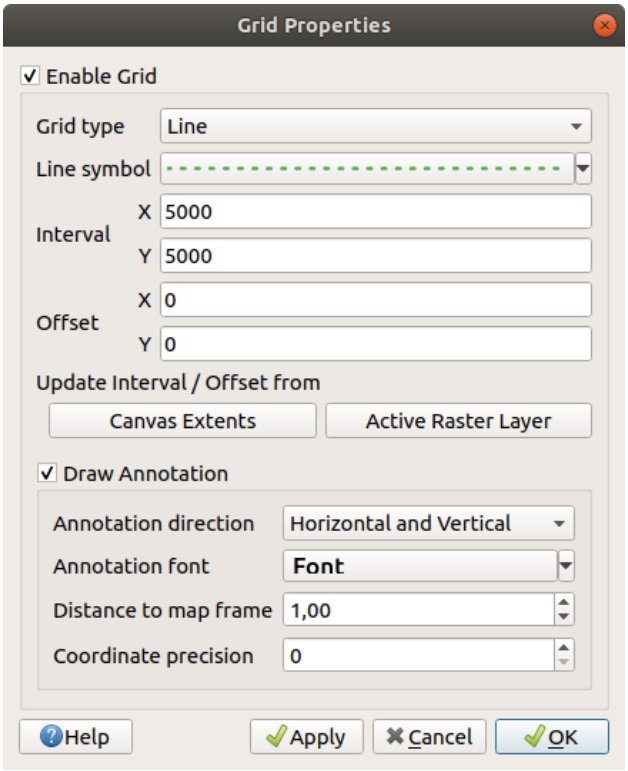
7.1.6 Decorating the map

Decorations include Grid, Title Label, Copyright Label, Image, North Arrow, Scale Bar and Layout Extents. They are used to 'decorate' the map by adding cartographic elements.

Grid

Grid allows you to add a coordinate grid and coordinate annotations to the map canvas.

1. Select menu option *View* ► *Decorations* ► *Grid…* to open the dialog.



Слика7.5: The Grid Dialog

2. Tick *Enable grid* and set grid definitions according to the layers loaded in the map canvas:
   - The *Grid type*: it can be *Line* or *Marker*
   - The associated *Line symbol* or *marker symbol* used to represent the grid marks
   - The *Interval X* and *Interval Y* between the grid marks, in map units
   - An *Offset X* and *Offset Y* distance of the grid marks from the bottom left corner of the map canvas, in map units
   - The interval and offset parameters can be set based on the:
     - *Canvas Extents*: generates a grid with an interval that is approximately 1/5 of the canvas width
     - *Active Raster Layer* resolution
3. Tick *Draw annotations* to display the coordinates of the grid marks and set:
   - The *Annotation direction*, ie how the labels would be placed relative to their grid line. It can be:
     - *Horizontal* or *Vertical* for all the labels
     - *Horizontal and Vertical*, ie each label is parallel to the grid mark it refers to

  - *Boundary direction*, ie each label follows the canvas boundary, and is perpendicular to the grid mark it refers to
- The *Annotation font* (text formatting, buffer, shadow…) using the *font selector widget*
- The *Distance to map frame*, margin between annotations and map canvas limits. Convenient when *exporting the map canvas* eg to an image format or PDF, and avoid annotations to be on the „paper" limits.
- The *Coordinate precision*

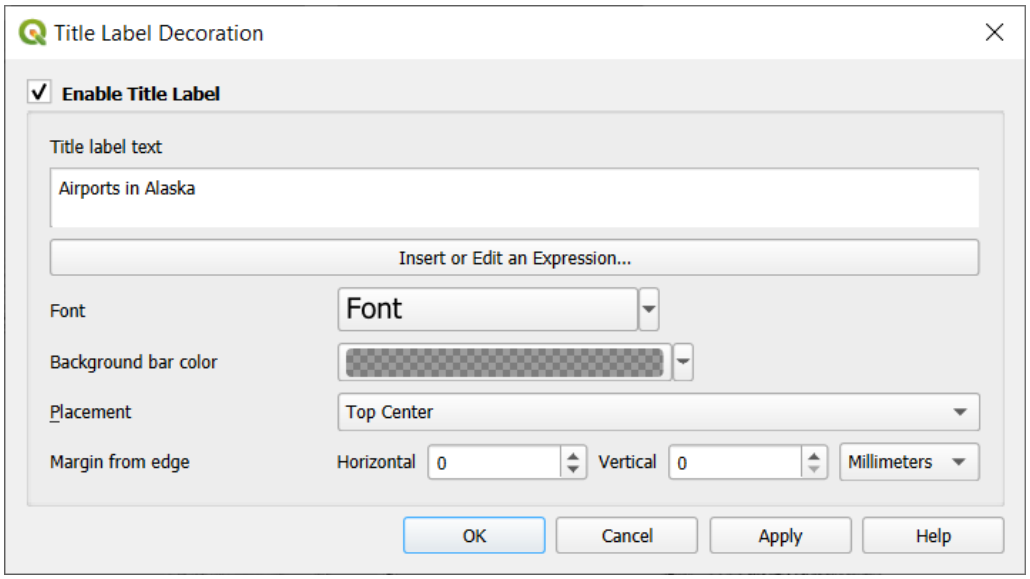
4. Click *Apply* to verify that it looks as expected or *OK* if you're satisfied.

Title Label

Title Label allows you to decorate your map with a **Title**.

To add a Title Label decoration:

1. Select menu option *View* ► *Decorations* ► *Title Label…* to open the dialog.



Слика7.6: The Title Decoration Dialog

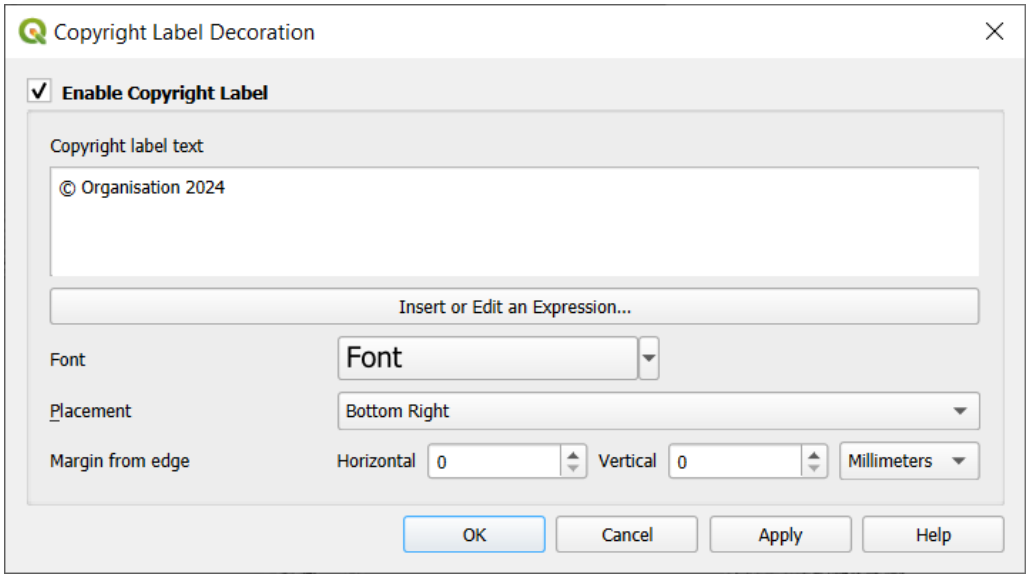
2. Make sure *Enable Title Label* is checked
3. Enter the title text you want to place on the map. You can make it dynamic using the *Insert or Edit an Expression…* button.
4. Choose the *Font* for the label using the *font selector widget* with full access to QGIS *text formatting* options. Quickly set the font color and opacity by clicking the black arrow to the right of the font combo box.
5. Select the *color* to apply to the title's *Background bar color*.
6. Choose the *Placement* of the label in the canvas: options are *Top left*, *Top Center* (default), *Top Right*, *Bottom left*, *Bottom Center* and *Bottom Right*.
7. Refine the placement of the item by setting a horizontal and/or vertical *Margin from Edge*. These values can be in **Millimeters** or **Pixels** or set as a **Percentage** of the width or height of the map canvas.
8. Click *Apply* to verify that it looks as expected or *OK* if you're satisfied.

Copyright Label

Copyright Label can be used to decorate your map with a **Copyright** label.

To add this decoration:

1. Select menu option *View* ► *Decorations* ► *Copyright Label…* to open the dialog.



Слика7.7: The Copyright Decoration Dialog

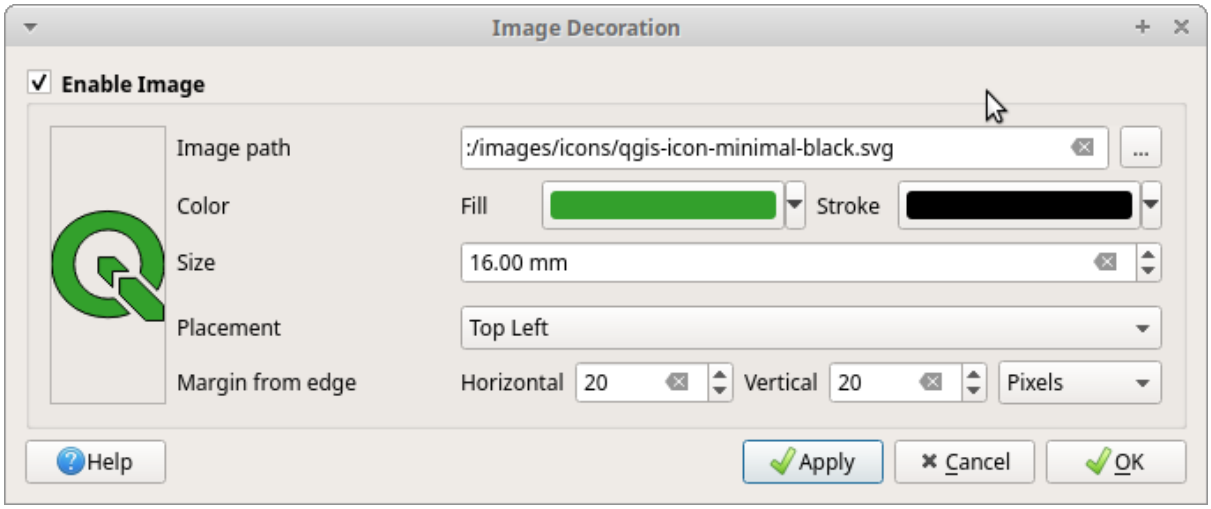
2. Make sure *Enable Copyright Label* is checked
3. Enter the copyright text you want to place on the map. You can make it dynamic using the *Insert or Edit an Expression…* button.
4. Choose the *Font* for the label using the *font selector widget* with full access to QGIS *text formatting* options. Quickly set the font color and opacity by clicking the black arrow to the right of the font combo box.
5. Choose the *Placement* of the label in the canvas: options are *Top left*, *Top Center*, *Top Right*, *Bottom left*, *Bottom Center*, and *Bottom Right* (default for Copyright decoration)
6. Refine the placement of the item by setting a horizontal and/or vertical *Margin from Edge*. These values can be in **Millimeters** or **Pixels** or set as a **Percentage** of the width or height of the map canvas.
7. Click *Apply* to verify that it looks as expected or *OK* if you're satisfied.

Image Decoration

Image allows you to add an image (logo, legend, ..) on the map canvas.

To add an image:

1. Select menu option *View* ► *Decorations* ► *Image…* to open the dialog.



Слика7.8: The Image Decoration Dialog

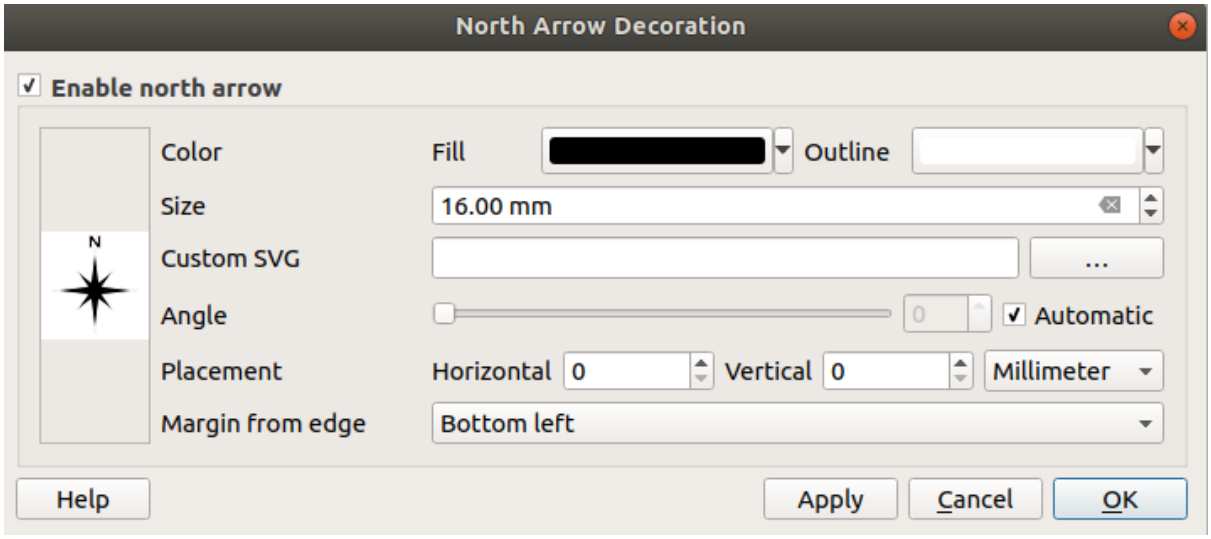
2. Make sure *Enable Image* is checked
3. Select a bitmap (e.g. png or jpg) or SVG image using the … Browse button
4. If you have chosen a parameter enabled SVG then you can also set a *Fill* or *Stroke* (outline) color. For bitmap images, the color settings are disabled.
5. Set a *Size* of the image in mm. The width of selected image is used to resize it to given *Size*.
6. Choose where you want to place the image on the map canvas with the *Placement* combo box. The default position is *Top Left*.
7. Set the *Horizontal* and *Vertical Margin from (Canvas) Edge*. These values can be set in **Millimeters**, **Pixels** or as a **Percentage** of the width or height of the map canvas.
8. Click *Apply* to verify that it looks as expected and *OK* if you're satisfied.

North Arrow

North Arrow allows you to add a north arrow on the map canvas.

To add a north arrow:

1. Select menu option *View* ► *Decorations* ► *North Arrow*… to open the dialog.



Слика7.9: The North Arrow Dialog

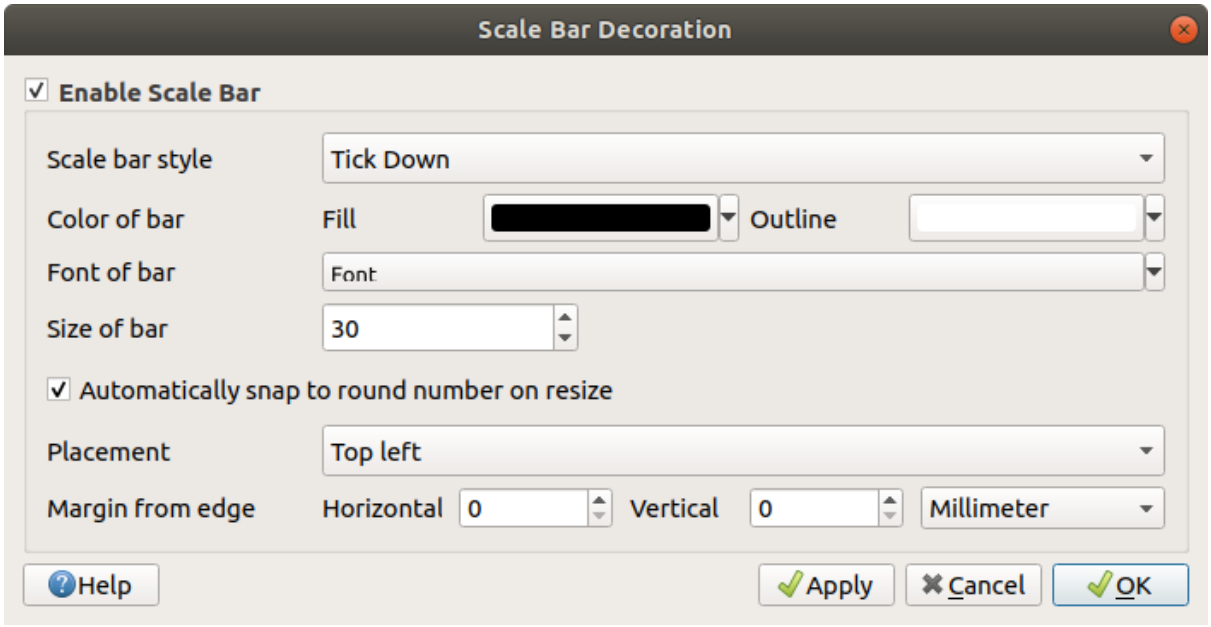
2. Make sure *Enable north arrow* is checked
3. Optionally change the color and size, or choose a custom SVG
4. Optionally change the angle or choose **Automatic** to let QGIS determine the direction
5. Optionally choose the placement from the Placement combo box
6. Optionally refine the placement of the arrow by setting a horizontal and/or vertical *Margin from (Canvas) Edge*. These values can be in **Millimeters** or **Pixels** or set as a **Percentage** of the width or height of the map canvas.
7. Click *Apply* to verify that it looks as expected and *OK* if you're satisfied.

Scale Bar

Scale Bar adds a simple scale bar to the map canvas. You can control the style and placement, as well as the labelling of the bar. The scale bar respects the active *project's distance unit* as defined in *Project properties* ► *General* ► *Units for distance measurement*.

To add a scale bar:

1. Select menu option *View* ► *Decorations* ► *Scale Bar…* to open the dialog



Слика7.10: The Scale Bar Dialog

2. Make sure *Enable scale bar* is checked
3. Choose a style from the *Scale bar style* combo box
4. Select the *Color of bar* by choosing a fill color (default: black) and an outline color (default: white). The scale bar fill and outline can be made opaque by clicking on the down arrow to the right of the color input.
5. Select the font for the scale bar from the *Font of bar* combo box
6. Set the *Size of bar* in the active unit
7. Optionally check *Automatically snap to round number on resize* to display easy-to-read values
8. Choose the placement from the *Placement* combo box
9. You can refine the placement of the item by setting a horizontal and/or vertical *Margin from (Canvas) Edge*. These values can be in **Millimeters** or **Pixels** or set as a **Percentage** of the width or height of the map canvas.
10. Click *Apply* to verify that it looks as expected or *OK* if you're satisfied.

Layout Extents

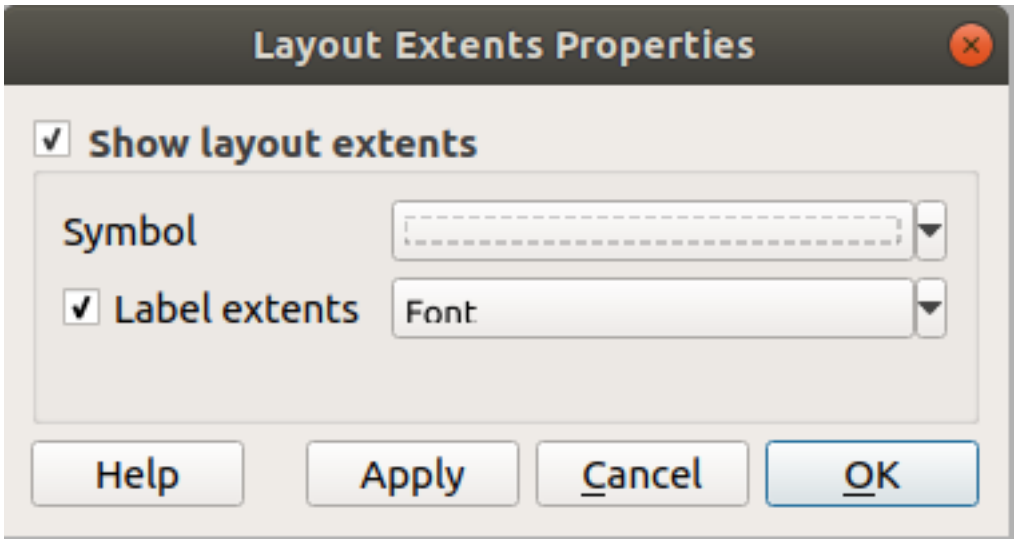
Layout Extents adds the extents of *map item(s)* in print layout(s) to the canvas. When enabled, the extents of all map items within all print layouts are shown using a lightly dotted border labeled with the name of the print layout and map item. You can control the style and labeling of the displayed layout extents. This decoration is useful when you are tweaking the positioning of map elements such as labels, and need to know the actual visible region of print layouts.



Слика7.11: Example of layout extents displayed in a QGIS project with two print layouts. The print layout named 'Sights' contains two map items, while the other print layout contains one map item.

To add layout extent(s):

1. Select *View* ► *Decorations* ► *Layout Extents* to open the dialog



Слика7.12: The Layout Extents Dialog

2. Make sure *Show layout extents* is checked.

3. Optionally change the symbol and labeling of the extents.
4. Click *Apply* to verify that it looks as expected and *OK* if you're satisfied.

Савет

Decorations Settings

When you save a QGIS project file, any changes you have made to Grid, North Arrow, Scale Bar, Copyright and Layout Extents will be saved in the project and restored the next time you load the project.

7.1.7 Annotation Tools

Annotations are another type of elements added onto the map canvas to provide additional information that can not be depicted by the rendered layers. Unlike *labels* that rely on attribute values stored in vector layers, annotations are independent details, stored within the project file itself.

Two families of annotations are available in QGIS:

- **Feature annotations**: they are actual georeferenced features of text, marker, line or polygon type stored within a special layer type called „annotation layer". They are tied to a particular geographic location, meaning that moving your map, changing the scale or changing projection won't cause your annotations to jump around the map. Rather, they'll be locked in place to the location you've drawn them.
- **Balloon annotations**: these are individual HTML or form annotations inside a bubble. They can be associated to any layer for their visibility and drawn on top of the map canvas. The size is dependent from the map canvas scale, and its position can be anchored.

Савет

Layout the map with annotations

You can print or export annotations with your map to various formats using:

- map canvas export tools available in the *Project* menu
- *print layout*, in which case you need to check *Draw map canvas items* in the corresponding map item properties

The *Annotations Toolbar* provides a set of tools to create and interact with both families of annotations.

Табела7.3: The Annotations Toolbar actions

Alatka	Upotreba	Scope
New Annotation Layer	Create a new layer to store annotations	*Feature annotations*
Main Annotation Layer Properties	Control settings of the Main Annotation Layer	
Modify Annotations	Select, move, resize and modify symbology properties of annotations	
Create Polygon Annotation	Create an annotation as a polygon feature	
Create Line Annotation	Create an annotation as a polyline feature	
Create Marker Annotation	Create an annotation as a point feature	
Create Text Annotation at Point	Create an annotation as a text label anchored at a point	
Create Text Annotation along Line	Create an annotation as a curved text along a line feature	

наставак на следећој страни

Табела 7.3 – настављено са претходне стране

Alatka	Upotreba	Scope
Create Text Annotation in Rectangle	Create an annotation as a text label in a rectangular text box	*Balloon annotations*
Create Picture Annotation	Create an annotation representing an `.SVG` or raster image	
HTML Annotation	Select and create annotation with a `HTML` formatted text	
Form Annotation	Select and create annotation showing attributes of a vector layer in a custom form file	

Белешка

Starting with QGIS 3.40, it is no longer possible to create balloon annotations of SVG or text type. When loading old projects, any of these annotations will automatically be converted to the newer picture and rectangular text annotation item types.

Feature Annotations

Feature annotations are stored in **annotation layers**. Unlike conventional layers, an annotation layer is available only in the current project and can contain features of different types (text, marker, line, polygon, picture). The layer has no attributes and no symbology associated, but instead each feature can be symbolized on an item-by-item basis, through *Layer Styling* panel.

Two types of annotation layer are available in QGIS:

- A common *Annotation Layer*: you can create one using the New Annotation Layer tool. It is listed in the *Layers* panel, allowing you to control its features' visibility, move it to show above or below particular layers in your map, like any common layer. Double-click the layer and you can access its properties.
- The *Main Annotation Layer*: By default, this is where annotations are stored when no annotation layer is available in the project or is selected at creation time. This layer is always drawn on the very top of your map and you won't see it listed in the *Layers* panel alongside the other layers in your project, meaning that its features are always visible. The *Main Annotation Layer Properties* entry on the *Annotations* toolbar helps you open its properties dialog.

Svojstva sloja

The properties dialog of an annotation layer provides the following tabs:

- *Information*: a read-only dialog representing an interesting place to quickly grab summarized information and metadata on the current layer. This may include the layer extent, count of items per annotation type and total count, CRS details, …
- *Source*: defines general settings for the annotation layer. You can:
  - Set a *Layer name* that will be used to identify the layer in the project (in the *Layers Panel*, with expressions, …)
  - Display the layer's *Assigned Coordinate Reference System (CRS)*: you can change the layer's CRS, selecting a recently used one in the drop-down list or clicking on Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the CRS applied to the layer is a wrong one or if none was applied.
- *Rendering*:
  - You can set the *Maximum (inclusive)* and *Minimum (exclusive)* scale, defining a range of scale in which features 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 *Visibility Scale Selector* for more information.

- *Linked layer*: allows you to optionally set a linked visibility layer for the annotation layer. If set, then the annotations will only be drawn when the linked layer is visible in the map.
- *Opacity*: You can make the underlying layer in the map canvas visible with this tool. Use the slider to adapt the visibility of your vector layer to your needs. You can also make a precise definition of the percentage of visibility in the menu beside the slider.
- *Blending mode* at the *Layer* level: You can achieve special rendering effects with these tools that you may previously only know from graphics programs. The pixels of your overlaying and underlying layers are mixed through the settings described in *Blending Modes*.
- Apply *paint effects* on all the layer features with the *Draw Effects* button.

Some of these options are accessible from the feature annotation *Symbology* properties.

Interacting with features

The Feature annotations have dedicated tools for creation depending on their type.

Табела7.4: Creating feature annotation

Family / based on	Annotation tool	Procedure
Geometrija	Create Polygon Annotation	This is like digitizing a point, line or polygon vector feature. All the usual shortcuts for creating features apply when creating annotation items. A line or polygon annotation is drawn by left-clicking once for each vertex, with a final right mouse click to complete the shape. Snapping can be enabled while you draw, you can use the *Advanced Digitizing Tools* to precisely place vertices, and even switch the *drawing tools* to for example the streaming mode for completely free-form shapes.
	Create Line Annotation	
	Create Marker Annotation	
Text	Create Text Annotation at Point	Left-click on the map canvas.
	Create Text Annotation along Line	Left-click to add vertices to the text basement line geometry, and right-click to end. As above, digitizing features capabilities are also available.
	Create Text Annotation In Rectangle	Draw the rectangle box of the text, by left-clicking to add a first corner and left-clicking again to add the opposite one.
Picture	Create Picture Annotation	Draw the rectangle box of the picture, by left-clicking to add a first corner and left-clicking again to add the opposite one. Then pick an image or SVG file from the file explorer.

Unlike common layers, an annotation layer does not need to be active before you select its features. Simply grab the Modify Annotations tool and you will be able to interact with any feature annotation:

- **Selection**: left-click on the annotation. By default, annotations are rendered in the order of creation, meaning that recent annotations will be placed on top of older. You may need to play with the Z-index property of features in order to select ones they would sit above.
- **Moving**: Left click on a selected annotation item to start moving it. A right-click or pressing `Esc` key cancels the move, while a second left click will confirm the move. The displacement can also be controlled pressing the cursor keys:
  - `Shift+key` for big movement

  - `Alt`+`key` for `1 px` movement

- **Geometry modification**: move over an annotation and purple squares are displayed on nodes of its underlying geometry. Left-click on the square, move and click again. For linear or polygonal annotations, double-clicking a segment will add a new vertex.
- **Delete**: Pressing the `Del` or `Backspace` key while an annotation is selected will delete that annotation
- *Change feature symbology*

Feature symbology

A selected annotation will display its *Symbology* properties in the *Layer styling* panel.

- For polygon, polyline and marker annotations, you can modify the appearance using full capabilities of the *symbol properties*
- For text-based annotations, an area with text editing tools helps you construct the string to display. It can be multiline, use HTML formatting and rely on QGIS expression functions. You can moreover modify the appearance using full capabilities of the *text format properties*.

  Depending on the type of annotation, additional options are available.

  - For text annotation at point, it is also possible to:
    - set the text *Alignment* (left, center or right of the anchor point)
    - configure whether the text should *Ignore map rotation* or *Rotate with map*. In both cases, a custom *Angle* can be set for the feature orientation.
  - For text annotation along a line, configure an *Offset from line* in the unit of your choice
  - For text annotation in rectangle, it is also possible to:
    - set the text *Horizontal alignment* (left, center, right or justify) and *Vertical alignment* (top, vertical center or bottom) in the rectangle box
    - configure for the rectangle, the *Margins* distance in the units of your choice, the *Frame* and *Background* colors.
- Picture annotations allow you to display a *Raster image* or an *SVG image*, using the *Remote or embedded file selector*. The placement and size of the picture can be set as:
  - *Scale dependent size*: the picture scales with the map, appearing larger when zooming in and smaller when zooming out. The size is defined by a single reference size in map units (e.g. meters or degrees, depending on the layer's CRS). With *Lock aspect ratio*, you can keep the annotation size proportional to the embedded picture size, or stretch the picture to fill the annotation extent.
  - *Fixed size*: the picture annotation maintains a constant size in screen or output units (e.g. pixels or millimeters), regardless of the map scale. The *Width* and *Height* of the annotation can be filled separatey or adjusted proportionally to the embedded picture size.
  - *Relative to map*: the picture annotation will always be rendered at the same position relative to the map canvas bounds, regardless of the map scale. The *Width* and *Height* of the annotation can be filled separatey or adjusted proportionally to the embedded picture size.

  Moreover, you can enable display of the *Frame* and *Background* of the rectangle box, and configure them using *fill symbols*.
- For text at point, text inside rectangle and picture annotations, you can *Show callout* when the picture or text anchor point is offset from its default placement. To create a callout for an annotation:
  1. Select the annotation item
  2. Move the anchor point, i.e., click the central and green `X` node and click at the new placement
  3. In the annotation properties, press the … button to configure the *callout properties*.

Савет

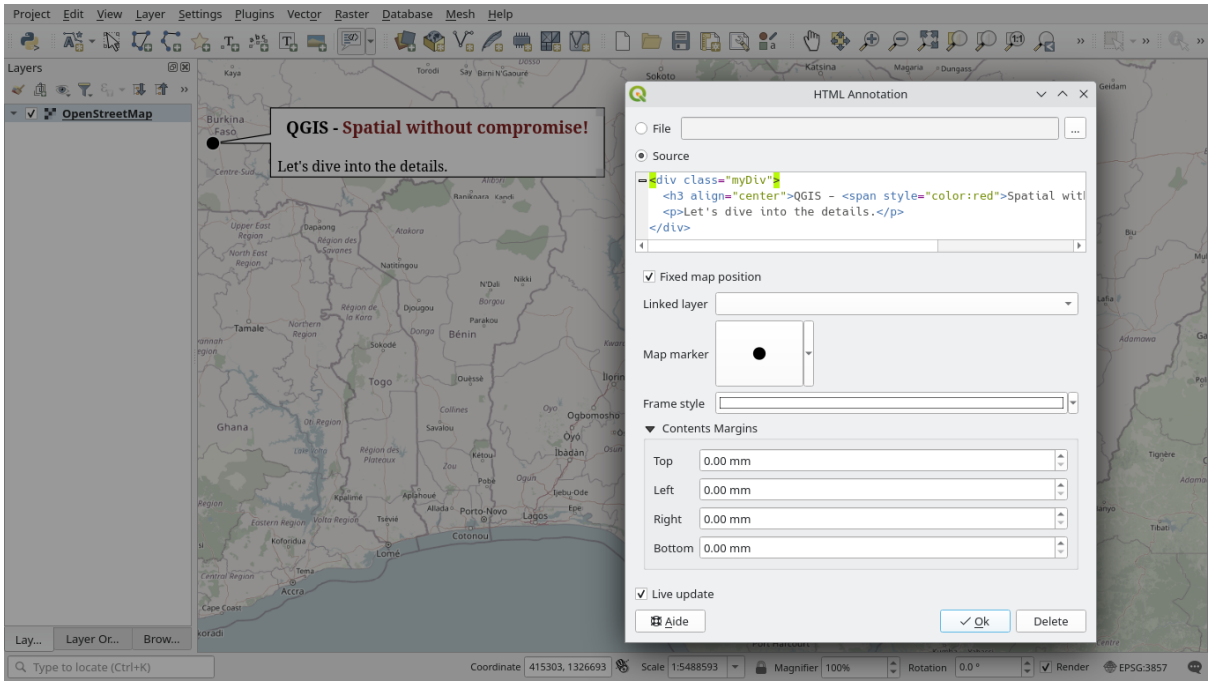
Applying a balloon callout to a text or picture annotation will make you get their old-style balloon annotation display.

- Configure a *Reference scale*: indicates the map scale at which symbol or text sizes which use paper-based units (such as millimeters or points) relate to. The sizes will be scaled accordingly whenever the map is viewed at a different scale. For instance, a line feature wide of 2mm at 1:2000 *Reference scale* will be rendered using 4mm when the map is viewed at 1:1000.
- Set a *Z-index*: a feature with a higher index is placed on top of features with lower index. A convenient setting for both feature display and selection.
- Modify some of the *Layer rendering* settings

Balloon annotations

You can add balloon annotations through *Edit* ► *Add Annotation* ► menu or from the *Annotations Toolbar*:

- 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.



Слика7.13: Example of a HTML balloon annotation

To add a balloon annotation, select the corresponding tool and click on the map canvas. An empty balloon is added. Double-click on it and a dialog opens with various options. This dialog is almost the same for all the annotation types:

- *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.
- *Linked layer*: associates the annotation with a map layer, making it visible only when that layer is visible.
- *Map marker*: using *QGIS symbols*, sets the symbol to display at the balloon anchor position (shown only when *Fixed map position* is checked).
- *Frame style*: sets the frame background color, transparency, stroke color or width of the balloon using QGIS symbols.
- *Contents margins*: sets interior margins of the annotation frame.
- *Live update* allows you to live preview your changes.

When a balloon annotation tool is active, you can also:

- Select an annotation
- Resize an annotation
- Move an annotation by map position (by dragging the map marker) or by moving only the balloon.
- Delete an annotation: select it and either press the `Del` or `Backspace` button, or double-click it and press the *Delete* button in the properties dialog.
- Right-click and in the contextual menu:
  - *Copy coordinate* of the annotation's map marker in various CRS
  - *Edit* properties of the annotation. Same as double-clicking the annotation.
  - *Delete* the annotation

7.1.8 Measuring

General information

QGIS provides four means of measuring geometries:

- interactive measurement tools
- measuring in the Field Calculator
- derived measurements in the *Identifying Features* tool
- the vector analysis tool: *Vector* ► *Geometry Tools* ► *Export/Add Geometry Columns*

Measuring works within projected coordinate systems (e.g., UTM) and unprojected data. The first three measuring tools behave equally to global project settings:

- Unlike most other GIS, the default measurement metric is ellipsoidal, using the ellipsoid defined in *Project* ► *Properties…* ► *General*. This is true both when geographic and projected coordinate systems are defined for the project.
- If you want to calculate the projected/planimetric area or distance using cartesian maths, the measurement ellipsoid has to be set to „None/Planimetric" (*Project* ► *Properties…* ► *General*). However, with a geographic (ie unprojected) CRS defined for the data and project, area and distance measurement will be ellipsoidal.

However, neither the identify tool nor the field calculator will transform your data to the project CRS before measuring. If you want to achieve this, you have to use the vector analysis tool: *Vector* ► *Geometry Tools* ► *Add Geometry Attributes*…. Here, measurement is planimetric, unless you choose the ellipsoidal measurement.

Measure length, areas, bearings and angles interactively

Click the icon in the Attribute toolbar to begin measurements. The down arrow near the icon switches between length, area, bearing or angle. The default unit used in the dialog is the one set in *Project* ► *Properties…* ► *General* menu.

For the *Measure Line* and the *Measure Area* the measurements can be done in *Cartesian* or *Ellipsoidal* measure.

Белешка

Configuring the measure tool

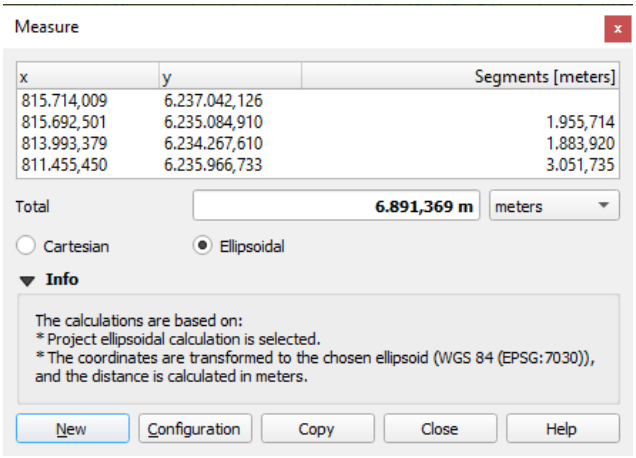
While measuring length or area, clicking the *Configuration* button at the bottom of the widget opens the *Settings* ► *Options* ► *Map Tools* menu, where you can select the rubberband color, the precision of the measurements and the unit behavior. You can also choose your preferred measurement or angle units, but keep in mind that those values are overridden in the current project by the selection made in the *Project* ► *Properties…* ► *General* menu, and by the selection made in the measurement widget.

All measuring modules use the snapping settings from the digitizing module (see section *Setting the snapping tolerance and search radius*). So, if you want to measure exactly along a line feature, or around a polygon feature, first set its layer snapping tolerance. Now, when using the measuring tools, each mouse click (within the tolerance setting) will snap to that layer.

The Measure Line measures distances between given points. The tool then allows you to click points on the map. Each segment length, as well as the total, shows up in the measure window. In the measure window, you will see coordinates for all your points and distances. Keep in mind that the first row will contain only coordinates, as it represents your starting point. Now it is possible to copy all your line measurements at once to the clipboard using the *Copy* button. Clicking the *Configuration* button you will access to *Measure Tool Copy Settings* where you can set up *copy options*. To stop measuring, click the right mouse button.

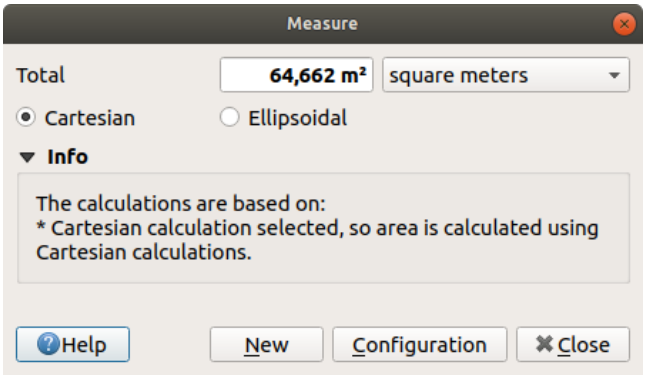
Note that you can use the drop-down list near the total to change the *measurement units* interactively while working with the measure tool. This unit is retained for the widget until a new project is created or another project is opened.

The *Info* section in the dialog explains how calculations are made according to the CRS settings available.



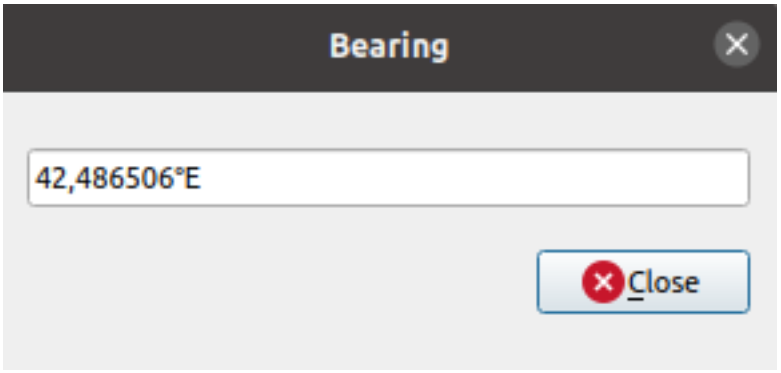
Слика7.14: Measure Distance

Measure Area: Areas can also be measured. In the measure window, the accumulated area size appears. Right-click to stop drawing. The Info section is also available as well as the ability to switch between different *area units*.



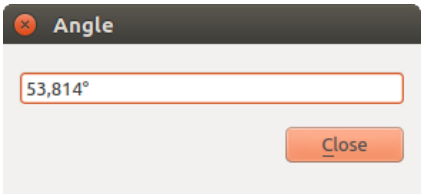
Слика7.15: Measure Area

Measure Bearing: You can also measure bearings. The cursor becomes cross-shaped. Click to draw the first point of the bearing, then move the cursor to draw the second point. The measurement is displayed in a pop-up dialog.



Слика7.16: Measure Bearing

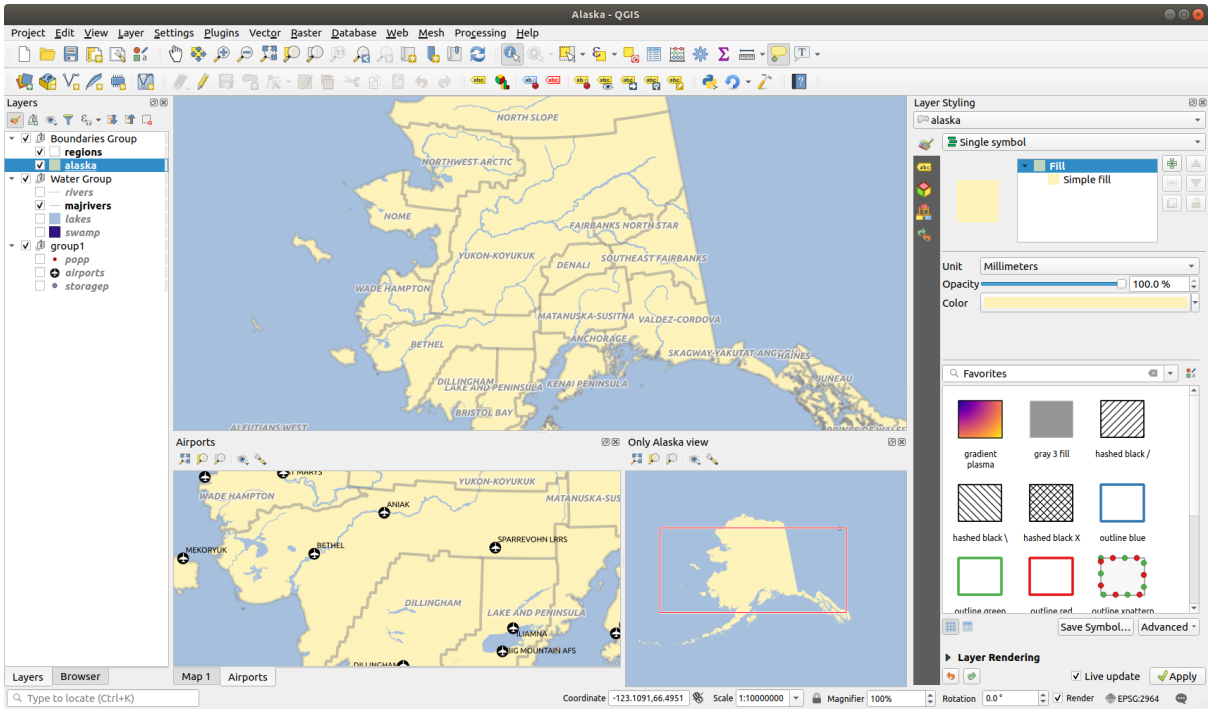
Measure Angle: You can also measure angles. The cursor becomes cross-shaped. Click to draw the first segment of the angle you wish to measure, then move the cursor to draw the desired angle. The measurement is displayed in a pop-up dialog.



Слика7.17: Measure Angle

7.1.9 Setting additional map views

It is also possible to open additional map views whose content could diverge from the *Layers* panel current state. To add a new map view, go to *View* ► *New Map View*. A new floating widget, mimicking the main map view's rendering, is added to QGIS. You can add as many map views as you need. They can be kept floating, placed side by side or stacked on top of each other.



Слика7.18: Multiple map views with different settings

At the top of an additional map canvas, there's a toolbar with the following capabilities:

- Zoom Full, Zoom to Selection and Zoom to Layer(s) to navigate within the view
- Set View Theme to select the *map theme* to display in the map view. If set to `(none)`, the view will follow the *Layers* panel changes.
- View settings to configure the map view:
  - *Synchronize view center with main map*: syncs the center of the map views without changing the scale. This allows you to have an overview style or magnified map which follows the main canvas center.
  - *Synchronize view to selection*: same as zoom to selection
  - *Scale*
  - *Rotation*
  - *Magnification*
  - *Synchronize scale* with the main map scale. A *Scale factor* can then be applied, allowing you to have a view which is e.g. always 2x the scale of the main canvas.
  - *Show annotations*
  - *Show cursor position*
  - *Show main canvas extent*
  - *Show labels*: allows to hide labels regardless they are set in the displayed layers' properties
  - *Change map CRS…*
  - *Rename view…*

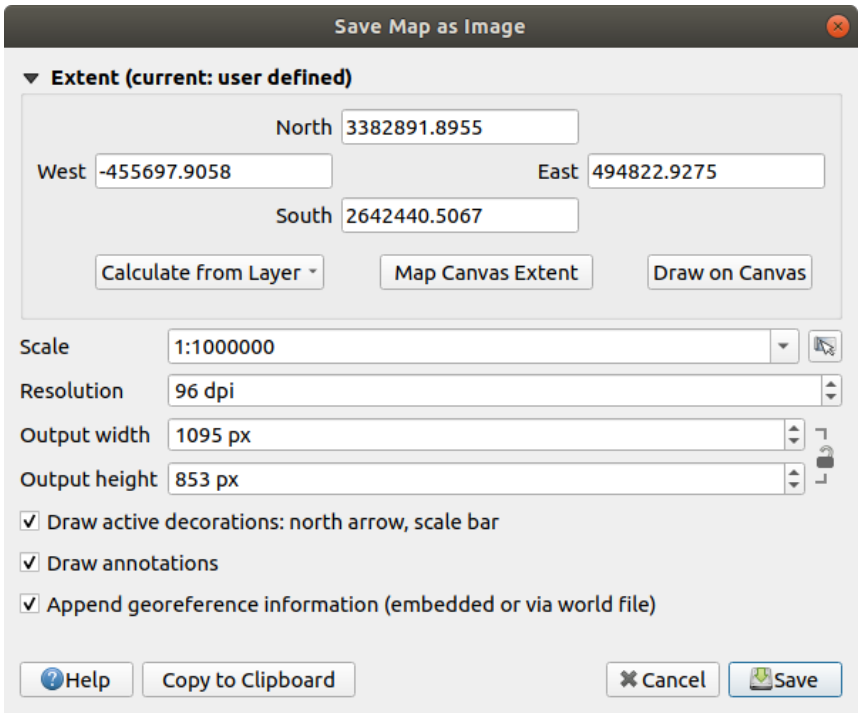
7.1.10 Exporting the map view

Maps you make can be layout and exported to various formats using the advanced capabilities of the *print layout or report*. It's also possible to directly export the current rendering, without a layout. This quick „screenshot" of the map view has some convenient features.

To export the map canvas with the current rendering:

1. Go to *Project* ► *Import/Export*
2. Depending on your output format, select either
   - *Export Map to Image…*
   - or *Export Map to PDF…*

The two tools provide you with a common set of options. In the dialog that opens:

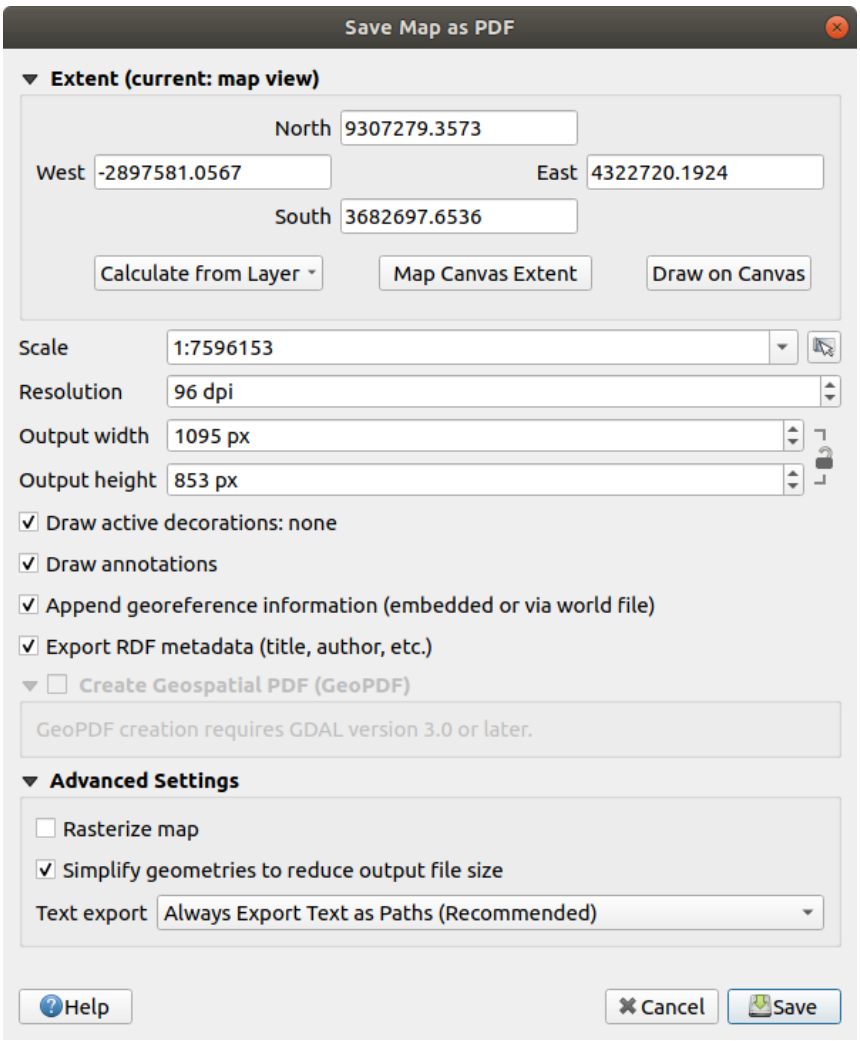


Слика7.19: The Save Map as Image dialog

1. Choose the *Extent* to export: it can be the current view extent (the default), the extent of a layer or a custom extent drawn over the map canvas. Coordinates of the selected area are displayed and manually editable.
2. Enter the *Scale* of the map or select it from the *predefined scales*: changing the scale will resize the extent to export (from the center).
3. Set the *Resolution* of the output
4. Control the *Output width* and *Output height* in pixels of the image: based by default on the current resolution and extent, they can be customized and will resize the map extent (from the center). The size ratio can be locked, which may be particularly convenient when drawing the extent on the canvas.
5. *Draw active decorations*: in use *decorations* (scale bar, title, grid, north arrow…) are exported with the map
6. *Draw annotations* to export any *annotation*
7. *Append georeference information (embedded or via world file)*: depending on the output format, a world file of the same name (with extension `PNGW` for `PNG` images, `JPGW` for `JPG`, …) is saved in the same folder

as your image. The `PDF` format embeds the information in the PDF file.

8. When exporting to PDF, more options are available in the *Save map as PDF…* dialog:



Слика7.20: The Save Map as PDF dialog

- *Export RDF metadata* of the document such as the title, author, date, description…
- *Create Geospatial PDF*: Generate a georeferenced PDF file. You can:
  - Choose the Geospatial PDF *Format*
  - *Include vector feature information* in the Geospatial PDF file: will include all the geometry and attribute information from features visible within the map in the output Geospatial PDF file.

> **Белешка**
>
> A Geospatial PDF file can also be used as a data source. For more on Geospatial PDF support in QGIS, see https://north-road.com/2019/09/03/qgis-3-10-loves-geospatialpdf/.

- *Rasterize map*
- *Simplify geometries to reduce output file size*: Geometries will be simplified while exporting the map by removing vertices that are not discernibly different at the export resolution (e.g. if the export

resolution is `300 dpi`, vertices that are less than `1/600 inch` apart will be removed). This can reduce the size and complexity of the export file (very large files can fail to load in other applications).

- Set the *Text export*: controls whether text labels are always or preferably exported as *text or outline objects*.

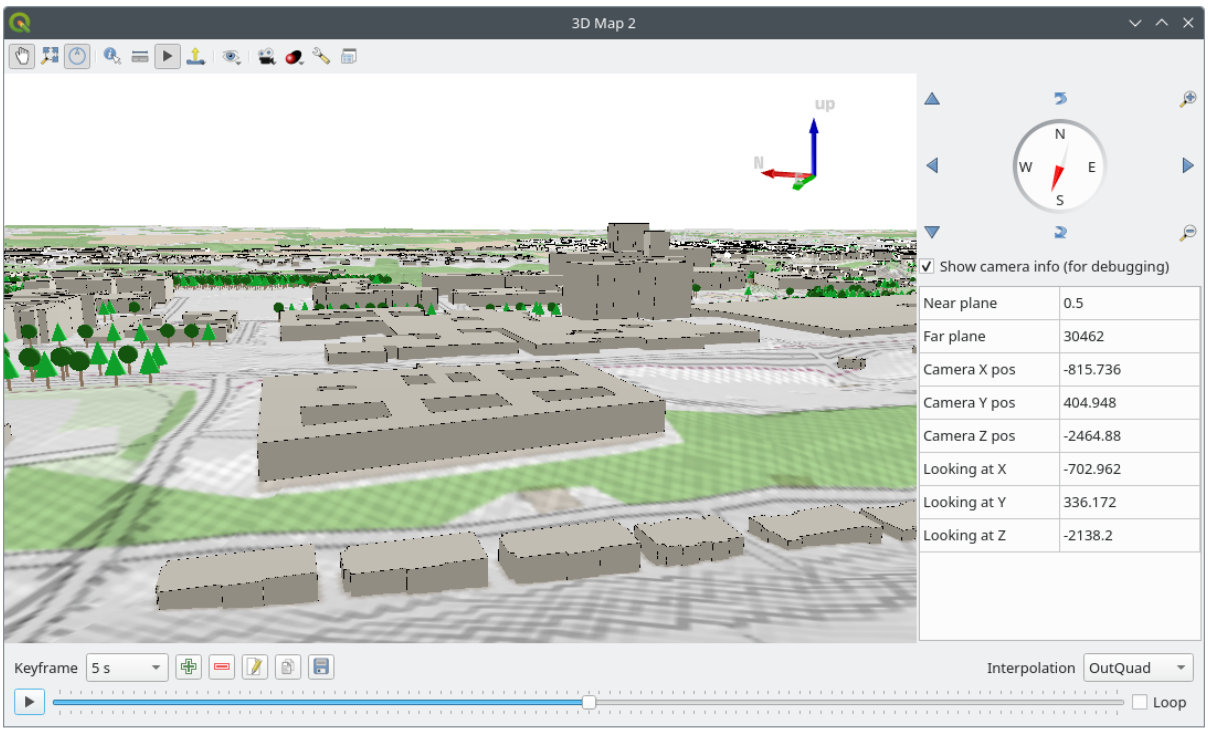
9. Click *Save* to select file location, name and format.

   When exporting to image, it's also possible to *Copy to clipboard* the expected result of the above settings and paste the map in another application such as LibreOffice, GIMP…

7.2 3D Map View

3D visualization support is offered through the 3D map view. You can create, manage and open 3D map views via *View* ► *3D Map Views* ► menu:

1. By clicking on *New 3D Map View* you can create a new 3D map view. A floating and dockable QGIS panel will appear (see *The 3D Map View dialog*). It has the same extent and view as the 2D main map canvas and provides a set of navigation tools to turn the view into 3D.
2. By clicking on *Manage 3D Map Views* you get in the 3D Map Views Manager. Here you get the ability to open, duplicate, remove and rename 3D map views.
3. If you created one or more 3D map views, you see them listed in *3D Map Views*. You can turn them on and off by clicking on. They will be saved by saving the project, even if they are turned off.



Слика7.21: The 3D Map View dialog

The following tools are provided at the top of the 3D map view panel:

- Camera Control: moves the view, keeping the same angle and direction of the camera
- Zoom Full: resizes the view to the whole layers' extent, or the *reference extent* if set

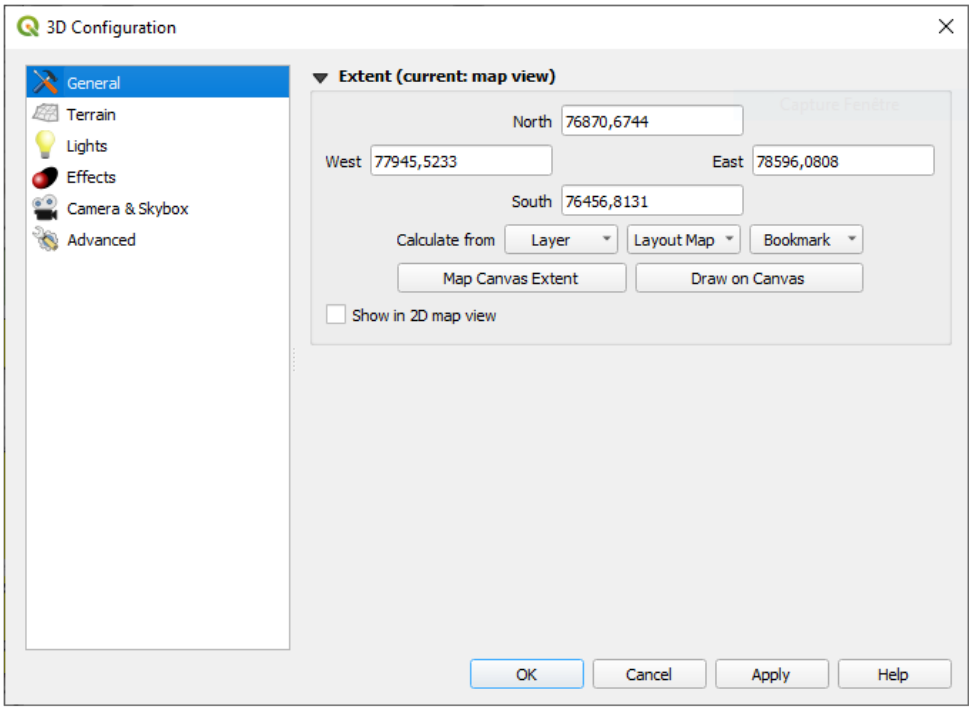
- Toggle On-Screen Notification: shows/hides the navigation widget (that is meant to ease controlling of the map view)
- Identify: returns information on the clicked point of the terrain or the clicked 3D feature(s) – More details at *Identifying Features*
- Measurement Line: measures the horizontal, vertical, and total 3D distance between points
  - Click to add points
  - Right-click to finish the measurement
  - Right-clicking twice or pressing *New* will reset the table
- Animations: shows/hides the *animation player* widget
- Export menu allows to export the scene to various formats:
  - Save as Image…: exports the current view to an image file format
  - Export 3D Scene: exports the current view as a 3D scene (`.obj` file), allowing post-processing in applications like Blender… The terrain and vector features are exported as 3D objects. The export settings, overriding the layers *properties* or map view *configuration*, include:
    - *Scene name* and destination *Folder*
    - *Terrain resolution*
    - *Terrain texture resolution*
    - *Model scale*
    - *Smooth edges*
    - *Export normals*
    - *Export textures*
- Set View Theme: Allows you to select the set of layers to display in the map view from predefined *map themes*.
- The Camera menu helps you control relation between the 2D and 3D views:
  - Synchronize the views (*2D map view follows 3D camera* and/or *3D camera follows 2D Map view*)
  - *Show visible camera area in 2D map view*
  - *Set 3D scene on 2D map view*: allows to clip the 3D scene and display only the terrain and features intersecting an extent drawn on the 2D map canvas. More options are available in the *General configuration* tab.
  - 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.
- Effects adds visual effects to the 3D rendering, such as showing *shadows*, *eye dome lighting* or *ambient occlusion*.
- The Configure… button opens the dialog to configure the 3D map view *settings*.
- Dock 3D Map View: switch from docked widget to top level window

7.2.1 Scene Configuration

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.

In the 3D Configuration window there are various options to fine-tune the 3D scene:

General



Слика7.22: The 3D Map General Configuration dialog

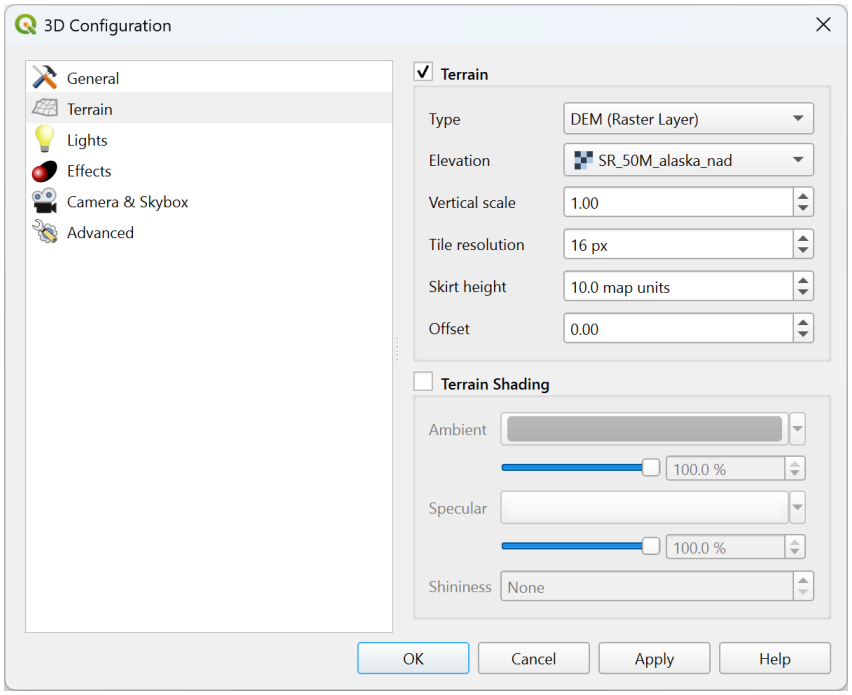
Under the *General* tab, you can:

- Limit the 3D scene's contents to a 2D map extent, using the *spatial extent selector*: terrain and features are clipped at the specified extent and only the parts falling within the extent are loaded in the scene. Elevation range of the terrain, mesh and pointcloud layers is also taken into account so that the camera is not positioned below the scene's contents, particularly when using the terrain's vertical scale setting to exaggerate the elevation differences.

  The set extent is also used as reference extent when pressing the Zoom full button in 3D map view.

- Check *Show in 2D map view* to display in the main map canvas a rubberband corresponding to the current extent of the 3D scene.

Terrain



Слика7.23: The 3D Map Terrain Configuration dialog

- *Terrain*: Before diving into the details, it is worth noting that the terrain in a 3D view is represented by a hierarchy of terrain tiles and as the camera moves closer to the terrain, existing tiles that do not have sufficient details are replaced by smaller tiles with more details. Each tile has mesh geometry derived from the elevation raster layer and texture from 2D map layers.
  - The elevation terrain *Type* can be:
    - a *Flat terrain*
    - a loaded *DEM (Raster Layer)*
    - an *Online* service, loading elevation tiles produced by Mapzen tools – more details at https://registry.opendata.aws/terrain-tiles/
    - a loaded *Mesh* dataset
    - a *Quantized Mesh* layer from the project
  - *Elevation*: Raster or mesh layer to be used for generation of the terrain. The raster layer must contain a band that represents elevation. For a mesh layer, the Z values of the vertices are used.
  - *Vertical scale*: Scale factor for vertical axis. Increasing the scale will exaggerate the height of the landforms.
  - *Tile resolution*: How many samples from the terrain raster layer to use for each tile. A value of 16px means that the geometry of each tile will consist of 16x16 elevation samples. Higher numbers create more detailed terrain tiles at the expense of increased rendering complexity.
  - *Skirt height*: Sometimes it is possible to see small cracks between tiles of the terrain. Raising this value will add vertical walls („skirts”) around terrain tiles to hide the cracks.
  - *Offset*: moves the terrain up or down, e.g. to adjust its elevation with respect to the ground level of other objects in the scene.

    This can be useful when there is a discrepancy between the height of the terrain and the height of layers in your scene (e.g. point clouds which use a relative vertical height only). In this case adjusting the terrain

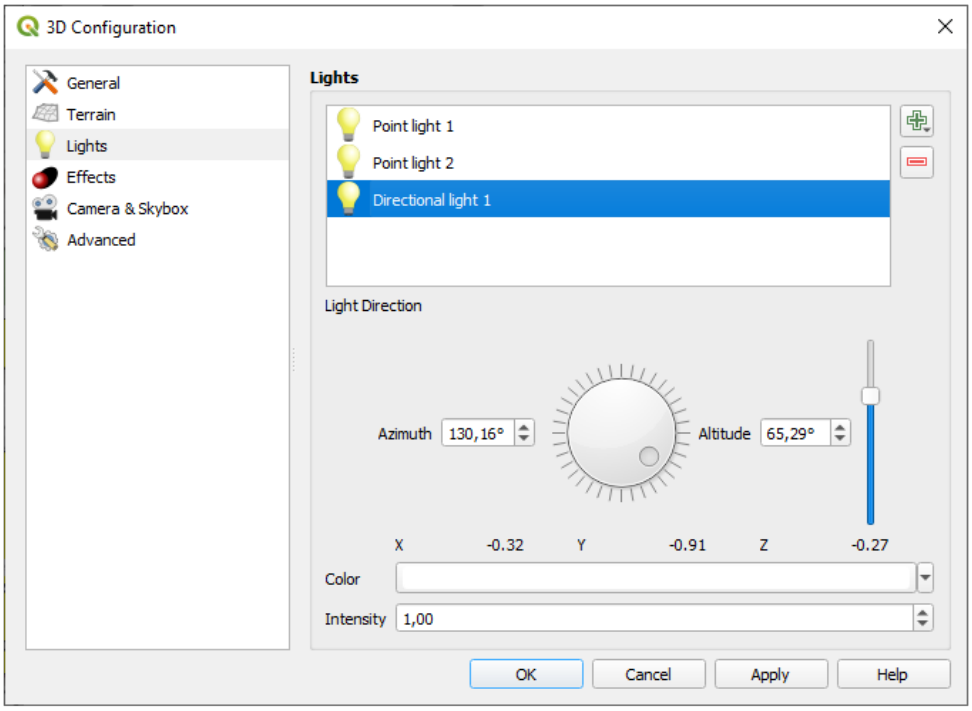
elevation manually to coincide with the elevation of objects in your scene can improve the navigation experience.

- When a mesh layer is used as terrain, you can configure the *Triangles settings* (wireframe display, smooth triangles, level of detail) and the *Rendering colors settings* (as a uniform color or *color ramp based*). More details in the *Mesh layer 3D properties* section.
- *Terrain shading*: Allows you to choose how the terrain should be rendered:
  - Shading disabled - terrain color is determined only from map texture
  - Shading enabled - terrain color is determined using Phong's shading model, taking into account map texture, the terrain normal vector, scene light(s) and the terrain material's *Ambient* and *Specular* colors and *Shininess*

Lights

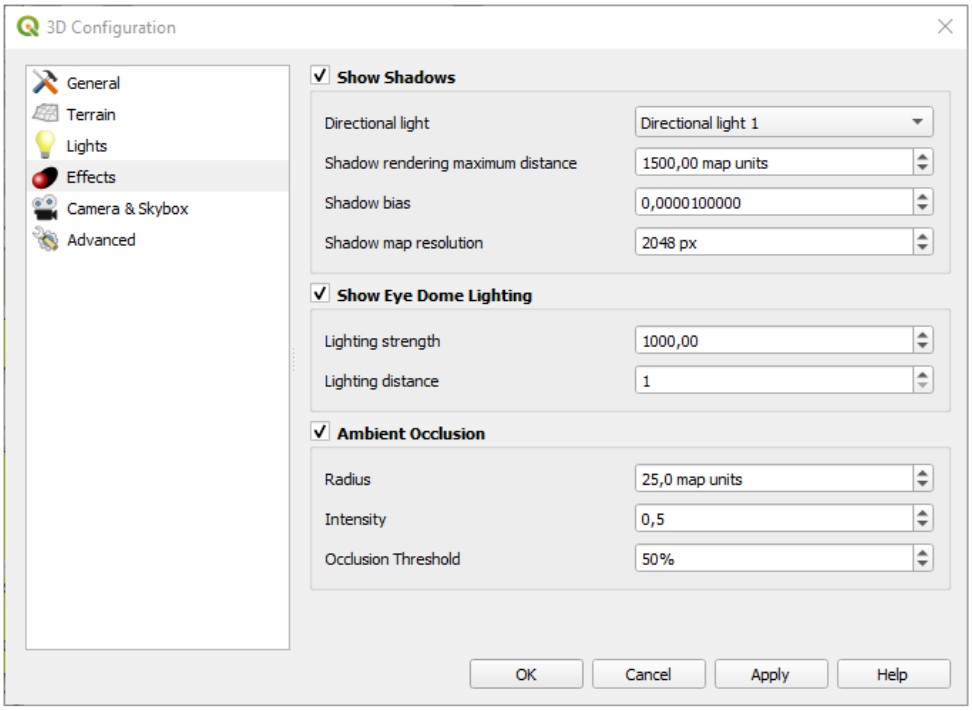
From the *Lights* tab, press the menu to add

- up to eight *Point lights*: emits light in all directions, like a sphere of light filling an area. Objects closer to the light will be brighter, and objects further away will be darker. A point light has a set position (*X*, *Y* and *Z*), a *Color*, an *Intensity* and an *Attenuation*
- up to four *Directional lights*: mimics the lighting that you would get from a giant flash light very far away from your objects, always centered and that never dies off (e.g. the sun). It emits parallel light rays in a single direction but the light reaches out into infinity. A directional light can be rotated given an *Azimuth*, have an *Altitude*, a *Color* and an *Intensity*.



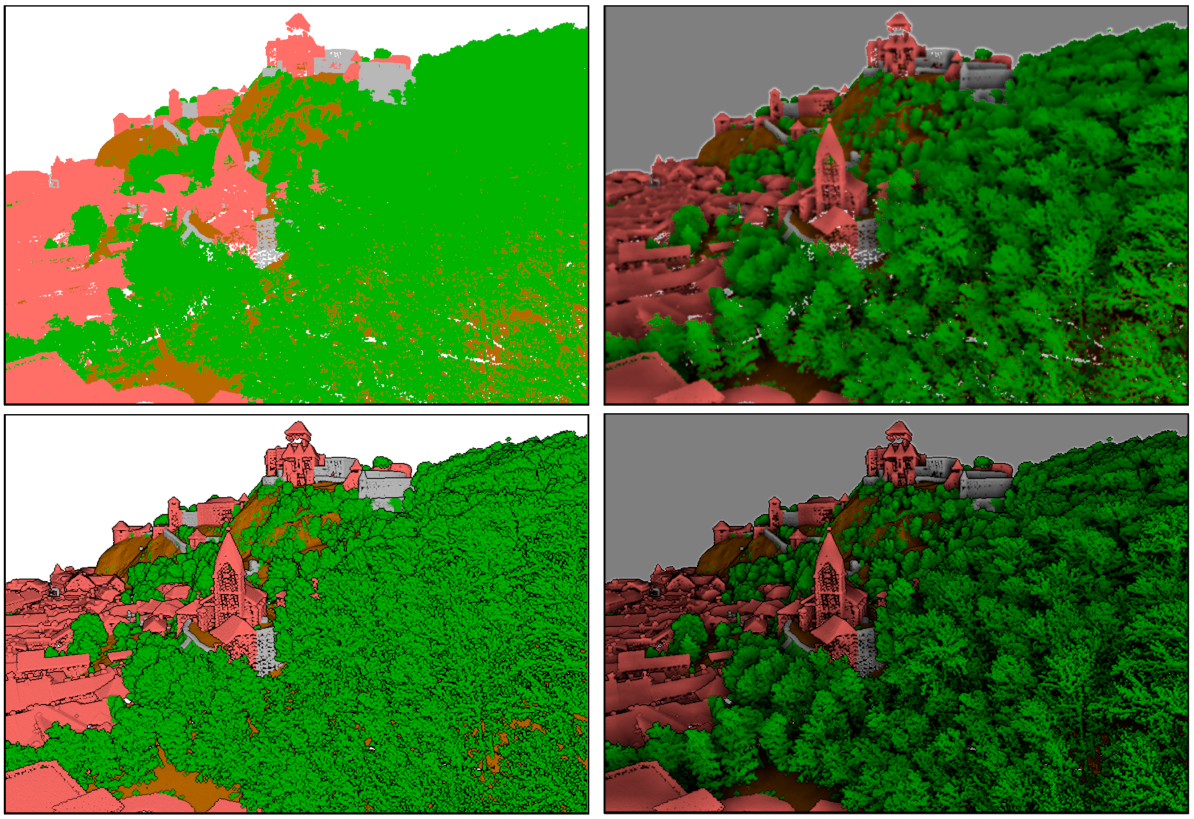
Слика7.24: The 3D Map Lights Configuration dialog

Effects



Слика7.25: The 3D Map Effects Configuration dialog

- Check *Show shadows* to display shadows within your scene, given:
  - a *Directional light*
  - a *Shadow rendering maximum distance*: to avoid rendering shadow of too distant objects, particularly when the camera looks up along the horizon
  - a *Shadow bias*: to avoid self-shadowing effects that could make some areas darker than others, due to differences between map sizes. The lower the better
  - a *Shadow map resolution*: to make shadows look sharper. It may result in less performance if the resolution parameter is too high.
- *Show Eye Dome Lighting* (EDL): a post processing effect which enhances depth perception. Each pixel's depth (distance off the camera) is compared to its neighboring pixels' depth and gets highlighted according to that depth difference, making the edges stand out. Affects the whole scene and can be combined with *Screen Space Ambient Occlusion*. Following parameters can be controlled:
  - *Lighting strength*: increases the contrast, allowing for better depth perception
  - *Lighting distance*: represents the distance of the used pixels off the center pixel and has the effect of making edges thicker.
- Add screen-space *Ambient Occlusion* (SSAO): a post processing effect which also enhances depth perception by applying a darker shading to areas which are less exposed to ambient lighting. Affects the whole scene and can be combined with *Eye dome Lighting*. Following parameters can be controlled:
  - *Radius*: how far we will reach to calculate ambient occlusion
  - *Intensity*: how strong the effect should be (higher values make things darker)
  - *Occlusion threshold*: how many neighboring points need to be occluded for the effect to appear (lower values than 50% will make the output darker, but possibly providing greater range of occlusion)

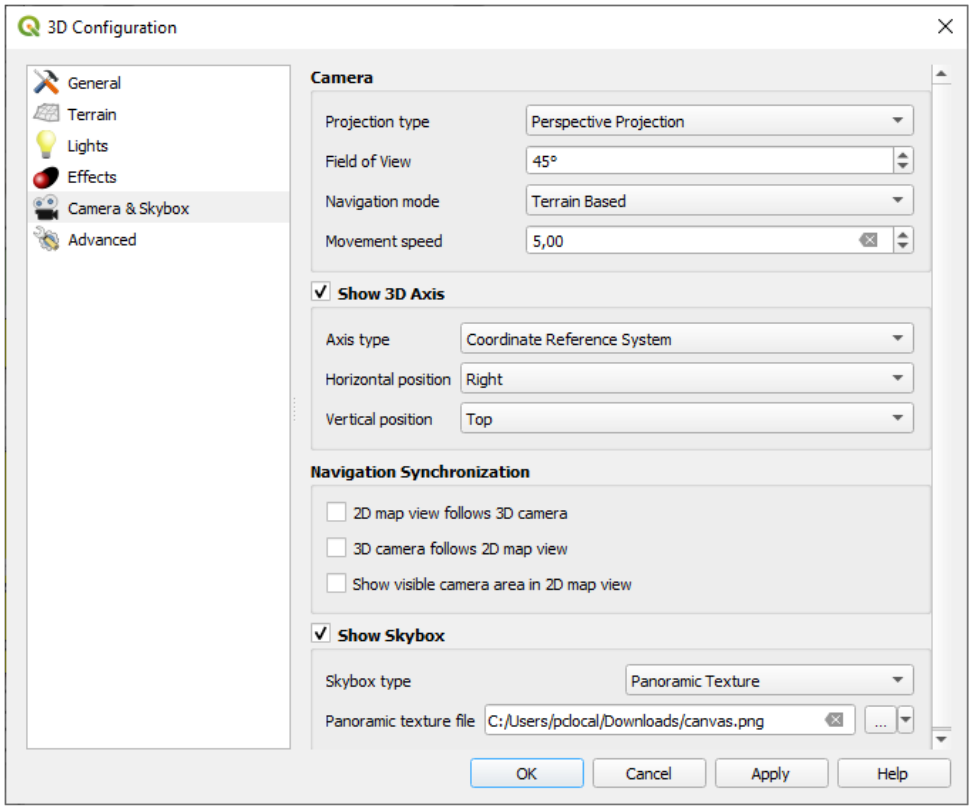


Слика7.26: Rendering Point clouds in 3D map using Eye Dome Lighting (EDL) and/or Screen-Space Ambient Occlusion (SSAO)

From top, left to right: No effect – SSAO only – EDL only – SSAO and EDL

Camera & Skybox

In this tab, you can control different parameters like camera, 3D axis, navigation synchronization and skybox.

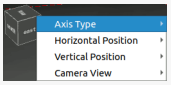


Слика7.27: The 3D Map Camera Configuration dialog

- The *Camera* parameter group overrides some *default camera settings* made in the *Settings* ► *Options* ► *3D* dialog.
- Check *Show 3D Axis* to enable 3D axis tool. This parameter group allows to set the axis type and its position.
  - With the *Coordinate Reference System* type an orthogonal axis will be represented.
  - With the *Cube* type, a 3D cube will be represented. The cube faces can be used to change the camera view: for example, click on the *north* face to set the camera to see from the north.

Савет

Right-click the 3D axis to quickly set its position and type, and the camera view.

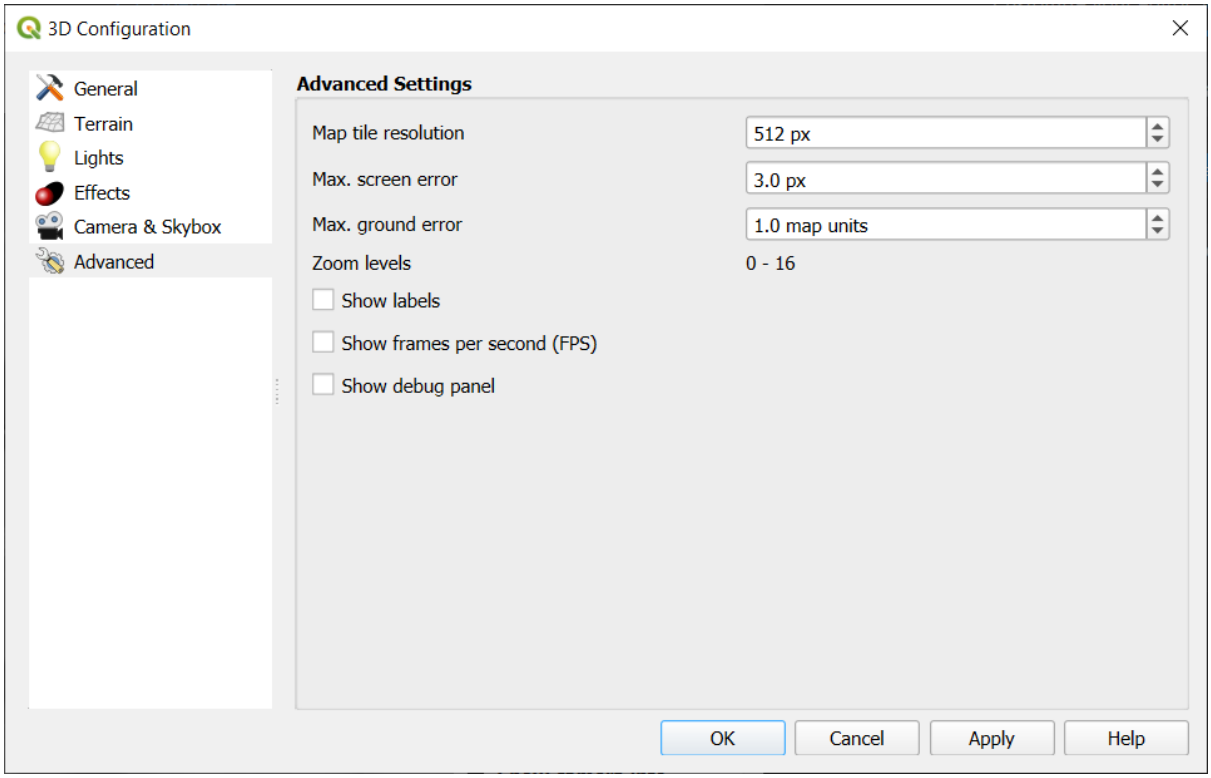


Слика7.28: The 3D Axis context menu

- The *Navigation Synchronization* parameter group adds options to synchronize 2D view with 3D camera position or 3D camera position with 2D view or bi directional synchronization. The last option displays the extent visible from the 3D camera over the 2D map view.
- Check *Show skybox* to enable skybox rendering in the scene. The skybox type can be:
  - *Panoramic texture*, with a single file providing sight on 360°
  - *Distinct faces*, with a texture file for each of the six sides of a box containing the scene

  Texture image files of the skybox can be files on the disk, remote URLs or embedded in the project (*more details*).

Advanced



Слика7.29: The 3D Map Advanced Configuration dialog

- *Map tile resolution*: Width and height of the 2D map images used as textures for the terrain tiles. 256px means that each tile will be rendered into an image of 256x256 pixels. Higher numbers create more detailed terrain tiles at the expense of increased rendering complexity.
- *Max. screen 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 expense of increased rendering complexity.
- *Max. ground error*: The resolution of the terrain tiles at which dividing tiles into more detailed ones will stop (splitting them would not introduce any extra detail anyway). This value limits the depth of the hierarchy of tiles: lower values make the hierarchy deep, increasing rendering complexity.
- *Zoom levels*: Shows the number of zoom levels (depends on the map tile resolution and max. ground error).
- ☐ *Show labels*: Toggles map labels on/off
- ☐ *Show frames per second (FPS)*
- ☐ *Show debug panel*: allows to show or hide a panel next to the 3D map view displaying options to better control map navigation and debug information:
  - ☐ *Show map tile info*: Include border and tile numbers for the terrain tiles (useful for troubleshooting terrain issues)
  - ☐ *Show bounding boxes*: Show 3D bounding boxes of the terrain tiles (useful for troubleshooting terrain issues)
  - ☐ *Show camera's view center*
  - ☐ *Show camera's rotation center*

- *Show light sources*: shows a sphere at light source origins, allowing easier repositioning and placement of light sources relative to the scene contents
- *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
- *Debug Shadow Map*: renders the scene as a red-black image from the point of view of the light used for shadows (for troubleshooting). The widget is set with a proportional *Size* to the 3D map view's, and docked in a *Corner*.
- *Debug Depth Map*: renders the scene's depth map as an image with nearer pixels being darker (for troubleshooting). The widget is set with a proportional *Size* to the 3D map view's, and docked in a *Corner*.
- *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

Белешка

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 Navigation options

To explore the map view in 3D:

- Tilt the terrain (rotating it around a horizontal axis that goes through the center of the window)
  - Press the Tilt up and Tilt down tools
  - Press `Shift` and use the up/down keys
  - Drag the mouse forward/backward with the middle mouse button pressed
  - Press `Shift` and drag the mouse forward/backward with the left mouse button pressed
- Rotate the terrain (around a vertical axis that goes through the center of the window)
  - Turn the compass of the navigation widget to the watching direction
  - Press `Shift` and use the left/right keys
  - Drag the mouse right/left with the middle mouse button pressed
  - Press `Shift` and drag the mouse right/left with the left mouse button pressed
- Change the camera position (and the view center), moving it around in a horizontal plan
  - Drag the mouse with the left mouse button pressed, and the Camera control button enabled
  - Press the directional arrows of the navigation widget
  - Use the up/down/left/right keys to move the camera forward, backward, right and left, respectively
- Change the camera altitude: press the `Page Up`/`Page Down` keys
- Change the camera orientation (the camera is kept at its position but the view center point moves)
  - Press `Ctrl` and use the arrow keys to turn the camera up, down, left and right

  - Press `Ctrl` and drag the mouse with the left mouse button pressed

- Zoom in and out
  - Press the corresponding  Zoom In and  Zoom Out tools of the navigation widget
  - Scroll the mouse wheel (keep `Ctrl` pressed results in finer zooms)
  - Drag the mouse with the right mouse button pressed to zoom in (drag down) and out (drag up)

To reset the camera view, click the  Zoom Full button on the top of the 3D canvas panel.

7.2.3 Creating an animation

An animation is based on a set of keyframes - camera positions at particular times. To create an animation:

1. Toggle on the  Animations tool, displaying the animation player widget
2. Click the  Add keyframe button and enter a *Keyframe time* in seconds. The *Keyframe* combo box now displays the time set.
3. Using the navigation tools, move the camera to the position to associate with the current keyframe time.
4. Repeat the previous steps to add as many keyframes (with time and position) as necessary.
5. Click the  button to preview the animation. QGIS will generate scenes using the camera positions/rotations at set times, and interpolating them in between these keyframes. Various *Interpolation* modes for animations are available (eg, linear, inQuad, outQuad, inCirc… – more details at https://doc.qt.io/archives/qt-5.15/qeasingcurve.html#EasingFunction-typedef).

   The animation can also be previewed by moving the time slider. Keeping the *Loop* box checked will repeatedly run the animation while clicking  stops a running animation.

Click  Export animation frames to generate a series of images representing the scene. Other than the filename *Template* and the *Output directory*, you can set the number of *Frames per second*, the *Output width* and *Output height*.

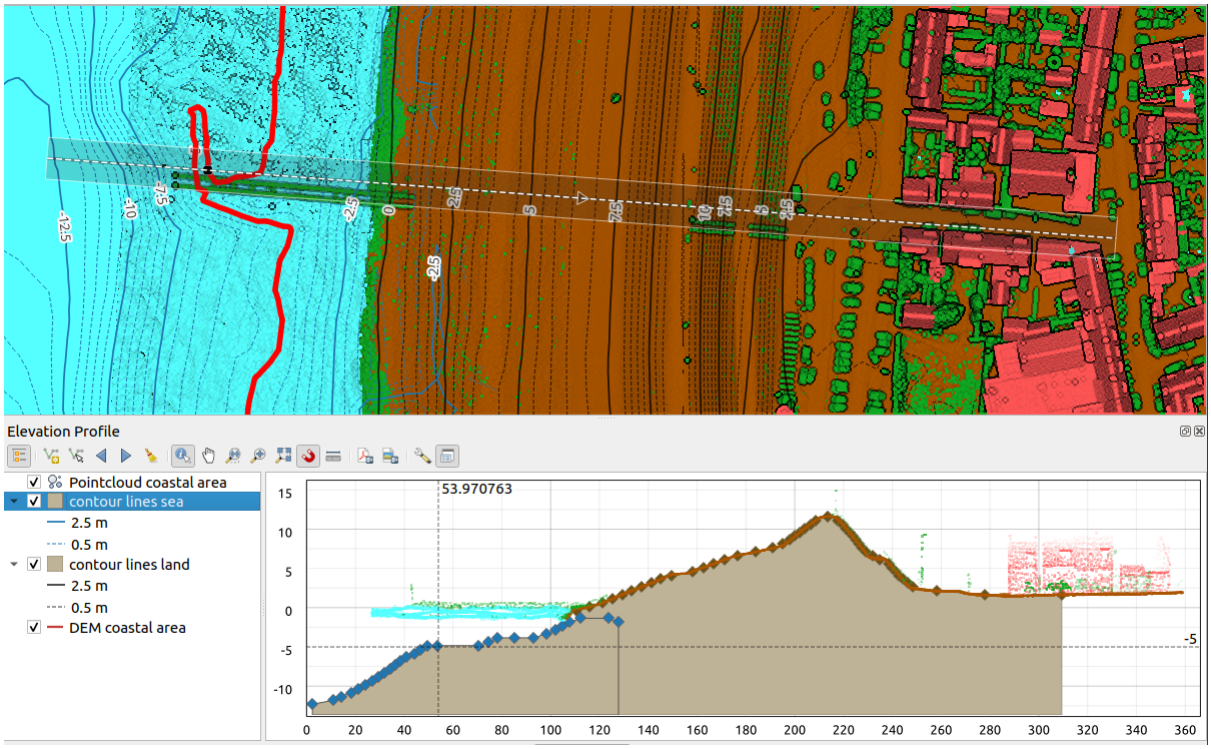
7.2.4 3D vector layers

A vector layer with elevation values can be shown in the 3D map view by checking *Enable 3D Renderer* in the *3D View* section of the vector layer properties. A number of options are available for controlling the rendering of the 3D vector layer.

7.3 Elevation Profile View

The *Elevation Profile* panel is a plotting tool for side view, for visualizing elevation data along a line. It supports vector, raster, mesh and point cloud layers. Data can be of 2D or 3D type.

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.



Слика7.30: Elevation Profile dialog embedded below main map canvas
Figure shows a coastline with a town behind a sea-dike. A larger tolerance applied to the elevation profile line returns more point cloud features.

7.3.1 The interface

At the top of the *Elevation Profile* panel, a toolbar provides you with the following tools:

Табела7.5: Elevation Profile View toolbar

Alatka	Prečica	Opis
Add Layers	hold `Ctrl` while drag-and-drop	Allows you to add new layers to a plot.
Show Layer Tree		Shows or hides a list of project layers to configure rendering in the profile view.
Capture Curve		Draws interactively a line over the map canvas to represent the profile curve.
Capture Curve From Feature		Picks an existing line feature on the map canvas and generate a profile curve along that line.
Nudge Left	`Ctrl+Alt+,`	Allows you to slowly move the capture line across the map to the left (e.g. to find the optimal profile line based on the elevation).
Nudge Right	`Ctrl+Alt+.`	Allows you to slowly move the capture line across the map to the right (e.g. to find the optimal profile line based on the elevation).
Clear		Removes the profile line and any plot displayed in the *Elevation Profile* view

наставак на следећој страни

Табела 7.5 – настављено са претходне стране

Alatka	Prečica	Opis
 Identify Features		Identifies features in the plot canvas via either a single click, or click-and-drag rectangle. Results are shown in the standard *Identify Results* dock.
 Pan	`Space`	Click and drag to pan the plot canvas. Can also be done with the middle mouse button.
 Zoom X Axis		Zooms in/out along the horizontal axis, keeping the vertical ratio
 zoom	`Ctrl+Space`	Click or click-and-drag a rectangle over the plot to zoom in. Press `Alt` and click to instead zoom out.
 Zoom Full		Zooms the *Elevation Profile* view to the extent of the capture line
 Enable Snapping		Allows to snap to the edge or vertex of the plot features in the profile view. Convenient for accurate retrieval of coordinates or distance measurements.
 Measure Distances		Measures horizontal and vertical distances
 Export As PDF		Exports plots to PDF (as high quality vector objects)
 Export As Image		Exports plots to several image formats
 Export Results		Exports plots as 3D features, 2D Profile and Distance/Elevation Table. (The results can be saved as DXF files, CSV files or any of the standard writable vector spatial formats). More details at *Exporting from the elevation profile*.
 Options ►		Provides access to configuration settings of the profile elevation line.
►  *Lock distance/elevation scales*		Ensures that the horizontal and vertical scales are always kept equal (so that e.g. a 45° slope will appear as a 45° slope in the profile view).
► *Distance units*		Allows to render distances in the profile chart with units other than the map canvas units.
► *Tolerance*		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.
► *Rename Profile…*		Allows to rename the profile view.
 Dock Elevation Profile View		Switch between docked and floating status of the view

In the bottom left, a copy of the *Layers* panel can be displayed pushing the  *Show Layer Tree* button. This is however an independent widget, with its own set of visible layers, in a custom stack order. It allows you to control layers rendering and behavior within the plot canvas:

- Tick the box next to the layer name to set whether it should be rendered in the plot canvas
- Drag-and-drop layers up or down to change the order of the layers
- style rendering of layers in the profile view: double-click a layer or right-click and select *Properties…* to open the layer's *Elevation* properties tab for configuration. A summary of elevation settings is displayed as tooltip when hovering over the layer.

On the right of the layer tree, the plot canvas is the main place you can preview the elevation profile of the enabled layers. It is based on a graduated grid in which the horizontal axis displays the length of the profile line and the vertical axis displays the Z elevation of the observed features. It also allows a set of interactions such as zooming, panning, measuring, identifying features, … using the tools at the top.

7.3.2 Creating an elevation profile

To create a profile view, you can:

1. Go to *View* ►  *Elevation Profile* menu. The *Elevation profile* panel opens.
2. Create the profile line along which the terrain and the features will be rendered. Select a drawing tool:
   -  Capture Curve: click left over the main map canvas to add vertices and click right to finalize a line that will be used as profile line
   - or  Capture Curve From Feature: click a line feature on the map canvas to select it as the profile line. If multiple features are present at the clicked point then a popup menu will appear allowing you to select among them.

   All the line digitizing capabilities such as the *snapping options*, *tracing*, *digitizing techniques* or the advanced *digitizing panel* are available for use.

   The plot canvas may start rendering some features.
3. The next step is to configure the elevation properties of the layers you want to visualize.
   1. Push on the  Show Layer Tree button to display the list of layers. By default, not all the layers of the project are loaded and referenced in the profile tool; e.g., raster layers which are not marked as having elevation data in their properties are ignored. Clicking the  Add layers button will show a filtered list of possible layers which can be added to the plot, but which currently aren't in the plot. Applying the dialog with selected layers will automatically mark them as having elevation data and immediately add them to the plot.

      Савет

      Holding `Ctrl` key, it is possible to drag and drop additional layers from the main layer tree to one in the elevation profile tool.

   2. Toggle visibility of the layers you are interested in. These are the only ones rendered in the profile view and selected layers can be different from the main *Layers* panel's.
   3. Double-click a layer name or right-click and select *Properties*. The *Elevation* properties tab of the layer opens. This is the place you configure how each feature or terrain should render on the profile view. Available properties depend on the layer type:
      - *Raster Elevation Properties*
      - *Vector Elevation Properties*
      - *Point Cloud Elevation Properties*
      - *Mesh Elevation Properties*

      The profile view starts rendering terrain or features of active layers crossing the given profile curve, as soon as they have elevation properties configured.
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.

Белешка

Limitations with polygon extrusion

Geometry extrusion can be set in the  *Elevation* properties of a layer, and rendered in the profile view. When tolerance is enabled, it is however not trivial to render extruded polygons, thus, for now, polygon extrusion is ignored.

7.3.3 Interacting with the profile Views

When an elevation profile line is created, the plot canvas zooms to its full extent. On the X-axis you can see the length of the profile and, on the Y-axis the height range between minimum and maximum height captured, both in map units.

When you move the mouse pointer in the elevation profile view, you will see two crossing dot lines:

- the vertical line shows the height information
- the horizontal line shows the distance from the beginning of the elevation profile Line

When you move the mouse pointer in the elevation profile canvas, you can also see a black dot move along the elevation profile line on the main map canvas. At the middle of the profile line, an arrow indicates its direction.

As for the main map canvas, QGIS provides means to navigate on the plot canvas:

-  Pan is used to move the elevation profile extent in any direction you want. Holding `Space` key while moving the mouse also shifts the plot canvas extent.
-  Zoom X Axis is used to zoom in along the horizontal axis, keeping the ratio of the vertical axis (the elevation) the same. Left click to stretch out the plot along the X axis, with the clicked point at the center of the axis, or drag a rectangle to stretch out the plot along the X axis to the rectangle width. Hold `Alt` while using  Zoom X Axis to zoom out along that axis.
-  Zoom is used to zoom in on a certain point (using left click), or to a certain extent (dragging a rectangle on the area). Hold `Alt` while using  Zoom to instead zoom out. In combination with the `Ctrl` key you can zoom in or out more smoothly.
-  Zoom Full is the default zoom level used at the beginning and shows the full extent of the profile line, with all returned features. Use it to reset the zoom level.

It is also possible to interact with the elements displayed in the plot canvas:

- Press  Enable snapping button to accurately catch points, vertices or edges of the features, for an accurate measurement or coordinates report.
-  Identify Features is used to identify features of the visible layers in the layer tree. You can drag a rectangle across several features in the profile view to query all of them. When compatible with the format (e.g. vector, point cloud), these features will be highlighted in the main map canvas.
-  Measurement Distances: click or select two points in the plot canvas to report the horizontal *Distance*, the *Elevation* and the *Total distance* between them, in map units.

- Nudge Left and Nudge right are used to shift the position of the elevation profile line in the map canvas to its left or right. The plot canvas will be redrawn, showing features and terrain overlapping the profile line buffer. The line is moved sideways using the *Tolerance* value in Options menu.

Упозорење

Closing an elevation profile view or the project currently removes the view from the project.

For more details, give a look to QGIS elevation profile/cross section tool – a deep dive!, a presentation done by Nyall Dawson.

7.3.4 Exporting from the elevation profile

The top toolbar of the *Elevation profile* dialog provides a variety of tools to export an elevation profile, allowing you to reuse the plots in another software:

- Export As PDF: Exports plots to PDF (as high quality vector objects). Following options are requested for generating the output file:
  - The *Page size*, selecting a page from a predefined list or setting custom dimensions, and the page orientation
  - The *Chart ranges* to control the extent of the chart to export:
    - on the X axis, the *Minimum distance* and *Maximum distance* from the profile curve starting point
    - On the Y axis, the *Minimum elevation* and *Maximum elevation*
  - The *Distance axis settings*: helps you configure the horizontal graduation of the grid over which the elevation profile is rendered. You can customize the *Label interval*, the *Major grid line interval* and the *Minor grid line interval*.
  - The *Elevation axis settings*, same as above, for vertical graduation
- Export As Image: Exports plots to an image file format. Specific *Export options* such as the *Image width* and *Image height* are requested, along with the *Chart ranges*, *Distance axis settings* and *Elevation axis settings* previously mentioned.
- Export results: Exports the profile results as a set of features of a vector layer (`.DXF`, `.CSV`, `.SHP`, `.GPKG`, …). Depending on the output format, multiple layers may be returned when the geometry types of exported features differ.
  - *Export 3D features*: Exports profile lines as 3D features, with elevation values taken from the elevation slices and stored in exported geometry Z values.
  - *Export 2D profile*: Exports profiles as 2D profile lines, with elevation stored in exported geometry Y dimension and distance in X dimension (as shown in the elevation profile widget).
  - *Export distance/elevation table*: Exports profiles as a table of sampled distance vs elevation values.

ГЛАВА 8

General Tools

8.1 Context help

Whenever you need help on a specific topic, you can access the corresponding page in the current User Manual via the *Help* button available in most dialogs — please note that third-party plugins can point to dedicated web pages.

8.2 Paneli

By default, QGIS provides many panels to work with. Some of these panels are described below while others may be found in different parts of the document. A complete list of default panels provided by QGIS is available via the *View* ► *Panels* ► menu and mentioned at *Paneli*.

8.2.1 Layers Panel

The *Layers* panel (also called the `map legend`) lists all the layers in the project and helps you manage their visibility and shape the map. You can show or hide the panel by pressing `Ctrl+1`.

QGIS provides a variety of ways to add layers to a project:

- using the *Add* button from the dedicated data provider tab in the *Data source manager* dialog
- from QGIS *Browser panel* or *DB Manager*: double-click, drag-and-drop files and layers onto QGIS or use the contextual menu
- drag-and-drop files from the Operating System files explorer onto QGIS

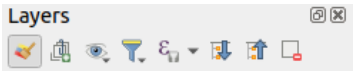
In all these scenarios, you can open one or many layers at a time. New layers are added to the *Layers* panel:

1. if dropped over the *Layers* panel, at the exact location they are released
2. in the other cases, at a location that respects the global *behavior used when adding new layers* setting
3. and in case of multiple layers, they are sorted in a way that increases the chance of their stacking being logical and features being visible as most as possible, using the following logic (top to bottom):
    - vector point layers
    - vector line layers
    - vector polygon layers
    - point cloud layers

- mesh layers
- raster layers

At the top of the *Layers* panel, a toolbar allows you to:

- Open the layer styling dock (F7): toggle the *Layer Styling* panel on and off.
- Add new group: see *Interact with groups and layers*
- Manage Map Themes: control visibility of layers and arrange them in different *map themes*.
- filter layers in the legend tree:
  - *Filter Legend by Map Content*: only the layers that are set visible and whose features intersect the current map canvas have their style rendered in the layers panel. Otherwise, a generic NULL symbol is applied to the layer. Based on the layer symbology, this is a convenient way to identify which kind of features from which layers cover your area of interest.
  - *Show Private Layers*: a convenient shortcut to display and interact with *private layers* in the *Layers* panel without modifying the project settings.
  - *Show Broken Layers Only*: only layers with broken data sources are displayed.
- Filter Legend by Expression: apply an expression to remove styles from the selected layer tree that have no feature satisfying the condition. This can be used to highlight features that are within a given area/feature of another layer. From the drop-down list, you can edit and clear the expression currently applied.
- Expand All or Collapse All layers and groups in the layers panel.
- Remove Layer/Group currently selected.



Слика8.1: Layer Toolbar in Layers Panel

Белешка

Tools to manage the layers panel are also available for map and legend items in print layouts

Configuring map themes

The Manage Map Themes drop-down button provides access to convenient shortcuts to manipulate visibility of the layers in the *Layers* panel:

- *Show All Layers*
- *Hide All Layers*
- *Show Selected Layers*
- *Hide Selected Layers*
- *Toggle Selected Layers*: changes the visibility of the first selected layer in the panel, and applies that state to the other selected layers. Also accessible through `Space` shortcut.
- *Toggle Selected Layers Independently*: changes the visibility status of each selected layer

-  *Hide Deselected Layers*

Beyond the simple control of layer visibility, the  Manage Map Themes menu allows you to configure **Map Themes** in the legend and switch from one map theme to another. A map theme is a **snapshot** of the current map legend that records:

- the layers set as visible in the *Layers* panel
- **and** for each visible layer:
  - the reference to the *style* applied to the layer
  - the visible classes of the style, ie the layer checked node items in the *Layers panel*. This applies to *symbologies* other than the single symbol rendering
  - the collapsed/expanded state of the layer node(s) and the group(s) it's placed inside

To create a map theme:

1. Check a layer you want to show
2. Configure the layer properties (symbology, diagram, labels…) as usual
3. Expand the *Style* ► menu at the bottom and click on *Add…* to store the settings as *a new style embedded in the project*

   Белешка

   A map theme does not remember the current details of the properties: only a reference to the style name is saved, so whenever you apply modifications to the layer while this style is enabled (eg change the symbology rendering), the map theme is updated with new information.

4. Repeat the previous steps as necessary for the other layers
5. If applicable, expand or collapse groups or visible layer nodes in the *Layers* panel
6. Click on the  Manage Map Themes button on top of the panel, and *Add Theme…*
7. Enter the map theme's name and click *OK*

The new theme is listed in the lower part of the  drop-down menu.

You can create as many map themes as you need: whenever the current combination in the map legend (visible layers, their active style, the map legend nodes) does not match any existing map theme contents as defined above, click on *Add Theme…* to create a new map theme, or use *Replace Theme* ► to update a map theme. You can rename the active map theme with *Rename Current Theme…* or use the *Remove Current Theme* button to delete it.

Map themes are helpful to switch quickly between different preconfigured combinations: select a map theme in the list to restore its combination. All configured themes are also accessible in the print layout, allowing you to create different map items based on specific themes and independent of the current main canvas rendering (see *Map item layers*).

Overview of the context menu of the Layers panel

At the bottom of the toolbar, the main component of the Layers panel is the frame listing all the layers added to the project, optionally organized in groups. A layer with a checked box next to it displays its contents overlapping the map canvas extent, unless a *scale-based visibility* is set. A layer can be selected and dragged up or down in the legend to change the Z-ordering. Z-ordering means that layers listed nearer the top of the legend are drawn over layers listed lower down in the legend. Also a layer or a group of layers can be dragged across several QGIS instances.

Белешка

The Z-ordering behavior can be overridden by the *Layer Order* panel.

Depending on the item selected in the panel, a right-click shows a dedicated set of options presented below. Some of these entries are not available when multiple selections are made (e.g., *Filter*, *Rename*, *Properties*).

Табела8.1: Contextual menus from *Layers* panel items

Option	Group	Vek-torski sloj	Ras-terski sloj	Mesh sloj	Sloj oblaka tačaka	3D Layer
Zoom to Layer(s)/Group	✓	✓	✓	✓	✓	✓
Zoom to Selection		✓				
Prikaži u pregledu		✓	✓	✓	✓	✓
Show Feature Count		✓				
Show Label		✓		✓		
Copy Layer/Group	✓	✓	✓	✓	✓	✓
Rename Layer/Group	✓	✓	✓	✓	✓	✓
Zoom to Native Resolution (100%)			✓			
Stretch Using Current Extent			✓			
Create Raster Attribute Table			✓			
Open Raster Attribute Table			✓			
Load Raster Attribute Table from VAT.DBF			✓			
Update SQL Layer…		✓				
Execute SQL…		✓				
Edit Virtual Layer…		✓				
Add Group	✓					
Duplicate Layer		✓	✓	✓	✓	✓
Remove Layer/Group…	✓	✓	✓	✓	✓	✓
Move Out of Group		✓	✓	✓	✓	
Move to Top	✓	✓	✓	✓	✓	
Move to Bottom	✓	✓	✓	✓	✓	
Check and all its Parents		✓	✓	✓	✓	
Group Selected		✓	✓	✓	✓	
Otvori atributsku tabelu		✓				
Uključi/isključi uređivanje		✓		✓		
Trenutne izmene ►		✓		✓		
Filter…		✓	✓		✓	
Change Data Source…		✓	✓	✓	✓	

наставак на следећој страни

Табела 8.1 – настављено са претходне стране

Option	Group	Vek-torski sloj	Ras-terski sloj	Mesh sloj	Sloj oblaka tačaka	3D Layer
Repair Data Source…		☑	☑	☑	☑	
Actions on selections ► (in edit mode)		☑				
► *Duplicate Feature*		☑				
► *Duplicate Feature and Digitize*		☑				
Set Layer Scale Visibility…		☑	☑	☑	☑	☑
Zoom to Visible Scale		☑	☑	☑	☑	
Layer CRS ►		☑	☑	☑	☑	☑
► *Set Project CRS from Layer*		☑	☑	☑	☑	☑
► *Set to..* (recent CRSs)				☑	☑	☑
► *Set Layer CRS…*		☑	☑	☑	☑	☑
Set Group CRS…	☑					
Set Group WMS Data…	☑					
☐ *Mutually Exclusive Group*	☑					
Check and all its children (Ctrl-click)	☑					
Uncheck and all its children (Ctrl--click)	☑					
Make Permanent		☑				
Izvezi ►	☑	☑	☑	☑	☑	☑
► *Save As…*			☑			
► *Save Features As…*		☑				
► *Save Selected Features As…*		☑				
► *Save As Layer Definition File…*	☑	☑	☑	☑	☑	☑
► *Save As QGIS Layer Style File…*		☑	☑	☑	☑	☑
Styles ►		☑	☑	☑	☑	☑
► *Copy Style*		☑	☑	☑	☑	☑
► *Paste Style*	☑	☑	☑	☑	☑	
► *Add…*		☑	☑	☑	☑	☑
► *Rename Current…*		☑	☑	☑	☑	☑
► *Edit symbol…*		☑				
► *Copy Symbol*		☑				
► *Paste Symbol*		☑				
Add Layer Notes…		☑	☑	☑	☑	☑
Edit Layer Notes…		☑	☑	☑	☑	
Remove Layer Notes		☑	☑	☑	☑	
Properties…		☑	☑	☑	☑	☑

For GRASS vector layers,  Toggle editing is not available. See section *Digitizing and editing a GRASS vector layer* for information on editing GRASS vector layers.

Interact with groups and layers

Layers in the legend window can be organized into groups. There are different ways to do this:

1. Press the  icon to add a new group. Type in a name for the group and press `Enter`. Now click on an existing layer and drag it onto the group.
2. Select more than one layer, then press the  icon. The selected layers are automatically inserted into the new group.
3. Select some layers, right-click in the legend window and choose *Group Selected*. The selected layers will automatically be placed in a new group.

To move a layer out of a group, drag it out, or right-click on it and choose *Move Out of Group*: the layer is moved from the group and placed above it. Groups can also be nested inside other groups. If a layer is placed in a nested group, *Move Out of Group* will move the layer out of all nested groups.

To move a group or layer to the top of the layer panel, either drag it to the top, or choose *Move to Top*. If you use this option on a layer nested in a group, the layer is moved to the top in its current group. The *Move to Bottom* option follows the same logic to move layers and groups down.

The checkbox for a group will show or hide the checked layers in the group with one click. With `Ctrl` pressed, the checkbox will also turn on or off all the layers in the group and its sub-groups.

`Ctrl`-click on a checked / unchecked layer will uncheck / check the layer and all its parents.

Enabling the **Mutually Exclusive Group** option means you can make a group have only one layer visible at the same time. Whenever a layer within the group is set visible the others will be toggled not visible.

It is possible to select more than one layer or group at the same time by holding down the `Ctrl` key while clicking additional layers. You can then move all selected layers to a new group at the same time.

You may also delete more than one layer or group at once by selecting several items with the `Ctrl` key and then pressing `Ctrl+D`: all selected layers or groups will be removed from the layers list.

More information on layers and groups using indicator icon

In some circumstances, formatting or icons next to the layer or group in the *Layers* panel change to give more information about the layer/group. These elements are:

-  to indicate that the layer is in edit mode and you can modify the data
-  to indicate that the layer being edited has some unsaved changes
-  to indicate *a filter* applied to the layer. Hover over the icon to see the filter expression and double-click to update the query
-  to identify layers that are *required* in the project, hence non removable
-  to identify an *embedded group or layer* and the path to their original project file
-  to identify a layer whose data source was not available at the project file opening (see *Handling broken file paths*). Click the icon to update the source path or select *Repair Data Source…* entry from the layer contextual menu.
-  to remind you that the layer is a *temporary scratch layer* and its content will be discarded when you close this project. To avoid data loss and make the layer permanent, click the icon to store the layer in any of the GDAL vector formats supported by QGIS.
-  to identify a layer used in *offline editing mode*.
-  to identify a layer that has no/unknown CRS

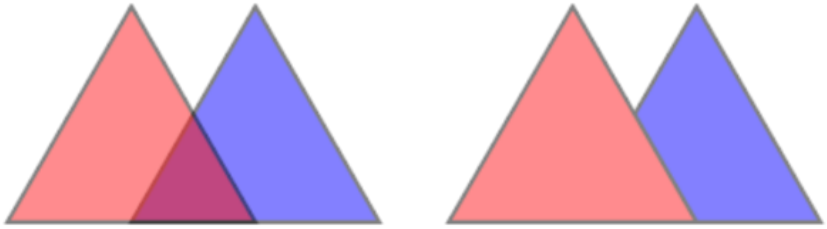
- for layers with coordinates stored in a coordinate reference system which is inherently low accuracy (requires the *corresponding setting* to be enabled)
- to identify a temporal layer controlled by canvas animation
- to identify a layer that has *notes* associated
- A grayed name, when the map canvas current scale is outside the layer's visibility scale range (as set in its *Rendering* properties). Select the contextual menu *Zoom to Visible Scale* option to zoom the map to the layer's nearest visibility scale bound.

Control layers rendering through grouping

Groups are a means of structuring layers within a tree in the project but they can also impact how their component layers are rendered, namely as a single flattened object during map renders.

The option for such a rendering is available within the *Layer Styling* panel whenever a group is selected. Under the Symbology tab, check *Render Layers as a Group* to enable a set of options to control the appearance of the child layers as a whole, instead of individual layers:

- *Opacity*: Features from child layers which are obscured by other child layers remain obscured, and the opacity applies to the „whole of group" only.



Слика8.2: Setting opacity on layers vs on a group
The image on the left shows two layers being rendered at 50% opacity (underlying features are visible, but semi-masked by the 50% red feature on top). The second image shows the result of setting the opacity on the group (parts of the blue underlying child layer is completely obscured by the red layer on top and then the result is rendered at 50% opacity).

- *Blend modes*: Just like opacity, setting a *blend mode* (like multiply, overlay, …) for an entire group results first in flattening features of child layers, with upper ones obscuring lower. The rendering is then obtained by blending the flat group and the layers sitting below the group.
  - When the child layers have blend modes assigned, it is applied before flattening but the scope is restricted to only affecting other child layers from that group, and not other layers sitting below the whole group.
  - Some more *blending modes* options are available for child layers in groups, in their *Symbology* tab which perform „clipping" style operations on other child layers during the render. You can e.g. clip the render of one layer's content by the content in a second „mask" layer.
- *Layer effects*: applies *effects* only to the flattened render of the child layers; So e.g. a drop shadow effect applied to the group would not be visible for obscured child layers.

When a group is set to *Render layers as a group*, then only the group will be shown in the *Layer Order* panel list. Group children will not be visible in this order list, as their ordering is determined by the placement of the group layer.

Editing layer style

From the *Layers* panel, you have shortcuts to change the layer rendering quickly and easily.

Right-click on a layer and select *Styles* ► in the list in order to:

- see the *styles* currently available for the layer. If you defined many styles for the layer, you can switch from one to another and your layer rendering will automatically be updated on the map canvas.
- copy part or all of the current style, and when applicable, paste a copied style from another layer
- *Rename current…* style
- *Add* a new style (which is actually a copy of the current one)
- or *Remove current* style (only when multiple styles are available).

> **Савет**
>
> **Quickly share a layer style**
>
> From the context menu, copy the style of a layer and paste it to a group or a selection of layers: the style is applied to all the layers that are of the same type (vector, raster, mesh, point cloud, …) as the original layer and, for vector layers, have the same geometry type (point, line or polygon).

When using a symbology based on features classification (e.g. *categorized*, *graduated* or *rule-based* for vector layers, or *classification* for point clouds), right-clicking a class entry in the *Layers* panels makes it possible to edit the visibility of the classes (and their features) and avoid (un)checking them one by one:

-  *Toggle Items*
-  *Show All Items*
-  *Hide All Items*

With vector layer, the contextual menu of a class leaf entry also gives access to:

-  *Select features*: selects in the layer all the features matching that class
-  *Show in attribute table*: opens an attribute table filtered to only the features matching that class
- update the *symbol color* using a **Color Wheel**. For convenience, the recently used colors are also available at the bottom of the color wheel.
- *Edit Symbol…*: opens the *Symbol Selector* dialog to change feature symbol (symbol, size, color…).
- *Copy Symbol*
- *Paste Symbol*

> **Савет**
>
> Double-clicking a class leaf entry also opens the *Symbol Selector* dialog.

8.2.2 Layer Styling Panel

The *Layer Styling* panel (also enabled with `Ctrl+3`) is a shortcut to some of the functionalities of the *Layer Properties* dialog. It provides a quick and easy way to define the rendering and the behavior of a layer, and to visualize its effects without having to open the layer properties dialog.

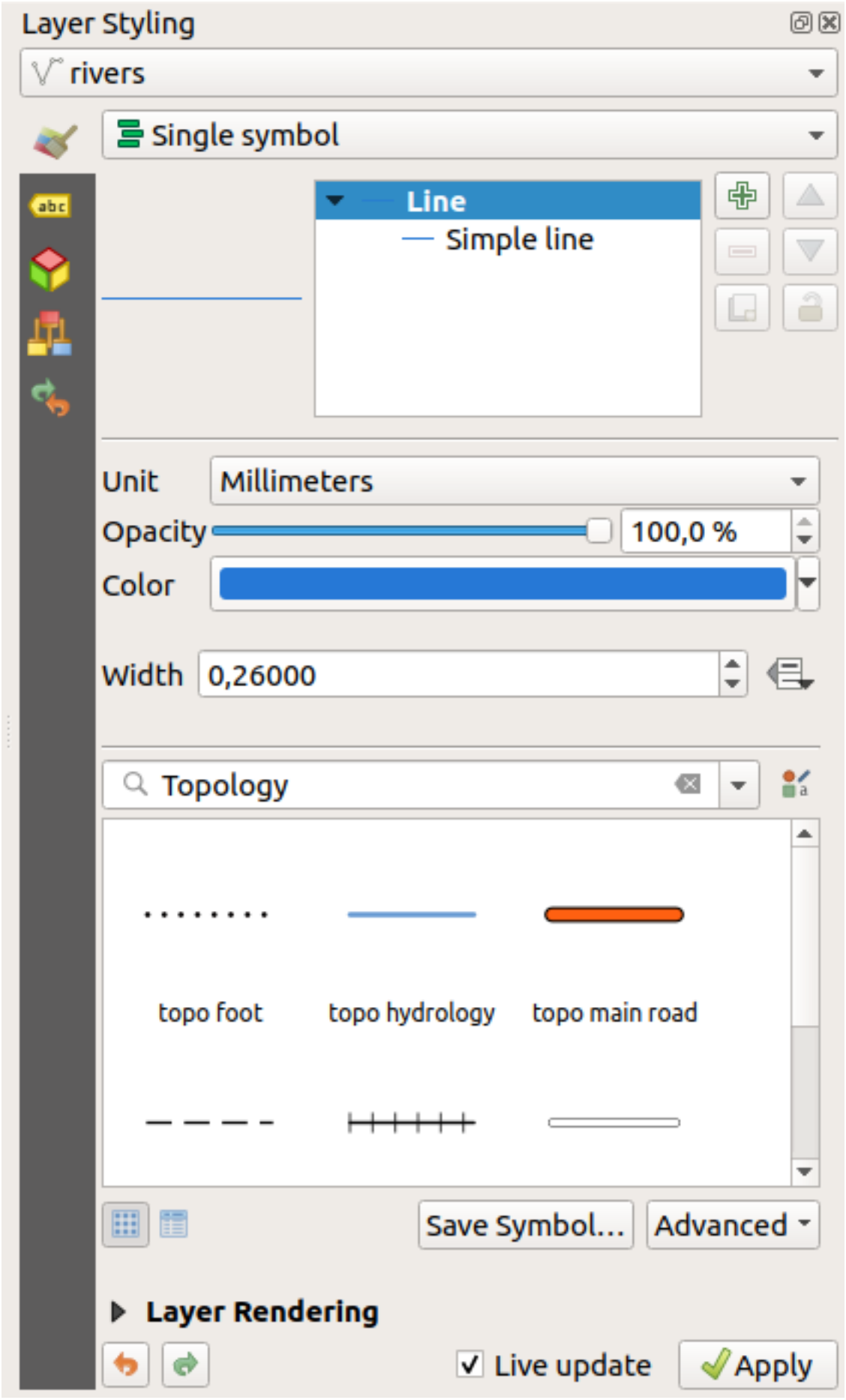
In addition to avoiding the blocking (or „modal") layer properties dialog, the layer styling panel also avoids cluttering the screen with dialogs, and contains most style functions (color selector, effects properties, rule edit, label

substitution…): e.g., clicking color buttons inside the layer style panel causes the color selector dialog to be opened inside the layer style panel itself rather than as a separate dialog.

From a drop-down list of current layers in the layer panel, select an item and:

- Depending on the active item, set:
  - *Symbology* for groups (see *Control layers rendering through grouping*)
  - *Symbology*, *Transparency*, and *Histogram* properties for raster layer. These options are the same as in the *Raster Properties Dialog*.
  - *Symbology*, *Labels*, *Mask* and *3D View* properties for vector layer. These options are the same as in the *The Vector Properties Dialog* and can be extended by custom properties introduced by third-party plugins.
  - *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 *Point Clouds Properties* or *3D Tiles Properties*.
  - *Symbology* and *Labels* properties for vector tiles layer. These options are the same as in the *Vector Tiles Dataset Properties*.
  - *Symbology* properties for annotations layer. These options are the same as in the *feature annotation properties*.
- Enable and configure *global map shading* properties
- Manage the associated style(s) in the *Style Manager* (more details at *Managing Custom Styles*).
- See the *History* of changes you applied to the layer style in the current project: you can therefore cancel or restore to any state by selecting it in the list and clicking *Apply*.

Another powerful feature of this panel is the *Live update* checkbox. Tick it to render your changes immediately on the map canvas: you no longer need to click the *Apply* button.



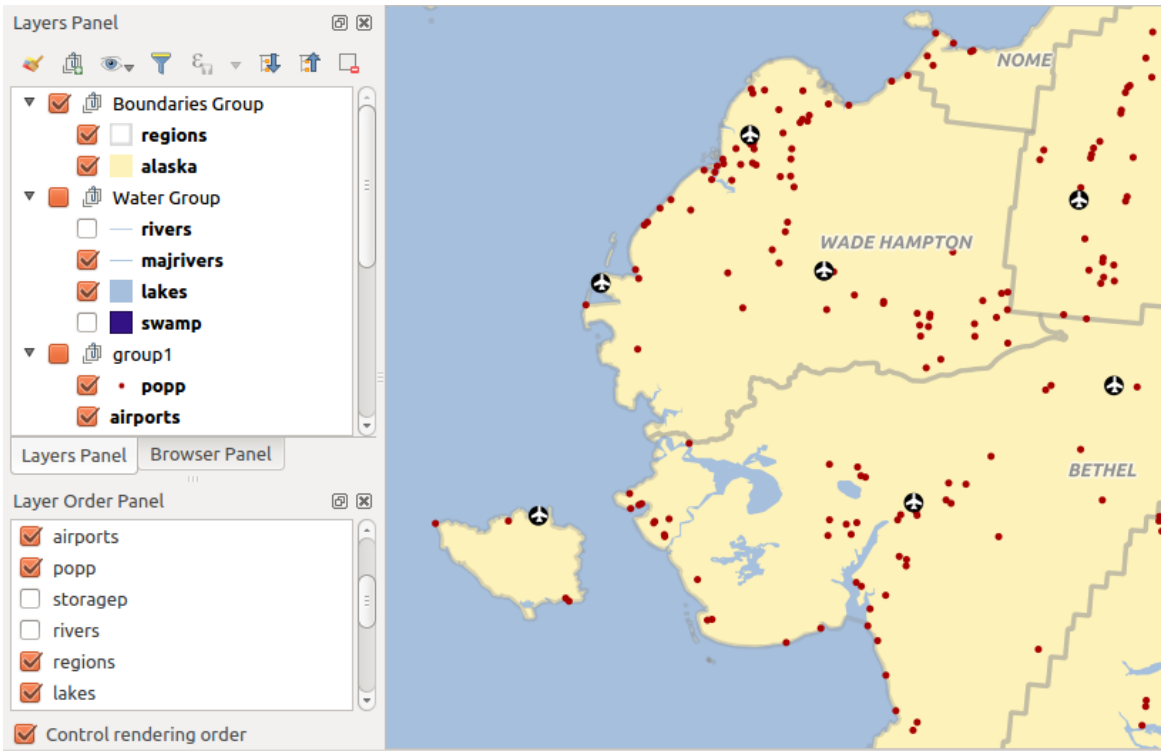
Слика8.3: Defining a layer's symbology from the layer styling panel

8.2.3 Layer Order Panel

By default, layers shown on the QGIS map canvas are drawn following their order in the *Layers* panel: the higher a layer is in the panel, the higher (hence, more visible) it'll be in the map view.

You can define a drawing order for the layers independent of the order in the layers panel with the *Layer Order* panel enabled in *View* ► *Panels* ► menu or with `Ctrl+9`. Check *Control rendering order* underneath the list of layers and reorganize the layers in the panel as you want. This order becomes the one applied to the map canvas. For example, in Сл.8.4, you can see that the `airports` features are displayed over the `alaska` polygon despite those layers' respective placement in the Layers panel.

Unchecking *Control rendering order* will revert to default behavior.



Слика8.4: Define a layer order independent of the legend

8.2.4 Overview Panel

The *Overview* panel (`Ctrl+8`) displays a map with a full extent view of some of the layers. The Overview map is filled with layers using the *Show in Overview* option from the *Layer* menu or in the layer contextual menu. Within the view, a red rectangle shows the current map canvas extent, helping you quickly to determine which area of the whole map you are currently viewing. If you click-and-drag the red rectangle in the overview frame, the main map view extent will update accordingly.

Note that labels are not rendered to the map overview even if the layers used in the map overview have been set up for labeling.

8.2.5 Log Messages Panel

When loading or processing some operations, you can track and follow messages that appear in different tabs using the *Log Messages Panel*. It can be activated using the most right icon in the bottom status bar.

8.2.6 Undo/Redo Panel

For each layer being edited, the *Undo/Redo* (`Ctrl+5`) panel shows the list of actions carried out, allowing you quickly to undo a set of actions by selecting the action listed above. More details at *Undo and Redo edits*.

8.2.7 Statistical Summary Panel

The *Statistics* panel (`Ctrl+6`) provides summarized information on any vector layer. This panel allows you to select:

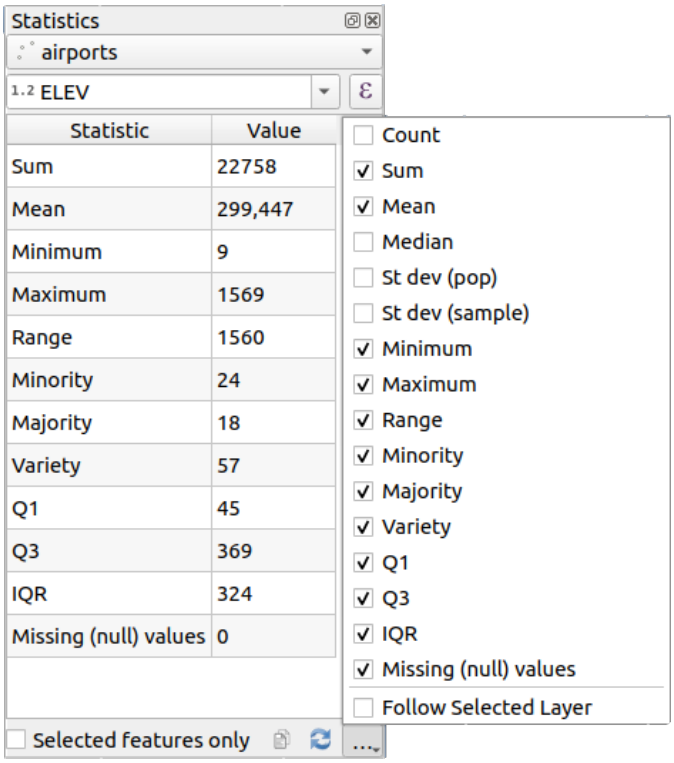
- the vector layer to compute the statistics on: it can be selected from the top drop-down menu, or synced with the active layer in the *Layers* panel using the *Follow selected layer* checkbox at the bottom of the statistics drop-down list
- the field or ε *expression* to use: for each layer, the last entry is remembered and automatically computed upon layer reselection.
- the statistics to return using the drop-down button at the bottom-right of the dialog. Depending on the field's (or expression's values) type, available statistics are:

Табела8.2: Statistics available for each field type

Statistika	Niska	Integer	Float	Datum
Count	☑	☑	☑	☑
Count Distinct Value	☑			☑
Count Missing value	☑	☑	☑	☑
Sum		☑	☑	
Mean		☑	☑	☑
Standard Deviation		☑	☑	
Standard Deviation on Sample		☑	☑	
Minimal value	☑	☑	☑	☑
Maximal value	☑	☑	☑	☑
Opseg		☑	☑	☑
Minority	☑	☑	☑	
Majority	☑	☑	☑	
Variety		☑	☑	
First Quartile		☑	☑	
Third Quartile		☑	☑	
Inter Quartile Range		☑	☑	
Minimum Length	☑			
Maximum Length	☑			
Mean Length	☑			

The statistical summary can be:

- returned for the whole layer or ☑ *Selected features only*
- copied to the clipboard and pasted as a table in another application
- recalculated using the button when the underlying data source changes (eg, new or removed features/fields, attribute modification)



Слика8.5: Show statistics on a field

8.2.8 Debugging/Development Tools Panel

The *Debugging/Development Tools* panel (`F12`) provides a unified place for handling and debugging actions within QGIS. Available tools are organized under following tabs:

- *API Documentation*: embeds developer oriented documentation
- *Network Logger*: records and displays network requests and interactions
- *Query Logger*: logs SQL commands issued by data providers and duration of execution
- *Profiler*: provides load times for every actions in order to detect causes of slow down

Белешка

Plugin authors can extend the panel with custom tabs for debugging and developing their own plugins. This is done using `registerDevToolWidgetFactory` method.

Dokumentacija API-ja

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.

Network Logger

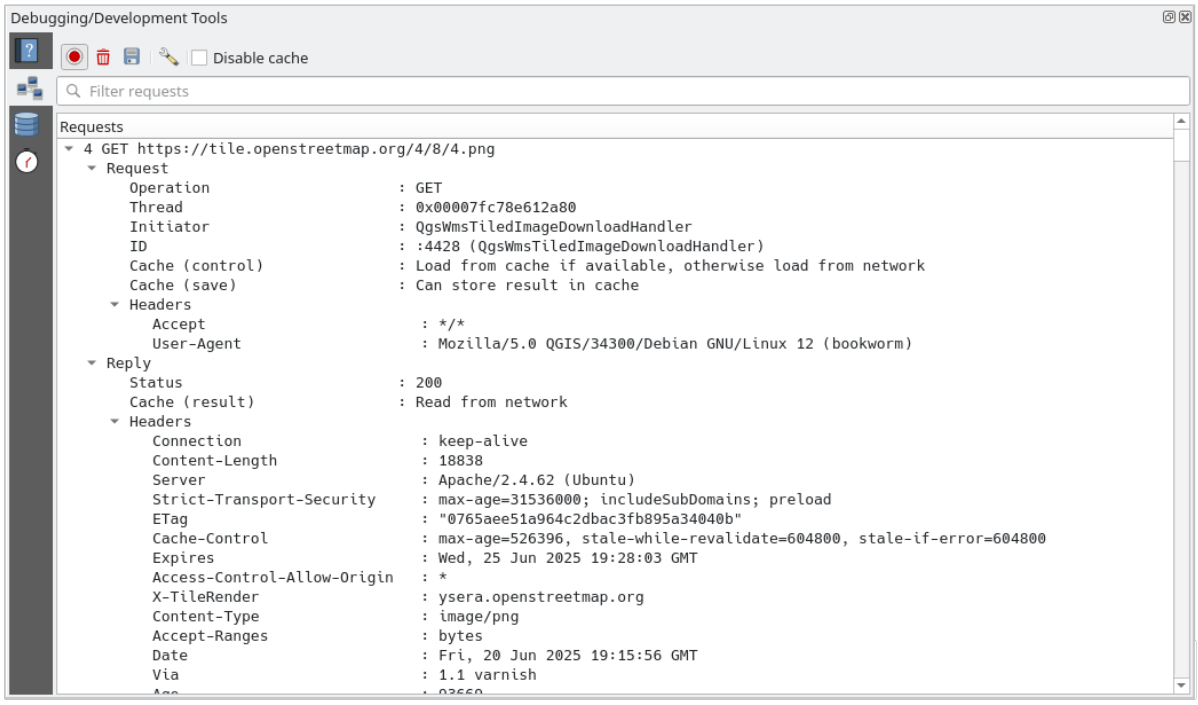
The *Network Logger* tab helps you record and display network requests, along with a whole load of useful details like request and reply status, header, errors, SSL configuration errors, timeouts, cache status, etc.

From its top toolbar, you can:

- *Record Log*: which will start or stop the logging.
- *Clear Log*: will clear the log history.
- *Save Log…*: will first show a big warning that the log is sensitive and should be treated as confidential and then allow you to save the log.
- Press the *Settings* drop-down menu to select whether to *Show Successful Requests*, *Show Timeouts* and *Show Replies Served from Cache*.
- *Disable cache*: will disable the cache so that every request has to be performed.
- *Filter requests* based on URL string subsets or request status

By right clicking on a request you can:

- *Open URL* which will open the URL in your default browser.
- *Copy URL*
- *Copy As cURL* to use it in the terminal.
- *Copy as JSON*: copies the tree values as a json string to the clipboard, for easy pasting in bug reports or for remote assistance.



Слика8.6: Network Logger output for GET Request

Query Logger

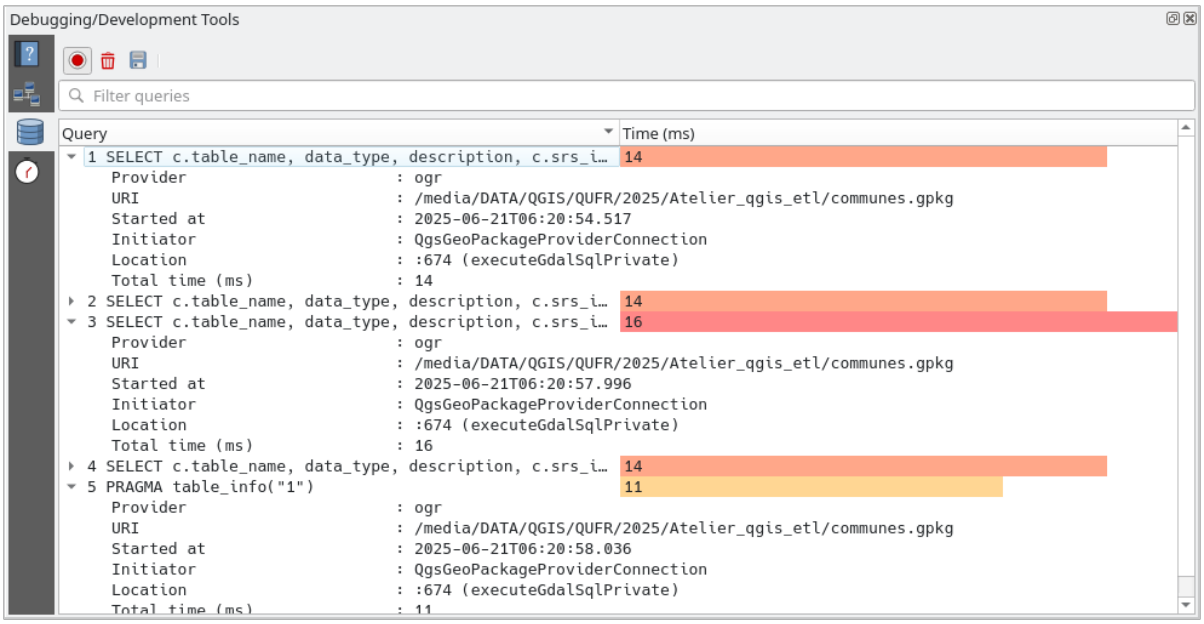
The *Query Logger* is the place to log the SQL commands sent by the data provider and the connections API to the backend database, together with their execution time as measured by QGIS (i.e. in the client that sent the commands). This can be useful when investigating performances of a particular layer during debugging or development of a QGIS algorithm or plugin.

From its top toolbar, you can:

- *Record Log*: which will start or stop the logging.
- *Clear Log*: will clear the log history.
- *Save Log…*: will first show a big warning that the log is sensitive and should be treated as confidential and then allow you to save the log.
- *Filter queries* based on the query string subsets or details such as provider type, start time, initiator, …

Right-click on a reported query, and you can:

- *Copy SQL* command called by QGIS on the database
- *Copy as JSON*: copies the tree values as a json string to the clipboard, for easy pasting in bug reports or for remote assistance.



Слика8.7: Query Logger output

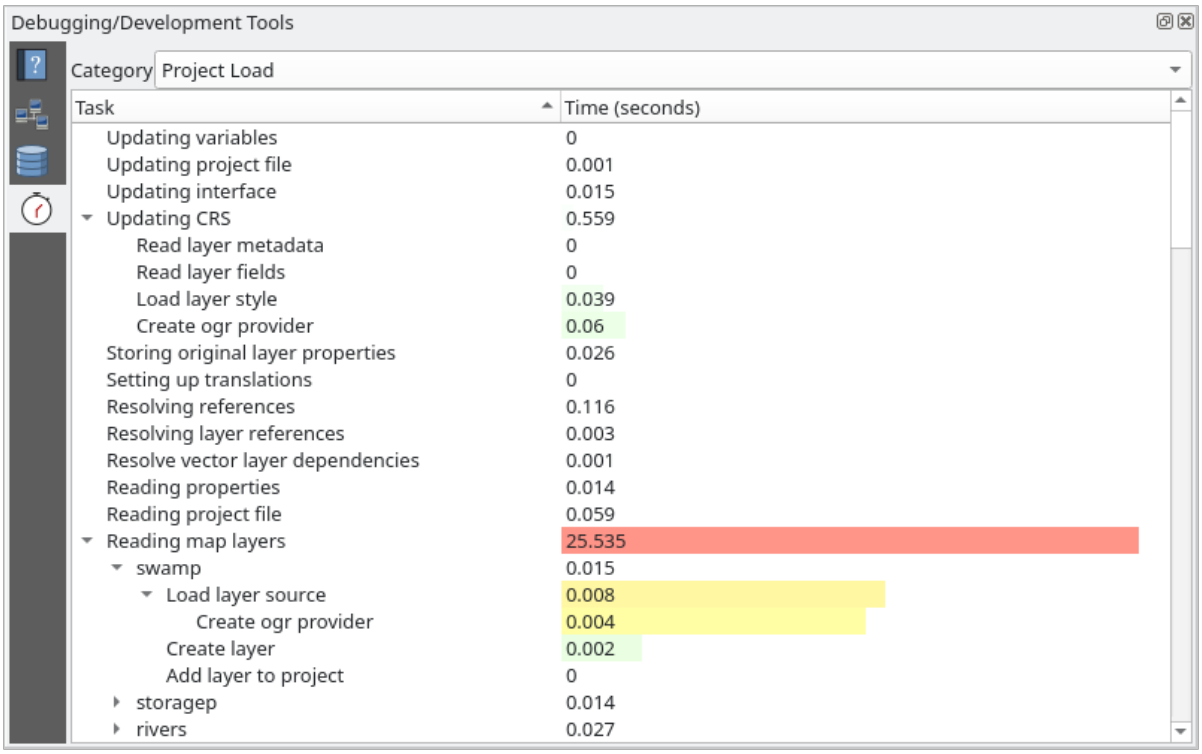
Profiler

The *Profiler* tab allows to get load times for every single operation involved in actions requested by the user. Depending on the context, these operations can be settings reading, menu, map canvas or 3D views creation, resolving map layers reference, bookmark or layout loading, … This helps identify causes of slow load times.

Default supported actions can be selected from the *Category* drop-down menu:

- *Startup* helps you diagnose (and fix) occasional long startup times of QGIS.
- *Project Load* allows you to get a breakdown of the various stages of project load, in order to identify the causes of slow project load times.

- *Map Render* tool allows you to identify pain points in your map rendering and track down exactly which layers are causing long map redraws.



Слика8.8: Profiler for QGIS Startup

8.3 Embedding layers from external projects

Sometimes, you'd like to keep some layers in different projects, but with the same style. You can either create a *default style* for these layers or embed them from another project to save time and effort.

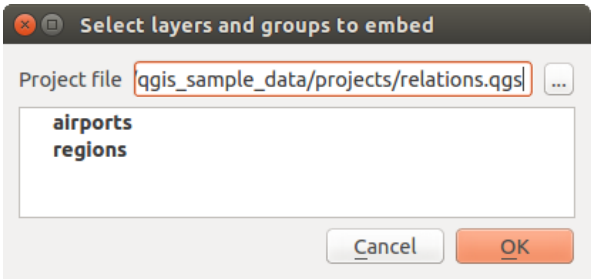
Embed layers and groups from an existing project has some advantages over styling:

- All types of layers (vector or raster, local or online…) can be added
- Fetching groups and layers, you can keep the same tree structure of the „background" layers in your different projects
- While the embedded layers are editable, you can't change their properties such as symbology, labels, forms, default values and actions, ensuring consistency across projects
- Modify the items in the original project and changes are propagated to all the other projects

If you want to embed content from other project files into your project, select *Layer* ► *Embed Layers and Groups*:

1. Click the … button to look for a project: you can see the content of the project (see Сл.8.9)
2. Hold down `Ctrl` (or `Cmd`) and click on the layers and groups you wish to retrieve
3. Kliknite na *OK*

The selected layers and groups are embedded in the *Layers* panel and displayed on the map canvas. An icon is added next to their name for recognition and hovering over displays a tooltip with the original project file path.



Слика8.9: Select layers and groups to embed

Like any other layer, an embedded layer can be removed from the project by right-clicking on the layer and clicking Remove.

Савет

Change rendering of an embedded layer

It's not possible to change the rendering of an embedded layer, unless you make the changes in the original project file. However, right-clicking on a layer and selecting *Duplicate* creates a layer which is fully-featured and not dependent on the original project. You can then safely remove the linked layer.

8.4 Interacting with features

8.4.1 Izbor objekata

QGIS provides several tools to select features on the map canvas. Selection tools are available in the *Edit* ► *Select* menu or in the *Selection Toolbar*.

Белешка

Selection tools work with the currently active layer.

Selecting manually on the map canvas

To select one or more features with the mouse, you can use one of the following tools:

- Select Features by area or single click
- Select Features by Polygon
- Select Features by Freehand
- Select Features by Radius

Белешка

Other than Select Features by Polygon, these manual selection tools allow you to select feature(s) on the map canvas with a single click.

Белешка

Use the  Select Features by Polygon tool to use an existing polygon feature (from any layer) to select overlapping features in the active layer. Right-click in the polygon and choose it from the context menu that shows a list of all the polygons that contain the clicked point. All the overlapping features from the active layer are selected.

Савет

Use the *Edit* ► *Select* ► *Reselect Features* tool to redo your latest selection. Very useful when you have painstakingly made a selection, and then click somewhere else accidentally and clear your selection.

While using the  *Select Feature(s)* tool, holding `Shift` or `Ctrl` (`Cmd` on macOS) toggles whether a feature is selected (i.e., either adds to the current selection or removes from it).

For the other tools, different behaviors can be performed by holding down:

- `Shift`: add features to the current selection
- `Ctrl`/`Cmd`: subtract features from the current selection
- `Ctrl`+`Shift`/`Cmd`+`Shift`: intersect with the current selection, i.e., only keep overlapping features from the current selection
- `Alt`: select features that are totally within the selection shape. Combined with `Shift` or `Ctrl`/`Cmd` keys, you can add or subtract features to/from the current selection.

Automatic selection

The other selection tools, most of them available from the *Attribute table*, perform a selection based on a feature's attribute or its selection state (note that attribute table and map canvas show the same information, so if you select one feature in the attribute table, it will be selected on the map canvas too):

-  Select Features By Expression… or press `Ctrl`+`F3` to select features using the expression dialog
-  Select Features By Value… or press `F3`
-  Deselect Features from All Layers or press `Ctrl`+`Alt`+`A` to deselect all selected features in all layers
-  Deselect Features from the Current Active Layer or press `Ctrl`+`Shift`+`A`
-  Select All Features or press `Ctrl`+`A` to select all features in the current layer
-  Invert Feature Selection to invert the selection in the current layer
-  Select by Location to select the features based on their spatial relationship with other features (in the same or another layer - see *Izbor po lokaciji*)
-  Select within distance to select features wherever they are within the specified maximum distance from referenced features - see *Izbor unutar rastojanja*)

For example, if you want to find regions that are boroughs from `regions.shp` of the QGIS sample data, you can:

1. Use the  Select features using an Expression icon
2. Expand the *Fields and Values* group
3. Double-click the field that you want to query („TYPE_2")
4. Click *All Unique* in the panel that shows up on the right

5. From the list, double-click 'Borough'. In the *Expression* editor field, write the following query:

```
"TYPE_2"  =  'Borough'
```

6. Click *Select Features*

From the expression builder dialog, you can also use *Function list* ► *Recent (Selection)* to make a selection that you have used before. The dialog remembers the last 20 expressions used. See *Expressions* for more information and examples.

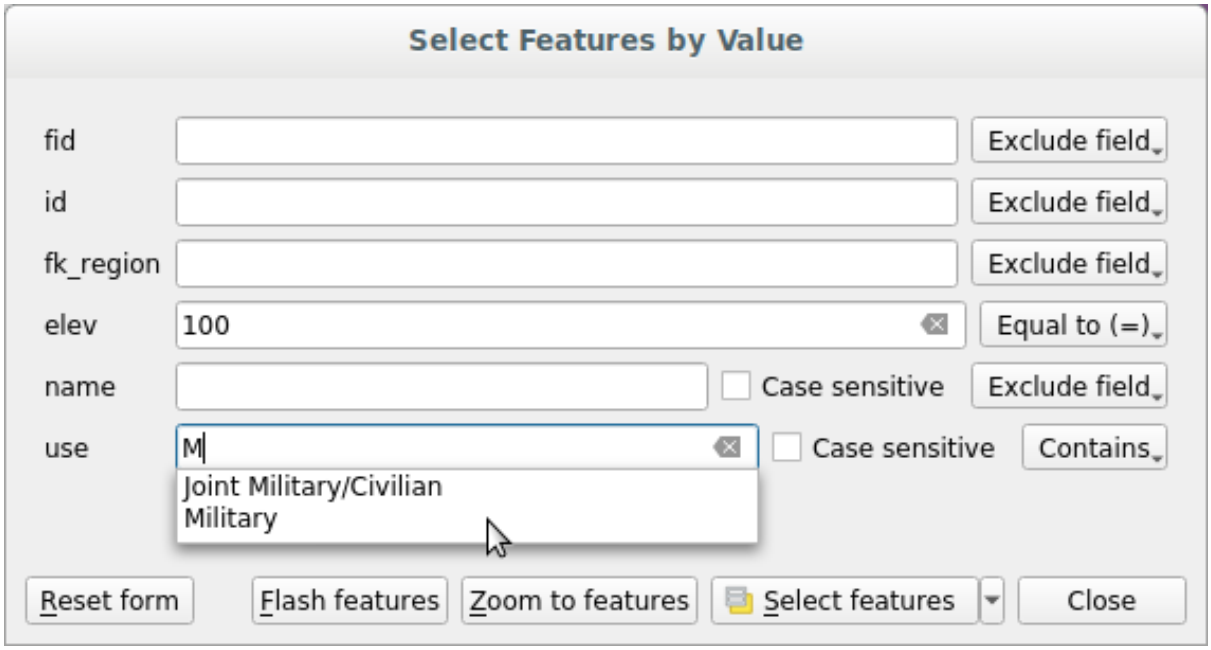
Савет

Save your selection into a new file

Users can save selected features into a **New Temporary Scratch Layer** or a **New Vector Layer** using *Edit* ► *Copy Features* and *Edit* ► *Paste Features as* in the desired format.

Select Features By Value

This selection tool opens the layer's feature form allowing the user to choose which value to look for for each field, whether the search should be case-sensitive, and the operation that should be used. The tool has also autocompletes, automatically filling the search box with existing values.



Слика8.10: Filter/Select features using form dialog

Alongside each field, there is a drop-down list with options to control the search behaviour:

Табела8.3: Query operators per data type

Field search option	Niska	Numeric	Datum
Exclude Field from the search	☑	☑	☑
Equal to (=)	☑	☑	☑
Not equal to (≠)	☑	☑	☑
Greater than (>)		☑	☑
Less than (<)		☑	☑
Greater than or equal to (≥)		☑	☑
Less than or equal to (≤)		☑	☑
Between (inclusive)		☑	☑
Not between (inclusive)		☑	☑
Contains	☑		
Does not contain	☑		
Is missing (null)	☑	☑	☑
Is not missing (not null)	☑	☑	☑
Starts with	☑		
Ends with	☑		

For string comparisons, it is also possible to use the ☑ *Case sensitive* option.

After setting all search options, click *Select features* to select the matching features. The drop-down options are:

- *Select features*
- *Add to current selection*
- *Remove from current selection*
- *Filter current selection*

You can also clear all search options using the *Reset form* button.

Once the conditions are set, you can also either:

- *Zoom to features* on the map canvas without the need of a preselection
- *Flash features*, highlighting the matching features. This is a handy way to identify a feature without selection or using the Identify tool. Note that the flash does not alter the map canvas extent and would be visible only if the feature is within the bounds of the current map canvas.

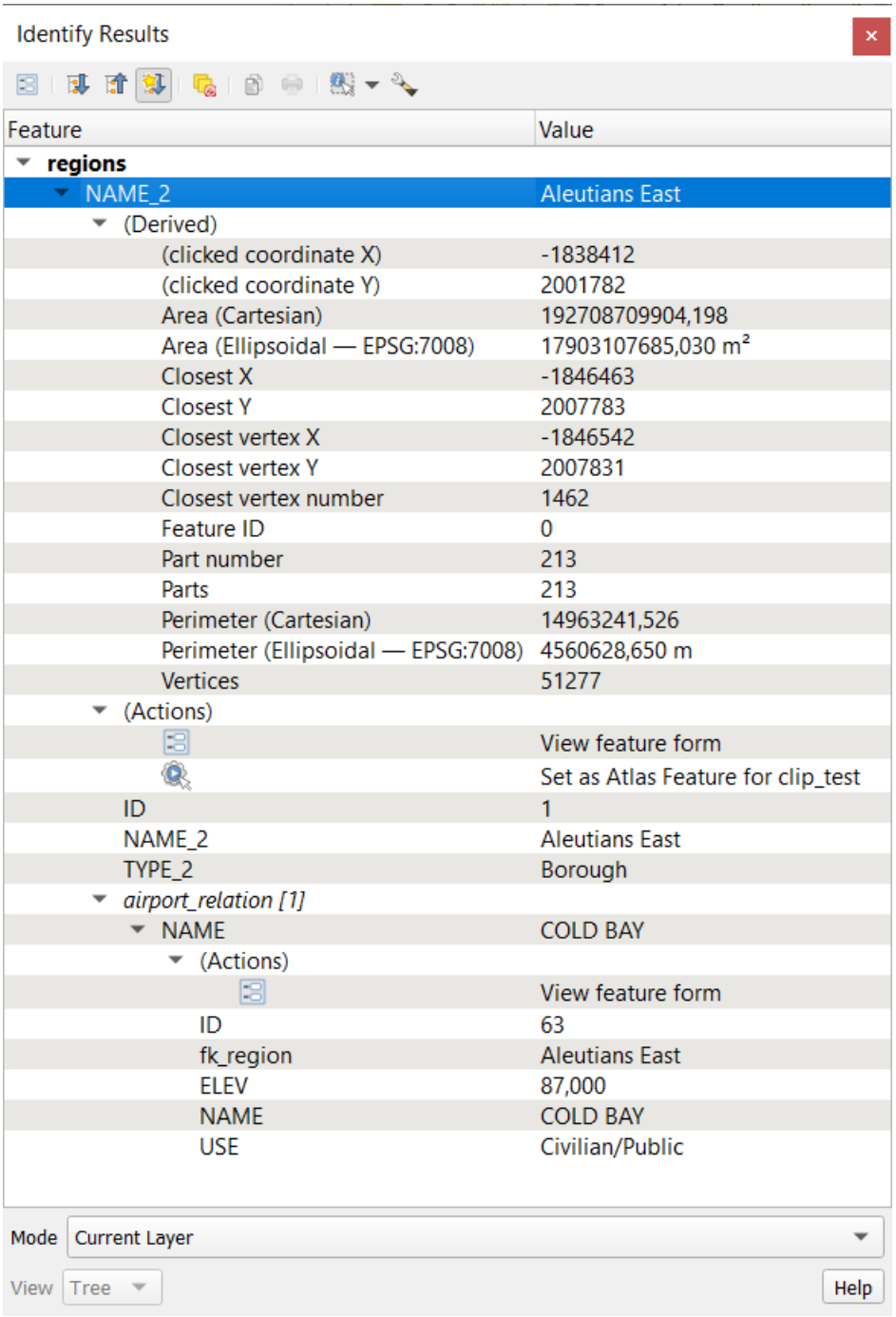
8.4.2 Identifying Features

The Identify Features tool allows you to interact with the map canvas and get information on features or pixels in a pop-up window. It can be used to query most of the layer types supported by QGIS (vector, raster, mesh, point cloud, wms, wfs, …). To identify an element, use either:

- *View* ► *Identify Features*
- `Ctrl+Shift+I` (or `Cmd+Shift+I`),
- Identify Features button on the *Attributes* toolbar

Then click on a feature or pixel of the active layer. The identified item gets highlighted in the map canvas while the *Identify Results* dialog opens with detailed information on it. The dialog also shows a set of buttons for advanced configuration.

The Identify Results dialog



Слика8.11: Dijalog Rezultati identifikacije

From bottom to top:

- The *View* controls the general aspect of the dialog and the formatting of the results; it can be set as:
  - **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).
  - **Table**: available only for raster-based layers, it allows to display the results as a table whose columns are `Layer`, `FID`, `Attribute` and `Value`
  - or **Graph**: available only for raster-based layers

- The *Mode* helps you select the layers from which results could be returned. These layers should be set visible, displaying data in the map canvas, and set *identifiable* from the *Project properties* ► *Data Sources* ► *Layers capabilities*. Available modes are:
  - **Current layer**: only the layer(s) selected in the *Layers* panel return results. If a group is selected, then results are picked from its leaf layers.
  - **Top down, stop at first**: results are from the layer of the top most feature or pixel under the mouse.
  - **Top down**: results are from the layers with feature or pixel under the mouse.
  - **Layer selection**: opens a contextual menu where the user selects the layer to identify features from. If only a single feature is under the mouse, then the results are automatically displayed.
- 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:
  - Open Form of the current feature
  - Expand tree
  - Collapse tree
  - Expand New Results by Default to define whether the next identified feature's information should be collapsed or expanded
  - Clear Results
  - Copy the identified feature to clipboard, suitable for pasting in a spreadsheet.
  - Print selected HTML response: a text-based formatting of the results to print on paper or save as a `.PDF` file
  - the *interactive identifying tools*: a drop-down menu with tools for selecting on the map canvas features or pixels to identify
  - Under Identify Settings, you can activate whether to:
    - *Auto open form for single feature results*: If checked, each time a single feature is identified, a form opens showing its attributes. This is a handy way to quickly edit a feature's attributes.
    - *Hide derived attributes from results* to only show fields actually defined in the layer
    - *Hide NULL values from results*
    - *Show Relations* to show relations in the vector layer *Identify results tree*
  - Help to access the current documentation

Using the Identify Features tool

In its default display (*View: Tree*), the *Identify Results* panel offers several tools to interact with the layers to query. A smart combination of these tools with the *target layers selector* may greatly improve identification operations:

- Identify Feature(s) by single click or click-and-drag

  - single click or click-and-drag: overlaying features in the target layers are returned
  - right-click: overlaying features from target layers are listed in the contextual menu, grouped by layers. You can then choose to:
    * display the result for a specific feature,
    * display the result for all the features of a specific layer,
    * for vector layers, it is also possible to open its attribute table filtered to the returned features
    * or show all of the returned features.

-  Identify Features on Mouse over: move over the map canvas and hovered items in the target layers get highlighted and returned in the results panel.
-  Identify Features by Polygon: returns items overlapping a drawn or selected polygon.
  - Draw a polygon (left click to add point, right click to close the polygon) and all the overlaying features from target layers are highlighted and returned in the results panel.
  - Right-click and you get the list of all visible polygon features in the project under the click. Pick an entry and QGIS will return all the features from the target layers that overlap the selected polygon.
-  Identify Features by Freehand: returns items overlapping a polygon drawn by freehand. Draw a polygon (left-click to start, move the pointer to shape the area and right-click to close the polygon). All the overlaying features from target layers are highlighted and returned in the results panel.
-  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.

Савет

Filter the layers to query with the Identify Features tool

Under *Project* ► *Properties…* ► *Data Sources* ► *Layer Capabilities*, uncheck the *Identifiable* column next to a layer to avoid it being queried when using the  Identify Features tool. This is a handy way to return features from only layers that are of interest to you.

Results contextual menu

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 Save and Share Layer Properties

8.5.1 Managing Custom Styles

When a vector layer is added to the map canvas, QGIS by default uses a random symbol/color to render its features. However, you can set a default symbol in *Project* ► *Properties…* ► *Default styles* that will be applied to each newly added layer according to its geometry type.

Most of the time, though, you'd rather have a custom and more complex style that can be applied automatically or manually to the layers (with less effort). You can achieve this by using the *Style* menu at the bottom of the Layer Properties dialog. This menu provides you with functions to create, load and manage styles.

A style stores various information set in the layer or project properties dialog to render or interact with the layer. This includes:

Табела8.4: Components of stored style and their references

Category	Vektor	Raster
Layer Configuration	*Layers capabilities*	
	Display Properties	*Display Properties*
Symbology	*Symbology Properties*	*Symbology Properties* ([1])
3D Symbology	*3D View Properties*	N/A
Labels	*Labels Properties*	N/A
Fields	*Fields Properties* and *Constraints*	N/A
Attributes Form	*Attributes Form Properties*	N/A
Actions	*Actions Properties*	N/A
Map Tips	*Display Properties*	*Display Properties*
Diagrams	*Diagrams Properties*	N/A
Attribute Table Configuration	*Rad sa atributskom tabelom*	
Rendering	*Rendering Properties*	*Rendering Properties* ([1])
Custom Properties	*Information Properties*	*Information Properties*
Geometry Options	*Digitizing Properties*	N/A
Relations	*Setting relations between multiple layers*	N/A
Temporal Properties	*Temporal Properties*	*Temporal Properties*
Legend Settings	*Legend Properties*	N/A
Elevation Properties	*Elevation Properties*	*Elevation Properties*
Notes	*Layer notes*	

By default, the style applied to a loaded layer is named `default`. Once you have got the ideal and appropriate rendering for your layer, you can save it by clicking the *Style* combo box and choosing:

- **Rename Current**: The active style is renamed and updated with the current options
- **Add**: A new style is created using the current options. By default, it will be saved in the QGIS project file. See below to save the style in another file or a database
- **Remove**: Delete unwanted style, in case you have more than one style defined for the layer.

At the bottom of the Style drop-down list, you can see the styles set for the layer with the active one checked.

Note that each time you validate the layer properties dialog, the active style is updated with the changes you've made.

You can create as many styles as you wish for a layer but only one can be active at a time. In combination with *Map Themes*, this offers a quick and powerful way to manage complex projects without the need to duplicate any layer in the map legend.

Белешка

Given that whenever you apply modifications to the layer properties, changes are stored in the active style, always ensure you are editing the right style to avoid mistakenly altering a style used in a *map theme*.

[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.

Савет

Manage styles from layer context menu

Right-click on the layer in the *Layers* panel to copy, paste, add or rename layer styles. For both vector and raster layers, you can choose which style categories to copy/paste. Choose *All Style Categories* or one of the available categories, depending on the layer type. Note that for raster layers, the *Symbology* and *Rendering* categories are always copied together.

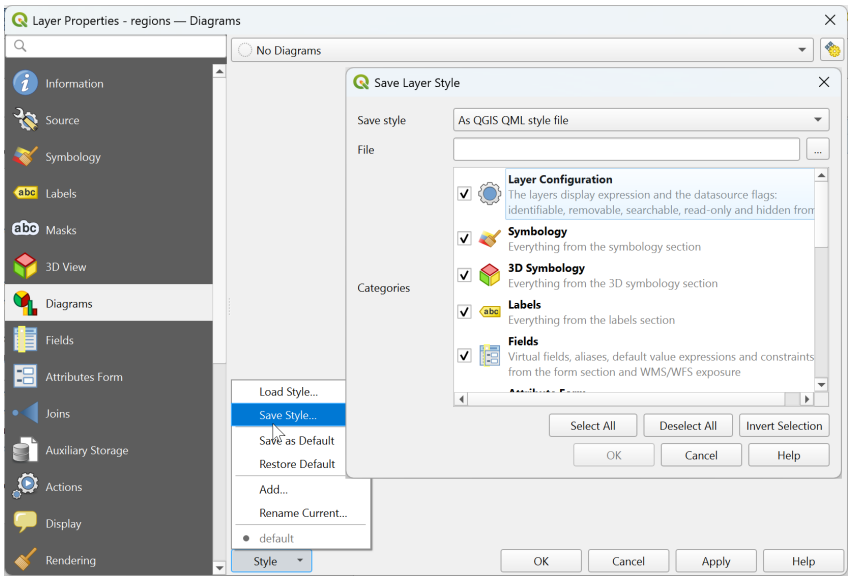
8.5.2 Storing Styles in a File or a Database

While styles created from the *Style* combo box are by default saved inside the project and can be copied and pasted from layer to layer in the project, it's also possible to save them outside the project so that they can be loaded in another project.

Save as text file

Clicking the  *Style* ► *Save Style*, you can save the style as a:

- QGIS layer style file (`.qml`)
- SLD file (`.sld`), only available for vector layers



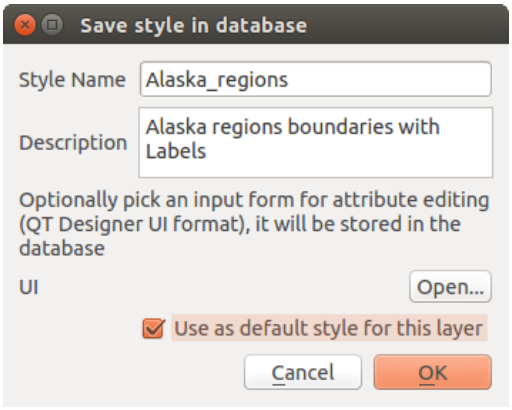
Слика8.12: Vector layer style combo box options

Used on file-based format layers (`.shp`, `.tab`…), *Save as Default* generates a `.qml` file for the layer (with the same name). SLDs can be exported from any type of renderer – single symbol, categorized, graduated or rule-based – but when importing an SLD, either a single symbol or rule-based renderer is created. This means that categorized or graduated styles are converted to rule-based. If you want to preserve those renderers, you have to use the QML format. On the other hand, it can be very handy sometimes to have this easy way of converting styles to rule-based.

Save in database

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.

You can save several styles for a single table in the database. However, each table can have only one default style. Default styles can be saved in the layer database or in `qgis.db`, a local SQLite database in the active *user profile* directory.



Слика8.13: Save Style in database Dialog

Савет

Sharing style files between databases

You can only save your style in a database if the layer comes from such a database. You can't mix databases (layer in Oracle and style in MS SQL Server for instance). Use instead a plain text file if you want the style to be shared among databases.

Белешка

You may encounter issues restoring the `layer_styles` table from a PostgreSQL database backup. Follow *QGIS layer_style table and database backup* to fix that.

Load style

When loading a layer in QGIS, if a default style already exists for this layer, QGIS loads the layer with this style. Also *Style* ► *Restore Default* looks for and loads that file, replacing the layer's current style.

Style ► *Load Style* helps you apply any saved style to a layer. While text-file styles (`.sld` or `.qml`) can be applied to any layer whatever its format, loading styles stored in a database is only possible if the layer is from the same database or the style is stored in the QGIS local database.

The *Database Styles Manager* dialog displays a list of styles related to the layer found in the database and all the other styles saved in it, with name and description.

Савет

Quickly share a layer style within the project

You can also share layer styles within a project without importing a file or database style: right-click on the layer in the *Layers Panel* and, from the *Styles* combo box , copy the style of a layer and paste it to a group or a selection of layers: the style is applied to all the layers that are of the same type (vector vs raster) as the original layer and, in the case of vector layers, have the same geometry type (point, line or polygon).

8.5.3 Layer definition file

Layer definitions can be saved as a `Layer Definition File` (`.qlr`) using *Export* ► *Save As Layer Definition File…* in the active layers' context menu. A layer definition file (`.qlr`) includes references to the data source of the layers and their styles. `.qlr` files are shown in the Browser Panel and can be used to add the layers (with the saved style) to the Layers Panel. You can also drag and drop `.qlr` files from the system file manager into the map canvas.

8.6 Documenting your data

In addition to displaying and symbolizing the data in the layers, QGIS allows you to fill:

- **metadata**: information to help people find and understand the dataset, how they can access and use it… these are properties of the datasource and can live out of the QGIS project. Metadata can also be defined for the project file itself.
- **notes**: instructions and comments regarding the layer in the current project

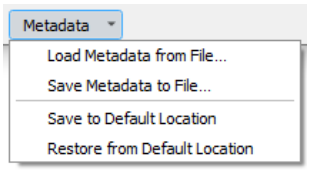
8.6.1 Metadata

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:

- the data *Identification*: basic attribution of the dataset (parent, identifier, title, abstract, language…);
- the *Categories* the data belongs to. Alongside the **ISO** categories, you can add custom ones;
- the *Keywords* to retrieve the data and associated concepts following a standard based vocabulary;
- the *Access* to the dataset (licenses, rights, fees, and constraints);
- the *Extent* of the dataset, either spatial one (CRS, map extent, altitudes) or temporal;
- the *Contact* of the owner(s) of the dataset;
- the *Links* to ancillary resources and related information;
- the *History* of the dataset.

A summary of the filled information is provided in the *Validation* tab and helps you identify potential issues related to the form. You can then either fix them or ignore them.

Metadata are saved in the project file by default, the *Metadata* drop-down offers options for loading/saving metadata from `.qmd` file and for loading/saving metadata in the „Default" location.



Слика8.14: Metadata load/save options

The „Default" location used by *Save to Default Location* and *Restore from Default Location* changes depending on the underlying data source and on its configuration:

- For PostgreSQL data sources if the configuration option *Allow saving/loading QGIS layer metadata in the database* is checked the metadata are stored inside a dedicated table in the database.
- For GeoPackage data sources *Save to Default Location* always saves the metadata in the internal metadata tables of the GeoPackage.

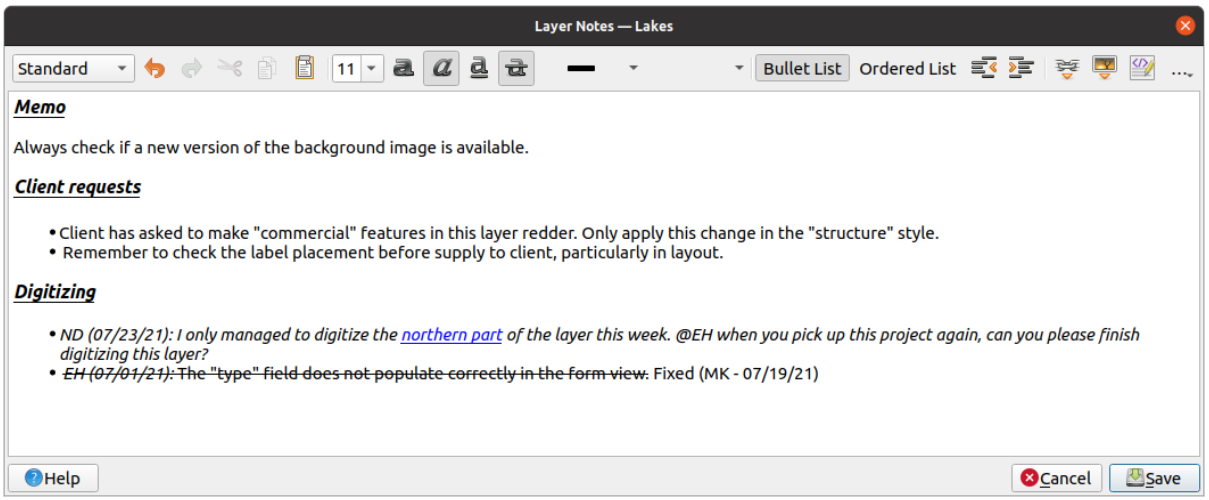
  When metadata are saved into the internal tables of PostgreSQL or GeoPackage they become available for search and filtering in the browser and in the *layer metadata search panel*.

- For all other file based data sources *Save to Default Location* saves the metadata in a `.qmd` file alongside the file.
- In all other cases *Save to Default Location* saves the metadata in a local `.sqlite` database.

8.6.2 Layer notes

Layer notes allow you to document the layer within the current project. They can be place to store important messages for users of the project like to do lists, instructions, warnings, …

From the layer's contextual menu in *Layers* panel, select *Add layer notes…* and fill the open dialog with necessary texts.



Слика8.15: Adding notes to a layer

The *Add layer notes* dialog provides a html-based multiline text box with a complete set of tools for:

- text manipulation: cut, copy, paste, undo, redo
- characters formatting, applied to all or parts of the contents: font size and color, bold, italic, underline, strikethrough, background color, URL highlighting
- paragraph structuring: bullet and numbered lists, indentation, predefined headings
- file insertion, even with drag-and-drop
- editing with HTML coding

From the … drop-down at the far right of the toolbar, you can:

- *Remove all formatting*
- *Remove character formatting*
- *Clear all content*

In the *Layers* panel, a layer with a note is assigned the  icon which, upon hover, displays the note. Click the icon to edit the note. You can as well right-click the layer and *Edit layer note…* or *Remove layer note*.

Белешка

Notes are part of the *layer style* and can be saved in the `.qml` or `.qlr` file. They can also be transferred from one layer to another while copy-pasting the layer style.

8.7 Storing values in Variables

In QGIS, you can use variables to store useful recurrent values (e.g. the project's title, or the user's full name) that can be used in expressions. Variables can be defined at the application's global level, project level, layer level, processing modeler level, layout level, and layout item's level. Just like CSS cascading rules, variables can be overwritten - e.g., a project level variable will overwrite any application global level variables set with the same name. You can use these variables to build text strings or other custom expressions using the `@` character before the variable name. For example in print layout creating a label with this content:

```
This map was made using QGIS [% @qgis_version %]. The project file for this
map is: [% @project_path %]
```

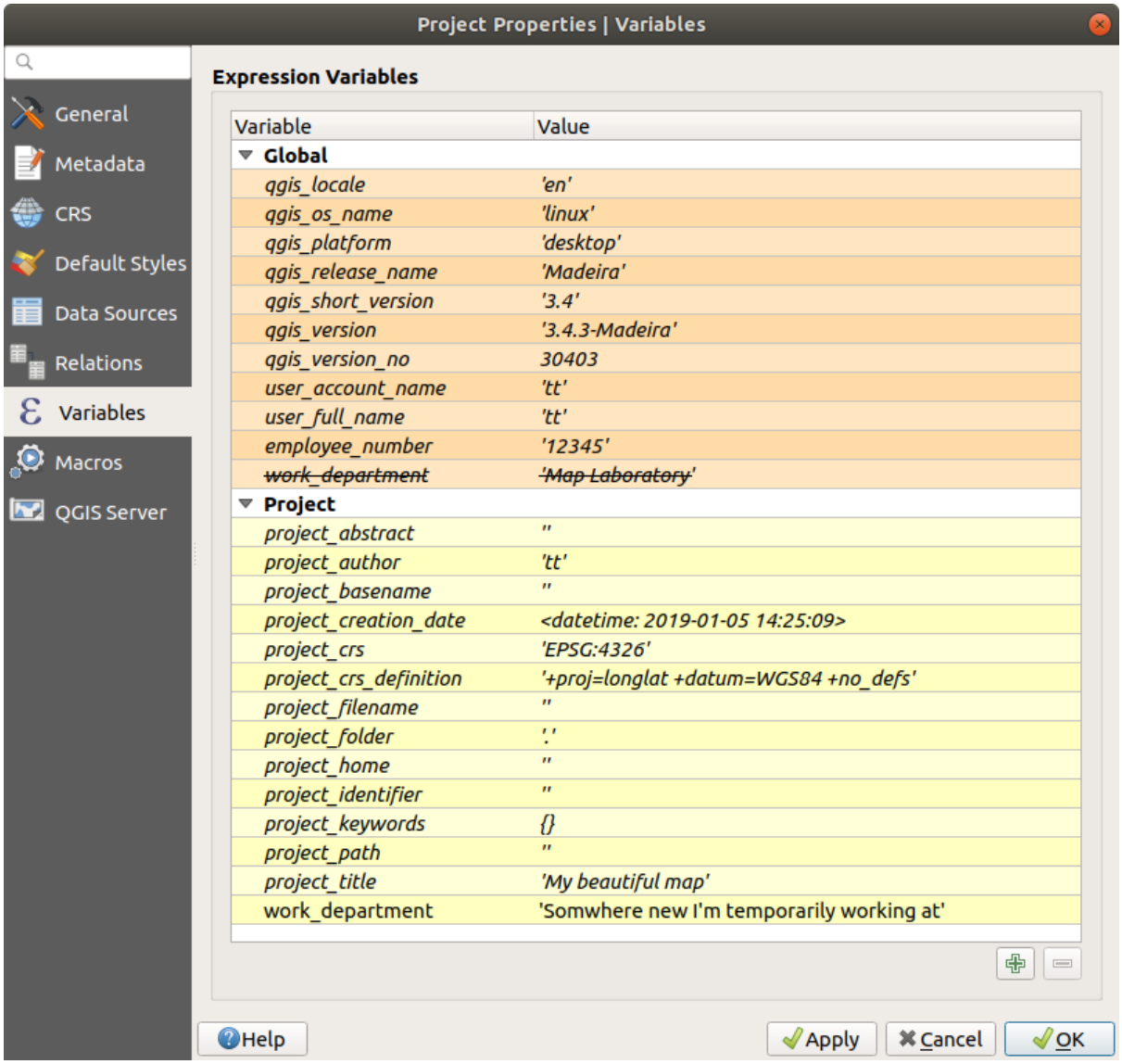
Will render the label like this:

```
This map was made using QGIS 3.4.4-Madeira. The project file for this map is:
/gis/qgis-user-conference-2019.qgs
```

Besides the *preset read-only variables*, you can define your own custom variables for any of the levels mentioned above. You can manage:

- **global variables** from the *Settings* ► *Options* menu
- **project variables** from the *Project Properties* dialog (see *Project Properties*)
- **vector layer variables** from the *Layer Properties* dialog (see *The Vector Properties Dialog*);
- **modeler variables** from the *Model Designer* dialog (see *Dizajner modela*);
- **layout variables** from the *Layout* panel in the Print layout (see *Panel Raspored*);
- and **layout item variables** from the *Item Properties* panel in the Print layout (see *Zajedničke opcije stavki rasporeda*).

To differentiate from editable variables, read-only variable names and values are displayed in italic. On the other hand, higher level variables overwritten by lower level ones are strike through.



Слика8.16: Variables editor at the project level

Белешка

You can read more about variables and find some examples in Nyall Dawson's Exploring variables in QGIS 2.12, part 1, part 2 and part 3 blog posts.

8.8 Common widgets

In QGIS, there are some options you'll often have to work with. For convenience, QGIS provides you with special widgets that are presented below.

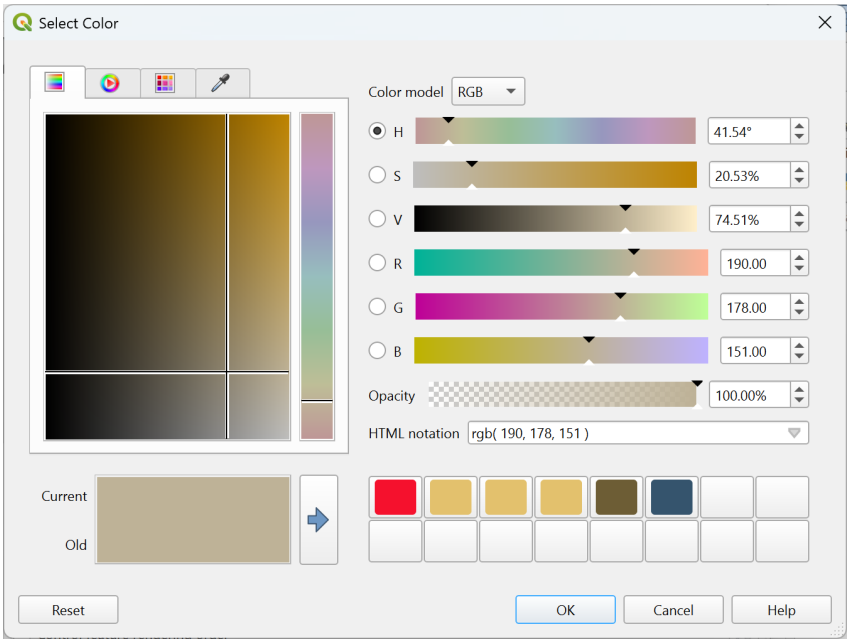
8.8.1 Color Selector

The color dialog

The *Select Color* dialog will appear whenever you click the  icon to choose a color. The features of this dialog depend on the state of the *Use native color chooser dialogs* parameter checkbox in *Settings* ► *Options…*

► *General*. When checked, the color dialog used is the native one of the OS on which QGIS is running. Otherwise, the QGIS custom color chooser is used.

The custom color chooser dialog has four different tabs which allow you to select colors by Color ramp, Color wheel, Color swatches or Color picker. With the first two tabs, you can browse to all possible color combinations and apply your choice to the item.

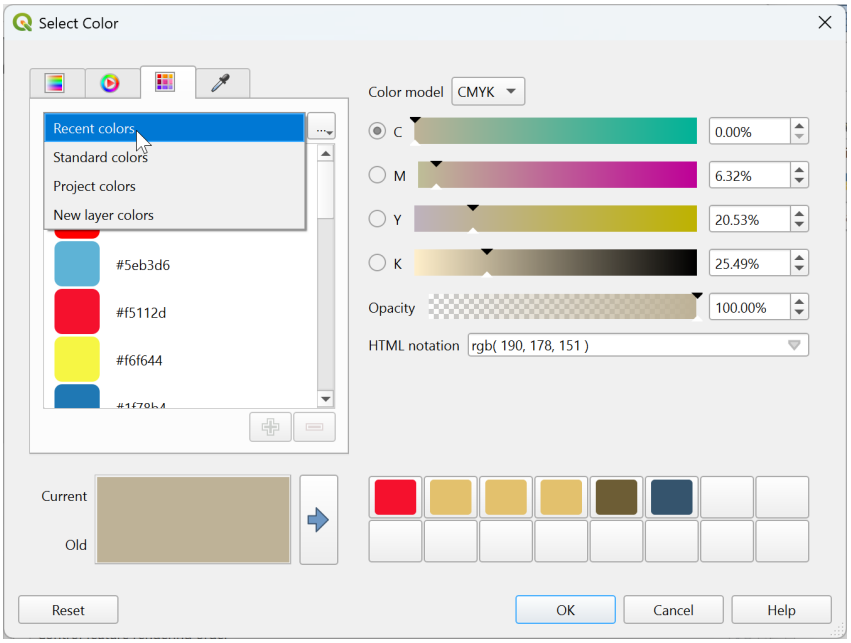


Слика8.17: Color selector ramp tab

In the Color swatches tab, you can choose from a list of color palettes (see *Colors settings* for details). All but the *Recent colors* palette can be modified with the Add current color and Remove selected color buttons at the bottom of the frame.

The … button next to the palette combo box also offers several options to:

- copy, paste, import or export colors
- create, import or remove color palettes
- add the custom palette to the color selector widget with the *Show in Color Buttons* item (see Сл.8.19)



Слика8.18: Color selector swatches tab

Another option is to use the  Color picker which allows you to sample a color from under your mouse cursor at any part of the QGIS UI or even from another application: press the space bar while the tab is active, move the mouse over the desired color and click on it or press the space bar again. You can also click the *Sample Color* button to activate the picker.

Whatever method you use, the selected color is always described through options of a *Color model* that defaults to the *project color model*:

- *RGB*: color is set using sliders for `HSV` (Hue, Saturation, Value) or `RGB` (Red, Green, Blue) values.
- *CMYK* (Cyan, Magenta, Yellow, Black)

Modifying a color is as simple as clicking on the color wheel or ramp or on any of the color parameters sliders. You can adjust such parameters with the spinbox beside or by scrolling the mouse wheel over the corresponding slider. You can also type the color in HTML notation. Finally, there is an *Opacity* slider to set transparency level.

The dialog also provides a visual comparison between the *Old* color (applied to object) and the *Current* one (being selected). Using drag-and-drop or pressing the  Add color to swatch button, any of these colors can be saved in a slot for easy access.

Савет

Quick color modification

Drag-and-drop a color selector widget onto another one to apply its color.

The color drop-down shortcut

Click the drop-down arrow to the right of the  color button to display a widget for quick color selection. This shortcut provides access to:

- a color wheel to pick a color from
- an alpha slider to change color opacity
- the color palettes previously set to *Show in Color Buttons*

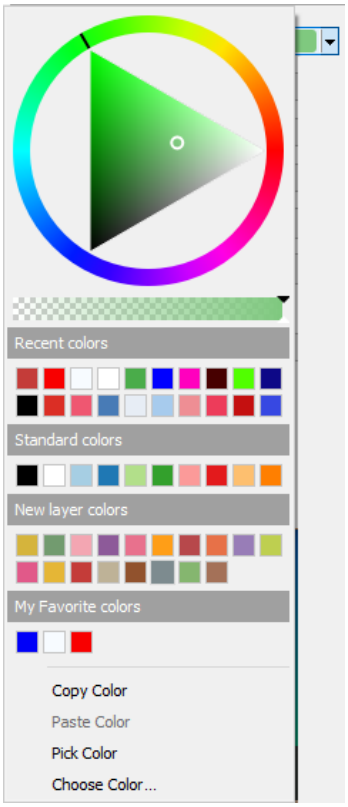
- copy the current color and paste it into another widget
- pick a color from anywhere on your computer display
- choose a color from the color selector dialog
- drag-and-drop the color from one widget to another for quick modification

Савет

Scroll the mouse wheel over a color selector widget to quickly modify the opacity of the associated color.

Белешка

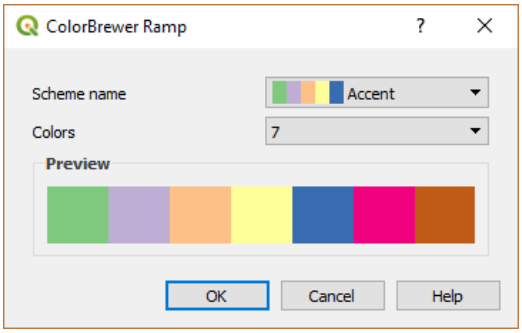
When the color widget is set to a *project color* through the data-defined override properties, the above functions for changing the color are unavailable. You'd first need to *Unlink color* or *Clear* the definition.



Слика8.19: Quick color selector menu

The color ramp drop-down shortcut

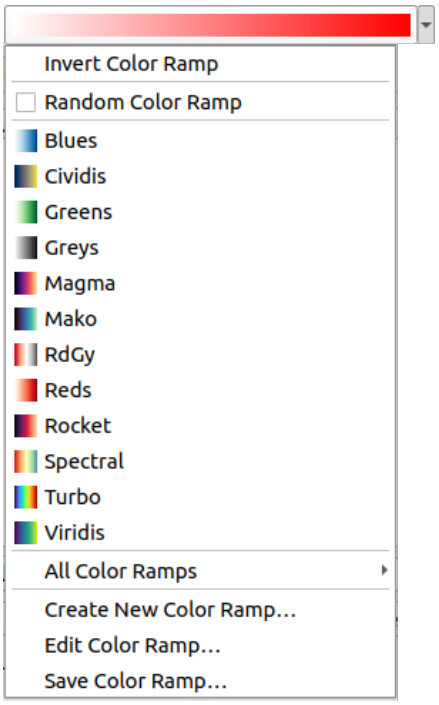
Color ramps are a practical way to apply a set of colors to one or many features. Their creation is described in the *Setting a Color Ramp* section. As for the colors, pressing the  color ramp button opens the corresponding color ramp type dialog allowing you to change its properties.



Слика8.20: Customizing a colorbrewer ramp

The drop-down menu to the right of the button gives quick access to a wider set of color ramps and options:

- *Invert Color Ramp*
- *Clear Current Ramp* to unset any assigned color ramp to the widget (available only in some contexts)
- *Random Colors*: available only in some contexts (e.g., when a color ramp is being used for a layer symbology), checking this entry creates and applies a color ramp with random colors. It also enables a *Shuffle random colors* entry to regenerate a new random color ramp if the current one is not satisfactory.
- a preview of the `gradient` or `catalog: cpt-city` color ramps flagged as **Favorites** in the *Style Manager* dialog
- *All Color Ramps* to access the compatible color ramps database
- *Create New Color Ramp...* of any supported type that could be used in the current widget (note that this color ramp will not be available elsewhere unless you save it in the library)
- *Edit Color Ramp...*, the same as clicking the whole color ramp button
- *Save Color Ramp...*, to save the current color ramp with its customizations in the style library



Слика8.21: Quick color ramp selection widget

8.8.2 Symbol Widget

The *Symbol* selector widget is a convenient shortcut when you want to set symbol properties of a feature. Clicking the drop-down arrow shows the following symbol options, together with the features of the *color drop-down widget*:

- *Configure Symbol…*: the same as pressing the symbol selector widget. It opens a dialog to set the *symbol parameters*.
- *Copy Symbol* from the current item
- *Paste Symbol* to the current item, speeding configuration
- *Clear Current Symbol* to unset any assigned symbol to the widget (available only in some contexts)

Савет

Scroll the mouse wheel over a marker or line symbol widget to quickly modify the size of the associated symbol.

8.8.3 Remote or embedded file selector

Along with the file selector widget, the … button will sometimes show a drop-down arrow. This is usually available when using:

- an SVG file in a symbol or a label
- a raster image to customize symbols, labels, textures or decorations

Pressing the arrow will provide you with a menu to:

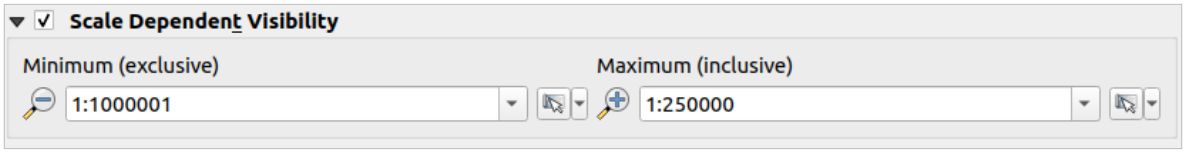
- *Select File…*, to load the file from the file system. The file is identified through the file path and QGIS needs to resolve the path in order to display the corresponding image
- *From URL…*, to load the file from a remote URL. As above, the image will only be loaded on successful retrieval of the remote resource
- *Embed File…*, to embed the file inside the current project, style database, or print layout template. The file is then always rendered as part of the item. This is a convenient way to create self-contained projects with custom symbols which can be easily shared amongst different users and installations of QGIS.
- *Extract Embedded File…*, to extract the embedded file from the widget and save it on disk.

8.8.4 Visibility Scale Selector

The visibility scale selector provides options to control the scales at which an element will be made visible in the map canvas. Out of the specified range of scales, the elements are not displayed. It can be applied e.g. to layers, labels or diagrams, from their *Rendering* properties tab.

1. Tick the  *Scale dependent visibility* box
2. Fill the *Minimum (exclusive)* box with the most zoomed out desired scale, typing the value or selecting it from the *predefined scales*
3. and/or fill the *Maximum (inclusive)* box with the most zoomed in desired scale

   The  Set to current canvas scale button next to the scale boxes sets the current map canvas scale as boundary of the range visibility. Press the arrow next to the button to access scales from layouts' maps and reuse them to fill the box.

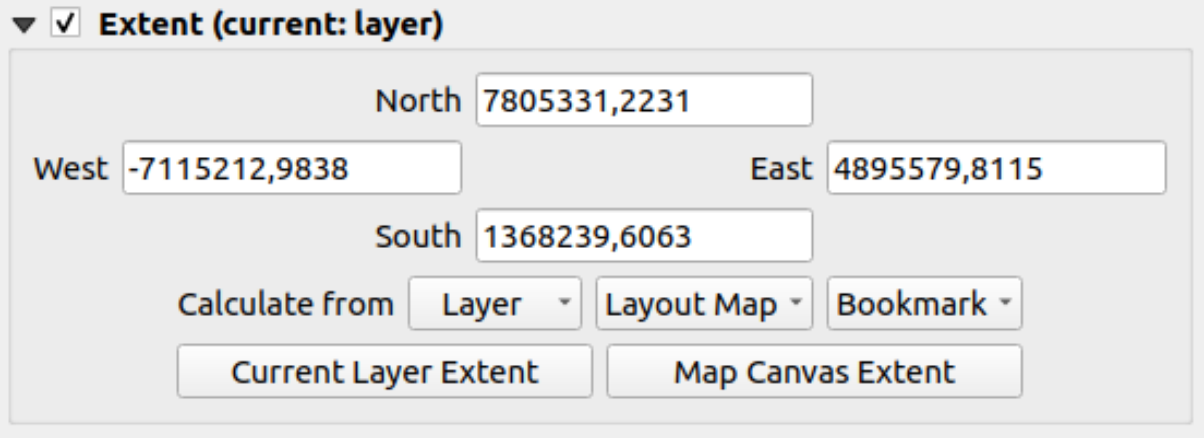


Слика8.22: Visibility scale selector widget

8.8.5 Spatial Extent Selector

The *Extent* selector widget is a convenient shortcut when you want to select a spatial extent to assign to a layer or to limit the actions to run on. Depending on the context, it offers selection between:

- *Current Layer Extent*: e.g. when exporting a layer
- *Calculate from Layer* ►: uses extent of a layer loaded in the current project
- Use current *Map Canvas Extent*
- *Draw on Canvas*: a rectangle whose coordinates are then used
- *Calculate from Bookmark*: uses extent of a saved *bookmark*
- *Calculate from Layout Map*: uses extent of a *layout map*
- Enter or edit the coordinates as `xmin, xmax, ymin, ymax`



Слика8.23: Extent selector widget

8.8.6 Font Selector

The *Font* selector widget is a convenient shortcut when you want to set font properties for textual information (feature labels, decoration labels, map legend text, …). Clicking the drop-down arrow shows some or all of the following options:



Слика8.24: Font selector drop-down menu

- *Clear Current Text Format* to unset any assigned text format to the widget (available only in some contexts)
- *Font Size* in the associated unit
- *Recent Fonts* ► menu with the active font checked (at the top)
- *Configure Format...*: same as pressing the font selector widget. It opens the *Text format* dialog, providing advanced formatting options such as color, opacity, orientation, HTML notation, buffer, background, shadow, …
- *Copy Format* of the text
- *Paste Format* to the text, speeding configuration
- the *color widget* for quick color setting

Савет

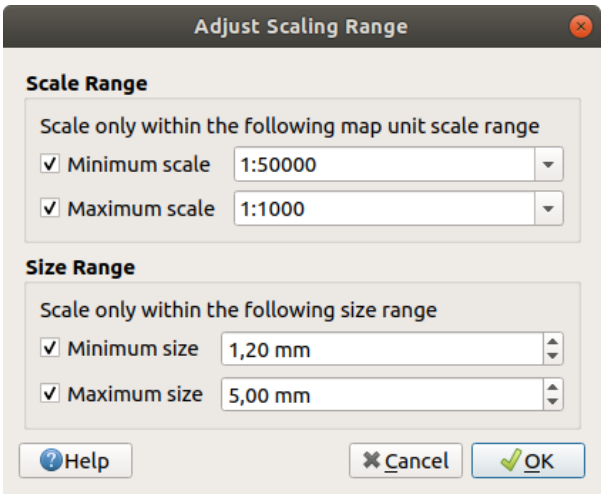
Scroll the mouse wheel over a font selector widget to quickly modify the font size of the associated text.

8.8.7 Unit Selector

Size properties of the items (labels, symbols, layout elements, …) in QGIS are not necessarily bound to either the project units or the units of a particular layer. For a large set of properties, the *Unit* selector drop-down menu allows you to tweak their values according to the rendering you want (based on screen resolution, paper size, or the terrain). Available units are:

- *Millimeters*
- *Points*
- *Pixels*
- *Inches*

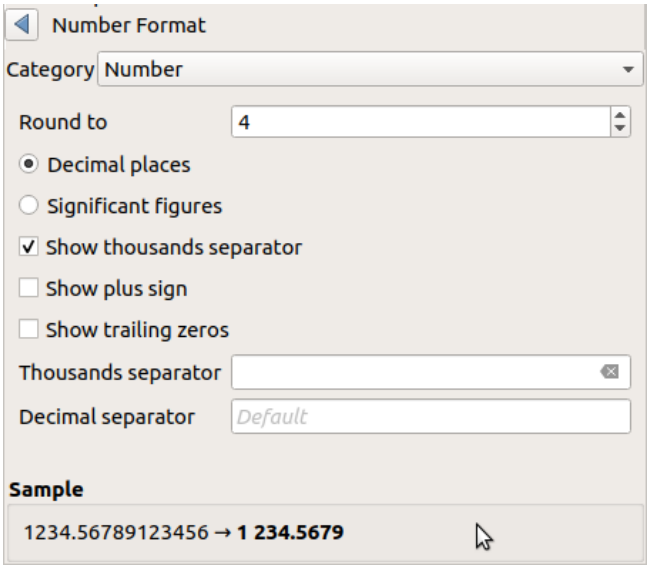
- *Percentage*: allows you to set some properties as a percent of another one. For example, this is useful for creation of text formats where the components (buffer size, shadow radius…) nicely scale as the text size is changed, instead of having constant buffer/shadow sizes. So you don't need to adjust those sizes, when the text size changes.
- *Meters at Scale*: This allows you to always set the size in meters, regardless of what the underlying map units are (e.g. they can be in inches, feet, geographic degrees, …). The size in meters is calculated based on the current project ellipsoid setting and a projection of the distances in meters at the center of the current map extent. For maps in a projected coordinate system this is calculated using projected units. For maps in a geographic (latitude/longitude) based system the size is approximated by calculating meter sizes using ellipsoidal calculations for the vertical scale of the map.
- and *Map Units*: The size is scaled according to the map view scale. Because this can lead to too big or too small values, use the button next to the entry to constrain the size to a range of values based on:
  - The *Minimum scale* and the *Maximum scale*: The value is scaled based on the map view scale until you reach any of these scale limits. Out of the range of scale, the value at the nearest scale limit is kept.
  - and/or The *Minimum size* and the *Maximum size* in `mm`: The value is scaled based on the map view scale until it reaches any of these limits; Then the limit size is kept.



Слика8.25: Adjust scaling range dialog

8.8.8 Number Formatting

Numeric formatters allow formatting of numeric values for display, using a variety of different formatting techniques (for instance scientific notation, currency values, percentage values, custom formatting, etc). One use of this is to set text in a layout scale bar or table, a color ramp legend, an elevation profile plot, …



Слика8.26: Formatting numeric value

Different categories of formats are supported. For most of them, you can set part or all of the following numeric options:

- *Show thousands separator*
- *Show plus sign*
- *Show trailing zeros*

But they can also have their custom settings. Provided categories are:

- *General*, the default category: has no setting and displays values as set in the parent widget properties or using the global settings.
- *Custom expression*: allows you to craft a custom *QGIS expression* to format numbers. The expression can use the `@value` variable to retrieve the value to be formatted, and then use any standard QGIS expression function to format this as desired.
- *Number*
  - The value can be *Round to* a self defined number of *Decimal places* or their *Significant figures*
  - customize the *Thousands separator* and *Decimal separator*
- *Bearing* for a text representation of a direction/bearing using:
  - *Format*: possible ranges of values are `0 to 180°, with E/W suffix`, `-180 to +180°` and `0 to 360°`
  - number of *Decimal places*
- *Currency* for a text representation of a currency value.
  - *Prefix*
  - *Suffix*
  - number of *Decimal places*
- *Fraction* for a vulgar fractional representation of a decimal value (e.g. *1/2* instead of *0.5*)
  - *Use unicode super/subscript* to show. For example $^{1}/_{2}$ instead of 1/2
  - *Use dedicated Unicode characters*

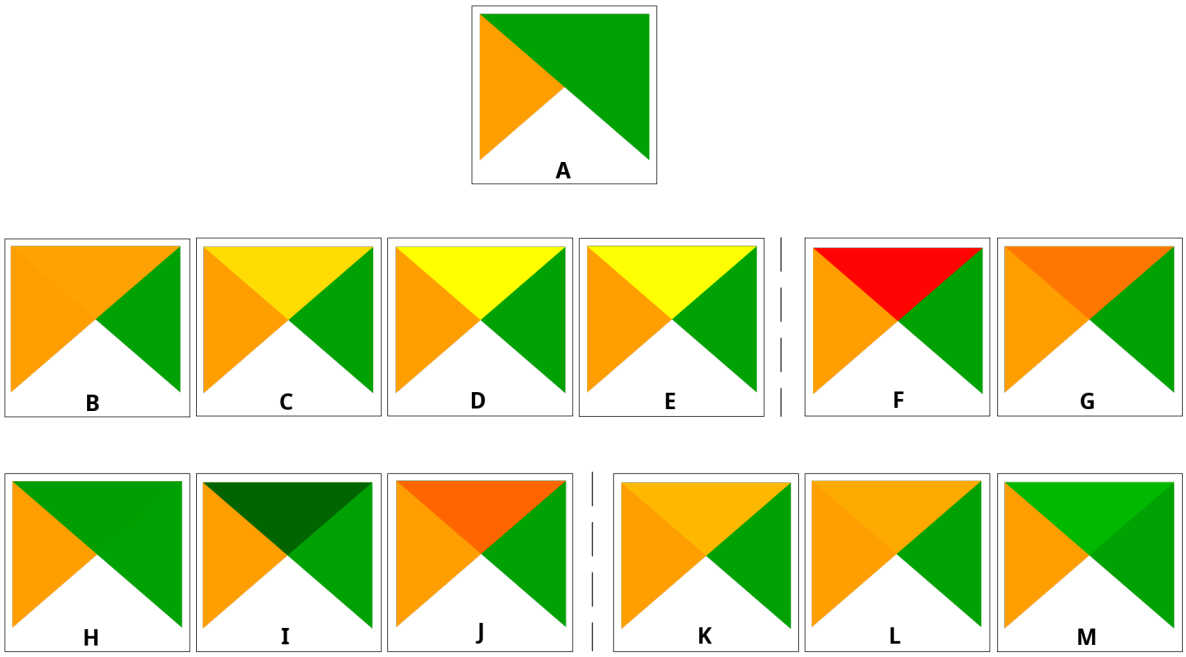
  - customize the *Thousands separator*
- *Percentage* - appends `%` to the values, with setting of:
  - number of *Decimal places*
  - *Scaling* to indicate whether the actual values already represent percentages (then they will be kept as is) or fractions (then they are converted)
- *Scientific* notation in the form `2.56e+03`. The number of *Decimal places* can be set.

A live preview of the settings is displayed under the *Sample* section.

8.8.9 Blending Modes

QGIS offers different options for special rendering effects with these tools that you may previously only know from graphics programs. Blending modes can be applied on layers and features, and also on print layout items:

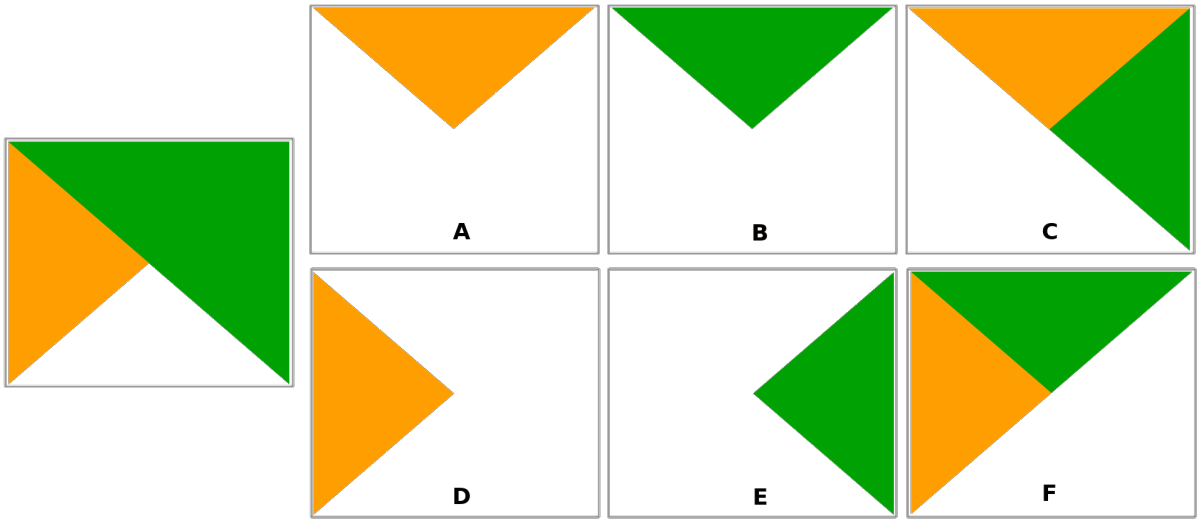
- *Normal*: This is the standard blend mode, which uses the alpha channel of the top pixel to blend with the pixel beneath it. The colors aren't mixed.
- *Lighten*: This selects the maximum of each component from the foreground and background pixels. Be aware that the results tend to be jagged and harsh.
- *Screen*: Light pixels from the source are painted over the destination, while dark pixels are not. This mode is most useful for mixing the texture of one item with another item (such as using a hillshade to texture another layer).
- *Dodge*: Brighten and saturate underlying pixels based on the lightness of the top pixel. Brighter top pixels cause the saturation and brightness of the underlying pixels to increase. This works best if the top pixels aren't too bright. Otherwise the effect is too extreme.
- *Addition*: Adds pixel values of one item to the other. In case of values above the maximum value (in the case of RGB), white is displayed. This mode is suitable for highlighting features.
- *Darken*: Retains the lowest values of each component of the foreground and background pixels. Like lighten, the results tend to be jagged and harsh.
- *Multiply*: Pixel values of the top item are multiplied with the corresponding values for the bottom item. The results are darker.
- *Burn*: Darker colors in the top item cause the underlying items to darken. Burn can be used to tweak and colorize underlying layers.
- *Overlay*: Combines multiply and screen blending modes. Light parts become lighter and dark parts become darker.
- *Soft light*: Very similar to overlay, but instead of using multiply/screen it uses color burn/dodge. This is supposed to emulate shining a soft light onto an image.
- *Hard light*: Hard light is also very similar to the overlay mode. It's supposed to emulate projecting a very intense light onto an image.
- *Difference*: Subtracts the top pixel from the bottom pixel, or the other way around, in order always to get a positive value. Blending with black produces no change, as the difference with all colors is zero.
- *Subtract*: Subtracts pixel values of one item from the other. In the case of negative values, black is displayed.



Слика8.27: Examples of blend modes applied to a green feature over an orange
A: Normal, **B**: Lighten, **C**: Screen, **D**: Dodge, **E**: Addition, **F**: Difference, **G**: Subtract, **H**: Darken, **I**: Multiply, **J**: Burn, **K**: Overlay, **L**: Soft light, **M**: Hard light

When a layer is part of a group that *renders layers as a group*, additional blending modes are available for the rendering. They provide methods to clip the render of one layer's content by the content in a second „mask" layer.

- *Masked By Below*: The output is the top pixel, where the opacity is reduced by that of the bottom pixel.
- *Mask Below*: The output is the bottom pixel, where the opacity is reduced by that of the top pixel.
- *Inverse Masked By Below*: The output is the top pixel, where the opacity is reduced by the inverse of the bottom pixel.
- *Inverse Mask Below*: The output is the bottom pixel, where the opacity is reduced by the inverse of the top pixel.
- *Paint Inside Below*: The top pixel is blended on top of the bottom pixel, with the opacity of the top pixel reduced by the opacity of the bottom pixel.
- *Paint Below Inside*: The bottom pixel is blended on top of the top pixel, with the opacity of the bottom pixel reduced by the opacity of the top pixel.



Слика8.28: Examples of blend clipping mode applied to top green layer in a group
A: Mask Below **B**: Masked By Below **C**: Paint Below Inside **D**: Inverse Mask Below **E**: Inverse Masked By Below **F**: Paint Inside Below

8.8.10 Data defined override setup

Next to many options in the vector layer properties dialog or settings in the print layout, you will find a Data defined override icon. Using *expressions* based on layer attributes or item settings, prebuilt or custom functions and *variables*, this tool allows you to set dynamic values for parameters. When enabled, the value returned by this widget is applied to the parameter regardless of its normal value (checkbox, textbox, slider…).

The data defined override widget

Clicking the Data defined override icon shows the following entries:

- *Description…* that indicates if the option is enabled, which input is expected, the valid input type and the current definition. Hovering over the widget also pops up this information.
- *Store data in the project*: a button allowing the property to be stored using the *Auxiliary Storage Properties* mechanism.
- *Field type*: an entry to select from the layer's fields that match the valid input type.
- *Color*: when the widget is linked to a color property, this menu gives access to the colors defined as part of the current *project's colors* scheme.
- *Variable*: a menu to access the available user-defined *variables*
- *Edit…* button to create or edit the expression to apply, using the *Expression String Builder* dialog. To help you correctly fill in the expression, a reminder of the expected output's format is provided in the dialog.
- *Paste* and *Copy* buttons.
- *Clear* button to remove the setup.
- For numeric and color properties, *Assistant…* to rescale how the feature data is applied to the property (more details *below*)

Савет

Use right-click to (de)activate the data override

When the data-defined override option is set up correctly the icon is yellow or . If it is broken, the icon is red or .

You can enable or disable a configured Data-defined override button by simply clicking the widget with the right mouse button.

Савет

Use middle-click to create or edit the expression to apply

You can directly open the *Expression String Builder* dialog to create or edit the expression to apply by simply clicking the Data-defined override widget with the middle mouse button.

Using the data-defined assistant interface

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 output value.

The assistant allows you to:

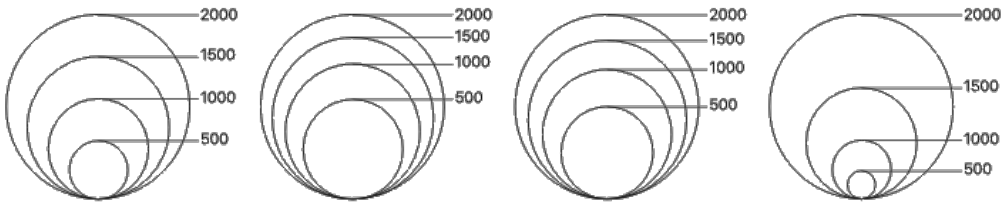
- Define the *Input* data, i.e.:
  - *Source*: the attribute to represent, using a field or an *expression*
  - the range of values to represent: you can manually enter the values or use the Fetch value range from layer button to fill these fields automatically with the minimum and maximum values returned by the *Source* expression applied to your data. Features whose values are out of the given range will be represented with the property of the closest limit of the output range.
- *Apply transform curve*: by default, output values are applied to input features following a specific scaling method (see below for setting). You can override this logic: enable the transform option, click on the graphic to add break point(s) and drag the point(s) to apply a custom distribution.
- Define the *Output* values: the options vary according to the parameter to define. You can globally set:
  - 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.

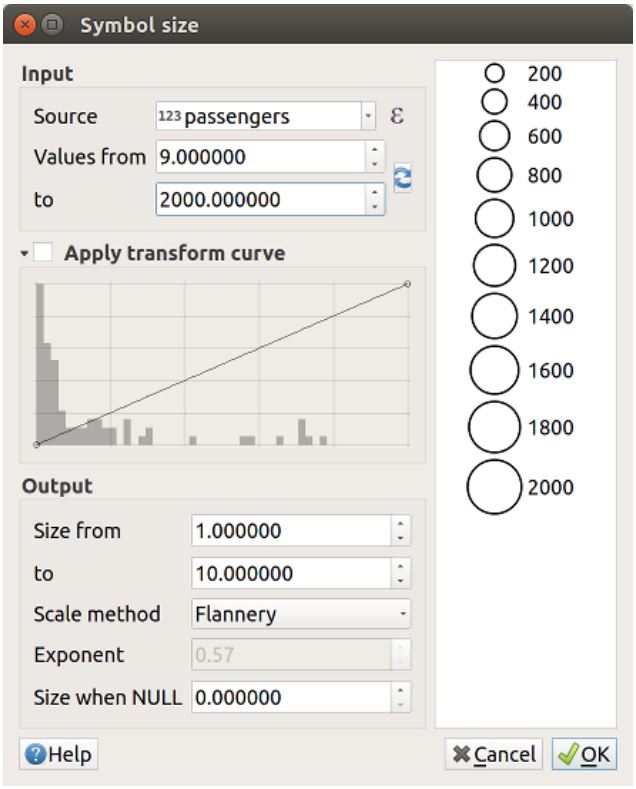
- for size (point symbols): an interpolation is done over a range of minimum and maximum values in order to assign each symbol the correct size „diameter". The *Scale method* field controls how the interpolation is done and what the output value represents.
  - **Radius**: linearly interpolates the diameter 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.



Слика8.29: Scaling values using different methods (from left to right): Radius - Surface - Flannery - Exponential of 2

When compatible with the property, a live-update preview is displayed in the right-hand side of the dialog to help you control the value scaling.



Слика8.30: Scaling feature size based on passengers field's value

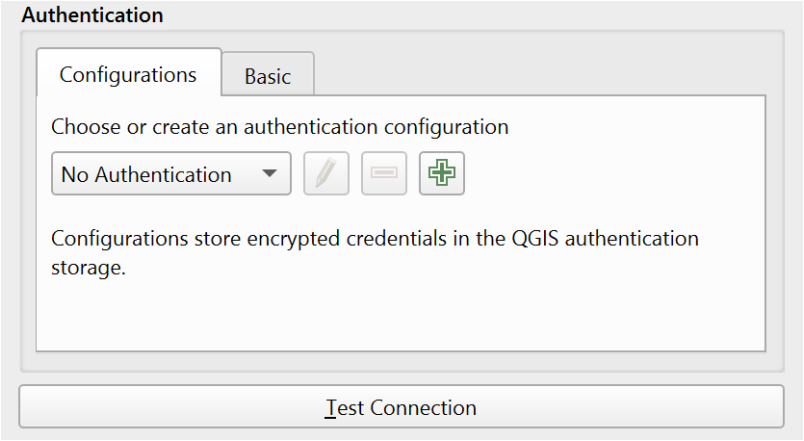
The values presented in the varying size assistant above will set the size 'Data-defined override' with the editable formula below:

```
coalesce(scale_polynomial("passengers", 9, 2000, 1, 10, 0.57), 0)
```

8.8.11 Authentication

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.



Слика8.31: 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.

Упозорење

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 *Sistem za autentifikaciju* 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 *Sistem za autentifikaciju*.

ГЛАВА 9

Напредујте уз изразе

9.1 Expressions

Based on layer data and prebuilt or user defined functions, **Expressions** offer a powerful way to manipulate attribute value, geometry and variables in order to dynamically change the geometry style, the content or position of the label, the value for diagram, the height of a layout item, select some features, create virtual field, …

Белешка

A list of the default functions and variables for writing expressions can be found at *Lista funkcija*, with detailed information and examples.

9.1.1 The Expression string builder

Main dialog to build expressions, the *Expression string builder* is available from many parts in QGIS and, can particularly be accessed when:

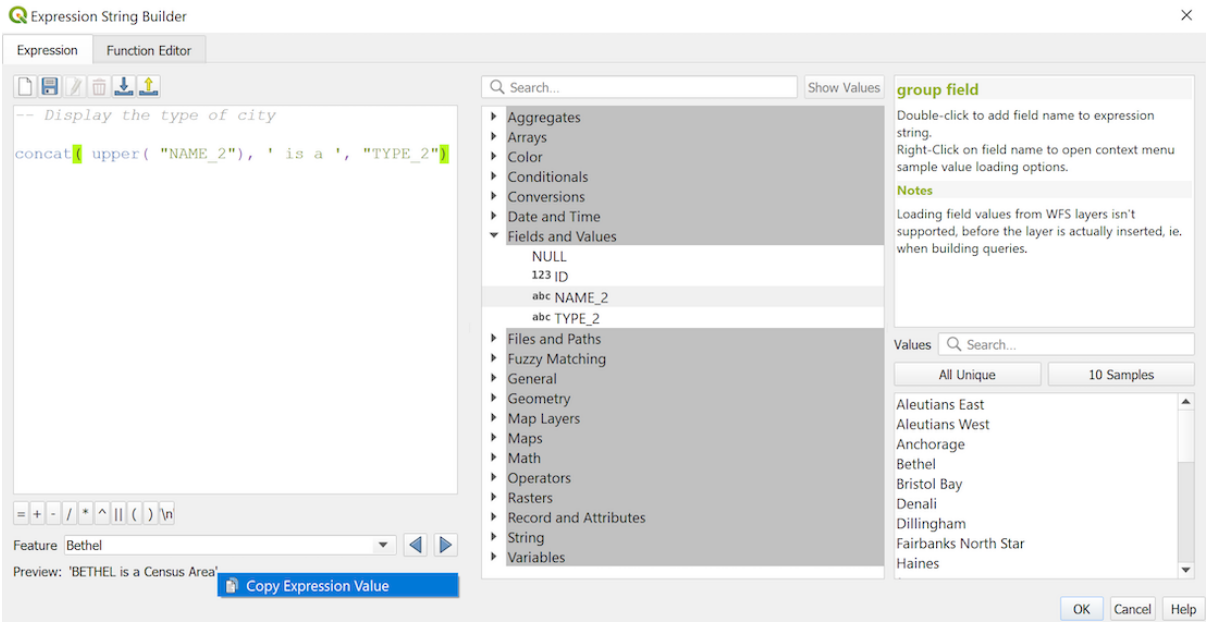
- clicking the ε button;
- *selecting features* with the Select By Expression… tool;
- *editing attributes* with e.g. the Field calculator tool;
- manipulating symbology, label or layout item parameters with the Data defined override tool (see *Data defined override setup*);
- building a *geometry generator* symbol layer;
- doing some *geoprocessing*.

The Expression builder dialog offers access to the:

- *Expression* tab which, thanks to a list of *predefined functions*, helps to write and check the expression to use;
- *Function Editor tab* which helps to extend the list of functions by creating custom ones.

The Interface

The *Expression* tab provides the main interface to write expressions using functions, layer fields and values. It contains the following widgets:



Слика9.1: The Expression tab

- An expression editor area for typing or pasting expressions. Autocompletion is available to speed expression writing:
  - Corresponding variables, function names and field names to the input text are shown below: use the `Up` and `Down` arrows to browse the items and press `Tab` to insert in the expression or simply click on the wished item.
  - Function parameters are shown while filling them.

  QGIS also checks the expression rightness and highlights all the errors using:
  - *Underline*: for unknown functions, wrong or invalid arguments;
  - *Marker*: for every other error (eg, missing parenthesis, unexpected character) at a single location.

Савет

Document your expression with comments

When using complex expression, it is good practice to add text either as a multiline comment or inline comments to help you remember.

```
/*
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
)
```

- Above the expression editor, a set of tools helps you:
  -  Clear the expression editor
  - Create and manage *user expressions*
- Under the expression editor, you find:
  - A set of basic operators to help you build the expression
  - An indication of the expected format of output when you are data-defining feature properties
  - A live *Output preview* of the expression (up to 60 characters), evaluated on the first feature of the Layer by default. To view output preview text exceeding 60 characters, you can hover your cursor over the text to display a tooltip pop-up containing the entire output preview. To copy the output preview text onto your clipboard, right-click on the output preview text and select  *Copy Expression Value*.

    You can browse and evaluate other features of the layer using the *Feature* combobox (the values are taken from the *display name* property of the layer).

    In case of error, it indicates it and you can access the details with the provided hyperlink.
- A function selector displays the list of functions, variables, fields… organized in groups. A search box is available to filter the list and quickly find a particular function or field. Double-clicking an item adds it to the expression editor.
- A help panel displays help for each selected item in the function selector.

> **Савет**
>
> Press `Ctrl+Click` when hovering a function name in an expression to automatically display its help in the dialog.

A field's values widget shown when a field is selected in the function selector helps to fetch features attributes:

- Look for a particular field value
- Display the list of *All Unique* or *10 Samples* values. Also available from right-click.

  When the field is mapped with another layer or a set of values, i.e. if the *field widget* is of *RelationReference*, *ValueRelation* or *ValueMap* type, it's possible to list all the values of the mapped field (from the referenced layer, table or list). Moreover, you can filter this list to  *Only show values in use* in the current field.

Double-clicking a field value in the widget adds it to the expression editor.

Савет

The right panel, showing functions help or field values, can be collapsed (invisible) in the dialog. Press the *Show Values* or *Show Help* button to get it back.

Writing an expression

QGIS expressions are used to select features or set values. Writing an expression in QGIS follows some rules:

1. **The dialog defines the context**: if you are used to SQL, you probably know queries of the type *select features from layer where condition* or *update layer set field = new_value where condition*. A QGIS expression also needs all these information but the tool you use to open the expression builder dialog provides parts of them. For example, giving a layer (`buildings`) with a field (`height`):
   - pressing the  Select by expression tool means that you want to „select features from buildings”. The **condition** is the only information you need to provide in the expression text widget, e.g. type `"height" > 20` to select buildings that are higher than 20.
   - with this selection made, pressing the  Field calculator button and choosing „height” as *Update existing field*, you already provide the command „update buildings set height = ??? where height > 20”. The only remaining bits you have to provide in this case is the **new value**, e.g. just enter `50` in the expression editor textbox to set the height of the previously selected buildings.
2. **Pay attention to quotes**: single quotes return a literal, so a text placed between single quotes (`'145'`) is interpreted as a string. Double quotes will give you the value of that text so use them for fields (`"myfield"`). Fields can also be used without quotes (`myfield`). No quotes for numbers (`3.16`).

Белешка

Functions normally take as argument a string for field name. Do:

```
attribute( @atlas_feature, 'height' ) -- returns the value stored in the
↪"height" attribute of the current atlas feature
```

And not:

```
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.
```

Савет

Use named parameters to ease expression reading

Some functions require many parameters to be set. The expression engine supports the use of named parameters. This means that instead of writing the cryptic expression `clamp( 1, 2, 9)`, you can use `clamp( min:=1, value:=2, max:=9)`. This also allows arguments to be switched, e.g. `clamp( value:=2, max:=9, min:=1)`. Using named parameters helps clarify what the arguments for an expression function refer to, which is helpful when you are trying to interpret an expression later!

Some use cases of expressions

- From the Field Calculator, calculate a „pop_density” field using the existing „total_pop” and „area_km2” fields:

```
"total_pop" / "area_km2"
```

- Label or categorize features based on their area:

```
CASE WHEN $area > 10 000 THEN 'Larger' ELSE 'Smaller' END
```

- Update the field „density_level" with categories according to the „pop_density" values:

```
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
```

- Apply a categorized style to all the features according to whether their average house price is smaller or higher than 10000€ per square metre:

```
"price_m2" > 10000
```

- Using the „Select By Expression…" tool, select all the features representing areas of "High population density" and whose average house price is higher than 10000€ per square metre:

```
"density_level" = 'High population density' and "price_m2" > 10000
```

  The previous expression could also be used to define which features to label or show on the map.

- Select features that overlap a natural zone from the „lands" layer:

```
overlay_intersects( layer:='lands', filter:="zone_type"='Natural' )
```

- Count for each feature the number of buildings they contain:

```
array_length( overlay_contains( layer:='buildings', expression:=@id ) )
```

- Create a different symbol (type) for the layer, using the geometry generator:

```
point_on_surface( @geometry )
```

- Given a point feature, generate a closed line (using `make_line`) around its geometry:

```
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 )␣
↪)
  )
)
```

- In a print layout label, display the name of the „airports" features that are within the layout „Map 1" item:

```
with_variable( 'extent',
               map_get( item_variables( 'Map 1' ), 'map_extent' ),
               aggregate( 'airports', 'concatenate', "NAME",
                          intersects( @geometry, @extent ), ' ,'
                        )
             )
```

Saving Expressions

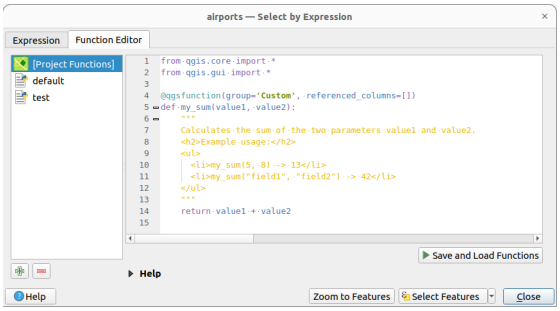
Using the Add current expression to user expressions button above the expression editor frame, you can save important expressions you want to have quick access to. These are available from the **User expressions** group in the middle panel. They are saved under the *user profile* (`<userprofile>/QGIS/QGIS3.ini` file) and available in all expression dialogs inside all projects of the current user profile.

A set of tools available above the expression editor frame helps you manage the user expressions:

- Add the current expression to user expressions: store the expression in the user profile. A label and a help text can be added for easy identification.
- Edit selected expression from user expressions, as well as their help and label
- Remove selected expression from user expressions
- Import user expressions from a `.json` file into the active user profile folder
- Export user expressions as a `.json` file; all the user expressions in the user profile `QGIS3.ini` file are shared

9.1.2 Function Editor

With the *Function Editor* tab, you are able to write your own functions in Python language. This provides a handy and comfortable way to address particular needs that would not be covered by the predefined functions.



Слика9.2: The Function Editor tab

To create a new function:

1. Press the New File button.
2. Select where to store the new function. You can choose to save it either in a *Function file* or in *Project functions*.
   - If you select *Function file*, you need to enter a name for the Python `.py` file, which is based on a QGIS template file and is stored in the `/python/expressions` folder under the active *user profile* directory, thus available for all projects you may open with that profile.

     Custom Python functions stored as Function files are auto loaded whenever you start a new project under the user profile. Be aware that for sharing such Python functions, you need to share the `.py` file in the `/python/expressions` folder.
   - If you select *Project functions*, the new function is stored in the project file and only available for that project.

     Custom Python functions stored as Project functions can be easily shared by sharing the project file where they were saved. When opening a project with project functions, QGIS can load or ignore them depending on whether *embedded Python code is enabled*.

   A new item is added in the left panel of the *Function Editor* tab.
3. The right panel displays the content of the file: a python script template.

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 settings*. Online API documentation for Qt and PyQGIS objects is also accessible pressing the `F1` key.

   You will find in the *Help* widget general information for setting the @qgsfunction decorator.

5. Press the  *Save and Load Functions* button. The function you wrote is added to the functions tree in the *Expression* tab, by default under the `Custom` group.

6. Enjoy your new function.

7. If the function requires improvements, enable the *Function Editor* tab, do the changes and press again the  *Save and Load Functions* button to make them available in the file, hence in any expression tab.

To delete a custom function:

1. Enable the *Function Editor* tab
2. Select the function in the list
3. Press the Remove selected function. The function is removed from the list and, depending on the storage, the corresponding `.py` file is deleted from the user profile folder, or the Python functions are removed from the project file.

Белешка

When a project file with Project functions is closed, the corresponding Project functions are unloaded and are no longer available in the QGIS session.

Primer

Here's a short example on how to create your own `my_sum` function that will operate with two values.

```
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
```

The `@qgsfunction` decorator accepts the following arguments:

- The `group` argument indicates the group in which the function should be listed in the Expression dialog.
- `usesgeometry`: Set this to `True` if the expression requires access to the features geometry. By default `False`.
- `handlesnull`: Set this to `True` if the expression has custom handling for NULL values. If `False` (default), the result will always be NULL as soon as any parameter is NULL.
- `referenced_columns`: An array of attribute names that are required to run the function. Defaults to `[QgsFeatureRequest.ALL_ATTRIBUTES]`. Specifying a subset of fields or an empty list will result in a faster execution.

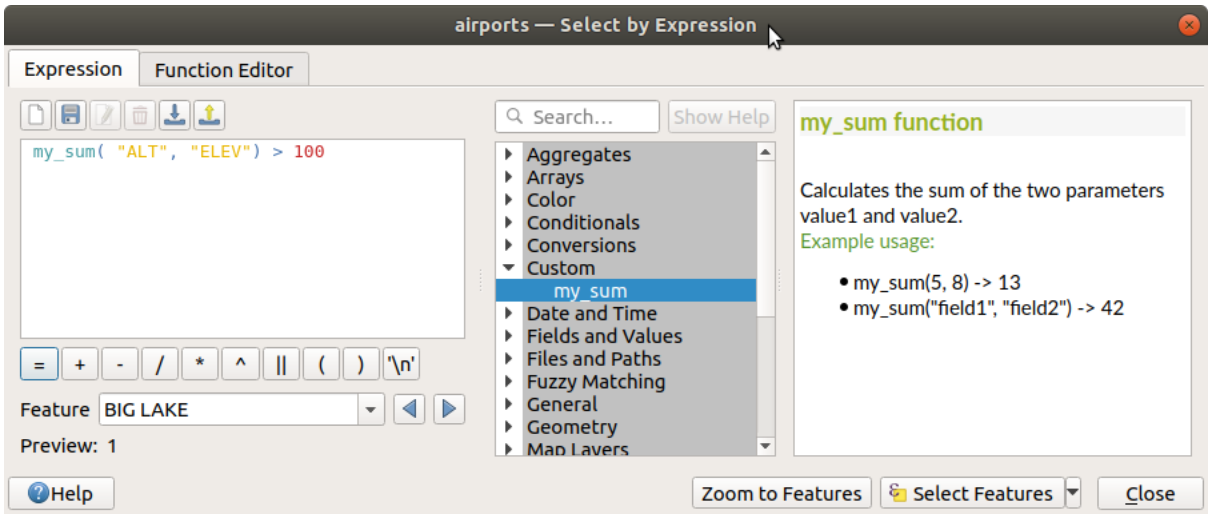
- `params_as_list`: Set this to `True` to pass the function parameters as a list. If `False` (default), the function will receive the parameters as individual arguments.

The function itself takes any number and type of parameters you want to pass to your function. These should be set before the following optional arguments:

- `feature`: the current feature
- `parent`: the `QgsExpression` object
- `context`: If there is an argument called `context` found at the last position, this variable will contain a `QgsExpressionContext` object, that gives access to various additional information like expression variables. E.g. `context.variable( 'layer_id' )`

If those arguments are present in the function signature, they will be automatically passed to the function, without the need to specify them in the expression.

The previous example function can then be used in expressions:



Слика9.3: Custom Function added to the Expression tab

Find more examples and additional arguments for custom functions at `qgsfunction`. Further information about creating Python code can be found in the PyQGIS-Developer-Cookbook.

9.2 Lista funkcija

Funkcije, operatori i promenljive dostupni u QGIS-u navedeni su ispod, grupisani po kategorijama.

9.2.1 Agregatne funkcije

Ova grupa sadrži funkcije koje agregiraju vrednosti preko slojeva i polja.

aggregate

Vraća agregatnu vrednost izračunatu pomoću objekata sa drugog sloja.

Sintaksa	aggregate(layer, aggregate, expression, [filter], [concatenator:=''], [order_by]) [] označava opcione argumente
Argumenti	• **layer** - niska koja predstavlja naziv sloja ili ID sloja • **aggregate** - niska koja odgovara agregatu koji se računa. Važeće opcije su: – count – count_distinct – count_missing – minimum ili min – maximum ili max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: prvi kvartil – q3: treći kvartil – iqr: interkvartilni raspon – min_length: minimalna dužina niske – max_length: maksimalna dužina niske – concatenate: spaja niske pomoću spajača – concatenate_unique: spaja jedinstvene niske pomoću spajača – collect: pravi agregiranu višedelnu geometriju – array_agg: pravi niz agregiranih vrednosti • **expression** - podizraz ili naziv polja koje se agregira • **filter** - opcioni izraz filtera za ograničavanje objekata koji se koriste za računanje agregata. Polja i geometrija su sa objekata na spojenom sloju. Izvornom objektu se može pristupiti promenljivom @parent. • **concatenator** - opciona niska za spajanje vrednosti za agregate 'concatenate' i 'concatenate_unique' • **order_by** - opcioni izraz filtera za uređivanje objekata koji se koriste za računanje agregata. Polja i geometrija su sa objekata na spojenom sloju. Podrazumevano se objekti vraćaju u neodređenom redosledu.
Primeri	• `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers")` → zbir svih vrednosti iz polja passengers na sloju rail_stations • `aggregate('rail_stations','sum', "passengers"/7)` → računa dnevni prosek za „passengers" tako što polje „passengers" deli sa 7 pre sabiranja vrednosti • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:="class">3)` → sabira sve vrednosti iz polja „passengers" samo sa objekata kod kojih je atribut „class" veći od 3 • `aggregate(layer:='rail_stations',aggregate:='concatenate', expression:="name", concatenator:=',')` → lista polja name razdvojena zarezima za sve objekte na sloju rail_stations • `aggregate(layer:='countries', aggregate:='max', expression:="code", filter:=intersects( @geometry, geometry(@parent) ) )` → Kod države koja se preseca, na sloju 'countries' • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:=contains( @atlas_geometry, @geometry ) )` → zbir svih vrednosti iz polja passengers na sloju rail_stations unutar trenutnog objekta atlasa • `aggregate(layer:='rail_stations', aggregate:='collect', expression:=centroid(@geometry), filter:="region_name" = attribute(@parent,'name') )` → agregira geometrije centroida objekata rail_stations iz iste regije kao trenutni objekat

array_agg

Vraća niz agregiranih vrednosti iz polja ili izraza.

Sintaksa	array_agg(expression, [group_by], [filter], [order_by]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata • **order_by** - opcioni izraz za uređivanje objekata koji se koriste za računanje agregata. Podrazumevano se objekti vraćaju u neodređenom redosledu.
Primeri	• `array_agg("name",group_by:="state")` → lista vrednosti name, grupisana po polju state

collect

Vraća višedelnu geometriju agregiranih geometrija iz izraza

Sintaksa	collect(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - izraz geometrije koji se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `collect( @geometry )` → višedelna geometrija agregiranih geometrija • `collect( centroid(@geometry), group_by:="region", filter:= "use" = 'civilian' )` → aggregated centroids of the civilian features based on their region value

concatenate

Vraća sve agregirane niske iz polja ili izraza spojene graničnikom.

Sintaksa	concatenate(expression, [group_by], [filter], [concatenator:=''], [order_by]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata • **concatenator** - opciona niska za spajanje vrednosti. Podrazumevano prazna. • **order_by** - opcioni izraz za uređivanje objekata koji se koriste za računanje agregata. Podrazumevano se objekti vraćaju u neodređenom redosledu.
Primeri	• `concatenate("town_name",group_by:="state",concatenator:=',')` → lista town_names razdvojena zarezima, grupisana po polju state

concatenate_unique

Vraća sve jedinstvene niske iz polja ili izraza spojene graničnikom.

Sintaksa	concatenate_unique(expression, [group_by], [filter], [concatenator:=”], [order_by]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata • **concatenator** - opciona niska za spajanje vrednosti. Podrazumevano prazna. • **order_by** - opcioni izraz za uređivanje objekata koji se koriste za računanje agregata. Podrazumevano se objekti vraćaju u neodređenom redosledu.
Primeri	• `concatenate_unique("town_name",group_by:="state", concatenator:=',')` → lista jedinstvenih town_names razdvojena zarezima, grupisana po polju state

count

Vraća broj podudarajućih objekata.

Sintaksa	count(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `count("stations",group_by:="state")` → broj stanica, grupisan po polju state

count_distinct

Vraća broj jedinstvenih vrednosti.

Sintaksa	count_distinct(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `count_distinct("stations",group_by:="state")` → broj jedinstvenih vrednosti stations, grupisan po polju state

count_missing

Vraća broj nedostajućih (NULL) vrednosti.

Sintaksa	count_missing(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `count_missing("stations",group_by:="state")` → broj nedostajućih (NULL) vrednosti stations, grupisan po polju state

iqr

Vraća izračunati interkvartilni raspon iz polja ili izraza.

Sintaksa	iqr(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `iqr("population",group_by:="state")` → interkvartilni raspon vrednosti population, grupisan po polju state

majority

Vraća agregatnu većinu vrednosti (najčešću vrednost) iz polja ili izraza.

Sintaksa	majority(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `majority("class",group_by:="state")` → najčešća vrednost class, grupisana po polju state

max_length

Vraća maksimalnu dužinu niski iz polja ili izraza.

Sintaksa	max_length(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `max_length("town_name",group_by:="state")` → maksimalna dužina town_name, grupisana po polju state

maximum

Vraća agregatnu maksimalnu vrednost iz polja ili izraza.

Sintaksa	maximum(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `maximum("population",group_by:="state")` → maksimalna vrednost population, grupisana po polju state

mean

Vraća agregatnu srednju vrednost iz polja ili izraza.

Sintaksa	mean(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `mean("population",group_by:="state")` → srednja vrednost population, grupisana po polju state

median

Vraća agregatnu medijanu iz polja ili izraza.

Sintaksa	median(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `median("population",group_by:="state")` → medijana vrednosti population, grupisana po polju state

min_length

Vraća minimalnu dužinu niski iz polja ili izraza.

Sintaksa	min_length(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `min_length("town_name",group_by:="state")` → minimalna dužina town_name, grupisana po polju state

minimum

Vraća agregatnu minimalnu vrednost iz polja ili izraza.

Sintaksa	minimum(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `minimum("population",group_by:="state")` → minimalna vrednost population, grupisana po polju state

minority

Vraća agregatnu manjinu vrednosti (najređu vrednost) iz polja ili izraza.

Sintaksa	minority(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `minority("class",group_by:="state")` → najređa vrednost class, grupisana po polju state

q1

Vraća izračunati prvi kvartil iz polja ili izraza.

Sintaksa	q1(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `q1("population",group_by:="state")` → prvi kvartil vrednosti population, grupisan po polju state

q3

Vraća izračunati treći kvartil iz polja ili izraza.

Sintaksa	q3(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `q3("population",group_by:="state")` → treći kvartil vrednosti population, grupisan po polju state

range

Vraća agregatni raspon vrednosti (maksimum - minimum) iz polja ili izraza.

Sintaksa	range(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `range("population",group_by:="state")` → raspon vrednosti population, grupisan po polju state

relation_aggregate

Vraća agregatnu vrednost izračunatu pomoću svih podudarajućih podređenih objekata iz relacije sloja.

Sintaksa	relation_aggregate(relation, aggregate, expression, [concatenator:=''], [order_by]) [] označava opcione argumente
Argumenti	• **relation** - niska koja predstavlja ID relacije • **aggregate** - niska koja odgovara agregatu koji se računa. Važeće opcije su: – count – count_distinct – count_missing – minimum ili min – maximum ili max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: prvi kvartil – q3: treći kvartil – iqr: interkvartilni raspon – min_length: minimalna dužina niske – max_length: maksimalna dužina niske – concatenate: spaja niske pomoću spajača – concatenate_unique: spaja jedinstvene niske pomoću spajača – collect: pravi agregiranu višedelnu geometriju – array_agg: pravi niz agregiranih vrednosti • **expression** - podizraz ili naziv polja koje se agregira • **concatenator** - opciona niska za spajanje vrednosti za agregat 'concatenate' • **order_by** - opcioni izraz za uređivanje objekata koji se koriste za računanje agregata. Polja i geometrija su sa objekata na spojenom sloju. Podrazumevano se objekti vraćaju u neodređenom redosledu.
Primeri	• `relation_aggregate(relation:='my_relation',aggregate:='mean', expression:="passengers")` → srednja vrednost svih podudarajućih podređenih objekata pomoću relacije 'my_relation' • `relation_aggregate('my_relation','sum', "passengers"/7)` → zbir polja passengers podeljen sa 7 za sve podudarajuće podređene objekte pomoću relacije 'my_relation' • `relation_aggregate('my_relation','concatenate', "towns", concatenator:=',')` → lista polja towns razdvojena zarezima za sve podudarajuće podređene objekte pomoću relacije 'my_relation' • `relation_aggregate('my_relation','array_agg', "id")` → niz polja id sa svih podudarajućih podređenih objekata pomoću relacije 'my_relation'

Više informacija: *Setting relations between multiple layers*

stdev

Vraća agregatnu standardnu devijaciju iz polja ili izraza.

Sintaksa	stdev(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `stdev("population",group_by:="state")` → standardna devijacija vrednosti population, grupisana po polju state

sum

Vraća agregatnu sumu vrednosti iz polja ili izraza.

Sintaksa	sum(expression, [group_by], [filter]) [] označava opcione argumente
Argumenti	• **expression** - podizraz polja koje se agregira • **group_by** - opcioni izraz za grupisanje agregatnih izračunavanja • **filter** - opcioni izraz za filtriranje objekata koji se koriste za računanje agregata
Primeri	• `sum("population",group_by:="state")` → zbir vrednosti population, grupisan po polju state

9.2.2 Funkcije za nizove

Ova grupa sadrži funkcije za pravljenje nizova i rad sa njima (poznatih i kao liste). Redosled vrednosti unutar niza je bitan, za razliku od *strukture podataka 'map'*, gde redosled parova ključ-vrednost nije bitan, a vrednosti se identifikuju svojim ključevima.

array

Vraća niz koji sadrži sve vrednosti prosleđene kao parametar.

Sintaksa	array(value1, value2, …)
Argumenti	• **value** - vrednost
Primeri	• `array(2,10)` → [2, 10] • `array(2,10)[0]` → 2

array_all

Vraća TRUE ako niz sadrži sve vrednosti zadatog niza.

Sintaksa	array_all(array_a, array_b)
Argumenti	• **array_a** - niz • **array_b** - niz vrednosti koje se traže
Primeri	• `array_all(array(1,2,3),array(2,3))` → TRUE • `array_all(array(1,2,3),array(1,2,4))` → FALSE

array_append

Vraća niz sa zadatom vrednošću dodatom na kraj.

Sintaksa	array_append(array, value)
Argumenti	• **array** - niz • **value** - vrednost koja se dodaje
Primeri	• `array_append(array(1,2,3),4)` → [1, 2, 3, 4]

array_cat

Vraća niz koji sadrži sve zadate nizove spojene.

Sintaksa	array_cat(array1, array2, …)
Argumenti	• **array** - niz
Primeri	• `array_cat(array(1,2),array(2,3))` → [1, 2, 2, 3]

array_contains

Vraća TRUE ako niz sadrži zadatu vrednost.

Sintaksa	array_contains(array, value)
Argumenti	• **array** - niz • **value** - vrednost koja se traži
Primeri	• `array_contains(array(1,2,3),2)` → TRUE

array_count

Broji koliko se puta zadata vrednost javlja u nizu.

Sintaksa	array_count(array, value)
Argumenti	• **array** - niz • **value** - vrednost koja se broji
Primeri	• `array_count(array('a', 'b', 'c', 'b'), 'b')` → 2

array_distinct

Vraća niz koji sadrži jedinstvene vrednosti zadatog niza.

Sintaksa	array_distinct(array)
Argumenti	• **array** - niz
Primeri	• `array_distinct(array(1,2,3,2,1))` → [1, 2, 3]

array_filter

Vraća niz sa samo onim stavkama za koje izraz daje true.

Sintaksa	array_filter(array, expression, [limit:=0]) [] označava opcione argumente
Argumenti	• **array** - niz • **expression** - izraz koji se izračunava na svakoj stavci. Promenljiva *@element* biće zamenjena trenutnom vrednošću. • **limit** - maksimalan broj elemenata koji se vraćaju. Koristite 0 za vraćanje svih vrednosti.
Primeri	• `array_filter(array(1,2,3),@element < 3)` → [1, 2] • `array_filter(array(1,2,3),@element < 3, 1)` → [1]

array_find

Vraća najmanji indeks (0 za prvi) vrednosti unutar niza. Vraća -1 ako vrednost nije pronađena.

Sintaksa	array_find(array, value)
Argumenti	• **array** - niz • **value** - vrednost koja se traži
Primeri	• `array_find(array('a', 'b', 'c'), 'b')` → 1 • `array_find(array('a', 'b', 'c', 'b'), 'b')` → 1

array_first

Vraća prvu vrednost niza.

Sintaksa	array_first(array)
Argumenti	• **array** - niz
Primeri	• `array_first(array('a','b','c'))` → 'a'

array_foreach

Vraća niz sa izrazom izračunatim na svakoj stavci.

Sintaksa	array_foreach(array, expression)
Argumenti	• **array** - niz • **expression** - izraz koji se izračunava na svakoj stavci. Promenljiva *@element* biće zamenjena trenutnom vrednošću, a promenljiva *@counter* trenutnim indeksom (počev od 0).
Primeri	• `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

Vraća N-tu vrednost (0 za prvu) ili poslednju -N-tu vrednost (-1 za poslednju) niza.

Sintaksa	array_get(array, pos)
Argumenti	• **array** - niz • **pos** - indeks koji se preuzima (počev od 0)
Primeri	• `array_get(array('a','b','c'),1)` → 'b' • `array_get(array('a','b','c'),-1)` → 'c'

Путоказ

Za dobijanje vrednosti iz niza možete koristiti i *operator indeksa ([])*.

array_insert

Vraća niz sa zadatom vrednošću dodatom na zadatu poziciju.

Sintaksa	array_insert(array, pos, value)
Argumenti	• **array** - niz • **pos** - pozicija na koju se dodaje (počev od 0) • **value** - vrednost koja se dodaje
Primeri	• `array_insert(array(1,2,3),1,100)` → [1, 100, 2, 3]

array_intersect

Vraća TRUE ako bar jedan element niza array1 postoji u array2.

Sintaksa	array_intersect(array1, array2)
Argumenti	• **array1** - niz • **array2** - drugi niz
Primeri	• `array_intersect(array(1,2,3,4),array(4,0,2,5))` → TRUE

array_last

Vraća poslednju vrednost niza.

Sintaksa	array_last(array)
Argumenti	• **array** - niz
Primeri	• `array_last(array('a','b','c'))` → 'c'

array_length

Vraća broj elemenata niza.

Sintaksa	array_length(array)
Argumenti	• **array** - niz
Primeri	• `array_length(array(1,2,3))` → 3

array_majority

Vraća najčešće vrednosti u nizu.

Sintaksa	array_majority(array, [option:='all']) [] označava opcione argumente
Argumenti	• **array** - niz • **option** - niska koja zadaje rukovanje povratnim vrednostima. Važeće opcije su: – all: Podrazumevano, sve najčešće vrednosti vraćaju se u nizu. – any: Vraća jednu od najčešćih vrednosti. – median: Vraća medijanu najčešćih vrednosti. Nearitmetičke vrednosti se zanemaruju. – real_majority: Vraća vrednost koja se javlja u više od polovine veličine niza.
Primeri	• `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

Vraća maksimalnu vrednost niza.

Sintaksa	array_max(array)
Argumenti	• **array** - niz
Primeri	• `array_max(array(0,42,4,2))` → 42

array_mean

Vraća srednju vrednost aritmetičkih vrednosti u nizu. Nenumeričke vrednosti u nizu se zanemaruju.

Sintaksa	array_mean(array)
Argumenti	• **array** - niz
Primeri	• `array_mean(array(0,1,7,66.6,135.4))` → 42 • `array_mean(array(0,84,'a','b','c'))` → 42

array_median

Vraća medijanu aritmetičkih vrednosti u nizu. Nearitmetičke vrednosti u nizu se zanemaruju.

Sintaksa	array_median(array)
Argumenti	• **array** - niz
Primeri	• `array_median(array(0,1,42,42,43))` → 42 • `array_median(array(0,1,2,42,'a','b'))` → 1.5

array_min

Vraća minimalnu vrednost niza.

Sintaksa	array_min(array)
Argumenti	• **array** - niz
Primeri	• `array_min(array(43,42,54))` → 42

array_minority

Vraća ređe vrednosti u nizu.

Sintaksa	array_minority(array, [option:='all']) [] označava opcione argumente
Argumenti	• **array** - niz • **option** - niska koja zadaje rukovanje povratnim vrednostima. Važeće opcije su: – all: Podrazumevano, sve ređe vrednosti vraćaju se u nizu. – any: Vraća jednu od ređih vrednosti. – median: Vraća medijanu ređih vrednosti. Nearitmetičke vrednosti se zanemaruju. – real_minority: Vraća vrednosti koje se javljaju u manje od polovine veličine niza.
Primeri	• `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

Vraća niz sa zadatom vrednošću dodatom na početak.

Sintaksa	array_prepend(array, value)
Argumenti	• **array** - niz • **value** - vrednost koja se dodaje
Primeri	• `array_prepend(array(1,2,3),0)` → [0, 1, 2, 3]

array_prioritize

Vraća niz sortiran prema redosledu zadatom u drugom nizu. Vrednosti koje postoje u prvom nizu, a nedostaju u drugom, biće dodate na kraj rezultata.

Sintaksa	array_prioritize(array, array_prioritize)
Argumenti	• **array** - niz • **array_prioritize** - niz sa vrednostima poređanim po prioritetu
Primeri	• `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

Vraća niz iz kojeg su uklonjene sve stavke sa zadatom vrednošću.

Sintaksa	array_remove_all(array, value)
Argumenti	• **array** - niz • **value** - vrednosti koje se uklanjaju
Primeri	• `array_remove_all(array('a','b','c','b'),'b')` → ['a', 'c']

array_remove_at

Vraća niz iz kojeg je uklonjena stavka na zadatom indeksu. Podržava pozitivan (0 za prvi element) i negativan indeks (poslednja -N-ta vrednost, -1 za poslednji element).

Sintaksa	array_remove_at(array, pos)
Argumenti	• **array** - niz • **pos** - pozicija koja se uklanja (počev od 0)
Primeri	• `array_remove_at(array(1, 2, 3), 1)` → [1, 3] • `array_remove_at(array(1, 2, 3), -1)` → [1, 2]

array_replace

Vraća niz sa zamenjenom zadatom vrednošću, nizom ili mapom vrednosti.

Varijanta sa vrednošću i nizom

Vraća niz u kojem je zadata vrednost ili niz vrednosti zamenjen drugom vrednošću ili nizom vrednosti.

Sintaksa	array_replace(array, before, after)
Argumenti	• **array** - ulazni niz • **before** - vrednost ili niz vrednosti koje se zamenjuju • **after** - vrednost ili niz vrednosti koje se koriste kao zamena
Primeri	• `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', '-']

Varijanta sa mapom

Vraća niz u kojem su zadati ključevi mape zamenjeni njihovim uparenim vrednostima.

Sintaksa	array_replace(array, map)
Argumenti	• **array** - ulazni niz • **map** - mapa koja sadrži ključeve i vrednosti
Primeri	• `array_replace(array('APP', 'SHOULD', 'ROCK'),map('APP','QGIS', 'SHOULD','DOES'))` → ['QGIS', 'DOES', 'ROCK']

Više informacija: *replace*, *regexp_replace*

array_reverse

Vraća zadati niz sa vrednostima u obrnutom redosledu.

Sintaksa	array_reverse(array)
Argumenti	• **array** - niz
Primeri	• `array_reverse(array(2,4,0,10))` → [10, 0, 4, 2]

array_slice

Vraća deo niza. Isečak je definisan argumentima start_pos i end_pos.

Sintaksa	array_slice(array, start_pos, end_pos)
Argumenti	• **array** - niz • **start_pos** - indeks početne pozicije isečka (počev od 0). Indeks start_pos uključen je u isečak. Ako koristite negativan start_pos, indeks se broji od kraja liste (počev od -1). • **end_pos** - indeks krajnje pozicije isečka (počev od 0). Indeks end_pos uključen je u isečak. Ako koristite negativan end_pos, indeks se broji od kraja liste (počev od -1).
Primeri	• `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

Vraća zadati niz sa sortiranim elementima.

Sintaksa	array_sort(array, [ascending:=true]) [] označava opcione argumente
Argumenti	• **array** - niz • **ascending** - postavite ovaj parametar na false da biste niz sortirali u opadajućem redosledu
Primeri	• `array_sort(array(3,2,1))` → [1, 2, 3]

array_sum

Vraća zbir aritmetičkih vrednosti u nizu. Nenumeričke vrednosti u nizu se zanemaruju.

Sintaksa	array_sum(array)
Argumenti	• **array** - niz
Primeri	• `array_sum(array(0,1,39.4,1.6,'a'))` → 42.0

array_to_string

Spaja elemente niza u nisku razdvojenu graničnikom, uz opcionu nisku za prazne vrednosti.

Sintaksa	array_to_string(array, [delimiter:=','], [emptyvalue:='']) [] označava opcione argumente
Argumenti	• **array** - ulazni niz • **delimiter** - graničnik u vidu niske koji se koristi za razdvajanje spojenih elemenata niza • **emptyvalue** - opciona niska koja se koristi kao zamena za prazna podudaranja (nulte dužine)
Primeri	• `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'

Više informacija: *string_to_array*

generate_series

Pravi niz koji sadrži niz brojeva.

Sintaksa	generate_series(start, stop, [step:=1]) [] označava opcione argumente
Argumenti	• **start** - prva vrednost niza • **stop** - vrednost koja završava niz kada se dostigne • **step** - vrednost koja se koristi kao korak između vrednosti
Primeri	• `generate_series(1,5)` → [1, 2, 3, 4, 5] • `generate_series(5,1,-1)` → [5, 4, 3, 2, 1]

geometries_to_array

Deli geometriju na jednostavnije geometrije u nizu.

Sintaksa	geometries_to_array(geometry)
Argumenti	• **geometry** - ulazna geometrija
Primeri	• `geometries_to_array(geom_from_wkt('MultiPoint (1 2, 5 21)'))` → Niz koji sadrži 'Point (1 2)' i 'Point (5 21)' • `geometries_to_array(geom_from_wkt('GeometryCollection (Polygon ((5 8, 4 1, 3 2, 5 8)),LineString (3 2, 4 2))'))` → niz sa poligonskom i linijskom geometrijom • `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))'))` → niz sa dve poligonske geometrije

regexp_matches

Vraća niz svih niski uhvaćenih grupama za hvatanje, redosledom kojim se same grupe pojavljuju u zadatom regularnom izrazu nad niskom.

Sintaksa	regexp_matches(string, regex, [emptyvalue:=”]) [] označava opcione argumente
Argumenti	• **string** - niska iz koje se hvataju grupe pomoću regularnog izraza • **regex** - regularni izraz koji se koristi za hvatanje grupa • **emptyvalue** - opciona niska koja se koristi kao zamena za prazna podudaranja (nulte dužine)
Primeri	• `regexp_matches('QGIS=>rocks','(.*)=>(.*)')` → [’QGIS’, ’rocks’] • `regexp_matches('key=>','(.*)=>(.*)','empty value')` → [’key’, ’empty value’]

Više informacija: *substr*, *regexp_substr*

string_to_array

Deli nisku u niz pomoću zadatog graničnika i opcione niske za prazne vrednosti.

Sintaksa	string_to_array(string, [delimiter:=’,’], [emptyvalue:=”]) [] označava opcione argumente
Argumenti	• **string** - ulazna niska • **delimiter** - graničnik u vidu niske koji se koristi za deljenje ulazne niske • **emptyvalue** - opciona niska koja se koristi kao zamena za prazna podudaranja (nulte dužine)
Primeri	• `string_to_array('1,2,3',',')` → [’1’, ’2’, ’3’] • `string_to_array('1,,3',',','0')` → [’1’, ’0’, ’3’]

Više informacija: *array_to_string*

9.2.3 Funkcije za boje

Ova grupa sadrži funkcije za rad sa bojama.

color_cmyk

Vraća predstavljanje boje u vidu niske na osnovu njenih cijan, magenta, žute i crne komponente

Sintaksa	color_cmyk(cyan, magenta, yellow, black)
Argumenti	• **cyan** - cijan komponenta boje, kao procentualni ceo broj od 0 do 100 • **magenta** - magenta komponenta boje, kao procentualni ceo broj od 0 do 100 • **yellow** - žuta komponenta boje, kao procentualni ceo broj od 0 do 100 • **black** - crna komponenta boje, kao procentualni ceo broj od 0 do 100
Primeri	• `color_cmyk(100,50,0,10)` → ’0,115,230’

color_cmyka

Vraća predstavljanje boje u vidu niske na osnovu njenih cijan, magenta, žute, crne i alfa (prozirnost) komponente

Sintaksa	color_cmyka(cyan, magenta, yellow, black, alpha)
Argumenti	• **cyan** - cijan komponenta boje, kao procentualni ceo broj od 0 do 100 • **magenta** - magenta komponenta boje, kao procentualni ceo broj od 0 do 100 • **yellow** - žuta komponenta boje, kao procentualni ceo broj od 0 do 100 • **black** - crna komponenta boje, kao procentualni ceo broj od 0 do 100 • **alpha** - alfa komponenta kao ceo broj od 0 (potpuno prozirno) do 255 (neprozirno).
Primeri	• `color_cmyka(100,50,0,10,200)` → ’0,115,230,200’

color_cmykf

Vraća objekat boje na osnovu njenih cijan, magenta, žute, crne i alfa komponente.

Sintaksa	color_cmykf(cyan, magenta, yellow, black, [alpha:=1.0]) [] označava opcione argumente
Argumenti	• **cyan** - cijan komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **magenta** - magenta komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **yellow** - žuta komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **black** - crna komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **alpha** - alfa komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0
Primeri	• `color_cmykf(1,0.9,0.81,0.62)` → CMYKA: 1.00,0.90,0.81,0.62,1.00

color_grayscale_average

Primenjuje filter sivih tonova na boju i vraća je. Vraćeni tip je isti kao tip argumenta boje, tj. predstavljanje boje u vidu niske ili objekat boje.

Sintaksa	color_grayscale_average(color)
Argumenti	• **color** - predstavljanje boje u vidu niske ili objekat boje
Primeri	• `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

Vraća predstavljanje boje u vidu niske na osnovu njenih atributa nijanse, zasićenja i svetline.

Sintaksa	color_hsl(hue, saturation, lightness)
Argumenti	• **hue** - nijansa boje, kao ceo broj od 0 do 360 • **saturation** - procenat zasićenja boje kao ceo broj od 0 do 100 • **lightness** - procenat svetline boje kao ceo broj od 0 do 100
Primeri	• `color_hsl(100,50,70)` → '166,217,140'

color_hsla

Vraća predstavljanje boje u vidu niske na osnovu njenih atributa nijanse, zasićenja, svetline i alfe (prozirnosti)

Sintaksa	color_hsla(hue, saturation, lightness, alpha)
Argumenti	• **hue** - nijansa boje, kao ceo broj od 0 do 360 • **saturation** - procenat zasićenja boje kao ceo broj od 0 do 100 • **lightness** - procenat svetline boje kao ceo broj od 0 do 100 • **alpha** - alfa komponenta kao ceo broj od 0 (potpuno prozirno) do 255 (neprozirno).
Primeri	• `color_hsla(100,50,70,200)` → '166,217,140,200'

color_hslf

Vraća objekat boje na osnovu njenih atributa nijanse, zasićenja i svetline.

Sintaksa	color_hslf(hue, saturation, lightness, [alpha:=1.0]) [] označava opcione argumente
Argumenti	• **hue** - nijansa boje, kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **saturation** - zasićenje boje kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **lightness** - svetlina boje kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **alpha** - alfa komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0
Primeri	• `color_hslf(0.3,0.52,0.7)` → HSLA: 0.30,0.52,0.70,1.00

color_hsv

Vraća predstavljanje boje u vidu niske na osnovu njenih atributa nijanse, zasićenja i vrednosti.

Sintaksa	color_hsv(hue, saturation, value)
Argumenti	• **hue** - nijansa boje, kao ceo broj od 0 do 360 • **saturation** - procenat zasićenja boje kao ceo broj od 0 do 100 • **value** - procenat vrednosti boje kao ceo broj od 0 do 100
Primeri	• `color_hsv(40,100,100)` → '255,170,0'

color_hsva

Vraća predstavljanje boje u vidu niske na osnovu njenih atributa nijanse, zasićenja, vrednosti i alfe (prozirnosti).

Sintaksa	color_hsva(hue, saturation, value, alpha)
Argumenti	• **hue** - nijansa boje, kao ceo broj od 0 do 360 • **saturation** - procenat zasićenja boje kao ceo broj od 0 do 100 • **value** - procenat vrednosti boje kao ceo broj od 0 do 100 • **alpha** - alfa komponenta kao ceo broj od 0 (potpuno prozirno) do 255 (neprozirno)
Primeri	• `color_hsva(40,100,100,200)` → '255,170,0,200'

color_hsvf

Vraća objekat boje na osnovu njenih atributa nijanse, zasićenja i vrednosti.

Sintaksa	color_hsvf(hue, saturation, value, [alpha:=1.0]) [] označava opcione argumente
Argumenti	• **hue** - nijansa boje, kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **saturation** - zasićenje boje kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **value** - vrednost boje kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **alpha** - alfa komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0
Primeri	• `color_hsvf(0.4,1,0.6)` → HSVA: 0.40,1.00,0.60,1.00

color_mix

Vraća boju koja meša crvenu, zelenu, plavu i alfa vrednost dve zadate boje na osnovu zadatog odnosa. Vraćeni tip je isti kao tip argumenata boje, tj. predstavljanje boje u vidu niske ili objekat boje.

Sintaksa	color_mix(color1, color2, ratio)
Argumenti	• **color1** - niska boje ili objekat boje • **color2** - niska boje ili objekat boje • **ratio** - odnos
Primeri	• `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

Vraća nisku koja predstavlja boju koja meša crvenu, zelenu, plavu i alfa vrednost dve zadate boje na osnovu zadatog odnosa.

Sintaksa	color_mix_rgb(color1, color2, ratio)
Argumenti	• **color1** - niska boje • **color2** - niska boje • **ratio** - odnos
Primeri	• `color_mix_rgb('0,0,0','255,255,255',0.5)` → '127,127,127,255'

color_part

Vraća određenu komponentu iz niske boje ili objekta boje, npr. crvenu komponentu ili alfa komponentu.

Sintaksa	color_part(color, component)
Argumenti	• **color** - niska boje ili objekat boje • **component** - niska koja odgovara komponenti boje koja se vraća. Važeće opcije su: – red: RGB crvena komponenta (0-255) – green: RGB zelena komponenta (0-255) – blue: RGB plava komponenta (0-255) – alpha: alfa vrednost (prozirnost) (0-255) – hue: HSV nijansa (0-360) – saturation: HSV zasićenje (0-100) – value: HSV vrednost (0-100) – hsl_hue: HSL nijansa (0-360) – hsl_saturation: HSL zasićenje (0-100) – lightness: HSL svetlina (0-100) – cyan: CMYK cijan komponenta (0-100) – magenta: CMYK magenta komponenta (0-100) – yellow: CMYK žuta komponenta (0-100) – black: CMYK crna komponenta (0-100)
Primeri	• `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

Vraća predstavljanje boje u vidu niske na osnovu njenih crvene, zelene i plave komponente.

Sintaksa	color_rgb(red, green, blue)
Argumenti	• **red** - crvena komponenta kao ceo broj od 0 do 255 • **green** - zelena komponenta kao ceo broj od 0 do 255 • **blue** - plava komponenta kao ceo broj od 0 do 255
Primeri	• `color_rgb(255,127,0)` → '255,127,0'

color_rgba

Vraća predstavljanje boje u vidu niske na osnovu njenih crvene, zelene, plave i alfa (prozirnost) komponente.

Sintaksa	color_rgba(red, green, blue, alpha)
Argumenti	• **red** - crvena komponenta kao ceo broj od 0 do 255 • **green** - zelena komponenta kao ceo broj od 0 do 255 • **blue** - plava komponenta kao ceo broj od 0 do 255 • **alpha** - alfa komponenta kao ceo broj od 0 (potpuno prozirno) do 255 (neprozirno).
Primeri	• `color_rgba(255,127,0,200)` → '255,127,0,200'

color_rgbf

Vraća objekat boje na osnovu njenih crvene, zelene, plave i alfa komponente.

Sintaksa	color_rgbf(red, green, blue, [alpha:=1.0]) [] označava opcione argumente
Argumenti	• **red** - crvena komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **green** - zelena komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **blue** - plava komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0 • **alpha** - alfa komponenta kao vrednost u pokretnom zarezu od 0.0 do 1.0
Primeri	• `color_rgbf(1.0,0.5,0)` → RGBA: 1.00,0.50,0.00,1.00

create_ramp

Vraća gradijentnu rampu od mape niski boja i koraka.

Sintaksa	create_ramp(map, [discrete:=false]) [] označava opcione argumente
Argumenti	• **map** - mapa niski boja i koraka • **discrete** - postavite ovaj parametar na true da biste napravili diskretnu rampu boja
Primeri	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → '255,0,0,255'

darker

Vraća tamniju (ili svetliju) boju. Vraćeni tip je isti kao tip argumenata boje, tj. predstavljanje boje u vidu niske ili objekat boje.

Sintaksa	darker(color, factor)
Argumenti	• **color** - niska boje ili objekat boje • **factor** - ceo broj koji odgovara faktoru potamnjivanja: – ako je faktor veći od 100, ova funkcija vraća tamniju boju (npr. postavljanje faktora na 200 vraća boju upola manje svetline); – ako je faktor manji od 100, vraćena boja je svetlija, ali se za tu svrhu preporučuje funkcija lighter(); – ako je faktor 0 ili negativan, vraćena vrednost nije određena.
Primeri	• `darker('200,10,30', 200)` → '100,5,15,255'

Više informacija: *lighter*

lighter

Vraća svetliju (ili tamniju) boju. Vraćeni tip je isti kao tip argumenata boje, tj. predstavljanje boje u vidu niske ili objekat boje.

Sintaksa	lighter(color, factor)
Argumenti	• **color** - niska boje ili objekat boje • **factor** - ceo broj koji odgovara faktoru posvetljivanja: – ako je faktor veći od 100, ova funkcija vraća svetliju boju (npr. postavljanje faktora na 150 vraća boju koja je 50% svetlija); – ako je faktor manji od 100, vraćena boja je tamnija, ali se za tu svrhu preporučuje funkcija darker(); – ako je faktor 0 ili negativan, vraćena vrednost nije određena.
Primeri	• `lighter('200,10,30', 200)` → '255,158,168,255'

Više informacija: *darker*

project_color

Vraća boju iz šeme boja projekta.

Sintaksa	project_color(name)
Argumenti	• **name** - naziv boje
Primeri	• `project_color('Logo color')` → '20,140,50'

Više informacija: *podešavanje boja projekta*

project_color_object

Vraća boju iz šeme boja projekta. Za razliku od project_color, koja vraća predstavljanje boje u vidu niske, project_color_object vraća objekat boje.

Sintaksa	project_color_object(name)
Argumenti	• **name** - naziv boje
Primeri	• `project_color_object('Logo color')` → RGBA: 0.08,0.55,0.20,1.00

ramp_color

Vraća nisku koja predstavlja boju sa rampe boja.

Varijanta sa sačuvanom rampom

Vraća nisku koja predstavlja boju sa sačuvane rampe

Sintaksa	ramp_color(ramp_name, value)
Argumenti	• **ramp_name** - naziv rampe boja kao niska, na primer 'Spectral' • **value** - pozicija na rampi sa koje se bira boja, kao realan broj između 0 i 1
Primeri	• `ramp_color('Spectral',0.3)` → '253,190,115,255'

Белешка

Rampe boja se razlikuju između QGIS instalacija. Ova funkcija možda neće dati očekivane rezultate ako QGIS projekat premeštate između instalacija.

Varijanta sa rampom napravljenom izrazom

Vraća nisku koja predstavlja boju sa rampe napravljene izrazom

Sintaksa	ramp_color(ramp, value)
Argumenti	• **ramp** - rampa boja • **value** - pozicija na rampi sa koje se bira boja, kao realan broj između 0 i 1
Primeri	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → '255,0,0,255'

Više informacija: *Setting a Color Ramp*, *The color ramp drop-down shortcut*

ramp_color_object

Vraća objekat boje sa rampe boja. Za razliku od ramp_color, koja vraća predstavljanje boje u vidu niske, ramp_color_object vraća objekat boje.

Varijanta sa sačuvanom rampom

Vraća objekat boje iz sačuvane rampe

Sintaksa	ramp_color_object(ramp_name, value)
Argumenti	• **ramp_name** - naziv rampe boja kao niska, na primer 'Spectral' • **value** - pozicija na rampi sa koje se bira boja, kao realan broj između 0 i 1
Primeri	• `ramp_color_object('Spectral',0.3)` → RGBA: 0.99,0.75,0.45,1.00

Белешка

Rampe boja se razlikuju između QGIS instalacija. Ova funkcija možda neće dati očekivane rezultate ako QGIS projekat premeštate između instalacija.

Varijanta sa rampom napravljenom izrazom

Vraća objekat boje sa rampe napravljene izrazom

Sintaksa	ramp_color_object(ramp, value)
Argumenti	• **ramp** - rampa boja • **value** - pozicija na rampi sa koje se bira boja, kao realan broj između 0 i 1
Primeri	• `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

Postavlja određenu komponentu boje za nisku boje ili objekat boje, npr. crvenu komponentu ili alfa komponentu.

Sintaksa	set_color_part(color, component, value)
Argumenti	• **color** - niska boje ili objekat boje • **component** - niska koja odgovara komponenti boje koja se postavlja. Važeće opcije su: – red: RGB crvena komponenta (0-255) – green: RGB zelena komponenta (0-255) – blue: RGB plava komponenta (0-255) – alpha: alfa vrednost (prozirnost) (0-255) – hue: HSV nijansa (0-360) – saturation: HSV zasićenje (0-100) – value: HSV vrednost (0-100) – hsl_hue: HSL nijansa (0-360) – hsl_saturation: HSL zasićenje (0-100) – lightness: HSL svetlina (0-100) – cyan: CMYK cijan komponenta (0-100) – magenta: CMYK magenta komponenta (0-100) – yellow: CMYK žuta komponenta (0-100) – black: CMYK crna komponenta (0-100) • **value** - nova vrednost za komponentu boje, u skladu sa gore navedenim opsezima
Primeri	• `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 Uslovne funkcije

Ova grupa sadrži funkcije za rukovanje uslovnim proverama u izrazima.

CASE

CASE se koristi za izračunavanje niza uslova i vraćanje rezultata za prvi ispunjeni uslov. Uslovi se izračunavaju redom, a ako je uslov tačan, izračunavanje se zaustavlja i vraća se odgovarajući rezultat. Ako nijedan od uslova nije tačan, vraća se vrednost iz ELSE klauzule. Osim toga, ako ELSE klauzula nije postavljena i nijedan uslov nije ispunjen, vraća se NULL.

CASE

WHEN *condition* THEN *result*

[…n]

[ELSE *result*]

END

[ ] označava opcione komponente

Argumenti	• **WHEN condition** - Uslovni izraz koji se izračunava • **THEN result** - Ako *condition* daje True, izračunava se i vraća *result*. • **ELSE result** - Ako nijedan od gornjih uslova nije dao True, izračunava se i vraća *result*.
Primeri	• `CASE WHEN "name" IS NULL THEN 'None' END` → Vraća nisku 'None' ako je polje „name" NULL • `CASE WHEN $area > 10000 THEN 'Big property' WHEN $area > 5000 THEN 'Medium property' ELSE 'Small property' END` → Returns the string 'Big property' if the area is bigger than 10000, 'Medium property' if the area is between 5000 and 10000, and 'Small property' for others

coalesce

Vraća prvu vrednost koja nije NULL iz liste izraza.

Ova funkcija može primiti bilo koji broj argumenata.

Sintaksa	coalesce(expression1, expression2, …)
Argumenti	• **expression** - bilo koji ispravan izraz ili vrednost, bez obzira na tip.
Primeri	• `coalesce(NULL, 2)` → 2 • `coalesce(NULL, 2, 3)` → 2 • `coalesce(7, NULL, 3*2)` → 7 • `coalesce("fieldA", "fallbackField", 'ERROR')` → vrednost polja fieldA ako nije NULL, u suprotnom vrednost polja „fallbackField", ili niska 'ERROR' ako su oba NULL

if

Testira uslov i vraća različit rezultat u zavisnosti od uslovne provere.

Sintaksa	if(condition, result_when_true, result_when_false)
Argumenti	• **condition** - uslov koji treba proveriti • **result_when_true** - rezultat koji se vraća kada je uslov tačan ili neka druga vrednost koja se ne pretvara u false. • **result_when_false** - rezultat koji se vraća kada je uslov netačan ili neka druga vrednost koja se pretvara u false, kao 0 ili ”. I NULL se pretvara u false.
Primeri	• `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

Vraća NULL vrednost ako je value1 jednako value2; u suprotnom vraća value1. Ovo se može koristiti za uslovnu zamenu vrednosti sa NULL.

Sintaksa	nullif(value1, value2)
Argumenti	• **value1** - Vrednost koja će se ili koristiti ili zameniti sa NULL. • **value2** - Kontrolna vrednost koja pokreće zamenu sa NULL.
Primeri	• `nullif('(none)', '(none)')` → NULL • `nullif('text', '(none)')` → ’text’ • `nullif("name", '')` → NULL, ako je name prazna niska (ili već NULL), a name u svim ostalim slučajevima.

regexp_match

Vraća prvu poziciju podudaranja sa regularnim izrazom unutar unicode niske, ili 0 ako podniska nije pronađena.

Sintaksa	regexp_match(input_string, regex)
Argumenti	• **input_string** - niska koja se testira regularnim izrazom • **regex** - Regularni izraz sa kojim se testira. Znakovi obrnute kose crte moraju biti dvostruko izbegnuti (npr. „\\s” za podudaranje sa belim znakom ili „\\b” za podudaranje sa granicom reči).
Primeri	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

try

Pokušava izraz i vraća njegovu vrednost ako nema greške. Ako izraz vrati grešku, vratiće se alternativna vrednost ako je zadata, u suprotnom funkcija vraća NULL.

Sintaksa	try(expression, [alternative]) [] označava opcione argumente
Argumenti	• **expression** - izraz koji treba pokrenuti • **alternative** - rezultat koji će biti vraćen ako izraz vrati grešku.
Primeri	• `try( to_int( '1' ), 0 )` → 1 • `try( to_int( 'a' ), 0 )` → 0 • `try( to_date( 'invalid_date' ) )` → NULL

9.2.5 Funkcije za pretvaranje

Ova grupa sadrži funkcije za pretvaranje jednog tipa podataka u drugi (npr. niska iz/u ceo broj, binarno iz/u nisku, niska u datum, …).

from_base64

Dekodira nisku u Base64 kodiranju u binarnu vrednost.

Sintaksa	from_base64(string)
Argumenti	• **string** - niska koja se dekodira
Primeri	• `from_base64('UUdJUw==')` → 'QGIS'

hash

Pravi heš od niske zadatim metodom. Jedan bajt (8 bitova) predstavlja se sa dve heksadecimalne "cifre", pa 'md4' (16 bajtova) daje heksadecimalnu nisku dužine 16 * 2 = 32 znaka, a 'keccak_512' (64 bajta) daje heksadecimalnu nisku dužine 64 * 2 = 128 znakova.

Sintaksa	hash(string, method)
Argumenti	• **string** - niska za koju se pravi heš • **method** - The hash method among 'md4', 'md5', 'sha1', 'sha224', 'sha384', 'sha512', 'sha3_224', 'sha3_256', 'sha3_384', 'sha3_512', 'keccak_224', 'keccak_256', 'keccak_384', 'keccak_512'
Primeri	• `hash('QGIS', 'md4')` → 'c0fc71c241cdebb6e888cbac0e2b68eb' • `hash('QGIS', 'md5')` → '57470aaa9e22adaefac7f5f342f1c6da' • `hash('QGIS', 'sha1')` → 'f87cfb2b74cdd5867db913237024e7001e62b114' • `hash('QGIS', 'sha224')` → '4093a619ada631c770f44bc643ead18fb393b93d6a6af1861fcfece0' • `hash('QGIS', 'sha256')` → 'eb045cba7a797aaa06ac58830846e40c8e8c780bc0676d3393605fae50c053 • `hash('QGIS', 'sha384')` → '91c1de038cc3d09fdd512e99f9dd9922efadc39ed21d3922e69a4305cc2550 • `hash('QGIS', 'sha512')` → 'c2c092f2ab743bf8edbeb6d028a745f30fc720408465ed369421f0a4e20fa5 • `hash('QGIS', 'sha3_224')` → '467f49a5039e7280d5d42fd433e80d203439e338eaabd701f0d6c17d' • `hash('QGIS', 'sha3_256')` → '540f7354b6b8a6e735f2845250f15f4f3ba4f666c55574d9e9354575de0 • `hash('QGIS', 'sha3_384')` → '96052da1e77679e9a65f60d7ead961b2879778231447863 86eb43647b0 • `hash('QGIS', 'sha3_512')` → '900d079dc69761da113980253aa8ac0414a8bd6d09879a916228f87437 • `hash('QGIS', 'keccak_224')` → '5b0ce6acef8b0a121d4ac4f3eaa8503c799ad4e26a3392d1fb201478' • `hash('QGIS', 'keccak_256')` → '991c520aa6815392de24087f61b2ae0fd56abbfeee4a8ca019c1011d3 • `hash('QGIS', 'keccak_384')` → 'c57a3aed9d856fa04e5eeee9b62b6e027cca81ba574116d3cc1f0d48a1

md5

Pravi md5 heš od niske.

Sintaksa	md5(string)
Argumenti	• **string** - niska za koju se pravi heš
Primeri	• `md5('QGIS')` → '57470aaa9e22adaefac7f5f342f1c6da'

sha256

Pravi sha256 heš od niske.

Sintaksa	sha256(string)
Argumenti	• **string** - niska za koju se pravi heš
Primeri	• `sha256('QGIS')` → 'eb045cba7a797aaa06ac58830846e40c8e8c780bc0676d3393605fae50c05309'

to_base64

Kodira binarnu vrednost u nisku koristeći Base64 kodiranje.

Sintaksa	to_base64(value)
Argumenti	• **value** - binarna vrednost koja se kodira
Primeri	• `to_base64('QGIS')` → 'UUdJUw=='

to_bool

Pretvara zadatu vrednost u logičku vrednost. Funkcija vraća false ako je vrednost NULL, prazna niska, prazna lista ili 0.

Sintaksa	to_bool(value)
Argumenti	• **value** - vrednost koja se pretvara u logičku vrednost
Primeri	• `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

Pretvara nisku u objekat datuma. Može se zadati opciona niska formata za parsiranje; pogledajte QDate::fromString ili dokumentaciju funkcije format_date za dodatnu dokumentaciju o formatu. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Sintaksa	to_date(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja vrednost datuma • **format** - format koji se koristi za pretvaranje niske u datum • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u datum. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.
Primeri	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, ako trenutni lokalitet koristi naziv 'June' za šesti mesec, u suprotnom dolazi do greške • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Više informacija: *format_date*

to_datetime

Pretvara nisku u datetime objekat. Može se zadati opciona niska formata za parsiranje; za dodatnu dokumentaciju o formatu pogledajte QDate::fromString, QTime::fromString ili dokumentaciju funkcije format_date. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Sintaksa	to_datetime(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja datetime vrednost • **format** - format koji se koristi za pretvaranje niske u datetime • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u datetime. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.
Primeri	• `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, ako trenutni lokalitet koristi naziv 'June' za šesti mesec, u suprotnom dolazi do greške • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm','fr')` → 2019-06-29T12:34

Više informacija: *format_date*

to_decimal

Pretvara koordinatu u stepenima, minutima i sekundama u njen decimalni ekvivalent.

Sintaksa	to_decimal(value)
Argumenti	• **value** - Niska sa stepenima, minutima, sekundama.
Primeri	• `to_decimal('6°21\'16.445')` → 6.3545680555

to_dm

Pretvara koordinatu u stepen, minut.

Sintaksa	to_dm(coordinate, axis, precision, [formatting:=NULL]) [] označava opcione argumente
Argumenti	• **coordinate** - Vrednost geografske širine ili dužine. • **axis** - Osa koordinate. Ili 'x' ili 'y'. • **precision** - Broj decimala. • **formatting** - Određuje tip formatiranja. Prihvatljive vrednosti su NULL (podrazumevano), 'aligned' ili 'suffix'.
Primeri	• `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

Pretvara koordinatu u stepen, minut, sekund.

Sintaksa	to_dms(coordinate, axis, precision, [formatting:=NULL]) [] označava opcione argumente
Argumenti	• **coordinate** - Vrednost geografske širine ili dužine. • **axis** - Osa koordinate. Ili 'x' ili 'y'. • **precision** - Broj decimala. • **formatting** - Određuje tip formatiranja. Prihvatljive vrednosti su NULL (podrazumevano), 'aligned' ili 'suffix'.
Primeri	• `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

Pretvara nisku u ceo broj. Ako se vrednost ne može pretvoriti u ceo broj, izraz je neispravan (npr. '123asd' je neispravno).

Sintaksa	to_int(string)
Argumenti	• **string** - niska koja se pretvara u ceo broj
Primeri	• `to_int('123')` → 123

to_interval

Pretvara nisku u tip interval. Može se koristiti za uzimanje dana, sati, meseci itd. iz datuma.

Sintaksa	to_interval(string)
Argumenti	• **string** - niska koja predstavlja interval. Dozvoljeni formati uključuju {n} days {n} hours {n} months.
Primeri	• `to_interval('1 day 2 hours')` → interval: 1.08333 days • `to_interval( '0.5 hours' )` → interval: 30 minuta • `to_datetime('2012-05-05 12:00:00') - to_interval('1 day 2 hours')` → 2012-05-04T10:00:00

to_real

Pretvara nisku u realan broj. Ako se vrednost ne može pretvoriti u realan broj, izraz je neispravan (npr. '123.56asd' je neispravno). Brojevi se zaokružuju nakon čuvanja izmena ako je preciznost manja od rezultata pretvaranja.

Sintaksa	to_real(string)
Argumenti	• **string** - niska koja se pretvara u realan broj
Primeri	• `to_real('123.45')` → 123.45

to_string

Pretvara broj u nisku. Pretvaranje ne uzima u obzir lokalitet; za alternativu koja uzima lokalitet u obzir pogledajte 'format_number'.

Sintaksa	to_string(number)
Argumenti	• **number** - Ceo ili realan broj. Broj koji se pretvara u nisku.
Primeri	• `to_string(1.23)` → '1.23'

Više informacija: *format_number*

to_time

Pretvara nisku u objekat vremena. Može se zadati opciona niska formata za parsiranje; za dodatnu dokumentaciju o formatu pogledajte QTime::fromString.

Sintaksa	to_time(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja vrednost vremena • **format** - format koji se koristi za pretvaranje niske u vreme • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u vreme
Primeri	• `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

Više informacija: *format_date*

9.2.6 Funkcije za CRS

Ova grupa sadrži funkcije za rad sa objektima koordinatnog referentnog sistema.

crs_from_text

Pravi koordinatni referentni sistem iz definicije u vidu niske. Niska može predstavljati ID autoriteta, WKT definiciju ili PROJ definiciju CRS-a.

Sintaksa	crs_from_text(definition)
Argumenti	• **definition** - definicija CRS-a
Primeri	• `crs_from_text('EPSG:3857')` → crs vrednost '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 vrednost 'EPSG:3857 - WGS 84 / Pseudo-Mercator'

crs_to_authid

Vraća nisku ID-ja autoriteta za koordinatni referentni sistem.

Sintaksa	crs_to_authid(crs)
Argumenti	• **crs** - crs vrednost
Primeri	• `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 Prilagođene funkcije

Ova grupa sadrži funkcije koje je napravio korisnik. Više detalja u *Function Editor*.

9.2.8 Funkcije za datum i vreme

Ova grupa sadrži funkcije za rad sa podacima o datumu i vremenu. Ova grupa deli nekoliko funkcija sa grupama *Funkcije za pretvaranje* (to_date, to_time, to_datetime, to_interval) i *Funkcije za niske* (format_date).

Белешка

Čuvanje datuma, datetime i intervala u poljima

Mogućnost direktnog čuvanja vrednosti *date*, *time* i *datetime* u poljima zavisi od provajdera izvora podataka (npr. Shapefile prihvata format *date*, ali ne i *datetime* ili *time*). Sledi nekoliko predloga za zaobilaženje ovog ograničenja:

- *date*, *datetime* i *time* mogu se pretvoriti i sačuvati u poljima tipa tekst pomoću funkcije *format_date()*.
- *Intervali* se mogu čuvati u poljima tipa ceo ili decimalan broj nakon korišćenja jedne od funkcija za izdvajanje datuma (npr. *day()* za dobijanje intervala izraženog u danima)

age

Vraća razliku između dva datuma ili datetime vrednosti.

Razlika se vraća kao `Interval` i mora se koristiti sa jednom od sledećih funkcija da bi se dobile korisne informacije:

- `year`
- `month`
- `week`
- `day`
- `hour`
- `minute`
- `second`

Sintaksa	age(datetime1, datetime2)
Argumenti	• **datetime1** - niska, datum ili datetime koji predstavlja kasniji datum • **datetime2** - niska, datum ili datetime koji predstavlja raniji datum
Primeri	• `day(age('2012-05-12','2012-05-02'))` → 10 • `hour(age('2012-05-12','2012-05-02'))` → 240

datetime_from_epoch

Vraća datetime čiji su datum i vreme broj milisekundi, msecs, koje su protekle od 1970-01-01T00:00:00.000, po koordiniranom svetskom vremenu (Qt.UTC), pretvoren u Qt.LocalTime.

Sintaksa	datetime_from_epoch(int)
Argumenti	• **int** - broj (milisekunde)
Primeri	• `datetime_from_epoch(1483225200000)` → 2017-01-01T00:00:00

day

Izdvaja dan iz datuma, ili broj dana iz intervala.

Varijanta sa datumom

Izdvaja dan iz datuma ili datetime.

Sintaksa	day(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `day('2012-05-12')` → 12

Varijanta sa intervalom

Računa dužinu intervala u danima.

Sintaksa	day(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj dana
Primeri	• `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

Vraća dan u nedelji za zadati datum ili datetime. Vraćena vrednost je u opsegu od 0 do 6, gde 0 odgovara nedelji, a 6 suboti.

Sintaksa	day_of_week(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `day_of_week(to_date('2015-09-21'))` → 1

epoch

Vraća interval u milisekundama između unix epohe i zadate vrednosti datuma.

Sintaksa	epoch(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `epoch(to_date('2017-01-01'))` → 1483203600000

format_date

Formatira tip datuma ili nisku u prilagođeni format niske. Koristi Qt formate datuma/vremena. Pogledajte QDateTime::toString.

Sintaksa

format_date(datetime, format, [language])
[] označava opcione argumente

Argumenti

- **datetime** - vrednost datuma, vremena ili datetime
- **format** - Šablon niske koji se koristi za formatiranje niske.

Expression	Izlaz
d	dan kao broj bez vodeće nule (1 do 31)
dd	dan kao broj sa vodećom nulom (01 do 31)
ddd	skraćeni lokalizovani naziv dana (npr. 'Mon' do 'Sun')
dddd	dugi lokalizovani naziv dana (npr. 'Monday' do 'Sunday')
M	mesec kao broj bez vodeće nule (1-12)
MM	mesec kao broj sa vodećom nulom (01-12)
MMM	skraćeni lokalizovani naziv meseca (npr. 'Jan' do 'Dec')
MMMM	dugi lokalizovani naziv meseca (npr. 'January' do 'December')
yy	godina kao dvocifren broj (00-99)
yyyy	godina kao četvorocifren broj

Ovi izrazi se mogu koristiti za vremenski deo niske formata:

Expression	Izlaz
h	sat bez vodeće nule (0 do 23 ili 1 do 12 ako je prikaz AM/PM)
hh	sat sa vodećom nulom (00 do 23 ili 01 do 12 ako je prikaz AM/PM)
H	sat bez vodeće nule (0 do 23, čak i uz AM/PM prikaz)
HH	sat sa vodećom nulom (00 do 23, čak i uz AM/PM prikaz)
m	minut bez vodeće nule (0 do 59)
mm	minut sa vodećom nulom (00 do 59)
s	sekunda bez vodeće nule (0 do 59)
ss	sekunda sa vodećom nulom (00 do 59)
z	milisekunde bez pratećih nula (0 do 999)
zzz	milisekunde sa pratećim nulama (000 do 999)
AP ili A	tumači se kao AM/PM vreme. *AP* mora biti ili 'AM' ili 'PM'.
ap ili a	Tumači se kao AM/PM vreme. *ap* mora biti ili 'am' ili 'pm'.

- **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za formatiranje datuma u prilagođenu nisku. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Primeri

- `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', ako je trenutni lokalitet neka engleska varijanta
- `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

Izdvaja deo sa satom iz datetime ili vremena, ili broj sati iz intervala.

Varijanta sa vremenom

Izdvaja deo sa satom iz vremena ili datetime.

Sintaksa	hour(datetime)
Argumenti	• **datetime** - vrednost vremena ili datetime
Primeri	• `hour( to_datetime('2012-07-22 13:24:57') )` → 13

Varijanta sa intervalom

Računa dužinu intervala u satima.

Sintaksa	hour(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj sati
Primeri	• `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

Pravi vrednost datuma od brojeva godine, meseca i dana.

Sintaksa	make_date(year, month, day)
Argumenti	• **year** - Broj godine. Godine 1 do 99 tumače se doslovno. Godina 0 je neispravna. • **month** - Broj meseca, gde je 1=januar • **day** - Broj dana, počev od 1 za prvi dan u mesecu
Primeri	• `make_date(2020,5,4)` → vrednost datuma 2020-05-04

make_datetime

Pravi datetime vrednost od brojeva godine, meseca, dana, sata, minuta i sekunde.

Sintaksa	make_datetime(year, month, day, hour, minute, second)
Argumenti	• **year** - Broj godine. Godine 1 do 99 tumače se doslovno. Godina 0 je neispravna. • **month** - Broj meseca, gde je 1=januar • **day** - Broj dana, počev od 1 za prvi dan u mesecu • **hour** - Broj sata • **minute** - Minuti • **second** - Sekunde (decimalne vrednosti uključuju milisekunde)
Primeri	• `make_datetime(2020,5,4,13,45,30.5)` → datetime vrednost 2020-05-04 13:45:30.500

make_interval

Pravi vrednost intervala od vrednosti godina, meseci, nedelja, dana, sati, minuta i sekundi.

Sintaksa	make_interval([years:=0], [months:=0], [weeks:=0], [days:=0], [hours:=0], [minutes:=0], [seconds:=0]) [] označava opcione argumente
Argumenti	• **years** - Broj godina (pretpostavlja se dužina godine od 365.25 dana). • **months** - Broj meseci (pretpostavlja se dužina meseca od 30 dana) • **weeks** - Broj nedelja • **days** - Broj dana • **hours** - Broj sati • **minutes** - Broj minuta • **seconds** - Broj sekundi
Primeri	• `make_interval(hours:=3)` → interval: 3 sata • `make_interval(days:=2, hours:=3)` → interval: 2.125 dana • `make_interval(minutes:=0.5, seconds:=5)` → interval: 35 sekundi

make_time

Pravi vrednost vremena od brojeva sata, minuta i sekunde.

Sintaksa	make_time(hour, minute, second)
Argumenti	• **hour** - Broj sata • **minute** - Minuti • **second** - Sekunde (decimalne vrednosti uključuju milisekunde)
Primeri	• `make_time(13,45,30.5)` → vrednost vremena 13:45:30.500

minute

Izdvaja deo sa minutima iz datetime ili vremena, ili broj minuta iz intervala.

Varijanta sa vremenom

Izdvaja deo sa minutima iz vremena ili datetime.

Sintaksa	minute(datetime)
Argumenti	• **datetime** - vrednost vremena ili datetime
Primeri	• `minute( to_datetime('2012-07-22 13:24:57') )` → 24

Varijanta sa intervalom

Računa dužinu intervala u minutima.

Sintaksa	minute(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj minuta
Primeri	• `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

Izdvaja deo sa mesecom iz datuma, ili broj meseci iz intervala.

Varijanta sa datumom

Izdvaja deo sa mesecom iz datuma ili datetime.

Sintaksa	month(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `month('2012-05-12')` → 05

Varijanta sa intervalom

Računa dužinu intervala u mesecima.

Sintaksa	month(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj meseci
Primeri	• `month(to_interval('3 months'))` → 3 • `month(age('2012-01-01','2010-01-01'))` → 4.03333

now

Vraća trenutni datum i vreme. Funkcija je statička i daje dosledne rezultate tokom izračunavanja. Vraćeno vreme je vreme kada je izraz pripremljen.

Sintaksa	now()
Primeri	• `now()` → 2012-07-22T13:24:57

second

Izdvaja deo sa sekundama iz datetime ili vremena, ili broj sekundi iz intervala.

Varijanta sa vremenom

Izdvaja deo sa sekundama iz vremena ili datetime.

Sintaksa	second(datetime)
Argumenti	• **datetime** - vrednost vremena ili datetime
Primeri	• `second( to_datetime('2012-07-22 13:24:57') )` → 57

Varijanta sa intervalom

Računa dužinu intervala u sekundama.

Sintaksa	second(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj sekundi
Primeri	• `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

Pretvara nisku u objekat datuma. Može se zadati opciona niska formata za parsiranje; pogledajte QDate::fromString ili dokumentaciju funkcije format_date za dodatnu dokumentaciju o formatu. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Sintaksa	to_date(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja vrednost datuma • **format** - format koji se koristi za pretvaranje niske u datum • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u datum. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.
Primeri	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, ako trenutni lokalitet koristi naziv 'June' za šesti mesec, u suprotnom dolazi do greške • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Više informacija: *format_date*

to_datetime

Pretvara nisku u datetime objekat. Može se zadati opciona niska formata za parsiranje; za dodatnu dokumentaciju o formatu pogledajte QDate::fromString, QTime::fromString ili dokumentaciju funkcije format_date. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Sintaksa	to_datetime(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja datetime vrednost • **format** - format koji se koristi za pretvaranje niske u datetime • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u datetime. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.
Primeri	• `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, ako trenutni lokalitet koristi naziv 'June' za šesti mesec, u suprotnom dolazi do greške • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm','fr')` → 2019-06-29T12:34

Više informacija: *format_date*

to_interval

Pretvara nisku u tip interval. Može se koristiti za uzimanje dana, sati, meseci itd. iz datuma.

Sintaksa	to_interval(string)
Argumenti	• **string** - niska koja predstavlja interval. Dozvoljeni formati uključuju {n} days {n} hours {n} months.
Primeri	• `to_interval('1 day 2 hours')` → interval: 1.08333 days • `to_interval( '0.5 hours' )` → interval: 30 minuta • `to_datetime('2012-05-05 12:00:00') - to_interval('1 day 2 hours')` → 2012-05-04T10:00:00

to_time

Pretvara nisku u objekat vremena. Može se zadati opciona niska formata za parsiranje; za dodatnu dokumentaciju o formatu pogledajte QTime::fromString.

Sintaksa	to_time(string, [format], [language]) [] označava opcione argumente
Argumenti	• **string** - niska koja predstavlja vrednost vremena • **format** - format koji se koristi za pretvaranje niske u vreme • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za pretvaranje niske u vreme
Primeri	• `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

Više informacija: *format_date*

week

Izdvaja broj nedelje iz datuma, ili broj nedelja iz intervala.

Varijanta sa datumom

Izdvaja broj nedelje iz datuma ili datetime.

Sintaksa	week(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `week('2012-05-12')` → 19

Varijanta sa intervalom

Računa dužinu intervala u nedeljama.

Sintaksa	week(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj meseci
Primeri	• `week(to_interval('3 weeks'))` → 3 • `week(age('2012-01-01','2010-01-01'))` → 104.285

year

Izdvaja deo sa godinom iz datuma, ili broj godina iz intervala.

Varijanta sa datumom

Izdvaja deo sa godinom iz datuma ili datetime.

Sintaksa	year(date)
Argumenti	• **date** - vrednost datuma ili datetime
Primeri	• `year('2012-05-12')` → 2012

Varijanta sa intervalom

Računa dužinu intervala u godinama.

Sintaksa	year(interval)
Argumenti	• **interval** - vrednost intervala iz koje se vraća broj godina
Primeri	• `year(to_interval('3 years'))` → 3 • `year(age('2012-01-01','2010-01-01'))` → 1.9986

Nekoliko primera:

Pored ovih funkcija, oduzimanje datuma, datetime ili vremena pomoću operatora - (minus) vraća interval.

Dodavanje ili oduzimanje intervala od datuma, datetime ili vremena, pomoću operatora + (plus) i – (minus), vraća datetime.

- Preuzmi broj dana do izdanja QGIS 3.0:

```
to_date('2017-09-29') - to_date(now())
-- Returns <interval: 203 days>
```

- Isto sa vremenom:

```
to_datetime('2017-09-29 12:00:00') - now()
-- Returns <interval: 202.49 days>
```

- Preuzmi datetime za 100 dana od sada:

```
now() + to_interval('100 days')
-- Returns <datetime: 2017-06-18 01:00:00>
```

9.2.9 Polja i vrednosti

Sadrži listu polja sa aktivnog sloja i posebne vrednosti. Lista polja uključuje ona sačuvana u skupu podataka, *virtuelna* i *pomoćna*, kao i ona iz *spojeva*.

Dvaput kliknite na naziv polja da biste ga dodali u svoj izraz. Možete i ukucati naziv polja (po mogućstvu unutar dvostrukih navodnika) ili njegov *alijas*.

Da biste preuzeli vrednosti polja za korišćenje u izrazu, izaberite odgovarajuće polje i, u prikazanom vidžetu, izaberite između *10 uzoraka* i *Sve jedinstvene*. Tražene vrednosti se zatim prikazuju, a možete koristiti polje *Pretraga* na vrhu liste da biste filtrirali rezultat. Vrednostima uzorka može se pristupiti i desnim klikom na polje.

Da biste dodali vrednost u izraz koji pišete, dvaput kliknite na nju u listi. Ako je vrednost tipa niska, treba je staviti pod jednostruke navodnike, u suprotnom navodnici nisu potrebni.

NULL

Jednako je NULL vrednosti.

Sintaksa	NULL
Primeri	• `NULL` → NULL vrednost

Белешка

Za proveru NULL vrednosti koristite izraz *IS NULL* ili *IS NOT NULL*.

9.2.10 Funkcije za datoteke i putanje

Ova grupa sadrži funkcije koje rade sa nazivima datoteka i putanja.

base_file_name

Vraća osnovni naziv datoteke bez direktorijuma ili sufiksa datoteke.

Sintaksa	base_file_name(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `base_file_name('/home/qgis/data/country_boundaries.shp')` → 'country_boundaries'

exif

Preuzima vrednosti exif oznaka iz datoteke slike.

Sintaksa	exif(path, [tag]) [] označava opcione argumente
Argumenti	• **path** - Putanja datoteke slike ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja. • **tag** - Oznaka koja se vraća. Ako je prazno, vratiće se mapa sa svim mogućim vrednostima Exif oznaka.
Primeri	• `exif('/my/photo.jpg','Exif.Image.Orientation')` → 0

file_exists

Vraća TRUE ako putanja datoteke postoji.

Sintaksa	file_exists(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `file_exists('/home/qgis/data/country_boundaries.shp')` → TRUE

file_name

Vraća naziv datoteke (uključujući ekstenziju), bez direktorijuma.

Sintaksa	file_name(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `file_name('/home/qgis/data/country_boundaries.shp')` → 'country_boundaries.shp'

file_path

Vraća komponentu direktorijuma iz putanje datoteke. Ne uključuje naziv datoteke.

Sintaksa	file_path(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `file_path('/home/qgis/data/country_boundaries.shp')` → '/home/qgis/data'

file_size

Vraća veličinu datoteke (u bajtovima).

Sintaksa	file_size(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `file_size('/home/qgis/data/country_boundaries.geojson')` → 5674

file_suffix

Vraća sufiks (ekstenziju) datoteke iz putanje datoteke.

Sintaksa	file_suffix(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `file_suffix('/home/qgis/data/country_boundaries.shp')` → 'shp'

is_directory

Vraća TRUE ako putanja odgovara direktorijumu.

Sintaksa	is_directory(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `is_directory('/home/qgis/data/country_boundaries.shp')` → FALSE • `is_directory('/home/qgis/data/')` → TRUE

is_file

Vraća TRUE ako putanja odgovara datoteci.

Sintaksa	is_file(path)
Argumenti	• **path** - putanja datoteke ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `is_file('/home/qgis/data/country_boundaries.shp')` → TRUE • `is_file('/home/qgis/data/')` → FALSE

9.2.11 Funkcije za forme

Ova grupa sadrži funkcije koje rade isključivo u kontekstu atributske forme. Na primer, u podešavanjima *vidžeta polja*.

current_parent_value

Upotrebljivo samo u kontekstu ugrađene forme, ova funkcija vraća trenutnu, nesačuvanu vrednost polja u nadređenoj formi koja se trenutno uređuje. To će se razlikovati od stvarnih vrednosti atributa nadređenog objekta za objekte koji se trenutno uređuju ili još nisu dodati u nadređeni sloj. Kada se koristi u izrazu filtera vidžeta value-relation, ovu funkciju treba obmotati u 'coalesce()' koji može preuzeti stvarni nadređeni objekat sa sloja kada se forma ne koristi u ugrađenom kontekstu.

Sintaksa	current_parent_value(field_name)
Argumenti	• **field_name** - naziv polja u trenutnoj nadređenoj formi
Primeri	• `current_parent_value( 'FIELD_NAME' )` → Trenutna vrednost polja 'FIELD_NAME' u nadređenoj formi.

current_value

Vraća trenutnu, nesačuvanu vrednost polja u formi ili redu tabele koji se trenutno uređuje. To će se razlikovati od stvarnih vrednosti atributa objekta za objekte koji se trenutno uređuju ili još nisu dodati na sloj.

Sintaksa	current_value(field_name)
Argumenti	• **field_name** - naziv polja u trenutnoj formi ili redu tabele
Primeri	• `current_value( 'FIELD_NAME' )` → Trenutna vrednost polja 'FIELD_NAME'.

9.2.12 Funkcije za približno podudaranje

Ova grupa sadrži funkcije za približna poređenja vrednosti.

hamming_distance

Vraća Hamingovo rastojanje između dve niske. To je jednako broju znakova na odgovarajućim pozicijama u ulaznim niskama na kojima se znaci razlikuju. Ulazne niske moraju biti iste dužine, a poređenje razlikuje velika i mala slova.

Sintaksa	hamming_distance(string1, string2)
Argumenti	• **string1** - niska • **string2** - niska
Primeri	• `hamming_distance('abc','xec')` → 2 • `hamming_distance('abc','ABc')` → 2 • `hamming_distance(upper('abc'),upper('ABC'))` → 0 • `hamming_distance('abc','abcd')` → NULL

levenshtein

Vraća Levenštajnovo rastojanje uređivanja između dve niske. To je jednako minimalnom broju izmena znakova (umetanja, brisanja ili zamena) potrebnih da se jedna niska promeni u drugu.

Levenštajnovo rastojanje je mera sličnosti između dve niske. Manja rastojanja znače da su niske sličnije, a veća rastojanja ukazuju na različitije niske. Rastojanje razlikuje velika i mala slova.

Sintaksa	levenshtein(string1, string2)
Argumenti	• **string1** - niska • **string2** - niska
Primeri	• `levenshtein('kittens','mitten')` → 2 • `levenshtein('Kitten','kitten')` → 1 • `levenshtein(upper('Kitten'),upper('kitten'))` → 0

longest_common_substring

Vraća najdužu zajedničku podnisku između dve niske. Ta podniska je najduža niska koja je podniska obe ulazne niske. Na primer, najduža zajednička podniska niski „ABABC” i „BABCA” je „BABC”. Podniska razlikuje velika i mala slova.

Sintaksa	longest_common_substring(string1, string2)
Argumenti	• **string1** - niska • **string2** - niska
Primeri	• `longest_common_substring('ABABC','BABCA')` → 'BABC' • `longest_common_substring('abcDeF','abcdef')` → 'abc' • `longest_common_substring(upper('abcDeF'),upper('abcdex'))` → 'ABCDE'

soundex

Vraća Soundex predstavljanje niske. Soundex je algoritam za fonetsko podudaranje, pa niske sa sličnim zvučanjem treba da budu predstavljene istim Soundex kodom.

Sintaksa	soundex(string)
Argumenti	• **string** - niska
Primeri	• `soundex('robert')` → 'R163' • `soundex('rupert')` → 'R163' • `soundex('rubin')` → 'R150'

9.2.13 Opšte funkcije

Ova grupa sadrži razne opšte funkcije.

env

Preuzima promenljivu okruženja i vraća njen sadržaj kao nisku. Ako promenljiva nije pronađena, vratiće se NULL. Ovo je zgodno za ubacivanje konfiguracije specifične za sistem, kao što su slova diskova ili prefiksi putanja. Definisanje promenljivih okruženja zavisi od operativnog sistema — proverite kod svog sistem administratora ili u dokumentaciji operativnog sistema kako se ovo podešava.

Sintaksa	env(name)
Argumenti	• **name** - Naziv promenljive okruženja koja se preuzima.
Primeri	• `env( 'LANG' )` → 'en_US.UTF-8' • `env( 'MY_OWN_PREFIX_VAR' )` → 'Z:' • `env( 'I_DO_NOT_EXIST' )` → NULL

eval

Izračunava izraz prosleđen u vidu niske. Korisno za proširivanje dinamičkih parametara prosleđenih kao kontekstne promenljive ili polja.

Sintaksa	eval(expression)
Argumenti	• **expression** - niska izraza
Primeri	• `eval('\'nice\'')` → 'nice' • `eval(@expression_var)` → [šta god da je rezultat izračunavanja @expression_var…]

eval_template

Izračunava šablon prosleđen u vidu niske. Korisno za proširivanje dinamičkih parametara prosleđenih kao kontekstne promenljive ili polja.

Sintaksa	eval_template(template)
Argumenti	• **template** - niska šablona
Primeri	• `eval_template('QGIS [% upper(\'rocks\') %]')` → QGIS ROCKS

is_layer_visible

Vraća TRUE ako je zadati sloj vidljiv.

Sintaksa	is_layer_visible(layer)
Argumenti	• **layer** - niska koja predstavlja naziv sloja ili ID sloja
Primeri	• `is_layer_visible('baseraster')` → TRUE

mime_type

Vraća mime tip binarnih podataka.

Sintaksa	mime_type(bytes)
Argumenti	• **bytes** - binarni podaci
Primeri	• `mime_type('<html><body></body></html>')` → text/html • `mime_type(from_base64('R0lGODlhAQABAAAAACH5BAEKAAEALAAAAAABAAEAAAIAOw==''))` → image/gif

var

Vraća vrednost sačuvanu u zadatoj promenljivoj.

Sintaksa	var(name)
Argumenti	• **name** - naziv promenljive
Primeri	• `var('qgis_version')` → '2.12'

Više informacija: Lista podrazumevanih *promenljivih*

with_variable

Ova funkcija postavlja promenljivu za bilo koji kod izraza koji se prosleđuje kao 3. argument. Ovo je korisno samo za složene izraze, gde je istu izračunatu vrednost potrebno koristiti na različitim mestima.

Sintaksa	with_variable(name, value, expression)
Argumenti	• **name** - naziv promenljive koja se postavlja • **value** - vrednost koja se postavlja • **expression** - izraz za koji će promenljiva biti dostupna
Primeri	• `with_variable('my_sum', 1 + 2 + 3, @my_sum * 2 + @my_sum * 5)` → 42

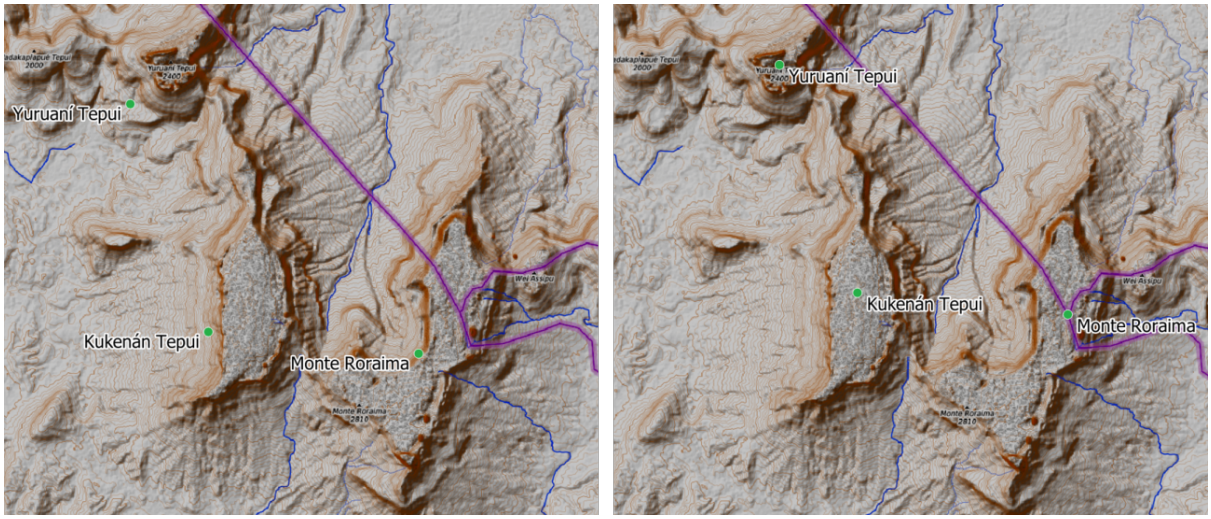
9.2.14 Funkcije za geometriju

Ova grupa sadrži funkcije koje rade sa objektima geometrije (npr. buffer, transform, $area).

affine_transform

Vraća geometriju nakon afine transformacije. Izračunavanja su u prostornom referentnom sistemu te geometrije. Operacije se izvode redosledom skaliranje, rotacija, translacija. Ako postoji Z ili M pomeraj, a koordinata nije prisutna u geometriji, biće dodata.

Sintaksa	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]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **delta_x** - translacija po x osi • **delta_y** - translacija po y osi • **rotation_z** - rotacija oko z ose u stepenima, suprotno od kazaljke na satu • **scale_x** - faktor skaliranja po x osi • **scale_y** - faktor skaliranja po y osi • **delta_z** - translacija po z osi • **delta_m** - translacija po m osi • **scale_z** - faktor skaliranja po z osi • **scale_m** - faktor skaliranja po m osi
Primeri	• `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)'



Слика9.4: Vektorski tačkasti sloj (zelene tačke) pre (levo) i posle (desno) afine transformacije (translacije).

angle_at_vertex

Vraća ugao simetrale (prosečan ugao) prema geometriji za zadato teme na linijskoj geometriji. Uglovi su u stepenima u smeru kazaljke na satu od severa.

Sintaksa	angle_at_vertex(geometry, vertex)
Argumenti	• **geometry** - geometrija tipa linestring • **vertex** - indeks temena, počev od 0; ako je vrednost negativna, izabrani indeks temena biće njihov ukupan broj umanjen za apsolutnu vrednost
Primeri	• `angle_at_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 45.0

apply_dash_pattern

Primenjuje šaru crtica na geometriju i vraća MultiLineString geometriju koja je ulazna geometrija iscrtana duž svake linije/prstena zadatom šarom.

Sintaksa	apply_dash_pattern(geometry, pattern, [start_rule:='no_rule'], [end_rule:='no_rule'], [adjustment:='both'], [pattern_offset:=0]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija (prihvata (multi)linestring ili (multi)poligone). • **pattern** - šara crtica, kao niz brojeva koji predstavljaju dužine crtica i razmaka. Mora sadržati paran broj elemenata. • **start_rule** - opciono pravilo za ograničavanje početka šare. Važeće vrednosti su 'no_rule', 'full_dash', 'half_dash', 'full_gap', 'half_gap'. • **end_rule** - opciono pravilo za ograničavanje kraja šare. Važeće vrednosti su 'no_rule', 'full_dash', 'half_dash', 'full_gap', 'half_gap'. • **adjustment** - opciono pravilo koje zadaje koji deo šare se prilagođava da bi odgovarao željenim pravilima šare. Važeće vrednosti su 'both', 'dash', 'gap'. • **pattern_offset** - Opciono rastojanje koje zadaje određeno rastojanje duž šare od kojeg se počinje.
Primeri	• `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

Vraća površinu trenutnog objekta. Površina koju računa ova funkcija poštuje i podešavanje elipsoida projekta i podešavanje jedinice površine. Na primer, ako je za projekat postavljen elipsoid, izračunata površina biće elipsoidna, a ako elipsoid nije postavljen, izračunata površina biće planimetrijska.

Sintaksa	$area
Primeri	• `$area` → 42

area

Vraća površinu poligonskog objekta geometrije. Izračunavanja su uvek planimetrijska u prostornom referentnom sistemu (SRS) te geometrije, a jedinice vraćene površine odgovaraće jedinicama tog SRS-a. Ovo se razlikuje od izračunavanja koja vrši funkcija $area, koja izvodi elipsoidna izračunavanja na osnovu podešavanja elipsoida i jedinice površine projekta.

Sintaksa	area(geometry)
Argumenti	• **geometry** - objekat poligonske geometrije
Primeri	• `area(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 8.0

azimuth

Vraća azimut u odnosu na sever kao ugao u radijanima meren u smeru kazaljke na satu od vertikale u point_a do point_b.

Sintaksa	azimuth(point_a, point_b)
Argumenti	• **point_a** - tačkasta geometrija • **point_b** - tačkasta geometrija
Primeri	• `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

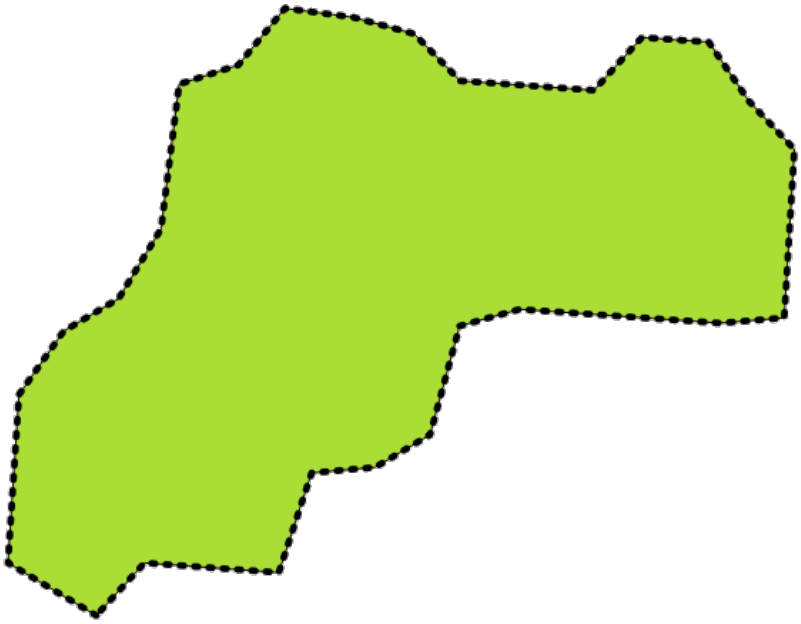
Vraća azimut u odnosu na sever kao ugao u radijanima meren na elipsoidu u smeru kazaljke na satu od vertikale u point_a do point_b.

Sintaksa	bearing(point_a, point_b, [source_crs], [ellipsoid]) [] označava opcione argumente
Argumenti	• **point_a** - tačkasta geometrija • **point_b** - tačkasta geometrija • **source_crs** - opciona niska ili objekat CRS-a koji predstavlja izvorni CRS tačaka. Podrazumevano se koristi CRS trenutnog sloja. • **ellipsoid** - opciona niska koja predstavlja akronim ili ID autoriteta (npr. 'EPSG:7030') elipsoida na kojem treba meriti azimut. Podrazumevano se koristi podešavanje elipsoida trenutnog projekta.
Primeri	• `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

Vraća zatvaranje kombinatorne granice geometrije (tj. topološku granicu geometrije). Na primer, poligonska geometrija imaće granicu koja se sastoji od linija za svaki prsten poligona. Neki tipovi geometrije nemaju definisanu granicu, npr. tačke ili kolekcije geometrija, i vratiće NULL.

Sintaksa	boundary(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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))'



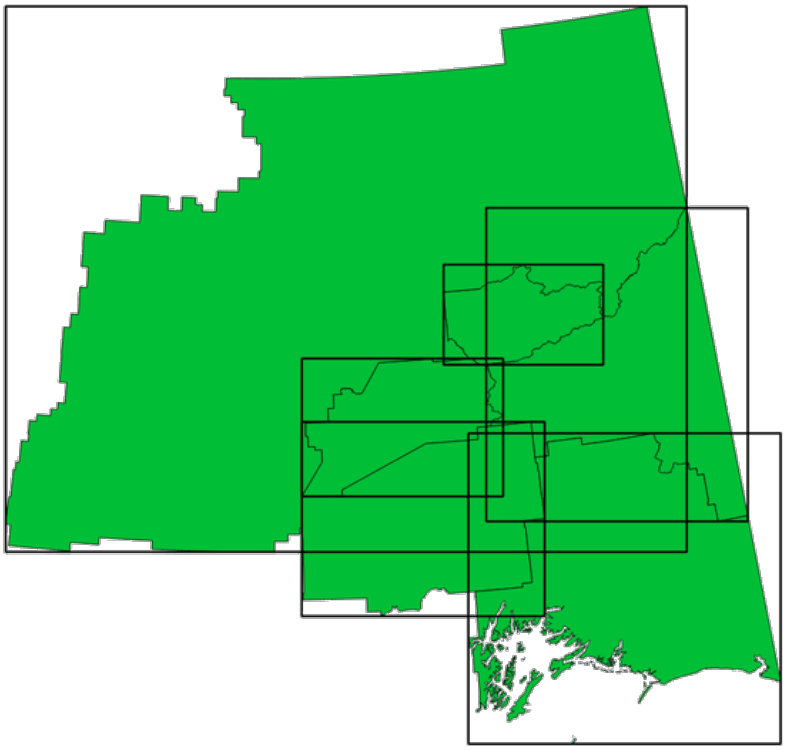
Слика9.5: Granica (crna isprekidana linija) izvornog poligonskog sloja

Više informacija: algoritam *Granica*

bounds

Vraća geometriju koja predstavlja granični pravougaonik ulazne geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	bounds(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `bounds(@geometry)` → granični pravougaonik geometrije trenutnog objekta • `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))'



Слика9.6: Crne linije predstavljaju granične pravougaonike svakog poligonskog objekta

Više informacija: algoritam *Granični pravougaonici*

bounds_height

Vraća visinu graničnog pravougaonika geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	bounds_height(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `bounds_height(@geometry)` → visina graničnog pravougaonika geometrije trenutnog objekta • `bounds_height(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 1

bounds_width

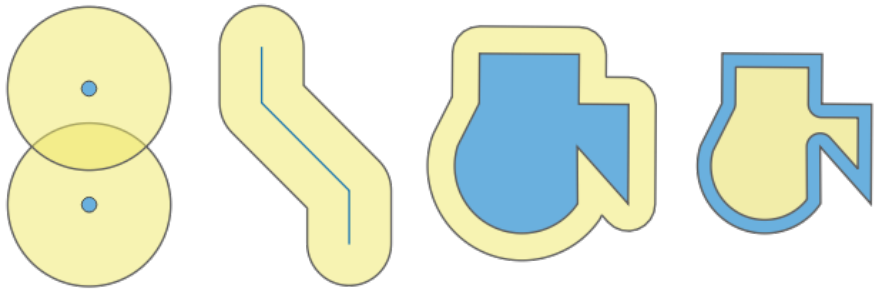
Vraća širinu graničnog pravougaonika geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	bounds_width(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `bounds_width(@geometry)` → širina graničnog pravougaonika geometrije trenutnog objekta • `bounds_width(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 2

buffer

Vraća geometriju koja predstavlja sve tačke čije je rastojanje od ove geometrije manje ili jednako zadatom rastojanju. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	buffer(geometry, distance, [segments:=8], [cap:='round'], [join:='round'], [miter_limit:=2]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **distance** - rastojanje bafera u jedinicama sloja • **segments** - broj segmenata koji se koristi za predstavljanje četvrtine kruga kada se koristi zaobljeni stil spoja. Veći broj daje glatkiji bafer sa više čvorova. • **cap** - stil završetka bafera. Važeće vrednosti su 'round', 'flat' ili 'square' • **join** - stil spoja za bafer. Važeće vrednosti su 'round', 'bevel' ili 'miter'. • **miter_limit** - ograničenje rastojanja oštrog spoja, za korišćenje kada je stil spoja postavljen na 'miter'
Primeri	• `buffer(@geometry, 10.5)` → poligon geometrije trenutnog objekta baferovan za 10.5 jedinica



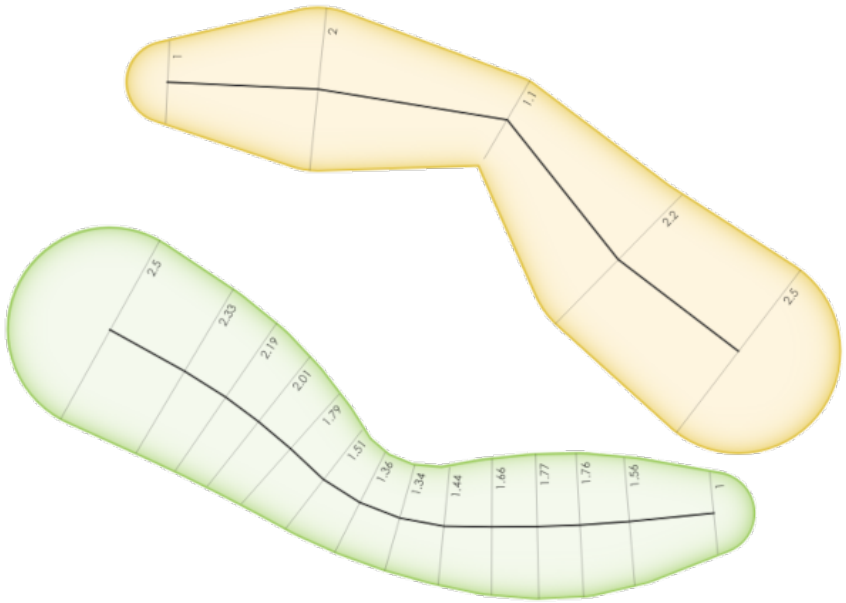
Слика9.7: Bafer (žuto) tačaka, linije, poligona sa pozitivnim baferom i poligona sa negativnim baferom

Više informacija: algoritam *Bafer*

buffer_by_m

Pravi bafer duž linijske geometrije, pri čemu prečnik bafera varira prema m vrednostima na temenima linije.

Sintaksa	buffer_by_m(geometry, [segments:=8]) [] označava opcione argumente
Argumenti	• **geometry** - ulazna geometrija. Mora biti (multi)linijska geometrija sa m vrednostima. • **segments** - number of segments to approximate quarter-circle curves in the buffer.
Primeri	• `buffer_by_m(geometry:=geom_from_wkt('LINESTRINGM(1 2 0.5, 4 2 0.2)'),segments:=8)` → Bafer promenljive širine koji počinje prečnikom 0.5 i završava se prečnikom 0.2 duž linijske geometrije.



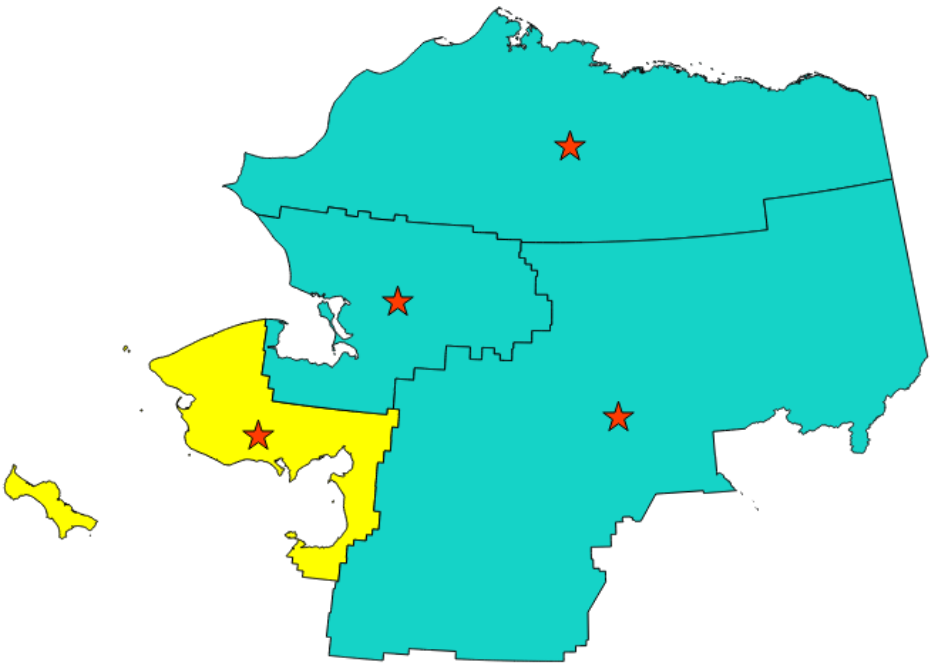
Слика9.8: Baferovanje linijskih objekata pomoću m vrednosti na temenima

Više informacija: algoritam *Bafer promenljive širine (po M vrednosti)*

centroid

Vraća geometrijski centar geometrije.

Sintaksa	centroid(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `centroid(@geometry)` → tačkasta geometrija



Слика9.9: Crvene zvezde predstavljaju centroide objekata ulaznog sloja.

Više informacija: algoritam *Centroidi*

close_line

Vraća zatvorenu liniju od ulazne linije dodavanjem prve tačke na kraj linije, ako već nije zatvorena. Ako geometrija nije linija ili višelinijska geometrija, rezultat će biti NULL.

Sintaksa	close_line(geometry)
Argumenti	• **geometry** - geometrija tipa line string
Primeri	• `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

Vraća tačku na geometry1 koja je najbliža geometry2.

Sintaksa	closest_point(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija na kojoj se traži najbliža tačka • **geometry2** - geometrija do koje se traži najbliža tačka
Primeri	• `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

Sakuplja skup geometrija u objekat višedelne geometrije.

Varijanta sa listom argumenata

Delovi geometrije zadaju se kao zasebni argumenti funkcije.

Sintaksa	collect_geometries(geometry1, geometry2, …)
Argumenti	• **geometry** - geometrija
Primeri	• `geom_to_wkt(collect_geometries(make_point(1,2), make_point(3,4), make_point(5,6)))` → 'MultiPoint ((1 2),(3 4),(5 6))'

Varijanta sa nizom

Delovi geometrije zadaju se kao niz delova geometrije.

Sintaksa	collect_geometries(array)
Argumenti	• **array** - niz objekata geometrije
Primeri	• `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))'

Više informacija: algoritam *Prikupi geometrije*

combine

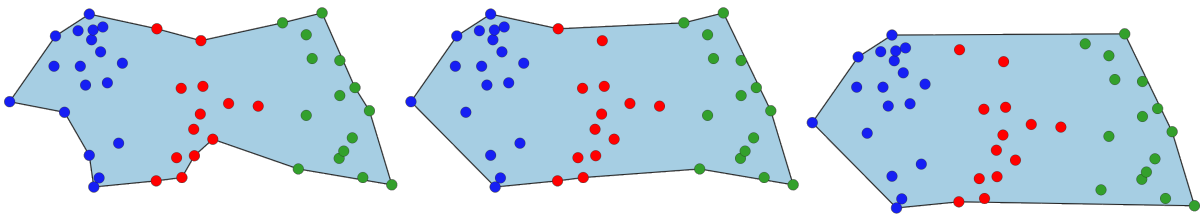
Vraća kombinaciju dve geometrije.

Sintaksa	combine(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Vraća moguće konkavan poligon koji sadrži sve tačke u geometriji

Sintaksa	concave_hull(geometry, target_percent, [allow_holes:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **target_percent** - procenat površine konveksnog omotača kojem rešenje teži. target_percent od 1 daje isti rezultat kao konveksni omotač. target_percent između 0 i 0.99 daje rezultat čija bi površina trebalo da bude manja od površine konveksnog omotača. • **allow_holes** - opcioni argument koji zadaje da li su dozvoljene rupe unutar izlazne geometrije. Podrazumevano FALSE; postavite na TRUE da biste dozvolili rupe u izlaznoj geometriji.
Primeri	• `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))'



Слика9.10: Konkavni omotači sa rastućim parametrom target_percent

Više informacija: *convex_hull*, algoritam *Konkavni omotač (po sloju)*

contains

Proverava da li jedna geometrija sadrži drugu. Vraća TRUE ako i samo ako nijedna tačka geometrije geometry2 ne leži izvan geometrije geometry1, a bar jedna tačka unutrašnjosti geometrije geometry2 leži u unutrašnjosti geometrije geometry1.

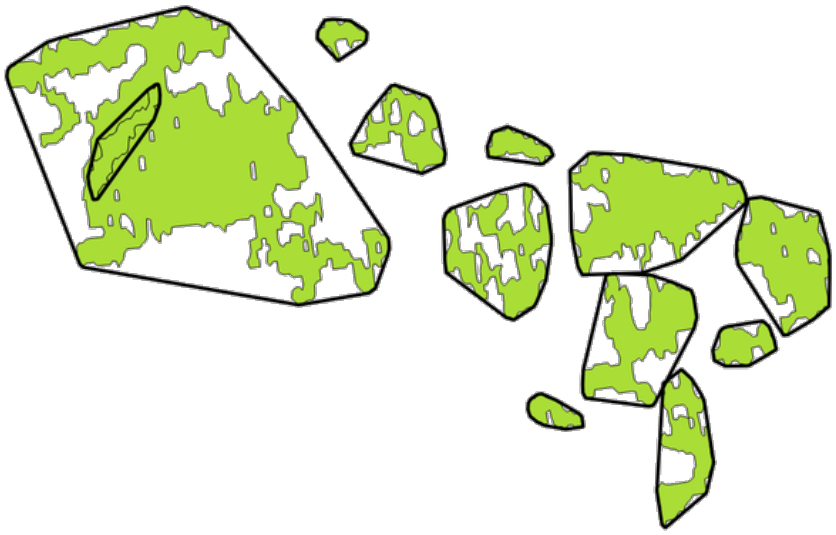
Sintaksa	contains(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `contains( geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))' ), geom_from_wkt( 'POINT(0.5 0.5 )' ) )` → TRUE • `contains( geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → FALSE

Više informacija: *overlay_contains*

convex_hull

Vraća konveksni omotač geometrije. On predstavlja minimalnu konveksnu geometriju koja obuhvata sve geometrije u skupu.

Sintaksa	convex_hull(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `geom_to_wkt( convex_hull( geom_from_wkt( 'LINESTRING(3 3, 4 4, 4 10)' ) ) )` → 'POLYGON((3 3, 4 10, 4 4, 3 3))'



Слика9.11: Crne linije označavaju konveksni omotač svakog objekta

Više informacija: *concave_hull*, algoritam *Konveksni omotač*

crosses

Proverava da li jedna geometrija prelazi preko druge. Vraća TRUE ako zadate geometrije imaju neke, ali ne sve, zajedničke unutrašnje tačke.

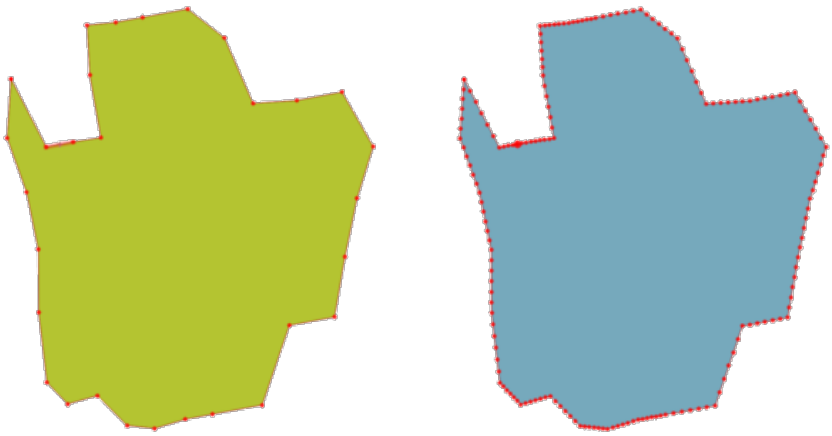
Sintaksa	crosses(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `crosses( geom_from_wkt( 'LINESTRING(3 5, 4 4, 5 3)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `crosses( geom_from_wkt( 'POINT(4 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → FALSE

Više informacija: *overlay_crosses*

densify_by_count

Uzima geometriju poligonskog ili linijskog sloja i pravi novu u kojoj geometrije imaju veći broj temena od originalne.

Sintaksa	densify_by_count(geometry, vertices)
Argumenti	• **geometry** - geometrija (prihvata (multi)linestring ili (multi)poligone). • **vertices** - broj temena koji se dodaje (po segmentu)
Primeri	• `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)



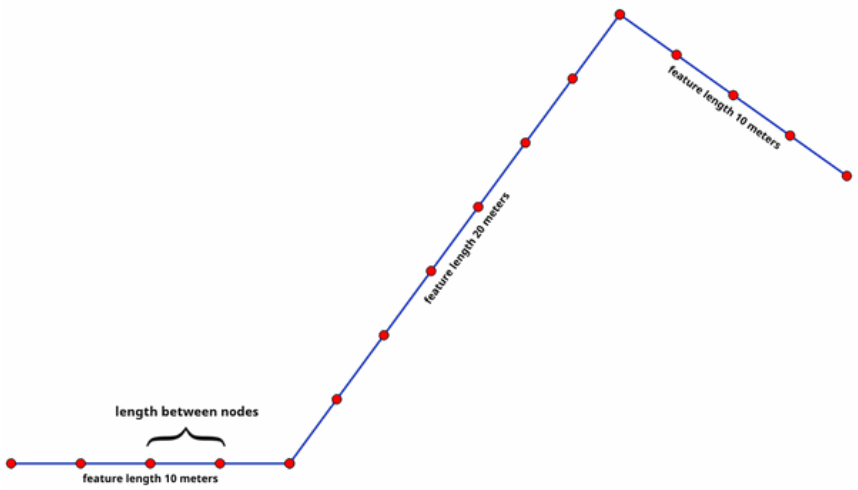
Слика9.12: Crvene tačke prikazuju temena pre i posle zgušnjavanja

Više informacija: algoritam *Zgusni po broju*

densify_by_distance

Uzima geometriju poligonskog ili linijskog sloja i pravi novu u kojoj su geometrije zgusnute dodavanjem dodatnih temena na ivicama koje imaju maksimalno rastojanje jednako zadatom intervalu.

Sintaksa	densify_by_distance(geometry, distance)
Argumenti	• **geometry** - geometrija (prihvata (multi)linestring ili (multi)poligone). • **distance** - maksimalno rastojanje intervala između temena u izlaznoj geometriji
Primeri	• `geom_to_wkt(densify_by_distance(geom_from_wkt('LINESTRING(1 1, 10 1)'), 4))` → LineString (1 1, 4 1, 7 1, 10 1)



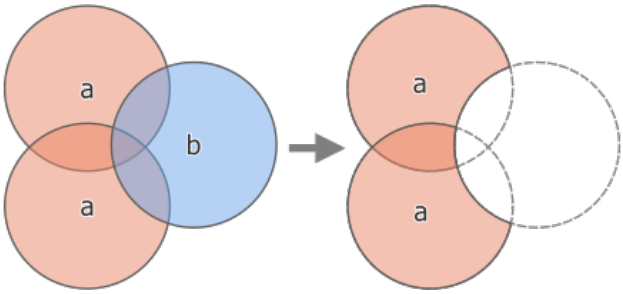
Слика9.13: Zgušnjavanje geometrije na zadatom intervalu

Više informacija: algoritam *Zgusni po intervalu*

difference

Vraća geometriju koja predstavlja onaj deo geometrije geometry1 koji se ne preseca sa geometry2.

Sintaksa	difference(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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)'



Слика9.14: Operacija razlike između ulaznog sloja 'a' sa dva objekta i preklapajućeg sloja 'b' sa jednim objektom (levo) — daje novi sloj sa izmenjenim objektima 'a' (desno)

Više informacija: algoritam *Razlika*

disjoint

Proverava da li se geometrije prostorno ne presecaju. Vraća TRUE ako geometrije ne dele nikakav zajednički prostor.

Sintaksa	disjoint(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Više informacija: *overlay_disjoint*

distance

Vraća minimalno rastojanje (na osnovu prostornog referentnog sistema) između dve geometrije u projektovanim jedinicama.

Sintaksa	distance(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `distance( geom_from_wkt( 'POINT(4 4)' ), geom_from_wkt( 'POINT(4 8)' ) )` → 4

distance_to_vertex

Vraća rastojanje duž geometrije do zadatog temena.

Sintaksa	distance_to_vertex(geometry, vertex)
Argumenti	• **geometry** - geometrija tipa linestring • **vertex** - indeks temena, počev od 0; ako je vrednost negativna, izabrani indeks temena biće njihov ukupan broj umanjen za apsolutnu vrednost
Primeri	• `distance_to_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 10.0

end_point

Vraća poslednji čvor iz geometrije.

Sintaksa	end_point(geometry)
Argumenti	• **geometry** - objekat geometrije
Primeri	• `geom_to_wkt(end_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → 'Point (0 2)'



Слика9.15: Krajnja tačka linijskog objekta

Više informacija: *start_point*, algoritam *Izdvoj određena temena*

exif_geotag

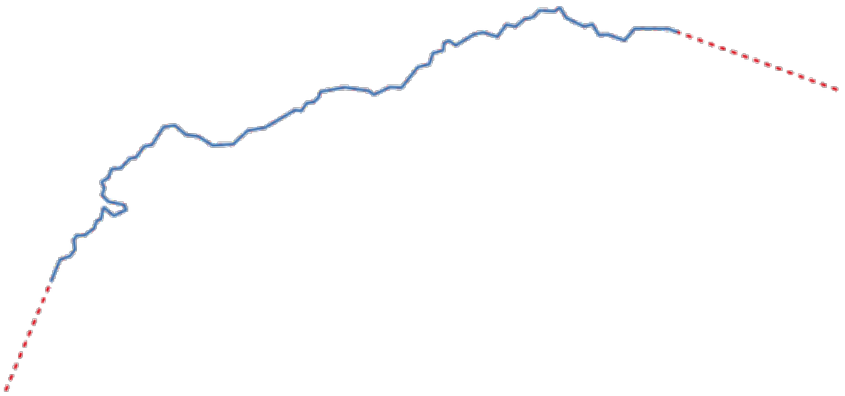
Pravi tačkastu geometriju od exif geooznaka datoteke slike.

Sintaksa	exif_geotag(path)
Argumenti	• **path** - Putanja datoteke slike ili vrednost sloja karte. Ako je zadata vrednost sloja karte, koristiće se izvor datoteke tog sloja.
Primeri	• `geom_to_wkt(exif_geotag('/my/photo.jpg'))` → 'Point (2 4)'

extend

Produžava početak i kraj linijske geometrije za zadati iznos. Linije se produžavaju koristeći pravac prvog i poslednjeg segmenta u liniji. Za multilinestring, produžavaju se svi delovi. Rastojanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	extend(geometry, start_distance, end_distance)
Argumenti	• **geometry** - (multi)linestring geometrija • **start_distance** - rastojanje za produžavanje početka linije • **end_distance** - rastojanje za produžavanje kraja linije.
Primeri	• `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))'



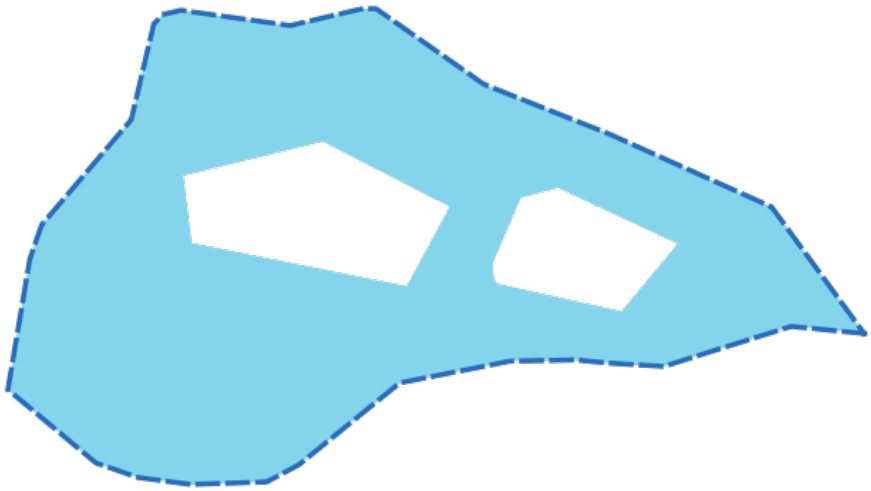
Слика9.16: Crvene crte predstavljaju početno i krajnje proširenje originalnog sloja

Više informacija: algoritam *Produži linije*

exterior_ring

Vraća liniju koja predstavlja spoljni prsten poligonske geometrije. Ako geometrija nije poligon, rezultat će biti NULL.

Sintaksa	exterior_ring(geometry)
Argumenti	• **geometry** - poligonska geometrija
Primeri	• `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)'



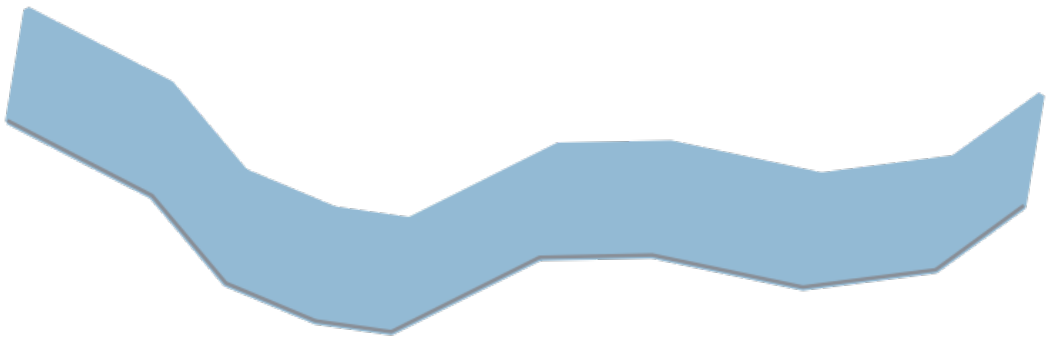
Слика9.17: Isprekidana linija predstavlja spoljni prsten poligona

Više informacija: algoritam *Granica*, *interior_ring_n*

extrude

Vraća izvučenu verziju ulazne (Multi-)Curve ili (Multi-)Linestring geometrije, sa proširenjem zadatim pomoću x i y.

Sintaksa	extrude(geometry, x, y)
Argumenti	• **geometry** - geometrija tipa kriva ili linestring • **x** - x proširenje, numerička vrednost • **y** - y proširenje, numerička vrednost
Primeri	• `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)))'



Слика9.18: Generisanje poligona izvlačenjem linije sa pomerajem po x i y pravcu

flip_coordinates

Vraća kopiju geometrije sa zamenjenim x i y koordinatama. Korisno za popravku geometrija kojima su vrednosti geografske širine i dužine zamenjene.

Sintaksa	flip_coordinates(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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)'

Više informacija: algoritam *Zameni X i Y koordinate*

force_polygon_ccw

Prisiljava geometriju da poštuje konvenciju po kojoj su spoljni prstenovi suprotno od kazaljke na satu, a unutrašnji u smeru kazaljke na satu.

Sintaksa	force_polygon_ccw(geometry)
Argumenti	• **geometry** - geometrija. Sve nepoligonske geometrije vraćaju se nepromenjene.
Primeri	• `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))'

Više informacija: *force_polygon_cw*, *force_rhr*

force_polygon_cw

Prisiljava geometriju da poštuje konvenciju po kojoj su spoljni prstenovi u smeru kazaljke na satu, a unutrašnji suprotno od kazaljke na satu.

Sintaksa	force_polygon_cw(geometry)
Argumenti	• **geometry** - geometrija. Sve nepoligonske geometrije vraćaju se nepromenjene.
Primeri	• `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))'

Više informacija: *force_polygon_ccw*, *force_rhr*

force_rhr

Prisiljava geometriju da poštuje pravilo desne ruke, po kojem se površina omeđena poligonom nalazi desno od granice. Konkretno, spoljni prsten je orijentisan u smeru kazaljke na satu, a unutrašnji prstenovi suprotno od kazaljke na satu. Zbog nedoslednosti u definiciji pravila desne ruke u nekim kontekstima, preporučuje se korišćenje eksplicitne funkcije force_polygon_cw.

Sintaksa	force_rhr(geometry)
Argumenti	• **geometry** - geometrija. Sve nepoligonske geometrije vraćaju se nepromenjene.
Primeri	• `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))'

Više informacija: algoritam *Nametni pravilo desne ruke*, *force_polygon_ccw*, *force_polygon_cw*

geom_from_gml

Vraća geometriju od GML predstavljanja geometrije.

Sintaksa	geom_from_gml(gml)
Argumenti	• **gml** - GML predstavljanje geometrije kao niska
Primeri	• `geom_from_gml('<gml:LineString srsName="EPSG:4326"><gml:coordinates>4, 4 5,5 6,6</gml:coordinates></gml:LineString>')` → objekat linijske geometrije

geom_from_wkb

Vraća geometriju napravljenu od Well-Known Binary (WKB) predstavljanja.

Sintaksa	geom_from_wkb(binary)
Argumenti	• **binary** - Well-Known Binary (WKB) predstavljanje geometrije (kao binarni blob)
Primeri	• `geom_from_wkb( geom_to_wkb( make_point(4,5) ) )` → objekat tačkaste geometrije

geom_from_wkt

Vraća geometriju napravljenu od Well-Known Text (WKT) predstavljanja.

Sintaksa	geom_from_wkt(text)
Argumenti	• **text** - Well-Known Text (WKT) predstavljanje geometrije
Primeri	• `geom_from_wkt( 'POINT(4 5)' )` → a geometry object

geom_to_wkb

Vraća Well-Known Binary (WKB) predstavljanje geometrije

Sintaksa	geom_to_wkb(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `geom_to_wkb( @geometry )` → binarni blob koji sadrži objekat geometrije

geom_to_wkt

Vraća Well-Known Text (WKT) predstavljanje geometrije bez SRID metapodataka.

Sintaksa	geom_to_wkt(geometry, [precision:=8]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **precision** - numerička preciznost
Primeri	• `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

Vraća geometriju trenutnog objekta. Može se koristiti za obradu drugim funkcijama. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske promenljive @geometry.**

Sintaksa	$geometry
Primeri	• `geom_to_wkt( $geometry )` → 'POINT(6 50)'

geometry

Vraća geometriju objekta.

Sintaksa	geometry(feature)
Argumenti	• **feature** - feature objekat
Primeri	• `geometry( @feature )` → geometrija trenutnog objekta. Bolje je koristiti @geometry. • `geom_to_wkt( geometry( get_feature_by_id( 'streets', 1 ) ) )` → geometrija u WKT-u objekta sa id-jem 1 na sloju „streets", npr. 'POINT(6 50)' • `intersects( @geometry, geometry( get_feature( 'streets', 'name', 'Main St.' ) ) )` → TRUE ako trenutni objekat prostorno preseca objekat pod nazivom 'Main St.' na sloju „streets"

geometry_n

Vraća određenu geometriju iz kolekcije geometrija, ili NULL ako ulazna geometrija nije kolekcija. Takođe vraća deo iz višedelne geometrije.

Sintaksa	geometry_n(geometry, index)
Argumenti	• **geometry** - kolekcija geometrija • **index** - indeks geometrije koja se vraća, gde je 1 prva geometrija u kolekciji
Primeri	• `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

Vraća nisku koja opisuje tip geometrije (Point, Line ili Polygon)

Sintaksa	geometry_type(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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

Vraća Hauzdorfovo rastojanje između dve geometrije. To je u osnovi mera koliko su dve geometrije slične ili različite, pri čemu manje rastojanje ukazuje na sličnije geometrije.

Funkcija se može izvršiti sa opcionim argumentom za udeo zgušnjavanja. Ako nije zadat, koristi se aproksimacija standardnog Hauzdorfovog rastojanja. Ta aproksimacija je tačna ili dovoljno bliska za veliki skup korisnih slučajeva. Primeri su:

- računanje rastojanja između linija koje su približno paralelne i približno iste dužine. To se javlja pri uparivanju linijskih mreža.
- Testiranje sličnosti geometrija.

Ako podrazumevana aproksimacija koju daje ovaj metod nije dovoljna, zadajte opcioni argument za udeo zgušnjavanja. Zadavanje ovog argumenta izvodi zgušnjavanje segmenata pre računanja diskretnog Hauzdorfovog rastojanja. Parametar zadaje udeo za koji se zgušnjava svaki segment. Svaki segment biće podeljen na određeni broj podsegmenata jednake dužine, čiji je udeo u ukupnoj dužini najbliži zadatom udelu. Smanjivanje parametra udela zgušnjavanja približava vraćeno rastojanje stvarnom Hauzdorfovom rastojanju geometrija.

Sintaksa	hausdorff_distance(geometry1, geometry2, [densify_fraction:=1]) [] označava opcione argumente
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija • **densify_fraction** - iznos udela zgušnjavanja
Primeri	• `hausdorff_distance( geometry1:= geom_from_wkt('LINESTRING (0 0, 2 1)'),geometry2:=geom_from_wkt('LINESTRING (0 0, 2 0)'))` → 1 • `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

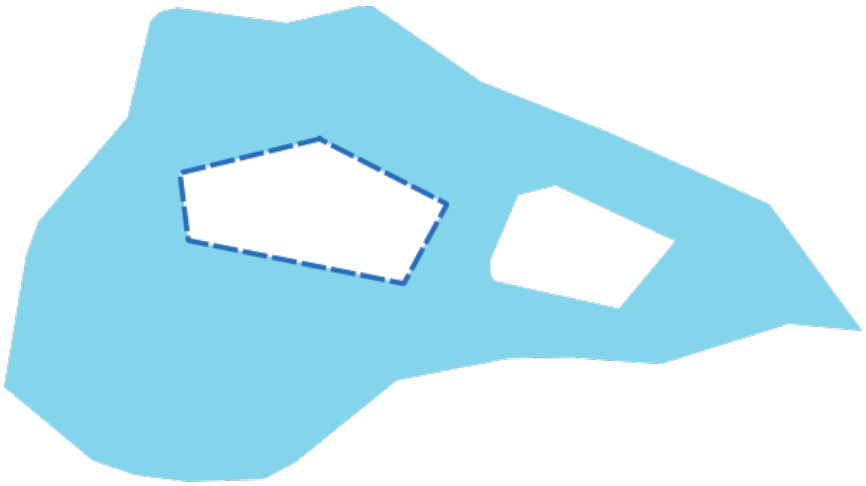
Vraća nagib meren od zenita (0) do nadira (180) na pravcu od point_a do point_b.

Sintaksa	inclination(point_a, point_b)
Argumenti	• **point_a** - tačkasta geometrija • **point_b** - tačkasta geometrija
Primeri	• `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

Vraća određeni unutrašnji prsten iz poligonske geometrije, ili NULL ako geometrija nije poligon.

Sintaksa	interior_ring_n(geometry, index)
Argumenti	• **geometry** - poligonska geometrija • **index** - indeks unutrašnjeg prstena koji se vraća, gde je 1 prvi unutrašnji prsten
Primeri	• `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)'



Слика9.19: Isprekidana linija predstavlja određeni unutrašnji prsten poligona

Više informacija: *exterior_ring*

intersection

Vraća geometriju koja predstavlja zajednički deo dve geometrije.

Sintaksa	intersection(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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)'

Više informacija: algoritam *Presek*

intersects

Proverava da li jedna geometrija preseca drugu. Vraća TRUE ako se geometrije prostorno presecaju (dele bilo koji deo prostora), a false ako ne.

Sintaksa	intersects(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Više informacija: *overlay_intersects*

intersects_bbox

Proverava da li granični pravougaonik jedne geometrije preklapa granični pravougaonik druge geometrije. Vraća TRUE ako geometrije prostorno presecaju definisani granični pravougaonik, a false ako ne.

Sintaksa	intersects_bbox(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Vraća TRUE ako je linija zatvorena (početna i krajnja tačka se poklapaju), ili false ako linija nije zatvorena. Ako geometrija nije linija, rezultat će biti NULL.

Sintaksa	is_closed(geometry)
Argumenti	• **geometry** - geometrija tipa line string
Primeri	• `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

Vraća TRUE ako je geometrija prazna (bez koordinata), false ako geometrija nije prazna i NULL ako geometrije nema. Pogledajte i is_empty_or_null.

Sintaksa	is_empty(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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

Više informacija: *is_empty_or_null*

is_empty_or_null

Vraća TRUE ako je geometrija NULL ili prazna (bez koordinata), u suprotnom false. Ova funkcija je kao izraz '@geometry IS NULL or is_empty(@geometry)'

Sintaksa	is_empty_or_null(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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

Više informacija: *is_empty*, *NULL*

is_multipart

Vraća TRUE ako je geometrija tipa Multi.

Sintaksa	is_multipart(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `is_multipart(geom_from_wkt('MULTIPOINT ((0 0),(1 1),(2 2))'))` → TRUE • `is_multipart(geom_from_wkt('POINT (0 0)'))` → FALSE

is_valid

Vraća TRUE ako je geometrija ispravna; ako je dobro oblikovana u 2D prema OGC pravilima.

Sintaksa	is_valid(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `is_valid(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2, 0 0)'))` → TRUE • `is_valid(geom_from_wkt('LINESTRING(0 0)'))` → FALSE

Više informacija: *make_valid*, algoritam *Proveri validnost*

$length

Vraća dužinu linije. Ako vam je potrebna dužina granice poligona, koristite $perimeter. Dužina koju računa ova funkcija poštuje i podešavanje elipsoida projekta i podešavanje jedinice rastojanja. Na primer, ako je za projekat postavljen elipsoid, izračunata dužina biće elipsoidna, a ako elipsoid nije postavljen, izračunata dužina biće planimetrijska.

Sintaksa	$length
Primeri	• `$length` → 42.4711

length

Vraća broj znakova u niski ili dužinu linijske geometrije.

Varijanta sa niskom

Vraća broj znakova u niski.

Sintaksa	length(string)
Argumenti	• **string** - niska čija se dužina broji
Primeri	• `length('hello')` → 5

Varijanta sa geometrijom

Računa dužinu objekta linijske geometrije. Izračunavanja su uvek planimetrijska u prostornom referentnom sistemu (SRS) te geometrije, a jedinice vraćene dužine odgovaraće jedinicama tog SRS-a. Ovo se razlikuje od izračunavanja koja vrši funkcija $length, koja izvodi elipsoidna izračunavanja na osnovu podešavanja elipsoida i jedinice rastojanja projekta.

Sintaksa	length(geometry)
Argumenti	• **geometry** - objekat linijske geometrije
Primeri	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

Više informacija: *straight_distance_2d*

length3D

Računa 3D dužinu objekta linijske geometrije. Ako geometrija nije 3D linijski objekat, vraća njenu 2D dužinu. Izračunavanja su uvek planimetrijska u prostornom referentnom sistemu (SRS) te geometrije, a jedinice vraćene dužine odgovaraće jedinicama tog SRS-a. Ovo se razlikuje od izračunavanja koja vrši funkcija $length, koja izvodi elipsoidna izračunavanja na osnovu podešavanja elipsoida i jedinice rastojanja projekta.

Sintaksa	length3D(geometry)
Argumenti	• **geometry** - objekat linijske geometrije
Primeri	• `length3D(geom_from_wkt('LINESTRINGZ(0 0 0, 3 0 4)'))` → 5.0

line_interpolate_angle

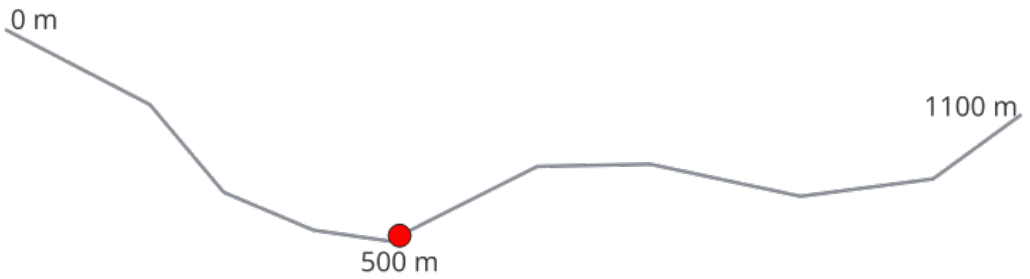
Vraća ugao paralelan geometriji na zadatom rastojanju duž linijske geometrije. Uglovi su u stepenima u smeru kazaljke na satu od severa.

Sintaksa	line_interpolate_angle(geometry, distance)
Argumenti	• **geometry** - geometrija tipa linestring • **distance** - rastojanje duž linije na kojem se interpolira ugao
Primeri	• `line_interpolate_angle(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),distance:=5)` → 90.0

line_interpolate_point

Vraća tačku interpoliranu na zadatom rastojanju duž linijske geometrije.

Sintaksa	line_interpolate_point(geometry, distance)
Argumenti	• **geometry** - geometrija tipa linestring • **distance** - rastojanje duž linije na kojem se interpolira
Primeri	• `geom_to_wkt(line_interpolate_point(geometry:=geom_from_wkt('LineString(0 0, 8 0)'), distance:=5))` → 'Point (5 0)' • `geom_to_wkt(line_interpolate_point(geometry:=geom_from_wkt('LineString(0 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 0, 1 0)'), distance:=2))` → NULL



Слика9.20: Interpolirana tačka na 500 m od početka linije

Više informacija: algoritam *Interpoliraj tačku na liniji*

line_interpolate_point_by_m

Vraća tačku interpoliranu prema podudarnoj M vrednosti duž linijske geometrije.

Sintaksa	line_interpolate_point_by_m(geometry, m, [use_3d_distance:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija tipa linestring • **m** - M vrednost • **use_3d_distance** - određuje da li se pri interpolaciji koriste 2D ili 3D rastojanja između temena (ova opcija se uzima u obzir samo za linije sa z vrednostima)
Primeri	• `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

Vraća rastojanje duž linije koje odgovara prvoj podudarnoj interpoliranoj M vrednosti.

Sintaksa	line_locate_m(geometry, m, [use_3d_distance:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija tipa linestring • **m** - M vrednost • **use_3d_distance** - određuje da li se pri interpolaciji koriste 2D ili 3D rastojanja između temena (ova opcija se uzima u obzir samo za linije sa z vrednostima)
Primeri	• `line_locate_m(geometry:=geom_from_wkt('LineStringM(0 0 0, 10 10 10)'),m:=5)` → 7.07106

line_locate_point

Vraća rastojanje duž linije koje odgovara najbližem položaju linije zadatoj tačkastoj geometriji.

Sintaksa	line_locate_point(geometry, point)
Argumenti	• **geometry** - geometrija tipa linestring • **point** - tačkasta geometrija do koje se traži najbliži položaj na liniji
Primeri	• `line_locate_point(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),point:=geom_from_wkt('Point(5 0)'))` → 5.0

line_merge

Vraća LineString ili MultiLineString geometriju u kojoj su sve povezane linije iz ulazne geometrije spojene u jednu liniju. Ova funkcija vraća NULL ako joj se prosledi geometrija koja nije LineString/MultiLineString.

Sintaksa	line_merge(geometry)
Argumenti	• **geometry** - geometrija tipa LineString/MultiLineString
Primeri	• `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

Vraća deo linijske (ili krive) geometrije koji se nalazi između zadatog početnog i krajnjeg rastojanja (mereno od početka linije). Z i M vrednosti se linearno interpoliraju iz postojećih vrednosti.

Sintaksa	line_substring(geometry, start_distance, end_distance)
Argumenti	• **geometry** - geometrija tipa linestring ili kriva • **start_distance** - rastojanje do početka podniske • **end_distance** - rastojanje do kraja podniske
Primeri	• `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)'



Слика9.21: Podniska linije sa početnim rastojanjem postavljenim na 0 metara i krajnjim rastojanjem na 250 metara.

Više informacija: algoritam *Podniska linije*

m

Vraća m (measure) vrednost tačkaste geometrije.

Sintaksa	m(geometry)
Argumenti	• **geometry** - tačkasta geometrija
Primeri	• `m( geom_from_wkt( 'POINTM(2 5 4)' ) )` → 4

m_at

Preuzima m koordinatu geometrije, ili NULL ako geometrija nema m vrednost.

Sintaksa	m_at(geometry, vertex)
Argumenti	• **geometry** - objekat geometrije • **vertex** - indeks temena geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `m_at(geom_from_wkt('LineStringZM(0 0 0 0, 10 10 0 5, 10 10 0 0)'), 1)` → 5

m_max

Vraća maksimalnu m (measure) vrednost geometrije.

Sintaksa	m_max(geometry)
Argumenti	• **geometry** - geometrija koja sadrži m vrednosti
Primeri	• `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

Vraća minimalnu m (measure) vrednost geometrije.

Sintaksa	m_min(geometry)
Argumenti	• **geometry** - geometrija koja sadrži m vrednosti
Primeri	• `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

Vraća ugao duže ose (u smeru kazaljke na satu, u stepenima od severa) orijentisanog minimalnog graničnog pravougaonika koji potpuno pokriva geometriju.

Sintaksa	main_angle(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `main_angle(geom_from_wkt('Polygon ((321577 129614, 321581 129618, 321585 129615, 321581 129610, 321577 129614))'))` → 38.66

make_circle

Pravi kružni poligon.

Sintaksa	make_circle(center, radius, [segments:=36]) [] označava opcione argumente
Argumenti	• **center** - centralna tačka kruga • **radius** - poluprečnik kruga • **segments** - opcioni argument za segmentaciju poligona. Podrazumevana vrednost je 36
Primeri	• `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

Pravi eliptični poligon.

Sintaksa	make_ellipse(center, semi_major_axis, semi_minor_axis, azimuth, [segments:=36]) [] označava opcione argumente
Argumenti	• **center** - centralna tačka elipse • **semi_major_axis** - velika poluosa elipse • **semi_minor_axis** - mala poluosa elipse • **azimuth** - orijentacija elipse • **segments** - opcioni argument za segmentaciju poligona. Podrazumevana vrednost je 36
Primeri	• `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

Pravi linijsku geometriju iz niza tačkastih geometrija.

Varijanta sa listom argumenata

Temena linije zadaju se kao zasebni argumenti funkcije.

Sintaksa	make_line(point1, point2, …)
Argumenti	• **point** - tačkasta geometrija (ili niz tačaka)
Primeri	• `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)'

Varijanta sa nizom

Temena linije zadaju se kao niz tačaka.

Sintaksa	make_line(array)
Argumenti	• **array** - niz tačaka
Primeri	• `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

Pravi tačkastu geometriju od x i y (i opcionih z i m) vrednosti.

Sintaksa	make_point(x, y, [z], [m]) [] označava opcione argumente
Argumenti	• **x** - x koordinata tačke • **y** - y koordinata tačke • **z** - opciona z koordinata tačke • **m** - opciona m vrednost tačke
Primeri	• `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

Pravi tačkastu geometriju od x, y koordinate i m vrednosti.

Sintaksa	make_point_m(x, y, m)
Argumenti	• **x** - x koordinata tačke • **y** - y koordinata tačke • **m** - m vrednost tačke
Primeri	• `geom_to_wkt(make_point_m(2,4,6))` → 'PointM (2 4 6)'

make_polygon

Pravi poligonsku geometriju od spoljnog prstena i opcionog niza unutrašnjih prstenova.

Sintaksa	make_polygon(outerRing, [innerRing1], [innerRing2], …) [] označava opcione argumente
Argumenti	• **outerRing** - zatvorena linijska geometrija za spoljni prsten poligona • **innerRing** - opciona zatvorena linijska geometrija za unutrašnji prsten
Primeri	• `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

Pravi pravougaonik od 3 tačke.

Sintaksa	make_rectangle_3points(point1, point2, point3, [option:=0]) [] označava opcione argumente
Argumenti	• **point1** - Prva tačka. • **point2** - Druga tačka. • **point3** - Treća tačka. • **option** - Opcioni argument za konstruisanje pravougaonika. Podrazumevana vrednost je 0. Vrednost može biti 0 (distance) ili 1 (projected). Opcija distance: Drugo rastojanje jednako je rastojanju između 2. i 3. tačke. Opcija projected: Drugo rastojanje jednako je rastojanju normalne projekcije 3. tačke na segment ili njegov produžetak.
Primeri	• `geom_to_wkt(make_rectangle_3points(make_point(0, 0), make_point(0,5), make_point(3, 3), 0))` → 'Polygon ((0 0, 0 5, 3.60555128 5, 3.60555128 0, 0 0))' • `geom_to_wkt(make_rectangle_3points(make_point(0, 0), make_point(0,5), make_point(3, 3), 1))` → 'Polygon ((0 0, 0 5, 3 5, 3 0, 0 0))'

make_regular_polygon

Pravi pravilan poligon.

Sintaksa	make_regular_polygon(center, radius, number_sides, [circle:=0]) [] označava opcione argumente
Argumenti	• **center** - centar pravilnog poligona • **radius** - druga tačka. Prva ako je pravilan poligon upisan. Sredina prve stranice ako je pravilan poligon opisan. • **number_sides** - Broj stranica/ivica pravilnog poligona • **circle** - Opcioni argument za konstruisanje pravilnog poligona. Podrazumevana vrednost je 0. Vrednost može biti 0 (upisan) ili 1 (opisan)
Primeri	• `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

Pravi kvadrat od dijagonale.

Sintaksa	make_square(point1, point2)
Argumenti	• **point1** - Prva tačka dijagonale • **point2** - Poslednja tačka dijagonale
Primeri	• `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

Pravi trougaoni poligon.

Sintaksa	make_triangle(point1, point2, point3)
Argumenti	• **point1** - prva tačka trougla • **point2** - druga tačka trougla • **point3** - treća tačka trougla
Primeri	• `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

Vraća ispravnu geometriju ili praznu geometriju ako geometrija nije mogla da se učini ispravnom.

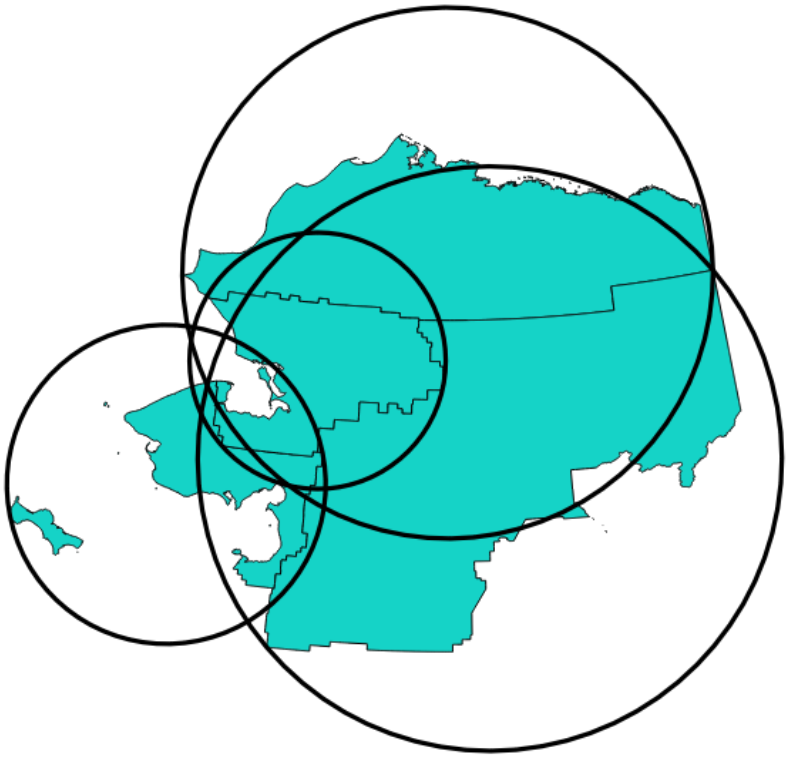
Sintaksa	make_valid(geometry, [method:='structure'], [keep_collapsed:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **method** - algoritam popravke. Može biti 'structure' ili 'linework'. Opcija 'linework' kombinuje sve prstenove u skup povezanih linija i zatim iz te mreže izdvaja ispravne poligone. Metod 'structure' prvo čini sve prstenove ispravnim, a zatim spaja ljuske i oduzima rupe od ljuski da bi dao ispravan rezultat. Pretpostavlja da su rupe i ljuske ispravno kategorizovane. • **keep_collapsed** - ako je postavljeno na true, zadržaće se komponente koje su se urušile u nižu dimenzionalnost. Na primer, prsten koji se urušio u liniju ili linija koja se urušila u tačku.
Primeri	• `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)'))` → Prazna geometrija

Više informacija: *is_valid*, algoritam *Popravi geometrije*

minimal_circle

Vraća minimalni opisani krug geometrije. On predstavlja minimalni krug koji obuhvata sve geometrije u skupu.

Sintaksa	minimal_circle(geometry, [segments:=36]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **segments** - opcioni argument za segmentaciju poligona. Podrazumevana vrednost je 36
Primeri	• `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))'



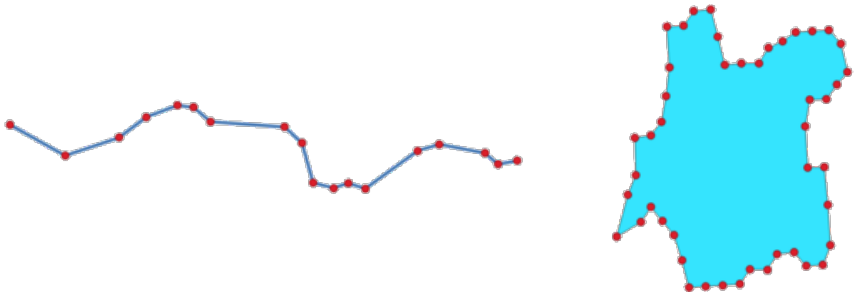
Слика9.22: Minimalni opisani krug svakog objekta

Više informacija: algoritam *Minimalni obuhvatajući krugovi*

nodes_to_points

Vraća višetačkastu geometriju koja se sastoji od svakog čvora u ulaznoj geometriji.

Sintaksa	nodes_to_points(geometry, [ignore_closing_nodes:=false]) [] označava opcione argumente
Argumenti	• **geometry** - objekat geometrije • **ignore_closing_nodes** - opcioni argument koji zadaje da li se uključuju duplirani čvorovi koji zatvaraju linije ili prstenove poligona. Podrazumevano false; postavite na true da biste izbegli uključivanje ovih dupliranih čvorova u izlaznu kolekciju.
Primeri	• `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))'



Слика9.23: Višetačkasti objekat izdvojen iz temena

Više informacija: algoritam *Izdvoj temena*

num_geometries

Vraća broj geometrija u kolekciji geometrija, ili broj delova u višedelnoj geometriji. Funkcija vraća NULL ako ulazna geometrija nije kolekcija.

Sintaksa	num_geometries(geometry)
Argumenti	• **geometry** - kolekcija geometrija ili višedelna geometrija
Primeri	• `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

Vraća broj unutrašnjih prstenova u poligonu ili kolekciji geometrija, ili NULL ako ulazna geometrija nije poligon ili kolekcija.

Sintaksa	num_interior_rings(geometry)
Argumenti	• **geometry** - ulazna geometrija
Primeri	• `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

Vraća broj temena u geometriji.

Sintaksa	num_points(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `num_points(@geometry)` → broj temena u geometriji trenutnog objekta

num_rings

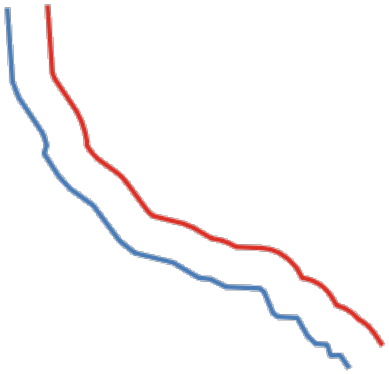
Vraća broj prstenova (uključujući spoljne prstenove) u poligonu ili kolekciji geometrija, ili NULL ako ulazna geometrija nije poligon ili kolekcija.

Sintaksa	num_rings(geometry)
Argumenti	• **geometry** - ulazna geometrija
Primeri	• `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

Vraća geometriju nastalu odmicanjem linijske geometrije u stranu. Rastojanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	offset_curve(geometry, distance, [segments:=8], [join:=1], [miter_limit:=2.0]) [] označava opcione argumente
Argumenti	• **geometry** - (multi)linestring geometrija • **distance** - rastojanje odmicanja. Pozitivne vrednosti daju bafer levo od linija, negativne desno • **segments** - broj segmenata koji se koristi za predstavljanje četvrtine kruga kada se koristi zaobljeni stil spoja. Veći broj daje glatkiju liniju sa više čvorova. • **join** - stil spoja za uglove, gde je 1 = zaobljen, 2 = oštar i 3 = zasečen • **miter_limit** - ograničenje odnosa oštrog spoja koje se koristi za veoma oštre uglove (samo pri korišćenju oštrih spojeva)
Primeri	• `offset_curve(@geometry, 10.5)` → linija odmaknuta ulevo za 10.5 jedinica • `offset_curve(@geometry, -10.5)` → linija odmaknuta udesno za 10.5 jedinica • `offset_curve(@geometry, 10.5, segments:=16, join:=1)` → linija odmaknuta ulevo za 10.5 jedinica, uz više segmenata radi glatkije krive • `offset_curve(@geometry, 10.5, join:=3)` → linija odmaknuta ulevo za 10.5 jedinica, uz zasečen spoj



Слика9.24: Plavo je izvorni sloj, crveno odmaknuti

Više informacija: algoritam *Pomeri linije*

order_parts

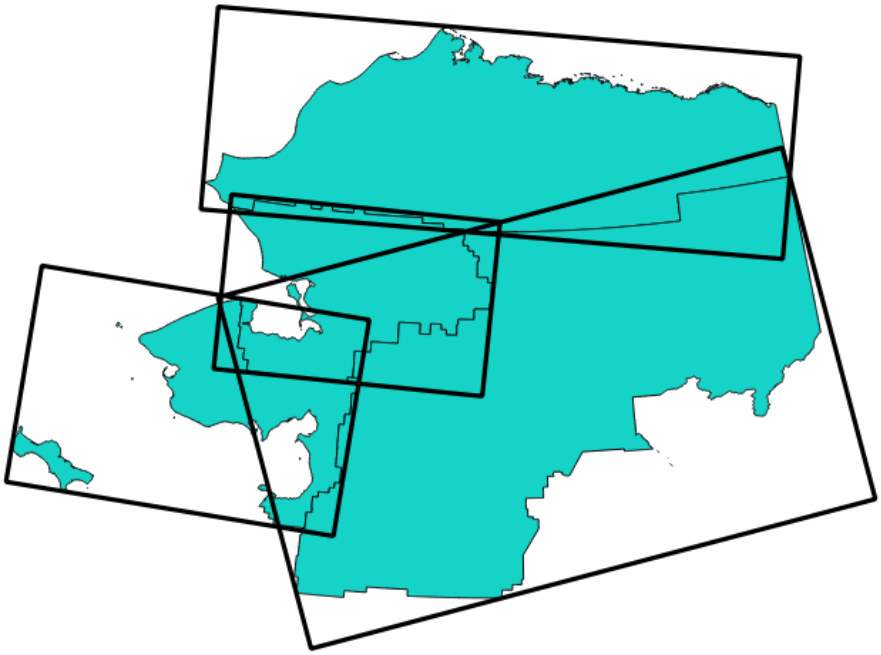
Uređuje delove višedelne geometrije prema zadatom kriterijumu

Sintaksa	order_parts(geometry, orderby, [ascending:=true]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija višestrukog tipa • **orderby** - niska izraza koja definiše kriterijum uređivanja • **ascending** - logička vrednost, True za rastuće, False za opadajuće
Primeri	• `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

Vraća geometriju koja predstavlja minimalni orijentisani granični pravougaonik ulazne geometrije.

Sintaksa	oriented_bbox(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `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))'



Слика9.25: Orijentisani minimalni granični pravougaonik

Više informacija: algoritam *Orijentisani minimalni granični pravougaonik*

overlaps

Proverava da li se jedna geometrija preklapa sa drugom. Vraća TRUE ako geometrije dele prostor, iste su dimenzije, ali nisu potpuno sadržane jedna u drugoj.

Sintaksa	overlaps(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Vraća da li trenutni objekat prostorno sadrži bar jedan objekat sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja sadržane u trenutnom objektu.

Više o osnovnom GEOS predikatu „Contains“ pročitajte u opisu PostGIS funkcije ST_Contains.

Sintaksa	overlay_contains(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_contains('regions')` → TRUE ako trenutni objekat prostorno sadrži neku regiju • `overlay_contains('regions', filter:= population > 10000)` → TRUE ako trenutni objekat prostorno sadrži regiju sa populacijom većom od 10000 • `overlay_contains('regions', name)` → niz naziva, za regije sadržane u trenutnom objektu • `array_to_string(overlay_contains('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije sadržane u trenutnom objektu • `array_sort(overlay_contains(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije sadržane u trenutnom objektu i sa populacijom većom od 10000 • `overlay_contains(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije sadržane u trenutnom objektu

Više informacija: *contains*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_crosses

Vraća da li trenutni objekat prostorno prelazi preko bar jednog objekta sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja preko kojih trenutni objekat prelazi.

Više o osnovnom GEOS predikatu „Crosses“ pročitajte u opisu PostGIS funkcije ST_Crosses.

Sintaksa	overlay_crosses(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_crosses('regions')` → TRUE ako trenutni objekat prostorno prelazi preko neke regije • `overlay_crosses('regions', filter:= population > 10000)` → TRUE ako trenutni objekat prostorno prelazi preko regije sa populacijom većom od 10000 • `overlay_crosses('regions', name)` → niz naziva, za regije preko kojih trenutni objekat prelazi • `array_to_string(overlay_crosses('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije preko kojih trenutni objekat prelazi • `array_sort(overlay_crosses(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije preko kojih trenutni objekat prelazi i sa populacijom većom od 10000 • `overlay_crosses(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije koje trenutni objekat preseca

Više informacija: *crosses*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_disjoint

Vraća da li je trenutni objekat prostorno razdvojen od svih objekata sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja koji su razdvojeni od trenutnog objekta.

Više o osnovnom GEOS predikatu „Disjoint“ pročitajte u opisu PostGIS funkcije ST_Disjoint.

Sintaksa	overlay_disjoint(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_disjoint('regions')` → TRUE ako je trenutni objekat prostorno razdvojen od svih regija • `overlay_disjoint('regions', filter:= population > 10000)` → TRUE ako je trenutni objekat prostorno razdvojen od svih regija sa populacijom većom od 10000 • `overlay_disjoint('regions', name)` → niz naziva, za regije prostorno razdvojene od trenutnog objekta • `array_to_string(overlay_disjoint('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije prostorno razdvojene od trenutnog objekta • `array_sort(overlay_disjoint(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije prostorno razdvojene od trenutnog objekta i sa populacijom većom od 10000 • `overlay_disjoint(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije prostorno razdvojene od trenutnog objekta

Više informacija: *disjoint*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_equals

Vraća da li je geometrija trenutnog objekta potpuno jednaka geometriji bar jednog objekta sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja čija je geometrija potpuno jednaka geometriji trenutnog objekta. Imajte u vidu da je redosled temena bitan.

Sintaksa	overlay_equals(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_equals('regions')` → TRUE ako je geometrija trenutnog objekta potpuno ista kao geometrija neke regije • `overlay_equals('regions', filter:= population > 10000)` → TRUE ako je geometrija trenutnog objekta potpuno ista kao geometrija neke regije čija je populacija veća od 10000 • `overlay_equals('regions', name)` → niz naziva, za regije potpuno jednake trenutnom objektu • `array_to_string(overlay_equals('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije potpuno jednake trenutnom objektu • `array_sort(overlay_equals(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije potpuno jednake trenutnom objektu i sa populacijom većom od 10000 • `overlay_equals(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije potpuno jednake trenutnom objektu

Белешка

Ova funkcija zahteva potpunu jednakost geometrija, što znači da su geometrije 'LINESTRING(0 0, 1 1)' i 'LINESTRING(1 1, 0 0)' različite.

Više informacija: *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_intersects

Vraća da li trenutni objekat prostorno preseca bar jedan objekat sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja koje trenutni objekat preseca.

Više o osnovnom GEOS predikatu „Intersects“ pročitajte u opisu PostGIS funkcije ST_Intersects.

Sintaksa	overlay_intersects(layer, [expression], [filter], [limit], [cache:=false], [min_overlap], [min_inscribed_circle_radius], [return_details], [sort_by_intersection_size]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka) • **min_overlap** - definiše opcioni filter isključivanja: – za poligone, minimalna površina u kvadratnim jedinicama trenutnog objekta za presek. Ako presek daje više poligona, presek će biti vraćen ako bar jedan od poligona ima površinu veću ili jednaku toj vrednosti – za linije, minimalna dužina u jedinicama trenutnog objekta. Ako presek daje više linija, presek će biti vraćen ako bar jedna od linija ima dužinu veću ili jednaku toj vrednosti. • **min_inscribed_circle_radius** - definiše opcioni filter isključivanja (samo za poligone): minimalni poluprečnik u jedinicama trenutnog objekta za maksimalni upisani krug preseka. Ako presek daje više poligona, presek će biti vraćen ako bar jedan od poligona ima poluprečnik maksimalnog upisanog kruga veći ili jednak toj vrednosti. Više o osnovnom GEOS predikatu pročitajte u opisu PostGIS funkcije ST_MaximumInscribedCircle. Ovaj argument zahteva GEOS >= 3.9. • **return_details** - Postavite na true da biste vratili listu mapa koje sadrže (nazivi ključeva pod navodnicima) 'id' objekta, 'result' izraza i vrednost 'overlap' (najvećeg elementa u slučaju višedelne geometrije). Kada je ciljni sloj poligonski, vraća se i 'radius' maksimalnog upisanog kruga. Važi samo kada se koristi sa parametrom expression • **sort_by_intersection_size** - važi samo kada se koristi sa izrazom; postavite na 'des' da biste vratili rezultate poređane po vrednosti preklapanja u opadajućem redosledu, ili na 'asc' za rastući redosled.
Primeri	• `overlay_intersects('regions')` → TRUE ako trenutni objekat prostorno preseca neku regiju • `overlay_intersects('regions', filter:= population > 10000)` → TRUE ako trenutni objekat prostorno preseca regiju sa populacijom većom od 10000 • `overlay_intersects('regions', name)` → niz naziva, za regije koje trenutni objekat preseca • `array_to_string(overlay_intersects('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije koje trenutni objekat preseca • `array_sort(overlay_intersects(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije koje trenutni objekat preseca i sa populacijom većom od 10000 • `overlay_intersects(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije koje trenutni objekat preseca • `overlay_intersects(layer:='regions', min_overlap:=0.54)` → TRUE ako trenutni objekat prostorno preseca regiju i površina preseka (bar jednog od delova u slučaju multipoligona) je veća ili jednaka 0.54 • `overlay_intersects(layer:='regions', min_inscribed_circle_radius:=0.54)` → TRUE ako trenutni objekat prostorno preseca regiju i poluprečnik maksimalnog upisanog kruga površine preseka (bar jednog od delova u slučaju višedelne geometrije) je veći ili jednak 0.54 • `overlay_intersects(layer:='regions', expression:= geom_to_wkt(@geometry), return_details:=true)` → niz mapa koje sadrže 'id', 'result', 'overlap' i 'radius' • `overlay_intersects(layer:='regions', expression:= geom_to_wkt(@geometry), sort_by_intersection_size:=true)` → niz geometrija (u WKT-u) poređanih po vrednosti preklapanja u opadajućem redosledu

Više informacija: *intersects*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_nearest

Vraća da li trenutni objekat ima objekat(e) sa ciljnog sloja unutar zadatog rastojanja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja unutar rastojanja od trenutnog objekta.

Napomena: Ova funkcija može biti spora i trošiti mnogo memorije za velike slojeve.

Sintaksa	overlay_nearest(layer, [expression], [filter], [limit:=1], [max_distance], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - ciljni sloj • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, koriste se svi objekti na ciljnom sloju. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraća se samo najbliži objekat. Ako je -1, vraćaju se svi podudarajući objekti. • **max_distance** - opciono rastojanje za ograničavanje pretrage podudarajućih objekata. Ako nije zadato, koriste se svi objekti na ciljnom sloju. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_nearest('airports')` → TRUE ako sloj „airports" ima bar jedan objekat • `overlay_nearest('airports', max_distance:= 5000)` → TRUE ako postoji aerodrom na rastojanju do 5000 jedinica karte od trenutnog objekta • `overlay_nearest('airports', name)` → naziv aerodroma najbližeg trenutnom objektu, kao niz • `array_to_string(overlay_nearest('airports', name))` → naziv aerodroma najbližeg trenutnom objektu, kao niska • `overlay_nearest(layer:='airports', expression:= name, max_distance:= 5000)` → naziv najbližeg aerodroma na rastojanju do 5000 jedinica karte od trenutnog objekta, kao niz • `overlay_nearest(layer:='airports', expression:="name", filter:= "Use"='Civilian', limit:=3)` → niz naziva, za najviše tri najbliža civilna aerodroma poređana po rastojanju • `overlay_nearest(layer:='airports', expression:="name", limit:= -1, max_distance:= 5000)` → niz naziva, za sve aerodrome na rastojanju do 5000 jedinica karte od trenutnog objekta, poređane po rastojanju

Više informacija: *rad sa nizovima*, algoritam *Spajanje atributa po najbližem*

overlay_touches

Vraća da li trenutni objekat prostorno dodiruje bar jedan objekat sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja koje trenutni objekat dodiruje.

Više o osnovnom GEOS predikatu „Touches" pročitajte u opisu PostGIS funkcije ST_Touches.

Sintaksa	overlay_touches(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_touches('regions')` → TRUE ako trenutni objekat prostorno dodiruje neku regiju • `overlay_touches('regions', filter:= population > 10000)` → TRUE ako trenutni objekat prostorno dodiruje regiju sa populacijom većom od 10000 • `overlay_touches('regions', name)` → niz naziva, za regije koje trenutni objekat dodiruje • `array_to_string(overlay_touches('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije koje trenutni objekat dodiruje • `array_sort(overlay_touches(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije koje trenutni objekat dodiruje i sa populacijom većom od 10000 • `overlay_touches(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije koje trenutni objekat dodiruje

Više informacija: *touches*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

overlay_within

Vraća da li je trenutni objekat prostorno unutar bar jednog objekta sa ciljnog sloja, ili niz rezultata zasnovanih na izrazu za objekte sa ciljnog sloja koji sadrže trenutni objekat.

Više o osnovnom GEOS predikatu „Within“ pročitajte u opisu PostGIS funkcije ST_Within.

Sintaksa	overlay_within(layer, [expression], [filter], [limit], [cache:=false]) [] označava opcione argumente
Argumenti	• **layer** - sloj čije se preklapanje proverava • **expression** - opcioni izraz koji se izračunava na objektima sa ciljnog sloja. Ako nije zadat, funkcija će samo vratiti logičku vrednost koja pokazuje da li postoji bar jedno podudaranje. • **filter** - opcioni izraz za filtriranje ciljnih objekata koji se proveravaju. Ako nije zadat, proveravaju se svi objekti. • **limit** - opcioni ceo broj za ograničavanje broja podudarajućih objekata. Ako nije zadat, vraćaju se svi podudarajući objekti. • **cache** - postavite na true da biste izgradili lokalni prostorni indeks (najčešće ovo nije poželjno, osim ako radite sa posebno sporim provajderom podataka)
Primeri	• `overlay_within('regions')` → TRUE ako je trenutni objekat prostorno unutar neke regije • `overlay_within('regions', filter:= population > 10000)` → TRUE ako je trenutni objekat prostorno unutar regije sa populacijom većom od 10000 • `overlay_within('regions', name)` → niz naziva, za regije koje sadrže trenutni objekat • `array_to_string(overlay_within('regions', name))` → niska kao lista naziva razdvojena zarezima, za regije koje sadrže trenutni objekat • `array_sort(overlay_within(layer:='regions', expression:="name", filter:= population > 10000))` → uređen niz naziva, za regije koje sadrže trenutni objekat i sa populacijom većom od 10000 • `overlay_within(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → niz geometrija (u WKT-u), za najviše dve regije koje sadrže trenutni objekat

Više informacija: *within*, *rad sa nizovima*, algoritam *Izbor po lokaciji*

$perimeter

Vraća dužinu obima trenutnog objekta. Obim koji računa ova funkcija poštuje i podešavanje elipsoida projekta i podešavanje jedinice rastojanja. Na primer, ako je za projekat postavljen elipsoid, izračunati obim biće elipsoidan, a ako elipsoid nije postavljen, izračunati obim biće planimetrijski.

Sintaksa	$perimeter
Primeri	• `$perimeter` → 42

perimeter

Vraća obim poligonskog objekta geometrije. Izračunavanja su uvek planimetrijska u prostornom referentnom sistemu (SRS) te geometrije, a jedinice vraćenog obima odgovaraće jedinicama tog SRS-a. Ovo se razlikuje od izračunavanja koja vrši funkcija $perimeter, koja izvodi elipsoidna izračunavanja na osnovu podešavanja elipsoida i jedinice rastojanja projekta.

Sintaksa	perimeter(geometry)
Argumenti	• **geometry** - objekat poligonske geometrije
Primeri	• `perimeter(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 12.0

point_n

Vraća određeni čvor iz geometrije.

Sintaksa	point_n(geometry, index)
Argumenti	• **geometry** - objekat geometrije • **index** - indeks čvora koji se vraća, gde je 1 prvi čvor; ako je vrednost negativna, izabrani indeks temena biće njihov ukupan broj umanjen za apsolutnu vrednost
Primeri	• `geom_to_wkt(point_n(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'),2))` → 'Point (4 0)'

Više informacija: algoritam *Izdvoj određena temena*

point_on_surface

Vraća tačku za koju je zagarantovano da leži na površini geometrije.

Sintaksa	point_on_surface(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `point_on_surface(@geometry)` → tačkasta geometrija

Više informacija: algoritam *Tačka na površini*

pole_of_inaccessibility

Računa približan pol nepristupačnosti za površ, što je najudaljenija unutrašnja tačka od granice te površi. Ova funkcija koristi algoritam 'polylabel' (Vladimir Agafonjkin, 2016), koji je iterativni pristup koji garantuje pronalaženje pravog pola nepristupačnosti u okviru zadate tolerancije. Preciznije tolerancije zahtevaju više iteracija i duže se računaju.

Sintaksa	pole_of_inaccessibility(geometry, tolerance)
Argumenti	• **geometry** - geometrija • **tolerance** - maksimalno rastojanje između vraćene tačke i stvarne lokacije pola
Primeri	• `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)'



Слика9.26: Pol nepristupačnosti

Više informacija: algoritam *Pol nepristupačnosti*

project

Vraća tačku projektovanu od početne tačke pomoću rastojanja, pravca (azimuta) i visinskog ugla u radijanima.

Sintaksa	project(point, distance, azimuth, [elevation]) [] označava opcione argumente
Argumenti	• **point** - početna tačka • **distance** - rastojanje projektovanja • **azimuth** - azimut u radijanima u smeru kazaljke na satu, gde 0 odgovara severu • **elevation** - ugao nagiba u radijanima
Primeri	• `geom_to_wkt(project(make_point(1, 2), 3, radians(270)))` → 'Point(-2, 2)'

Više informacija: algoritam *Projektuj tačke (kartezijanski)*

relate

Testira predstavljanje odnosa između dve geometrije po modelu Dimensionally Extended 9-Intersection Model (DE-9IM).

Varijanta sa relacijom

Vraća predstavljanje odnosa između dve geometrije po modelu Dimensionally Extended 9-Intersection Model (DE--9IM).

Sintaksa	relate(geometry, geometry)
Argumenti	• **geometry** - geometrija • **geometry** - geometrija
Primeri	• `relate( geom_from_wkt( 'LINESTRING(40 40,120 120)' ), geom_from_wkt( 'LINESTRING(40 40,60 120)' ) )` → 'FF1F00102'

Varijanta sa podudaranjem šare

Proverava da li DE-9IM odnos između dve geometrije odgovara zadatoj šari.

Sintaksa	relate(geometry, geometry, pattern)
Argumenti	• **geometry** - geometrija • **geometry** - geometrija • **pattern** - DE-9IM šara sa kojom se traži podudaranje
Primeri	• `relate( geom_from_wkt( 'LINESTRING(40 40,120 120)' ), geom_from_wkt( 'LINESTRING(40 40,60 120)' ), '**1F001**' )` → TRUE

reverse

Obrće smer linije ili obrće tekstualnu nisku.

Varijanta sa niskom

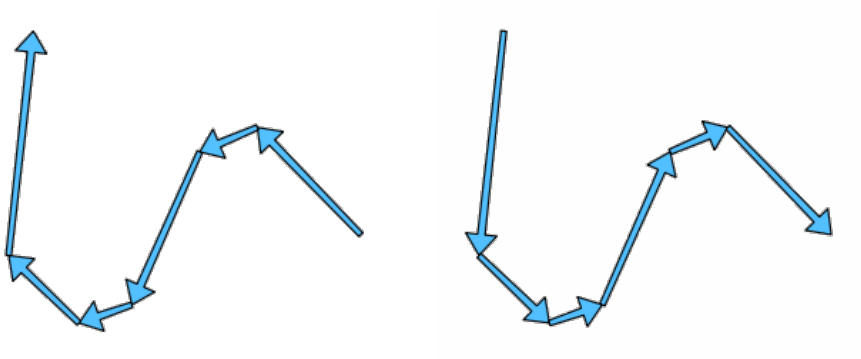
Obrće redosled znakova u niski.

Sintaksa	reverse(string)
Argumenti	• **string** - niska koja se obrće
Primeri	• `reverse('hello')` → 'olleh'

Varijanta sa geometrijom

Obrće smer linije obrtanjem redosleda njenih temena.

Sintaksa	reverse(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → 'LINESTRING(2 2, 1 1, 0 0)'



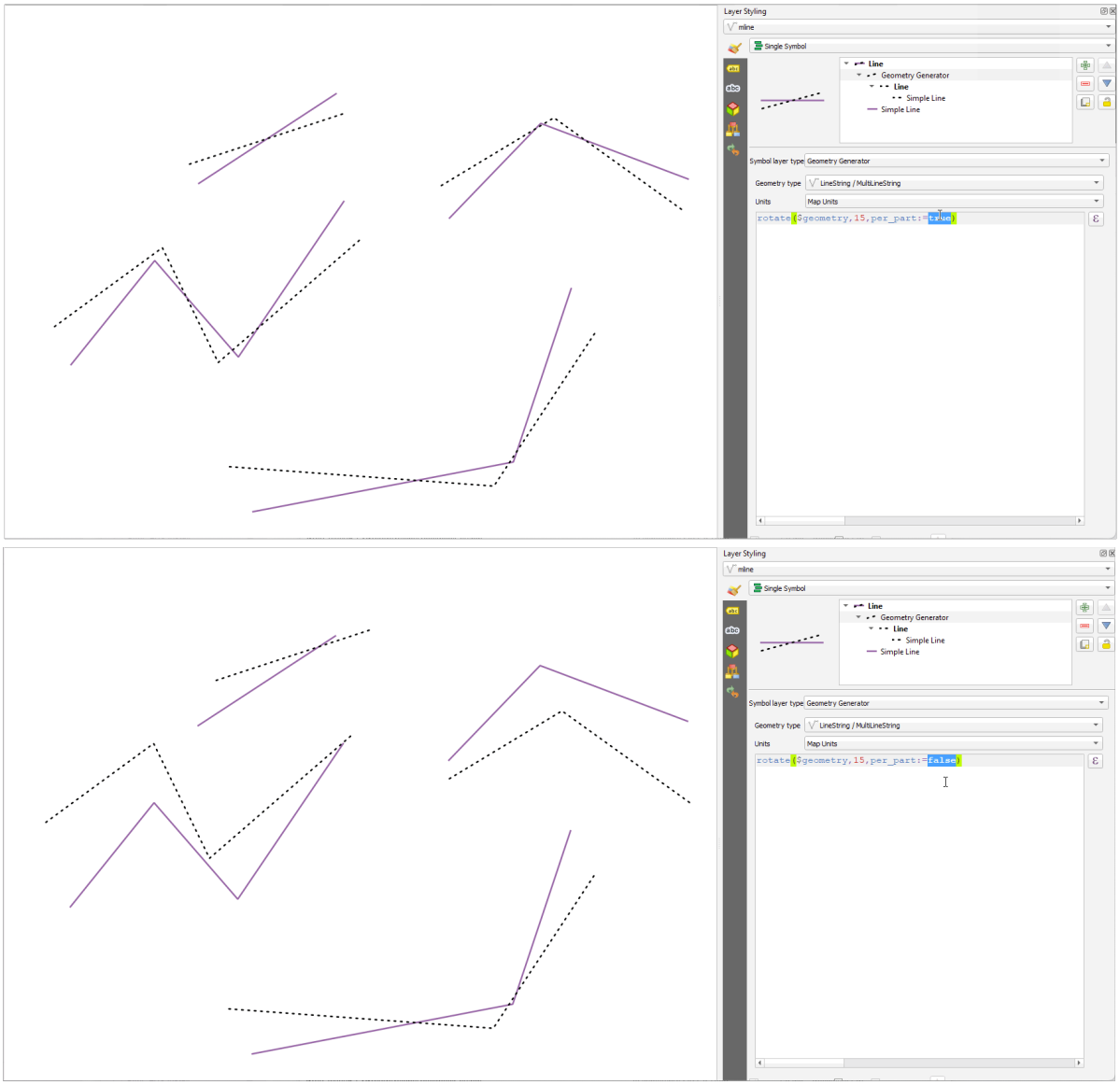
Слика9.27: Obrtanje smera linije

Više informacija: algoritam *Obrni smer linije*

rotate

Vraća rotiranu verziju geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	rotate(geometry, rotation, [center:=NULL], [per_part:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **rotation** - rotacija u smeru kazaljke na satu, u stepenima • **center** - centralna tačka rotacije. Ako nije zadata, koristi se centar graničnog pravougaonika geometrije. • **per_part** - primeni rotaciju po delu. Ako je true, rotacija će se primeniti oko centra graničnog pravougaonika svakog dela kada je ulazna geometrija višedelna, a eksplicitna centralna tačka rotacije nije zadata.
Primeri	• `rotate(@geometry, 45, make_point(4, 5))` → geometrija rotirana za 45 stepeni u smeru kazaljke na satu oko tačke (4, 5) • `rotate(@geometry, 45)` → geometrija rotirana za 45 stepeni u smeru kazaljke na satu oko centra svog graničnog pravougaonika



Слика9.28: Rotacija objekata

roundness

Računa koliko je oblik poligona blizak krugu. Funkcija vraća 1 kada je oblik poligona savršen krug i 0 kada je potpuno spljošten.

Sintaksa	roundness(geometry)
Argumenti	• **geometry** - poligon
Primeri	• `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

Više informacija: algoritam *Zaobljenost*

scale

Vraća skaliranu verziju geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	scale(geometry, x_scale, y_scale, [center:=NULL]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **x_scale** - faktor skaliranja po x osi • **y_scale** - faktor skaliranja po y osi • **center** - centralna tačka skaliranja. Ako nije zadata, koristi se centar graničnog pravougaonika geometrije.
Primeri	• `scale(@geometry, 2, 0.5, make_point(4, 5))` → geometrija dvostruko skalirana horizontalno i prepolovljena vertikalno, oko tačke (4, 5) • `scale(@geometry, 2, 0.5)` → geometrija dvostruko horizontalno i prepolovljena vertikalno, oko centra svog graničnog pravougaonika

segments_to_lines

Vraća višelinijsku geometriju koja se sastoji od linije za svaki segment u ulaznoj geometriji.

Sintaksa	segments_to_lines(geometry)
Argumenti	• **geometry** - objekat geometrije
Primeri	• `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))'

Više informacija: algoritam *Rasturi linije*

shared_paths

Vraća kolekciju koja sadrži putanje zajedničke za dve ulazne geometrije. One koje idu u istom smeru nalaze se u prvom elementu kolekcije, a one koje idu u suprotnom smeru u drugom elementu. Same putanje date su u smeru prve geometrije.

Sintaksa	shared_paths(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija tipa LineString/MultiLineString • **geometry2** - geometrija tipa LineString/MultiLineString
Primeri	• `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

Vraća najkraću liniju koja spaja geometry1 sa geometry2. Rezultujuća linija počinje na geometry1 i završava se na geometry2.

Sintaksa	shortest_line(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija od koje se traži najkraća linija • **geometry2** - geometrija do koje se traži najkraća linija
Primeri	• `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

Uprošćava geometriju uklanjanjem čvorova pomoću praga zasnovanog na rastojanju (tj. Daglas-Pekerov algoritam). Algoritam čuva velika odstupanja u geometrijama i smanjuje broj temena u skoro pravim segmentima.

Sintaksa	simplify(geometry, tolerance)
Argumenti	• **geometry** - geometrija • **tolerance** - maksimalno odstupanje od pravih segmenata da bi tačke bile uklonjene
Primeri	• `geom_to_wkt(simplify(geometry:=geom_from_wkt('LineString(0 0, 5 0.1, 10 0)'),tolerance:=5))` → 'LineString(0 0, 10 0)'



Слика9.29: Sleva nadesno, izvorni sloj i rastuće tolerancije uprošćavanja

Više informacija: algoritam *Uprosti*

simplify_vw

Uprošćava geometriju uklanjanjem čvorova pomoću praga zasnovanog na površini (tj. Visvalingam-Vajatov algoritam). Algoritam uklanja temena koja stvaraju male površine u geometrijama, npr. uske šiljke ili skoro prave segmente.

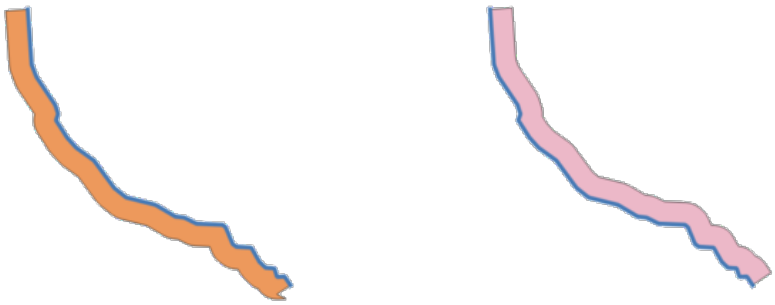
Sintaksa	simplify_vw(geometry, tolerance)
Argumenti	• **geometry** - geometrija • **tolerance** - mera maksimalne površine koju stvara čvor da bi taj čvor bio uklonjen
Primeri	• `geom_to_wkt(simplify_vw(geometry:=geom_from_wkt('LineString(0 0, 5 0, 5.01 10, 5.02 0, 10 0)'),tolerance:=5))` → 'LineString(0 0, 10 0)'

Više informacija: algoritam *Uprosti*

single_sided_buffer

Vraća geometriju nastalu baferovanjem samo jedne strane linijske geometrije. Rastojanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	single_sided_buffer(geometry, distance, [segments:=8], [join:=1], [miter_limit:=2.0]) [] označava opcione argumente
Argumenti	• **geometry** - (multi)linestring geometrija • **distance** - rastojanje bafera. Pozitivne vrednosti daju bafer levo od linija, negativne desno • **segments** - broj segmenata koji se koristi za predstavljanje četvrtine kruga kada se koristi zaobljeni stil spoja. Veći broj daje glatkiji bafer sa više čvorova. • **join** - stil spoja za uglove, gde je 1 = zaobljen, 2 = oštar i 3 = zasečen • **miter_limit** - ograničenje odnosa oštrog spoja koje se koristi za veoma oštre uglove (samo pri korišćenju oštrih spojeva)
Primeri	• `single_sided_buffer(@geometry, 10.5)` → linija baferovana ulevo za 10.5 jedinica • `single_sided_buffer(@geometry, -10.5)` → linija baferovana udesno za 10.5 jedinica • `single_sided_buffer(@geometry, 10.5, segments:=16, join:=1)` → linija baferovana ulevo za 10.5 jedinica, uz više segmenata radi glatkijeg bafera • `single_sided_buffer(@geometry, 10.5, join:=3)` → linija baferovana ulevo za 10.5 jedinica, uz zasečen spoj



Слика9.30: Bafer sa leve i sa desne strane na istom vektorskom linijskom sloju

Više informacija: algoritam *Jednostrani bafer*

sinuosity

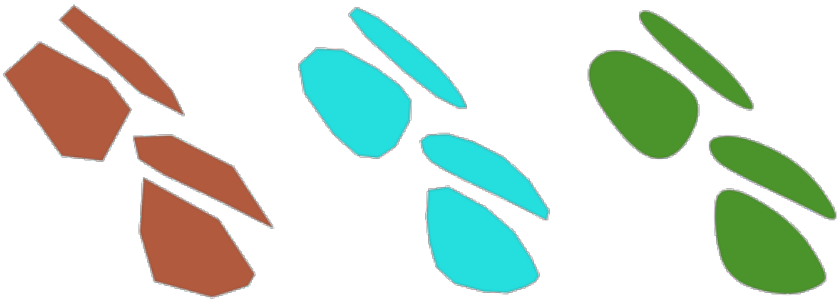
Vraća sinuoznost krive, što je odnos dužine krive i pravog (2D) rastojanja između njenih krajnjih tačaka.

Sintaksa	sinuosity(geometry)
Argumenti	• **geometry** - Ulazna kriva (circularstring, linestring)
Primeri	• `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

Zaglađuje geometriju dodavanjem dodatnih čvorova koji zaobljavaju uglove geometrije. Ako ulazne geometrije sadrže Z ili M vrednosti, i one će biti zaglađene, a izlazna geometrija zadržaće istu dimenzionalnost kao ulazna.

Sintaksa	smooth(geometry, [iterations:=1], [offset:=0.25], [min_length:=-1], [max_angle:=180]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **iterations** - broj iteracija zaglađivanja koje se primenjuju. Veći brojevi daju glatkije, ali složenije geometrije. • **offset** - vrednost između 0 i 0.5 koja određuje koliko tesno zaglađena geometrija prati originalnu geometriju. Manje vrednosti daju tešnje zaglađivanje, veće vrednosti daju labavije zaglađivanje. • **min_length** - minimalna dužina segmenata na koje se primenjuje zaglađivanje. Ovaj parametar se može koristiti da bi se izbeglo postavljanje prekomernih dodatnih čvorova u kraćim segmentima geometrije. • **max_angle** - maksimalan ugao u čvoru na koji se primenjuje zaglađivanje (0-180). Smanjivanjem maksimalnog ugla mogu se sačuvati namerno oštri uglovi u geometriji. Na primer, vrednost od 80 stepeni zadržaće prave uglove u geometriji.
Primeri	• `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)'



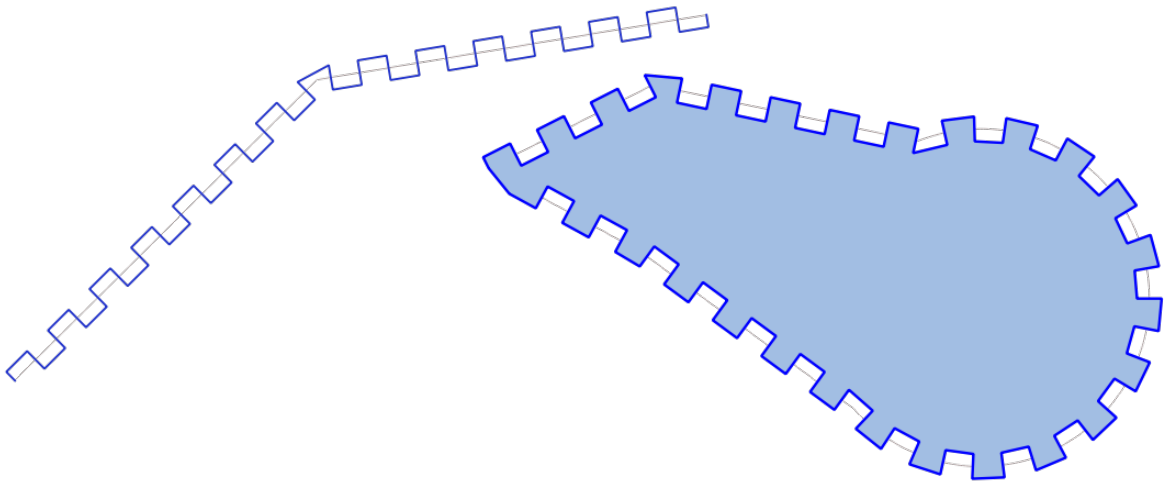
Слика9.31: Povećanje broja iteracija daje glatkije geometrije

Više informacija: algoritam *Zagladi*

square_wave

Konstruiše pravougaone/četvrtaste talase duž granice geometrije.

Sintaksa	square_wave(geometry, wavelength, amplitude, [strict:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **wavelength** - talasna dužina pravougaonog talasnog oblika • **amplitude** - amplituda pravougaonog talasnog oblika • **strict** - Podrazumevano se argument wavelength tretira kao „maksimalna talasna dužina", pri čemu se stvarna talasna dužina dinamički prilagođava tako da se duž granica geometrije napravi tačan broj pravougaonih talasa. Ako je argument strict postavljen na true, talasna dužina će se koristiti tačno, a za poslednji talasni oblik može se koristiti nepotpuna šara.
Primeri	• `square_wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Pravougaoni talasi talasne dužine 3 i amplitude 1 duž linije • `geom_to_wkt(square_wave(geom_from_wkt('LineString(0 0, 5 0)'), 3, 1, false))` → 'LineString (0 0, 0 1, 1.25 1, 1.25 -1, 2.5 -1, 2.5 1, 3.75 1, 3.75 -1, 5 -1, 5 0)' • `geom_to_wkt(square_wave(geom_from_wkt('LineString(0 0, 5 0)'), 3, 1, true))` → 'LineString (0 0, 0 1, 1.5 1, 1.5 -1, 3 -1, 3 1, 4.5 1, 5 0)'

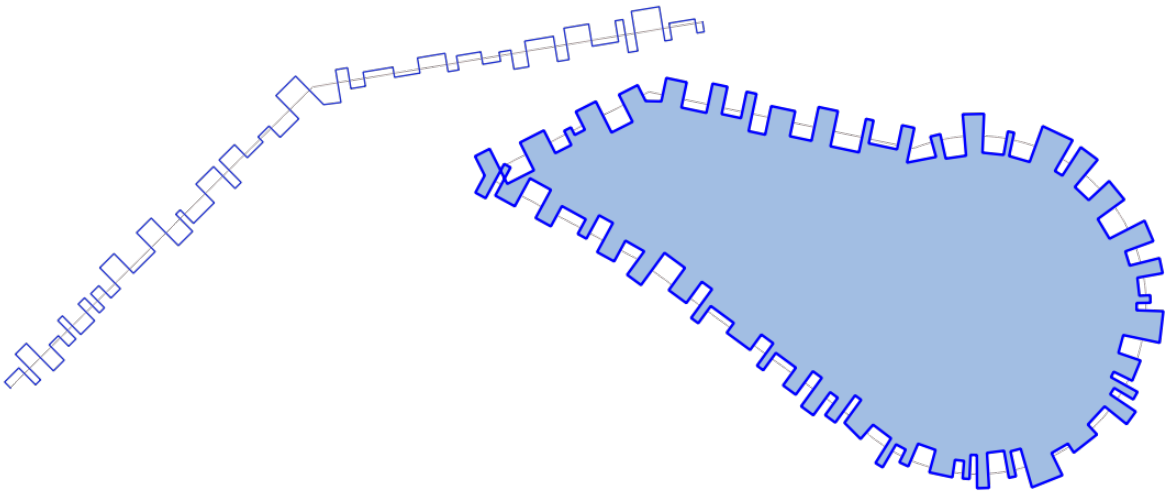


Слика9.32: Simbolizacija objekata pravougaonim talasima

square_wave_randomized

Konstruiše nasumične pravougaone/četvrtaste talase duž granice geometrije.

Sintaksa	square_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed:=0]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **min_wavelength** - minimum wavelength of waves • **max_wavelength** - maksimalna talasna dužina talasa • **min_amplitude** - minimalna amplituda talasa • **max_amplitude** - maksimalna amplituda talasa • **seed** - zadaje slučajno seme za generisanje talasa. Ako je seme 0, generisaće se potpuno slučajan skup talasa.
Primeri	• `square_wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Nasumično dimenzionisani pravougaoni talasi sa talasnim dužinama između 2 i 3 i amplitudama između 0.1 i 0.2 duž linije



Слика9.33: Simbolizacija objekata nasumičnim pravougaonim talasima

start_point

Vraća prvi čvor iz geometrije.

Sintaksa	start_point(geometry)
Argumenti	• **geometry** - objekat geometrije
Primeri	• `geom_to_wkt(start_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → 'Point (4 0)'



Слика9.34: Početna tačka linijskog objekta

Više informacija: *end_point*, algoritam *Izdvoj određena temena*

straight_distance_2d

Vraća direktno/euklidsko rastojanje između prvog i poslednjeg temena geometrije. Geometrija mora biti kriva (circularstring, linestring).

Sintaksa	straight_distance_2d(geometry)
Argumenti	• **geometry** - Geometrija.
Primeri	• `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

Više informacija: *length*

sym_difference

Vraća geometriju koja predstavlja delove dve geometrije koji se ne presecaju.

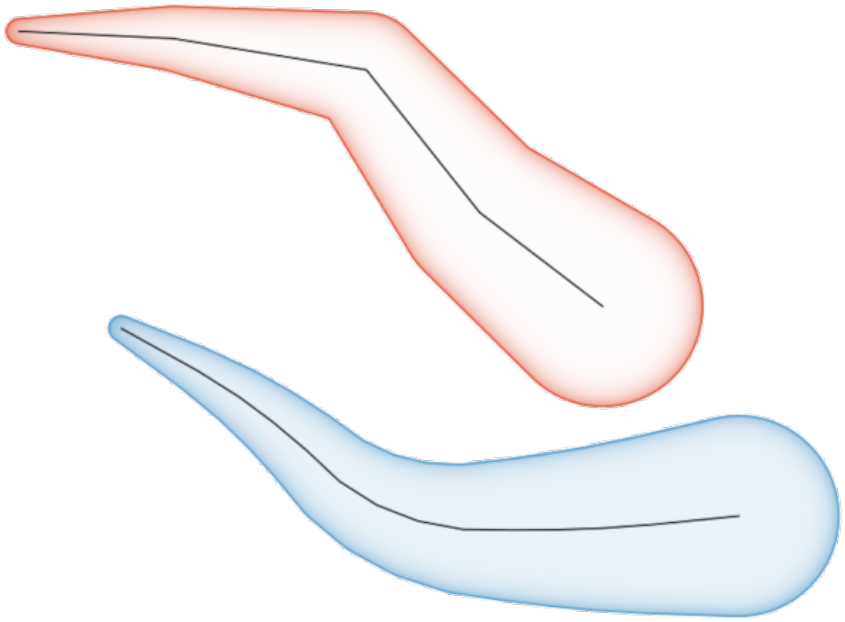
Sintaksa	sym_difference(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `geom_to_wkt( sym_difference( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'LINESTRING(3 3, 8 8)' ) ) )` → 'LINESTRING(5 5, 8 8)'

Više informacija: algoritam *Simetrična razlika*

tapered_buffer

Pravi bafer duž linijske geometrije, pri čemu se prečnik bafera ravnomerno menja duž dužine linije.

Sintaksa	tapered_buffer(geometry, start_width, end_width, [segments:=8]) [] označava opcione argumente
Argumenti	• **geometry** - ulazna geometrija. Mora biti (multi)linijska geometrija. • **start_width** - širina bafera na početku linije, • **end_width** - širina bafera na kraju linije. • **segments** - number of segments to approximate quarter-circle curves in the buffer.
Primeri	• `tapered_buffer(geometry:=geom_from_wkt('LINESTRING(1 2, 4 2)'), start_width:=1,end_width:=2,segments:=8)` → Sužavajući bafer koji počinje prečnikom 1 i završava se prečnikom 2 duž linijske geometrije.



Слика9.35: Sužavajući bafer na linijskim objektima

Više informacija: algoritam *Suženi baferi*

touches

Proverava da li jedna geometrija dodiruje drugu. Vraća TRUE ako geometrije imaju bar jednu zajedničku tačku, ali im se unutrašnjosti ne presecaju.

Sintaksa	touches(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Više informacija: *overlay_touches*

transform

Vraća geometriju transformisanu iz izvornog CRS-a u ciljni CRS.

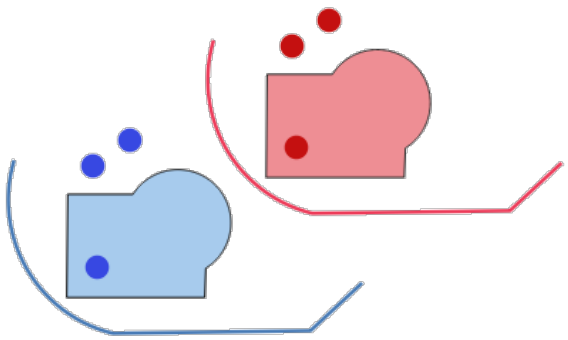
Sintaksa	transform(geometry, source_auth_id, dest_auth_id)
Argumenti	• **geometry** - geometrija • **source_auth_id** - definicija izvornog CRS-a ili objekat CRS-a • **dest_auth_id** - definicija ciljnog CRS-a ili objekat CRS-a
Primeri	• `geom_to_wkt( transform( make_point(488995.53240249, 7104473.38600835), 'EPSG:2154', 'EPSG:4326' ) )` → 'POINT(0 51)'

Više informacija: algoritam *Reprojektovanje sloja*

translate

Vraća transliranu verziju geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	translate(geometry, dx, dy)
Argumenti	• **geometry** - geometrija • **dx** - delta x • **dy** - delta y
Primeri	• `translate(@geometry, 5, 10)` → geometrija istog tipa kao original



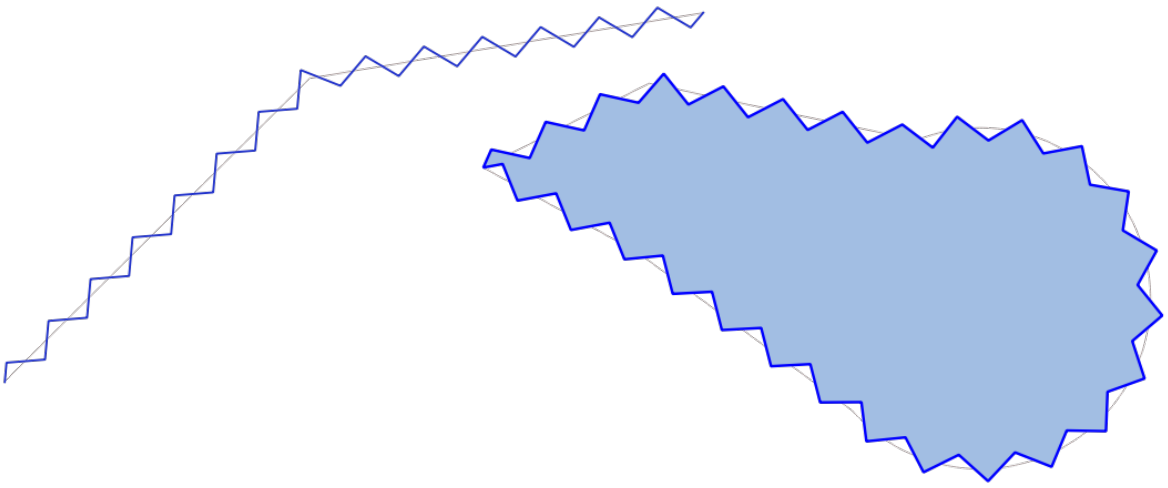
Слика9.36: Translacija objekata

Više informacija: algoritam *Transliraj*

triangular_wave

Konstruiše trougaone talase duž granice geometrije.

Sintaksa	triangular_wave(geometry, wavelength, amplitude, [strict:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **wavelength** - talasna dužina trougaonog talasnog oblika • **amplitude** - amplituda trougaonog talasnog oblika • **strict** - Podrazumevano se argument wavelength tretira kao „maksimalna talasna dužina", pri čemu se stvarna talasna dužina dinamički prilagođava tako da se duž granica geometrije napravi tačan broj trougaonih talasa. Ako je argument strict postavljen na true, talasna dužina će se koristiti tačno, a za poslednji talasni oblik može se koristiti nepotpuna šara.
Primeri	• `triangular_wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Trougaoni talasi talasne dužine 3 i amplitude 1 duž linije • `geom_to_wkt(triangular_wave(geom_from_wkt('LineString(0 0, 5 0)'), 3, 1, false))` → 'LineString (0 0, 0.625 1, 1.875 -1, 3.125 1, 4.375 -1, 5 0)' • `geom_to_wkt(triangular_wave(geom_from_wkt('LineString(0 0, 5 0)'), 3, 1, true))` → 'LineString (0 0, 0.75 1, 2.25 -1, 3.75 1, 5 0)'

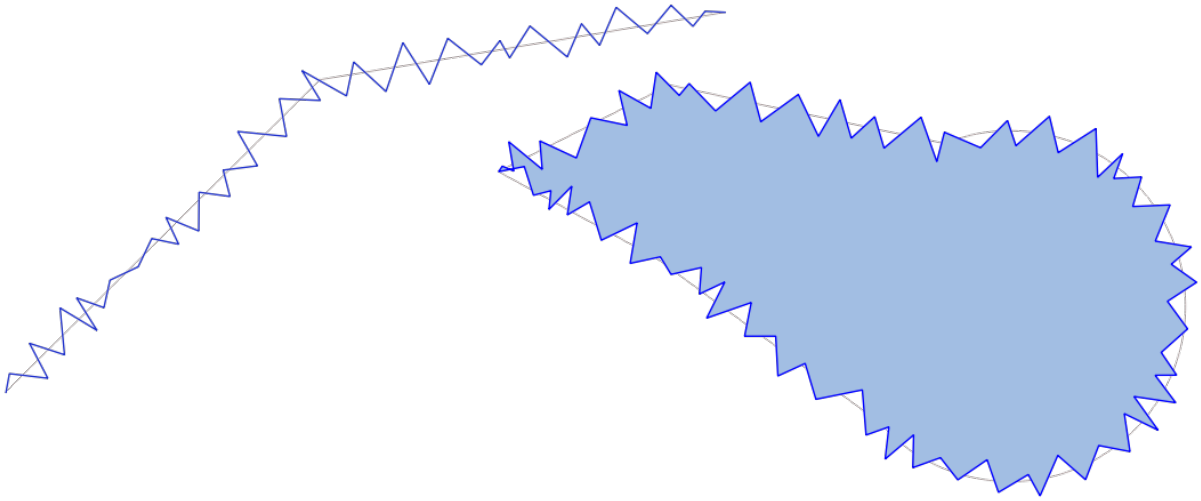


Слика9.37: Simbolizacija objekata trougaonim talasima

triangular_wave_randomized

Konstruiše nasumične trougaone talase duž granice geometrije.

Sintaksa	triangular_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed:=0]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **min_wavelength** - minimum wavelength of waves • **max_wavelength** - maksimalna talasna dužina talasa • **min_amplitude** - minimalna amplituda talasa • **max_amplitude** - maksimalna amplituda talasa • **seed** - zadaje slučajno seme za generisanje talasa. Ako je seme 0, generisaće se potpuno slučajan skup talasa.
Primeri	• `triangular_wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Nasumično dimenzionisani trougaoni talasi sa talasnim dužinama između 2 i 3 i amplitudama između 0.1 i 0.2 duž linije



Слика9.38: Simbolizacija objekata nasumičnim trougaonim talasima

union

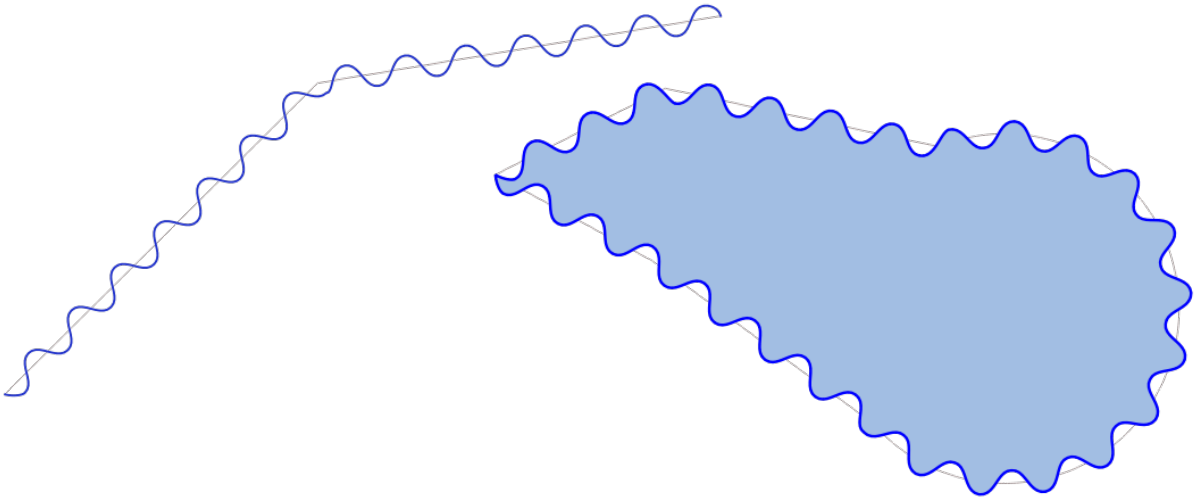
Vraća geometriju koja predstavlja uniju skupova tačaka geometrija.

Sintaksa	union(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `geom_to_wkt( union( make_point(4, 4), make_point(5, 5) ) )` → 'MULTIPOINT(4 4, 5 5)'

wave

Konstruiše zaobljene (sinusoidne) talase duž granice geometrije.

Sintaksa	wave(geometry, wavelength, amplitude, [strict:=false]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **wavelength** - talasna dužina sinusoidnog talasnog oblika • **amplitude** - amplituda sinusoidnog talasnog oblika • **strict** - Podrazumevano se argument wavelength tretira kao „maksimalna talasna dužina“, pri čemu se stvarna talasna dužina dinamički prilagođava tako da se duž granica geometrije napravi tačan broj talasa. Ako je argument strict postavljen na true, talasna dužina će se koristiti tačno, a za poslednji talasni oblik može se koristiti nepotpuna šara.
Primeri	• `wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Sinusoidni talasi talasne dužine 3 i amplitude 1 duž linije

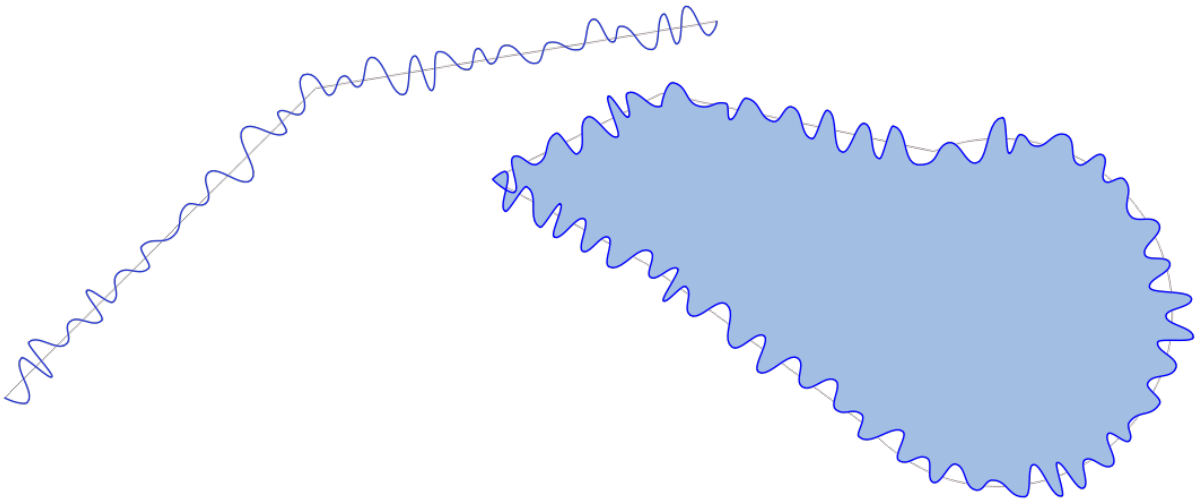


Слика9.39: Simbolizacija objekata talasima

wave_randomized

Konstruiše nasumične zakrivljene (sinusoidne) talase duž granice geometrije.

Sintaksa	wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed:=0]) [] označava opcione argumente
Argumenti	• **geometry** - geometrija • **min_wavelength** - minimum wavelength of waves • **max_wavelength** - maksimalna talasna dužina talasa • **min_amplitude** - minimalna amplituda talasa • **max_amplitude** - maksimalna amplituda talasa • **seed** - zadaje slučajno seme za generisanje talasa. Ako je seme 0, generisaće se potpuno slučajan skup talasa.
Primeri	• `wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Nasumično dimenzionisani zakrivljeni talasi sa talasnim dužinama između 2 i 3 i amplitudama između 0.1 i 0.2 duž linije

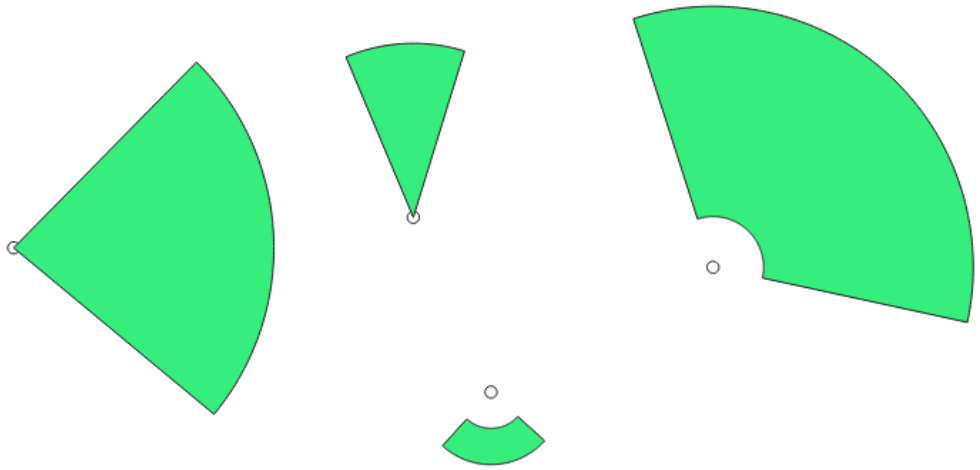


Слика9.40: Simbolizacija objekata nasumičnim talasima

wedge_buffer

Vraća bafer u obliku isečka koji polazi od tačkaste geometrije.

Sintaksa	wedge_buffer(center, azimuth, width, outer_radius, [inner_radius:=0.0]) [] označava opcione argumente
Argumenti	• **center** - centralna tačka (koordinatni početak) bafera. Mora biti tačkasta geometrija. • **azimuth** - ugao (u stepenima) ka kojem je usmerena sredina isečka. • **width** - širina bafera (u stepenima). Imajte u vidu da će se isečak protezati do polovine ugaone širine sa svake strane pravca azimuta. • **outer_radius** - spoljni poluprečnik za bafere • **inner_radius** - opcioni unutrašnji poluprečnik za bafere
Primeri	• `wedge_buffer(center:=geom_from_wkt('POINT(1 2)'),azimuth:=90, width:=180,outer_radius:=1)` → Bafer u obliku isečka sa centrom u tački (1,2), okrenut ka istoku, širine 180 stepeni i spoljnog poluprečnika 1.



Слика9.41: Baferovanje objekata u obliku isečka

Više informacija: algoritam *Napravi klinaste bafere*

within

Proverava da li je jedna geometrija unutar druge. Vraća TRUE ako je geometry1 potpuno unutar geometry2.

Sintaksa	within(geometry1, geometry2)
Argumenti	• **geometry1** - geometrija • **geometry2** - geometrija
Primeri	• `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

Više informacija: *overlay_within*

$x

Vraća x koordinatu trenutnog tačkastog objekta. Ako je objekat višetačkasti, vratiće se x koordinata prve tačke. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske funkcije x() sa promenljivom @geometry.**

Sintaksa	$x
Primeri	• `$x` → 42

Više informacija: *x*

x

Vraća x koordinatu tačkaste geometrije, ili x koordinatu centroida za netačkastu geometriju.

Sintaksa	x(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `x( geom_from_wkt( 'POINT(2 5)' ) )` → 2 • `x( @geometry )` → x koordinata centroida trenutnog objekta

$x_at

Preuzima x koordinatu geometrije trenutnog objekta. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske funkcije x_at sa promenljivom @geometry.**

Sintaksa	$x_at(vertex)
Argumenti	• **vertex** - indeks temena trenutne geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `$x_at(1)` → 5

Više informacija: *x_at*

x_at

Preuzima x koordinatu geometrije.

Sintaksa	x_at(geometry, vertex)
Argumenti	• **geometry** - objekat geometrije • **vertex** - indeks temena geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `x_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 4

x_max

Vraća maksimalnu x koordinatu geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	x_max(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `x_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 4

x_min

Vraća minimalnu x koordinatu geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	x_min(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `x_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 2

$y

Vraća y koordinatu trenutnog tačkastog objekta. Ako je objekat višetačkasti, vratiće se y koordinata prve tačke. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske funkcije y() sa promenljivom @geometry.**

Sintaksa	$y
Primeri	• `$y` → 42

Više informacija: *y*

y

Vraća y koordinatu tačkaste geometrije, ili y koordinatu centroida za netačkastu geometriju.

Sintaksa	y(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `y( geom_from_wkt( 'POINT(2 5)' ) )` → 5 • `y( @geometry )` → y koordinata centroida trenutnog objekta

$y_at

Preuzima y koordinatu geometrije trenutnog objekta. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske funkcije y_at sa promenljivom @geometry.**

Sintaksa	$y_at(vertex)
Argumenti	• **vertex** - indeks temena trenutne geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `$y_at(1)` → 2

Više informacija: *y_at*

y_at

Preuzima y koordinatu geometrije.

Sintaksa	y_at(geometry, vertex)
Argumenti	• **geometry** - objekat geometrije • **vertex** - indeks temena geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `y_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 5

y_max

Vraća maksimalnu y koordinatu geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	y_max(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `y_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 8

y_min

Vraća minimalnu y koordinatu geometrije. Izračunavanja su u prostornom referentnom sistemu te geometrije.

Sintaksa	y_min(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `y_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 5

$z

Vraća z vrednost trenutnog tačkastog objekta ako je 3D. Ako je objekat višetačkasti, vratiće se z vrednost prve tačke. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske funkcije z() sa promenljivom @geometry.**

Sintaksa	$z
Primeri	• `$z` → 123

z

Vraća z koordinatu tačkaste geometrije, ili NULL ako geometrija nema z vrednost.

Sintaksa	z(geometry)
Argumenti	• **geometry** - tačkasta geometrija
Primeri	• `z( geom_from_wkt( 'POINTZ(2 5 7)' ) )` → 7

z_at

Preuzima z koordinatu geometrije, ili NULL ako geometrija nema z vrednost.

Sintaksa	z_at(geometry, vertex)
Argumenti	• **geometry** - objekat geometrije • **vertex** - indeks temena geometrije (indeksi počinju od 0; negativne vrednosti se primenjuju od poslednjeg indeksa, počev od -1)
Primeri	• `z_at(geom_from_wkt('LineStringZ(0 0 0, 10 10 5, 10 10 0)'), 1)` → 5

z_max

Vraća maksimalnu z koordinatu geometrije, ili NULL ako geometrija nema z vrednost.

Sintaksa	z_max(geometry)
Argumenti	• **geometry** - geometrija sa z koordinatom
Primeri	• `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

Vraća minimalnu z koordinatu geometrije, ili NULL ako geometrija nema z vrednost.

Sintaksa	z_min(geometry)
Argumenti	• **geometry** - geometrija sa z koordinatom
Primeri	• `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 Funkcije za prikaz

Ova grupa sadrži funkcije za rad sa svojstvima stavki štampanog prikaza.

item_variables

Vraća mapu promenljivih iz stavke prikaza unutar ovog štampanog prikaza.

Sintaksa	item_variables(id)
Argumenti	• **id** - ID stavke prikaza
Primeri	• `map_get( item_variables('Map 0'), 'map_scale')` → razmera stavke 'Map 0' u trenutnom štampanom prikazu

Više informacija: Lista podrazumevanih *promenljivih*

map_credits

Vraća listu niski sa zaslugama (pravima korišćenja) za slojeve prikazane u prikazu ili u određenoj stavci karte u prikazu.

Sintaksa	map_credits([id], [include_layer_names:=false], [layer_name_separator:=': ']) [] označava opcione argumente
Argumenti	• **id** - ID stavke karte. Ako nije zadat, koristiće se slojevi sa svih karata u prikazu. • **include_layer_names** - Postavite na true da biste uključili nazive slojeva ispred njihovih niski sa zaslugama • **layer_name_separator** - Niska koja se umeće između naziva slojeva i njihovih niski sa zaslugama, ako je include_layer_names true
Primeri	• `array_to_string( map_credits() )` → lista zasluga slojeva razdvojena zarezima za sve slojeve prikazane u svim stavkama karte u prikazu, npr. 'CC-BY-NC, CC-BY-SA' • `array_to_string( map_credits( 'Main Map' ) )` → lista zasluga slojeva razdvojena zarezima za slojeve prikazane u stavci prikaza 'Main Map', npr. 'CC-BY-NC, CC-BY-SA' • `array_to_string( map_credits( 'Main Map', include_layer_names := true, layer_name_separator := ': ' ) )` → lista naziva slojeva i njihovih zasluga razdvojena zarezima, za slojeve prikazane u stavci prikaza 'Main Map', npr. 'Railway lines: CC-BY-NC, Basemap: CC-BY-SA'

Ova funkcija zahteva da *svojstva metapodataka* slojeva budu popunjena.

9.2.16 Slojevi karte

Ova grupa sadrži listu slojeva dostupnih u trenutnom projektu i, za svaki sloj, njihova polja (sačuvana u skupu podataka, virtuelna ili pomoćna, kao i ona iz spojeva). Sa poljima se radi na isti način kao što je opisano u *Polja i vrednosti*, osim što će dvoklik dodati naziv u izraz kao nisku (pod jednostrukim navodnicima), a ne kao referencu na polje, budući da ne pripadaju aktivnom sloju. Ovo je zgodan način za pisanje izraza koji se odnose na različite slojeve, kao pri *agregatnim*, *atributskim* ili *prostornim* upitima.

Takođe pruža neke zgodne funkcije za rad sa slojevima.

decode_uri

Uzima sloj i dekodira uri osnovnog provajdera podataka. Zavisi od provajdera podataka koji su podaci dostupni.

Sintaksa	decode_uri(layer, [part]) [] označava opcione argumente
Argumenti	• **layer** - Sloj za koji treba dekodirati uri. • **part** - Deo uri-ja koji se vraća. Ako nije zadat, vratiće se mapa sa svim delovima uri-ja.
Primeri	• `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

Vraća odgovarajuće svojstvo sloja ili vrednost metapodatka.

Sintaksa	layer_property(layer, property, [translate:=true]) [] označava opcione argumente
Argumenti	• **layer** - niska koja predstavlja naziv sloja ili ID sloja • **property** - niska koja odgovara svojstvu koje se vraća. Važeće opcije su: – name: naziv sloja – id: ID sloja – title: niska naslova iz metapodataka – abstract: niska apstrakta iz metapodataka – keywords: ključne reči iz metapodataka – data_url: URL iz metapodataka – attribution: niska atribucije iz metapodataka – attribution_url: URL atribucije iz metapodataka – source: izvor sloja – min_scale: minimalna razmera prikaza za sloj – max_scale: maksimalna razmera prikaza za sloj – is_editable: da li je sloj u režimu uređivanja – crs: CRS sloja – crs_definition: puna definicija CRS-a sloja – crs_description: opis CRS-a sloja – crs_ellipsoid: akronim elipsoida CRS-a sloja – extent: obuhvat sloja (kao objekat geometrije) – distance_units: jedinice rastojanja sloja – type: tip sloja, npr. Vector ili Raster – storage_type: format skladištenja (samo vektorski slojevi) – geometry_type: tip geometrije, npr. Point (samo vektorski slojevi) – feature_count: približan broj objekata na sloju (samo vektorski slojevi) – path: Putanja datoteke do izvora podataka sloja. Dostupno samo za slojeve zasnovane na datotekama. • **translate** - da li se vraća prevedena niska. Trenutno utiče samo na svojstvo 'type'. Podrazumevano se vraćaju prevedene niske. Postavljanje na false onemogućava prevođenje i uvek vraća engleske niske.
Primeri	• `layer_property('streets','title')` → 'Basemap Streets' • `layer_property('airports','feature_count')` → 120 • `layer_property('landsat','crs')` → 'EPSG:4326'

Više informacija: svojstva *vektorskog*, *rasterskog* i *mrežnog* sloja

load_layer

Učitava sloj po URI-ju izvora i nazivu provajdera.

Sintaksa	load_layer(uri, provider)
Argumenti	• **uri** - niska URI-ja izvora sloja • **provider** - naziv provajdera podataka sloja
Primeri	• `layer_property(load_layer('c:/data/roads.shp', 'ogr'), 'feature_count')` → broj objekata sa vektorskog sloja c:/data/roads.shp

9.2.17 Funkcije za mape

Ova grupa sadrži funkcije za pravljenje ključeva i vrednosti struktura podataka tipa mapa (poznatih i kao rečnici, parovi ključ-vrednost ili asocijativni nizovi) i rad sa njima. Za razliku od *strukture podataka lista*, gde je redosled vrednosti bitan, redosled parova ključ-vrednost u objektu mape nije relevantan, a vrednosti se identifikuju svojim ključevima.

from_json

Učitava nisku u JSON formatu.

Sintaksa	from_json(string)
Argumenti	• **string** - JSON niska
Primeri	• `from_json('{"1":"one","2":"two"}')` → { '1': 'one', '2': 'two' } • `from_json('[1,2,3]')` → [1,2,3]

hstore_to_map

Pravi mapu od niske u hstore formatu.

Sintaksa	hstore_to_map(string)
Argumenti	• **string** - ulazna niska
Primeri	• `hstore_to_map('qgis=>rocks')` → { 'qgis': 'rocks' }

map

Vraća mapu koja sadrži sve ključeve i vrednosti prosleđene kao parove parametara.

Sintaksa	map(key1, value1, key2, value2, …)
Argumenti	• **key** - ključ (niska) • **value** - vrednost
Primeri	• `map('1','one','2', 'two')` → { '1': 'one', '2': 'two' } • `map('1','one','2', 'two')['1']` → 'one'

map_akeys

Vraća sve ključeve mape kao niz.

Sintaksa	map_akeys(map)
Argumenti	• **map** - mapa
Primeri	• `map_akeys(map('1','one','2','two'))` → ['1', '2']

map_avals

Vraća sve vrednosti mape kao niz.

Sintaksa	map_avals(map)
Argumenti	• **map** - mapa
Primeri	• `map_avals(map('1','one','2','two'))` → ['one', 'two']

map_concat

Vraća mapu koja sadrži sve stavke zadatih mapa. Ako dve mape sadrže isti ključ, uzima se vrednost druge mape.

Sintaksa	map_concat(map1, map2, …)
Argumenti	• **map** - mapa
Primeri	• `map_concat(map('1','one', '2','overridden'),map('2','two', '3', 'three'))` → { '1': 'one', '2': 'two', '3': 'three' }

map_delete

Vraća mapu sa obrisanim zadatim ključem i njegovom odgovarajućom vrednošću.

Sintaksa	map_delete(map, key)
Argumenti	• **map** - mapa • **key** - ključ koji se briše
Primeri	• `map_delete(map('1','one','2','two'),'2')` → { '1': 'one' }

map_exist

Vraća TRUE ako zadati ključ postoji u mapi.

Sintaksa	map_exist(map, key)
Argumenti	• **map** - mapa • **key** - ključ koji se traži
Primeri	• `map_exist(map('1','one','2','two'),'3')` → FALSE

map_get

Vraća vrednost mape na osnovu njenog ključa. Vraća NULL ako ključ ne postoji.

Sintaksa	map_get(map, key)
Argumenti	• **map** - mapa • **key** - ključ koji se traži
Primeri	• `map_get(map('1','one','2','two'),'2')` → 'two' • `map_get( item_variables('Map 0'), 'map_scale')` → razmera stavke 'Map 0' (ako postoji) u trenutnom štampanom prikazu

Путоказ

Za dobijanje vrednosti iz mape možete koristiti i *operator indeksa ([])*.

map_insert

Vraća mapu sa dodatim parom ključ/vrednost. Ako ključ već postoji, njegova vrednost se prepisuje.

Sintaksa	map_insert(map, key, value)
Argumenti	• **map** - mapa • **key** - ključ koji se dodaje • **value** - vrednost koja se dodaje
Primeri	• `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

Vraća mapu sa svim ključevima kojima je dodat zadati prefiks.

Sintaksa	map_prefix_keys(map, prefix)
Argumenti	• **map** - mapa • **prefix** - niska
Primeri	• `map_prefix_keys(map('1','one','2','two'), 'prefix-')` → { 'prefix-1': 'one', 'prefix-2': 'two' }

map_to_hstore

Spaja elemente mape u nisku u hstore formatu.

Sintaksa	map_to_hstore(map)
Argumenti	• **map** - ulazna mapa
Primeri	• `map_to_hstore(map('1','one','2','two'))` → '„1"=>"one"','„2"=>"two"'

map_to_html_dl

Spaja elemente mape u nisku HTML liste definicija.

Sintaksa	map_to_html_dl(map)
Argumenti	• **map** - ulazna mapa
Primeri	• `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

Spaja elemente mape u nisku HTML tabele.

Sintaksa	map_to_html_table(map)
Argumenti	• **map** - ulazna mapa
Primeri	• `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></tbod

to_json

Pravi nisku u JSON formatu od mape, niza ili druge vrednosti.

Sintaksa	to_json(value)
Argumenti	• **value** - Ulazna vrednost
Primeri	• `to_json(map('1','one','2','two'))` → {„1”:”one”,”2”:”two”} • `to_json(array(1,2,3))` → [1,2,3]

url_encode

Vraća URL-kodiranu nisku iz mape. Pretvara sve znakove u njihov ispravno kodiran oblik, dajući potpuno usaglašenu nisku upita.

Imajte u vidu da se znak plus ’+’ ne pretvara.

Sintaksa	url_encode(map)
Argumenti	• **map** - mapa.
Primeri	• `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 Matematičke funkcije

Ova grupa sadrži matematičke funkcije (npr. kvadratni koren, sin i cos).

abs

Vraća apsolutnu vrednost broja.

Sintaksa	abs(value)
Argumenti	• **value** - broj
Primeri	• `abs(-2)` → 2

acos

Vraća arkus kosinus vrednosti u radijanima.

Sintaksa	acos(value)
Argumenti	• **value** - kosinus ugla u radijanima
Primeri	• `acos(0.5)` → 1.0471975511966

asin

Vraća arkus sinus vrednosti u radijanima.

Sintaksa	asin(value)
Argumenti	• **value** - sinus ugla u radijanima
Primeri	• `asin(1.0)` → 1.5707963267949

atan

Vraća arkus tangens vrednosti u radijanima.

Sintaksa	atan(value)
Argumenti	• **value** - tangens ugla u radijanima
Primeri	• `atan(0.5)` → 0.463647609000806

atan2

Vraća arkus tangens od dy/dx koristeći znake dva argumenta za određivanje kvadranta rezultata.

Sintaksa	atan2(dy, dx)
Argumenti	• **dy** - razlika y koordinate • **dx** - razlika x koordinate
Primeri	• `atan2(1.0, 1.732)` → 0.523611477769969

ceil

Zaokružuje broj naviše.

Sintaksa	ceil(value)
Argumenti	• **value** - broj
Primeri	• `ceil(4.9)` → 5 • `ceil(-4.9)` → -4

clamp

Ograničava ulaznu vrednost na zadati opseg.

Sintaksa	clamp(minimum, input, maximum)
Argumenti	• **minimum** - najmanja vrednost koju *input* sme da uzme. • **input** - vrednost koja će biti ograničena na opseg zadat sa *minimum* i *maximum* • **maximum** - najveća vrednost koju *input* sme da uzme
Primeri	• `clamp(1,5,10)` → 5 *input* je između 1 i 10 pa se vraća nepromenjen • `clamp(1,0,10)` → 1 *input* je manji od minimalne vrednosti 1, pa funkcija vraća 1 • `clamp(1,11,10)` → 10 *input* je veći od maksimalne vrednosti 10, pa funkcija vraća 10

cos

Vraća kosinus ugla.

Sintaksa	cos(angle)
Argumenti	• **angle** - ugao u radijanima
Primeri	• `cos(1.571)` → 0.000796326710733263

degrees

Pretvara iz radijana u stepene.

Sintaksa	degrees(radians)
Argumenti	• **radians** - numerička vrednost
Primeri	• `degrees(3.14159)` → 180 • `degrees(1)` → 57.2958

exp

Vraća eksponent vrednosti.

Sintaksa	exp(value)
Argumenti	• **value** - broj čiji se eksponent vraća
Primeri	• `exp(1.0)` → 2.71828182845905

floor

Zaokružuje broj naniže.

Sintaksa	floor(value)
Argumenti	• **value** - broj
Primeri	• `floor(4.9)` → 4 • `floor(-4.9)` → -5

ln

Vraća prirodni logaritam vrednosti.

Sintaksa	ln(value)
Argumenti	• **value** - numerička vrednost
Primeri	• `ln(1)` → 0 • `ln(2.7182818284590452354)` → 1

log

Vraća vrednost logaritma prosleđene vrednosti i osnove.

Sintaksa	log(base, value)
Argumenti	• **base** - bilo koji pozitivan broj • **value** - bilo koji pozitivan broj
Primeri	• `log(2, 32)` → 5 • `log(0.5, 32)` → -5

log10

Vraća vrednost logaritma sa osnovom 10 prosleđenog izraza.

Sintaksa	log10(value)
Argumenti	• **value** - bilo koji pozitivan broj
Primeri	• `log10(1)` → 0 • `log10(100)` → 2

max

Vraća najveću vrednost u skupu vrednosti.

Sintaksa	max(value1, value2, …)
Argumenti	• **value** - broj
Primeri	• `max(2,10.2,5.5)` → 10.2 • `max(20.5,NULL,6.2)` → 20.5

min

Vraća najmanju vrednost u skupu vrednosti.

Sintaksa	min(value1, value2, …)
Argumenti	• **value** - broj
Primeri	• `min(20.5,10,6.2)` → 6.2 • `min(2,-10.3,NULL)` → -10.3

pi

Vraća vrednost broja pi za izračunavanja.

Sintaksa	pi()
Primeri	• `pi()` → 3.14159265358979

radians

Pretvara iz stepeni u radijane.

Sintaksa	radians(degrees)
Argumenti	• **degrees** - numerička vrednost
Primeri	• `radians(180)` → 3.14159 • `radians(57.2958)` → 1

rand

Vraća slučajan ceo broj u opsegu zadatom argumentima minimum i maximum (uključivo). Ako je zadato seme, vraćena vrednost će uvek biti ista, u zavisnosti od semena.

Sintaksa	rand(min, max, [seed:=NULL]) [] označava opcione argumente
Argumenti	• **min** - ceo broj koji predstavlja najmanji željeni mogući slučajan broj • **max** - ceo broj koji predstavlja najveći željeni mogući slučajan broj • **seed** - bilo koja vrednost koja se koristi kao seme
Primeri	• `rand(1, 10)` → 8

randf

Vraća slučajan broj u pokretnom zarezu u opsegu zadatom argumentima minimum i maximum (uključivo). Ako je zadato seme, vraćena vrednost će uvek biti ista, u zavisnosti od semena.

Sintaksa	randf([min:=0.0], [max:=1.0], [seed:=NULL]) [] označava opcione argumente
Argumenti	• **min** - broj u pokretnom zarezu koji predstavlja najmanji željeni mogući slučajan broj • **max** - broj u pokretnom zarezu koji predstavlja najveći željeni mogući slučajan broj • **seed** - bilo koja vrednost koja se koristi kao seme
Primeri	• `randf(1, 10)` → 4.59258286403147

round

Zaokružuje broj na zadati broj decimalnih mesta.

Sintaksa	round(value, [places:=0]) [] označava opcione argumente
Argumenti	• **value** - decimalan broj koji se zaokružuje • **places** - Opcioni ceo broj koji predstavlja broj mesta na koji se zaokružuju decimale. Može biti negativan.
Primeri	• `round(1234.567, 2)` → 1234.57 • `round(1234.567)` → 1235 • `round(1234.567, -1)` → 1230

scale_exponential

Preslikava zadatu vrednost iz ulaznog domena u izlazni opseg pomoću eksponencijalne krive. Ova funkcija se može koristiti za blago uvođenje ili izvođenje vrednosti u zadati izlazni opseg.

Sintaksa	scale_exponential(value, domain_min, domain_max, range_min, range_max, exponent)
Argumenti	• **value** - Vrednost u ulaznom domenu. Funkcija će vratiti odgovarajuću skaliranu vrednost u izlaznom opsegu. • **domain_min** - Zadaje minimalnu vrednost u ulaznom domenu, najmanju vrednost koju ulazna vrednost sme da uzme. • **domain_max** - Zadaje maksimalnu vrednost u ulaznom domenu, najveću vrednost koju ulazna vrednost sme da uzme. • **range_min** - Zadaje minimalnu vrednost u izlaznom opsegu, najmanju vrednost koju funkcija sme da vrati. • **range_max** - Zadaje maksimalnu vrednost u izlaznom opsegu, najveću vrednost koju funkcija sme da vrati. • **exponent** - Pozitivna vrednost (veća od 0) koja određuje način preslikavanja ulaznih vrednosti u izlazni opseg. Veliki eksponenti izazvaće 'blago uvođenje' izlaznih vrednosti, koje počinje sporo, a zatim se ubrzava kako se ulazne vrednosti približavaju maksimumu domena. Manji eksponenti (manji od 1) izazvaće 'blago izvođenje', gde preslikavanje počinje brzo, ali usporava kako se približava maksimumu domena.
Primeri	• `scale_exponential(5,0,10,0,100,2)` → 3.030 blago uvođenje, uz eksponent 2 • `scale_exponential(3,0,10,0,100,0.5)` → 87.585 blago izvođenje, uz eksponent 0.5

scale_linear

Preslikava zadatu vrednost iz ulaznog domena u izlazni opseg pomoću linearne interpolacije.

Sintaksa	scale_linear(value, domain_min, domain_max, range_min, range_max)
Argumenti	• **value** - Vrednost u ulaznom domenu. Funkcija će vratiti odgovarajuću skaliranu vrednost u izlaznom opsegu. • **domain_min** - Zadaje minimalnu vrednost u ulaznom domenu, najmanju vrednost koju ulazna vrednost sme da uzme. • **domain_max** - Zadaje maksimalnu vrednost u ulaznom domenu, najveću vrednost koju ulazna vrednost sme da uzme. • **range_min** - Zadaje minimalnu vrednost u izlaznom opsegu, najmanju vrednost koju funkcija sme da vrati. • **range_max** - Zadaje maksimalnu vrednost u izlaznom opsegu, najveću vrednost koju funkcija sme da vrati.
Primeri	• `scale_linear(5,0,10,0,100)` → 50 • `scale_linear(0.2,0,1,0,360)` → 72 skaliranje vrednosti između 0 i 1 u ugao između 0 i 360 • `scale_linear(1500,1000,10000,9,20)` → 9.6111111 skaliranje populacije koja varira između 1000 i 10000 u veličinu fonta između 9 i 20

scale_polynomial

Preslikava zadatu vrednost iz ulaznog domena u izlazni opseg pomoću polinomijalne krive. Ova funkcija se može koristiti za blago uvođenje ili izvođenje vrednosti u zadati izlazni opseg.

Sintaksa	scale_polynomial(value, domain_min, domain_max, range_min, range_max, exponent)
Argumenti	• **value** - Vrednost u ulaznom domenu. Funkcija će vratiti odgovarajuću skaliranu vrednost u izlaznom opsegu. • **domain_min** - Zadaje minimalnu vrednost u ulaznom domenu, najmanju vrednost koju ulazna vrednost sme da uzme. • **domain_max** - Zadaje maksimalnu vrednost u ulaznom domenu, najveću vrednost koju ulazna vrednost sme da uzme. • **range_min** - Zadaje minimalnu vrednost u izlaznom opsegu, najmanju vrednost koju funkcija sme da vrati. • **range_max** - Zadaje maksimalnu vrednost u izlaznom opsegu, najveću vrednost koju funkcija sme da vrati. • **exponent** - Pozitivna vrednost (veća od 0) koja određuje način preslikavanja ulaznih vrednosti u izlazni opseg. Veliki eksponenti izazvaće 'blago uvođenje' izlaznih vrednosti, koje počinje sporo, a zatim se ubrzava kako se ulazne vrednosti približavaju maksimumu domena. Manji eksponenti (manji od 1) izazvaće 'blago izvođenje', gde preslikavanje počinje brzo, ali usporava kako se približava maksimumu domena.
Primeri	• `scale_polynomial(5,0,10,0,100,2)` → 25 blago uvođenje, uz eksponent 2 • `scale_polynomial(3,0,10,0,100,0.5)` → 54.772 blago izvođenje, uz eksponent 0.5

sin

Vraća sinus ugla.

Sintaksa	sin(angle)
Argumenti	• **angle** - ugao u radijanima
Primeri	• `sin(1.571)` → 0.999999682931835

sqrt

Vraća kvadratni koren vrednosti.

Sintaksa	sqrt(value)
Argumenti	• **value** - broj
Primeri	• `sqrt(9)` → 3

tan

Vraća tangens ugla.

Sintaksa	tan(angle)
Argumenti	• **angle** - ugao u radijanima
Primeri	• `tan(1.0)` → 1.5574077246549

9.2.19 Funkcije za mreže

Ova grupa sadrži funkcije koje računaju ili vraćaju vrednosti vezane za mreže.

$face_area

Vraća površinu trenutnog lica mreže. Površina koju računa ova funkcija poštuje i podešavanje elipsoida projekta i podešavanje jedinice površine. Na primer, ako je za projekat postavljen elipsoid, izračunata površina biće elipsoidna, a ako elipsoid nije postavljen, izračunata površina biće planimetrijska.

Sintaksa	$face_area
Primeri	• `$face_area` → 42

$face_index

Vraća indeks trenutnog lica mreže.

Sintaksa	$face_index
Primeri	• `$face_index` → 4581

$vertex_as_point

Vraća trenutno teme kao tačkastu geometriju.

Sintaksa	$vertex_as_point
Primeri	• `geom_to_wkt( $vertex_as_point )` → 'POINT(800 1500 41)'

$vertex_index

Vraća indeks trenutnog temena mreže.

Sintaksa	$vertex_index
Primeri	• `$vertex_index` → 9874

$vertex_x

Vraća X koordinatu trenutnog temena mreže.

Sintaksa	$vertex_x
Primeri	• `$vertex_x` → 42.12

$vertex_y

Vraća Y koordinatu trenutnog temena mreže.

Sintaksa	$vertex_y
Primeri	• `$vertex_y` → 12.24

$vertex_z

Vraća Z vrednost trenutnog temena mreže.

Sintaksa	$vertex_z
Primeri	• `$vertex_z` → 42

9.2.20 Operatori

Ova grupa sadrži operatore (npr. +, -, *). Imajte u vidu da za većinu matematičkih funkcija ispod, ako je jedan od ulaza NULL, rezultat je NULL.

%

Ostatak deljenja. Uzima znak deljenika.

Sintaksa	a % b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `9 % 2` → 1 • `9 % -2` → 1 • `-9 % 2` → -1 • `5 % NULL` → NULL

*

Množenje dve vrednosti

Sintaksa	a * b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 * 4` → 20 • `5 * NULL` → NULL

+

Sabiranje dve vrednosti. Ako je jedna od vrednosti NULL, rezultat će biti NULL.

Sintaksa	a + b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `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

Više informacija: *concat*, ||

-

Oduzimanje dve vrednosti. Ako je jedna od vrednosti NULL, rezultat će biti NULL.

Sintaksa	a - b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `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

/

Deljenje dve vrednosti

Sintaksa	a / b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 / 4` → 1.25 • `5 / NULL` → NULL

//

Celobrojno deljenje dve vrednosti

Sintaksa	a // b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `9 // 2` → 4

<

Poredi dve vrednosti i daje 1 ako je leva vrednost manja od desne.

Sintaksa	a < b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 < 4` → FALSE • `5 < 5` → FALSE • `4 < 5` → TRUE

<=

Poredi dve vrednosti i daje 1 ako je leva vrednost manja ili jednaka desnoj.

Sintaksa	a <= b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 <= 4` → FALSE • `5 <= 5` → TRUE • `4 <= 5` → TRUE

<>

Poredi dve vrednosti i daje 1 ako nisu jednake.

Sintaksa	a <> b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 <> 4` → TRUE • `4 <> 4` → FALSE • `5 <> NULL` → NULL • `NULL <> NULL` → NULL

=

Poredi dve vrednosti i daje 1 ako su jednake.

Sintaksa	a = b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 = 4` → FALSE • `4 = 4` → TRUE • `5 = NULL` → NULL • `NULL = NULL` → NULL

>

Poredi dve vrednosti i daje 1 ako je leva vrednost veća od desne.

Sintaksa	a > b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 > 4` → TRUE • `5 > 5` → FALSE • `4 > 5` → FALSE

>=

Poredi dve vrednosti i daje 1 ako je leva vrednost veća ili jednaka desnoj.

Sintaksa	a >= b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 >= 4` → TRUE • `5 >= 5` → TRUE • `4 >= 5` → FALSE

AND

Vraća TRUE kada su uslovi a i b tačni.

Sintaksa	a AND b
Argumenti	• **a** - uslov • **b** - uslov
Primeri	• `TRUE AND TRUE` → TRUE • `TRUE AND FALSE` → FALSE • `4 = 2+2 AND 1 = 1` → TRUE • `4 = 2+2 AND 1 = 2` → FALSE

BETWEEN

Vraća TRUE ako je vrednost unutar zadatog opsega. Opseg se smatra uključivim u odnosu na granice. Za proveru isključenja može se koristiti NOT BETWEEN.

Sintaksa	value BETWEEN lower_bound AND higher_bound
Argumenti	• **value** - vrednost koja se poredi sa opsegom. Može biti niska, broj ili datum. • **lower_bound AND higher_bound** - granice opsega
Primeri	• `'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

Белешка

value BETWEEN lower_bound AND higher_bound je isto što i „*value* >= *lower_bound AND value* <= *higher_bound*”.

Više informacija: *NOT BETWEEN*

ILIKE

Vraća TRUE ako se prvi parametar podudara sa zadatom šarom, bez razlikovanja velikih i malih slova. Umesto ILIKE može se koristiti LIKE da bi podudaranje razlikovalo velika i mala slova. Radi i sa brojevima.

Sintaksa	string/number ILIKE pattern
Argumenti	• **string/number** - niska koja se pretražuje • **pattern** - šara koja se traži, možete koristiti ’%’ kao džoker, ’_’ kao jedan znak i ’\\’ za izbegavanje ovih posebnih znakova.
Primeri	• `'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

Vraća TRUE ako se vrednost nalazi u listi vrednosti.

Sintaksa	a IN b
Argumenti	• **a** - vrednost • **b** - lista vrednosti
Primeri	• `'A' IN ('A','B')` → TRUE • `'A' IN ('C','B')` → FALSE

IS

Vraća TRUE ako je a isto što i b.

Sintaksa	a IS b
Argumenti	• **a** - bilo koja vrednost • **b** - bilo koja vrednost
Primeri	• `'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, ako vaša geometrija nije NULL

IS NOT

Vraća TRUE ako a nije isto što i b.

Sintaksa	a IS NOT b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `'a' IS NOT 'b'` → TRUE • `'a' IS NOT 'a'` → FALSE • `4 IS NOT 2+2` → FALSE

LIKE

Vraća TRUE ako se prvi parametar podudara sa zadatom šarom. Radi i sa brojevima.

Sintaksa	string/number LIKE pattern
Argumenti	• **string/number** - vrednost • **pattern** - šara sa kojom se poredi vrednost, možete koristiti '%' kao džoker, '_' kao jedan znak i '\\' za izbegavanje ovih posebnih znakova.
Primeri	• `'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

Negira uslov.

Sintaksa	NOT a
Argumenti	• **a** - uslov
Primeri	• `NOT 1` → FALSE • `NOT 0` → TRUE

NOT BETWEEN

Vraća TRUE ako vrednost nije unutar zadatog opsega. Opseg se smatra uključivim u odnosu na granice.

Sintaksa	value NOT BETWEEN lower_bound AND higher_bound
Argumenti	• **value** - vrednost koja se poredi sa opsegom. Može biti niska, broj ili datum. • **lower_bound AND higher_bound** - granice opsega
Primeri	• `'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

Белешка

value NOT BETWEEN lower_bound AND higher_bound je isto što i „*value* < *lower_bound OR value* > *higher_bound*”.

Više informacija: *BETWEEN*

OR

Vraća TRUE kada je uslov a ili b tačan.

Sintaksa	a OR b
Argumenti	• **a** - uslov • **b** - uslov
Primeri	• `4 = 2+2 OR 1 = 1` → TRUE • `4 = 2+2 OR 1 = 2` → TRUE • `4 = 2 OR 1 = 2` → FALSE

[]

Operator indeksa. Vraća element iz niza ili mape.

Sintaksa	[index]
Argumenti	• **index** - indeks niza ili vrednost ključa mape
Primeri	• `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

Više informacija: *array_get*, *map_get*

^

Stepenovanje dve vrednosti.

Sintaksa	a ^ b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `5 ^ 4` → 625 • `5 ^ NULL` → NULL

||

Spaja dve vrednosti u nisku.

Ako je jedna od vrednosti NULL, rezultat će biti NULL. Za drugačije ponašanje pogledajte funkciju CONCAT.

Sintaksa	a || b
Argumenti	• **a** - vrednost • **b** - vrednost
Primeri	• `'Here' || ' and ' || 'there'` → 'Here and there' • `'Nothing' || NULL` → NULL • `'Dia: ' || "Diameter"` → 'Dia: 25' • `1 || 2` → '12'

Više informacija: *concat*, *+*

~

Vrši podudaranje regularnog izraza nad vrednošću tipa niska. Znakovi obrnute kose crte moraju biti dvostruko izbegnuti (npr. „\\s" za podudaranje sa belim znakom).

Sintaksa	string ~ regex
Argumenti	• **string** - Vrednost tipa niska • **regex** - Regularni izraz. Kose crte moraju biti izbegnute, npr. \\d.
Primeri	• `'hello' ~ 'll'` → TRUE • `'hello' ~ '^ll'` → FALSE • `'hello' ~ 'llo$'` → TRUE • `'abc123' ~ '\\d+'` → TRUE

Više informacija: *regexp_match*

9.2.21 Funkcije za obradu

Ova grupa sadrži funkcije koje rade sa algoritmima obrade.

parameter

Vraća vrednost ulaznog parametra algoritma obrade.

Sintaksa	parameter(name)
Argumenti	• **name** - naziv odgovarajućeg ulaznog parametra
Primeri	• `parameter('BUFFER_SIZE')` → 5.6

9.2.22 Funkcije za rastere

Ova grupa sadrži funkcije za rad sa rasterskim slojem.

raster_attributes

Vraća mapu sa nazivima polja kao ključevima i vrednostima rasterske atributske tabele kao vrednostima, iz stavke atributske tabele koja odgovara zadatoj rasterskoj vrednosti.

Sintaksa	raster_attributes(layer, band, value)
Argumenti	• **layer** - naziv ili id rasterskog sloja • **band** - broj kanala za pretragu u pripadajućoj atributskoj tabeli. • **value** - rasterska vrednost
Primeri	• `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

Vraća statistiku sa rasterskog sloja.

Sintaksa	raster_statistic(layer, band, property)
Argumenti	• **layer** - niska koja predstavlja naziv rasterskog sloja ili ID sloja • **band** - ceo broj koji predstavlja broj kanala sa rasterskog sloja, počev od 1 • **property** - niska koja odgovara svojstvu koje se vraća. Važeće opcije su: – min: minimalna vrednost – max: maksimalna vrednost – avg: prosečna (srednja) vrednost – stdev: standardna devijacija vrednosti – range: raspon vrednosti (max - min) – sum: zbir svih vrednosti sa rastera
Primeri	• `raster_statistic('lc',1,'avg')` → Prosečna vrednost iz kanala 1 rasterskog sloja 'lc' • `raster_statistic('ac2010',3,'min')` → Minimalna vrednost iz kanala 3 rasterskog sloja 'ac2010'

raster_value

Vraća rastersku vrednost pronađenu u zadatoj tački.

Sintaksa	raster_value(layer, band, point)
Argumenti	• **layer** - naziv ili id rasterskog sloja • **band** - broj kanala iz kojeg se uzorkuje vrednost. • **point** - tačkasta geometrija (za višedelne geometrije sa više od jednog dela vratiće se NULL)
Primeri	• `raster_value('dem', 1, make_point(1,1))` → 25 • `raster_value('ndvi', 2, @layer_cursor_point)` → 25

9.2.23 Funkcije za zapise i atribute

Ova grupa sadrži funkcije koje rade sa identifikatorima zapisa.

attribute

Vraća atribut objekta.

Varijanta 1

Vraća vrednost atributa trenutnog objekta.

Sintaksa	attribute(attribute_name)
Argumenti	• **attribute_name** - naziv atributa koji se vraća
Primeri	• `attribute( 'name' )` → vrednost sačuvana u atributu 'name' za trenutni objekat

Varijanta 2

Omogućava zadavanje ciljnog objekta i naziva atributa.

Sintaksa	attribute(feature, attribute_name)
Argumenti	• **feature** - objekat • **attribute_name** - naziv atributa koji se vraća
Primeri	• `attribute( @atlas_feature, 'name' )` → vrednost sačuvana u atributu 'name' za trenutni objekat atlasa

attributes

Vraća mapu koja sadrži sve atribute objekta, sa nazivima polja kao ključevima mape.

Varijanta 1

Vraća mapu svih atributa trenutnog objekta.

Sintaksa	attributes()
Primeri	• `attributes()['name']` → vrednost sačuvana u atributu 'name' za trenutni objekat

Varijanta 2

Omogućava zadavanje ciljnog objekta.

Sintaksa	attributes(feature)
Argumenti	• **feature** - objekat
Primeri	• `attributes( @atlas_feature )['name']` → vrednost sačuvana u atributu 'name' za trenutni objekat atlasa

Više informacija: *Funkcije za mape*

$currentfeature

Vraća trenutni objekat koji se izračunava. Ovo se može koristiti sa funkcijom 'attribute' za izračunavanje vrednosti atributa trenutnog objekta. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske promenljive @feature.**

Sintaksa	$currentfeature
Primeri	• `attribute( $currentfeature, 'name' )` → vrednost sačuvana u atributu 'name' za trenutni objekat

display_expression

Vraća izraz prikaza za zadati objekat na sloju. Izraz se podrazumevano izračunava. Može se koristiti bez argumenata, sa jednim ili sa više, detalji su ispod.

Bez parametara

Ako se pozove bez parametara, funkcija izračunava izraz prikaza trenutnog objekta na trenutnom sloju.

Sintaksa	display_expression()
Primeri	• `display_expression()` → Izraz prikaza trenutnog objekta na trenutnom sloju.

Jedan parametar 'feature'

Ako se pozove samo sa parametrom 'feature', funkcija će izračunati zadati objekat sa trenutnog sloja.

Sintaksa	display_expression(feature)
Argumenti	• **feature** - Objekat koji treba izračunati.
Primeri	• `display_expression(@atlas_feature)` → Izraz prikaza trenutnog objekta atlasa.

Parametri layer i feature

Ako se funkcija pozove i sa slojem i sa objektom, izračunaće zadati objekat sa zadatog sloja.

Sintaksa	display_expression(layer, feature, [evaluate:=true]) [] označava opcione argumente
Argumenti	• **layer** - Sloj (ili njegov ID ili naziv) • **feature** - Objekat koji treba izračunati. • **evaluate** - Da li izraz mora biti izračunat. Ako je false, izraz će biti vraćen samo kao doslovna niska (koja se kasnije može izračunati funkcijom 'eval').
Primeri	• `display_expression( 'streets', get_feature_by_id('streets', 1))` → Izraz prikaza objekta sa ID-jem 1 na sloju 'streets'. • `display_expression('a_layer_id', @feature, 'False')` → Izraz prikaza zadatog objekta, bez izračunavanja.

feature_id

Vraća jedinstveni ID objekta, ili NULL ako objekat nije ispravan.

Sintaksa	feature_id(feature)
Argumenti	• **feature** - feature objekat
Primeri	• `feature_id( @feature )` → ID trenutnog objekta

Više informacija: *get_feature_by_id*

get_feature

Vraća prvi objekat sloja koji odgovara zadatoj vrednosti atributa.

Varijanta sa jednom vrednošću

Uz ID sloja zadaju se jedna kolona i vrednost.

Sintaksa	get_feature(layer, attribute, value)
Argumenti	• **layer** - naziv ili ID sloja • **attribute** - naziv atributa koji se koristi za podudaranje • **value** - vrednost atributa za podudaranje
Primeri	• `get_feature('streets','name','main st')` → prvi objekat pronađen na sloju „streets" sa vrednošću „main st" u polju „name"

Varijanta sa mapom

Uz ID sloja zadaje se mapa koja sadrži kolone (ključ) i njihove odgovarajuće vrednosti koje se koriste.

Sintaksa	get_feature(layer, attribute)
Argumenti	• **layer** - naziv ili ID sloja • **attribute** - Mapa koja sadrži parove kolone i vrednosti koji se koriste
Primeri	• `get_feature('streets',map('name','main st','lane_num','4'))` → prvi objekat pronađen na sloju „streets" sa vrednošću „main st" u polju „name" i vrednošću „4" u polju „lane_num"

get_feature_by_id

Vraća objekat sa zadatim id-jem na sloju.

Sintaksa	get_feature_by_id(layer, feature_id)
Argumenti	• **layer** - sloj, naziv sloja ili id sloja • **feature_id** - id objekta koji treba vratiti
Primeri	• `get_feature_by_id('streets', 1)` → objekat sa id-jem 1 na sloju „streets”

Više informacija: *feature_id*

$id

Vraća id objekta u trenutnom redu. **UPOZORENJE: Ova funkcija je zastarela. Preporučuje se korišćenje zamenske promenljive @id.**

Sintaksa	$id
Primeri	• `$id` → 42

Više informacija: *feature_id*, *get_feature_by_id*

is_attribute_valid

Vraća TRUE ako određeni atribut objekta ispunjava sva ograničenja.

Sintaksa	is_attribute_valid(attribute, [feature], [layer], [strength]) [] označava opcione argumente
Argumenti	• **attribute** - naziv atributa • **feature** - Objekat. Ako nije zadat, koristiće se trenutni objekat. • **layer** - Vektorski sloj. Ako nije zadat, koristiće se trenutni sloj. • **strength** - Postavite na 'hard' ili 'soft' da biste suzili na određeni tip ograničenja. Ako nije postavljeno, funkcija vraća FALSE ako ne prođe bilo tvrdo bilo meko ograničenje.
Primeri	• `is_attribute_valid('HECTARES')` → TRUE ako vrednost trenutnog objekta u polju „HECTARES” ispunjava sva ograničenja. • `is_attribute_valid('HOUSES',get_feature('my_layer', 'FID', 10), 'my_layer')` → FALSE ako vrednost u polju „HOUSES” objekta sa „FID”=10 na sloju 'my_layer' ne ispunjava sva ograničenja.

Više informacija: *Constraints*

is_feature_valid

Vraća TRUE ako objekat ispunjava sva ograničenja polja.

Sintaksa	is_feature_valid([feature], [layer], [strength]) [] označava opcione argumente
Argumenti	• **feature** - Objekat. Ako nije zadat, koristiće se trenutni objekat. • **layer** - Vektorski sloj. Ako nije zadat, koristiće se trenutni sloj. • **strength** - Postavite na 'hard' ili 'soft' da biste suzili na određeni tip ograničenja. Ako nije postavljeno, funkcija vraća FALSE ako ne prođe bilo tvrdo bilo meko ograničenje.
Primeri	• `is_feature_valid(strength:='hard')` → TRUE ako sva polja trenutnog objekta ispunjavaju svoja tvrda ograničenja. • `is_feature_valid(get_feature('my_layer', 'FID', 10), 'my_layer')` → FALSE ako sva polja objekta sa „FID”=10 na sloju 'my_layer' ne ispunjavaju sva ograničenja.

Više informacija: *Constraints*

is_selected

Vraća TRUE ako je objekat izabran. Može se koristiti bez argumenata, sa jednim ili sa dva, detalji su ispod.

Bez parametara

Ako se pozove bez parametara, funkcija vraća TRUE ako je trenutni objekat na trenutnom sloju izabran.

Sintaksa	is_selected()
Primeri	• `is_selected()` → TRUE ako je trenutni objekat na trenutnom sloju izabran.

Jedan parametar 'feature'

Ako se pozove samo sa parametrom 'feature', funkcija vraća TRUE ako je zadati objekat sa trenutnog sloja izabran.

Sintaksa	is_selected(feature)
Argumenti	• **feature** - Objekat za koji treba proveriti izbor.
Primeri	• `is_selected(@atlas_feature)` → TRUE ako je trenutni objekat atlasa izabran. • `is_selected(get_feature('streets', 'name', 'Main St.'))` → TRUE ako je jedinstveni objekat pod nazivom „Main St." na aktivnom sloju „streets" izabran. • `is_selected(get_feature_by_id('streets', 1))` → TRUE ako je objekat sa id-jem 1 na aktivnom sloju „streets" izabran.

Dva parametra

Ako se funkcija pozove i sa slojem i sa objektom, vratiće TRUE ako je zadati objekat sa zadatog sloja izabran.

Sintaksa	is_selected(layer, feature)
Argumenti	• **layer** - Sloj (njegov ID ili naziv) na kojem će se proveriti izbor. • **feature** - Objekat za koji treba proveriti izbor.
Primeri	• `is_selected( 'streets', get_feature('streets', 'name', "street_name"))` → TRUE ako je ulica trenutne zgrade izabrana (pod pretpostavkom da sloj zgrada ima polje pod nazivom 'street_name', a sloj 'streets' polje pod nazivom 'name' sa jedinstvenim vrednostima). • `is_selected( 'streets', get_feature_by_id('streets', 1))` → TRUE ako je objekat sa id-jem 1 na sloju „streets" izabran.

maptip

Vraća savet na karti za zadati objekat na sloju. Izraz se podrazumevano izračunava. Može se koristiti bez argumenata, sa jednim ili sa više, detalji su ispod.

Bez parametara

Ako se pozove bez parametara, funkcija izračunava savet na karti za trenutni objekat na trenutnom sloju.

Sintaksa	maptip()
Primeri	• `maptip()` → Savet na karti za trenutni objekat na trenutnom sloju.

Jedan parametar 'feature'

Ako se pozove samo sa parametrom 'feature', funkcija će izračunati zadati objekat sa trenutnog sloja.

Sintaksa	maptip(feature)
Argumenti	• **feature** - Objekat koji treba izračunati.
Primeri	• `maptip(@atlas_feature)` → Savet na karti za trenutni objekat atlasa.

Parametri layer i feature

Ako se funkcija pozove i sa slojem i sa objektom, izračunaće zadati objekat sa zadatog sloja.

Sintaksa	maptip(layer, feature, [evaluate:=true]) [] označava opcione argumente
Argumenti	• **layer** - Sloj (ili njegov ID ili naziv) • **feature** - Objekat koji treba izračunati. • **evaluate** - Da li izraz mora biti izračunat. Ako je false, izraz će biti vraćen samo kao doslovna niska (koja se kasnije može izračunati funkcijom 'eval_template').
Primeri	• `maptip('streets', get_feature_by_id('streets', 1))` → Savet na karti za objekat sa ID-jem 1 na sloju 'streets'. • `maptip('a_layer_id', @feature, 'False')` → Savet na karti za zadati objekat, bez izračunavanja.

num_selected

Vraća broj izabranih objekata na zadatom sloju. Podrazumevano radi na sloju na kojem se izraz izračunava.

Sintaksa	num_selected([layer:=current layer]) [] označava opcione argumente
Argumenti	• **layer** - Sloj (ili njegov id ili naziv) na kojem će se proveriti izbor.
Primeri	• `num_selected()` → Broj izabranih objekata na trenutnom sloju. • `num_selected('streets')` → Broj izabranih objekata na sloju streets

represent_attributes

Vraća mapu sa nazivima atributa kao ključevima i podešenim vrednostima predstavljanja kao vrednostima. Vrednost predstavljanja za atribute zavisi od podešenog tipa vidžeta za svaki atribut. Može se koristiti bez argumenata, sa jednim ili sa više, detalji su ispod.

Bez parametara

Ako se pozove bez parametara, funkcija vraća predstavljanje atributa trenutnog objekta na trenutnom sloju.

Sintaksa	represent_attributes()
Primeri	• `represent_attributes()` → Predstavljanje atributa trenutnog objekta.

Jedan parametar 'feature'

Ako se pozove samo sa parametrom 'feature', funkcija vraća predstavljanje atributa zadatog objekta sa trenutnog sloja.

Sintaksa	represent_attributes(feature)
Argumenti	• **feature** - Objekat koji treba izračunati.
Primeri	• `represent_attributes(@atlas_feature)` → Predstavljanje atributa zadatog objekta sa trenutnog sloja.

Parametri layer i feature

Ako se pozove sa parametrima 'layer' i 'feature', funkcija vraća predstavljanje atributa zadatog objekta sa zadatog sloja.

Sintaksa	represent_attributes(layer, feature)
Argumenti	• **layer** - Sloj (ili njegov ID ili naziv). • **feature** - Objekat koji treba izračunati.
Primeri	• `represent_attributes('atlas_layer', @atlas_feature)` → Predstavljanje atributa zadatog objekta sa zadatog sloja.

Više informacija: *represent_value*

represent_value

Vraća podešenu vrednost predstavljanja za vrednost polja. Zavisi od podešenog tipa vidžeta. Često je korisno za vidžete 'Value Map'.

Sintaksa	represent_value(value, [fieldName]) [] označava opcione argumente
Argumenti	• **value** - Vrednost koja treba da se razreši. Najverovatnije polje. • **fieldName** - Naziv polja za koje treba učitati konfiguraciju vidžeta.
Primeri	• `represent_value("field_with_value_map")` → Opis za vrednost • `represent_value('static value', 'field_name')` → Opis za statičku vrednost

Više informacija: *tipovi vidžeta*, *represent_attributes*

sqlite_fetch_and_increment

Upravlja automatski uvećavajućim vrednostima u sqlite bazama podataka.

SQlite podrazumevane vrednosti mogu se primeniti samo pri umetanju i ne mogu se unapred preuzeti.

Zbog toga je nemoguće dobiti uvećani primarni ključ preko AUTO_INCREMENT pre nego što se red kreira u bazi podataka. Napomena: kod postgresa ovo radi preko opcije *evaluate default values*.

Pri dodavanju novih objekata sa relacijama, veoma je zgodno moći već dodati decu za roditelja, dok je forma roditelja još otvorena, a nadređeni objekat još nije upisan.

Da bi se zaobišlo ovo ograničenje, ova funkcija se može koristiti za upravljanje vrednostima sekvence u zasebnoj tabeli u formatima zasnovanim na sqlite-u, kao što je gpkg.

Tabela sekvence biće filtrirana po id-ju sekvence (filter_attribute i filter_value), a trenutna vrednost id_field biće uvećana za 1 i vraćena.

Ako je za dodatne kolone potrebno zadati vrednosti, u tu svrhu se može koristiti mapa default_values.

Napomena

Ova funkcija menja ciljnu sqlite tabelu. Namenjena je za korišćenje sa konfiguracijama podrazumevanih vrednosti atributa.

Kada je parametar database sloj, a sloj je u režimu transakcije, vrednost će biti preuzeta samo jednom tokom trajanja transakcije, keširana i uvećana. Zbog toga nije bezbedno raditi nad istom bazom podataka iz više procesa paralelno.

Sintaksa	sqlite_fetch_and_increment(database, table, id_field, filter_attribute, filter_value, [default_values]) [] označava opcione argumente
Argumenti	• **database** - Putanja do sqlite datoteke ili geopackage sloja • **table** - Naziv tabele koja upravlja sekvencama • **id_field** - Naziv polja koje sadrži trenutnu vrednost • **filter_attribute** - Naziv polja koje sadrži jedinstveni identifikator ove sekvence. Mora imati UNIQUE indeks. • **filter_value** - Naziv sekvence koja se koristi. • **default_values** - Mapa sa podrazumevanim vrednostima za dodatne kolone u tabeli. Vrednosti moraju biti u potpunosti pod navodnicima. Funkcije su dozvoljene.
Primeri	• `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

Više informacija: *Data Sources Properties*, *Setting relations between multiple layers*

uuid

Generiše univerzalno jedinstveni identifikator (UUID) za svaki red pomoću Qt metoda QUuid::createUuid.

Sintaksa	uuid([format:='WithBraces']) [] označava opcione argumente
Argumenti	• **format** - Format u kojem će UUID biti formatiran. 'WithBraces', 'WithoutBraces' ili 'Id128'.
Primeri	• `uuid()` → '{0bd2f60f-f157-4a6d-96af-d4ba4cb366a1}' • `uuid('WithoutBraces')` → '0bd2f60f-f157-4a6d-96af-d4ba4cb366a1' • `uuid('Id128')` → '0bd2f60ff1574a6d96afd4ba4cb366a1'

9.2.24 Relacije

Ova grupa sadrži listu *relacija* dostupnih u trenutnom projektu, sa njihovim opisom. Omogućava brz pristup ID-ju relacije za pisanje izraza (npr. pomoću funkcije *relation_aggregate*) ili prilagođavanje forme.

9.2.25 Funkcije za senzore

Ova grupa sadrži funkcije za rad sa senzorima.

sensor_data

Vraća poslednju zabeleženu vrednost (ili vrednosti kao mapu za senzore koji prijavljuju više vrednosti) sa registrovanog senzora.

Sintaksa	sensor_data(name, [expiration]) [] označava opcione argumente
Argumenti	• **name** - naziv senzora • **expiration** - najveći dozvoljeni broj milisekundi od poslednje zabeležene vrednosti
Primeri	• `sensor_data('geiger_1')` → '2000'

9.2.26 Funkcije za niske

Ova grupa sadrži funkcije koje rade sa niskama (npr. koje zamenjuju, pretvaraju u velika slova).

ascii

Vraća unicode kod povezan sa prvim znakom niske.

Sintaksa	ascii(string)
Argumenti	• **string** - niska koja se pretvara u unicode kod
Primeri	• `ascii('Q')` → 81

char

Vraća znak povezan sa unicode kodom.

Sintaksa	char(code)
Argumenti	• **code** - broj unicode koda
Primeri	• `char(81)` → 'Q'

concat

Spaja više niski u jednu. NULL vrednosti se pretvaraju u prazne niske. Druge vrednosti (kao brojevi) pretvaraju se u niske.

Sintaksa	concat(string1, string2, …)
Argumenti	• **string** - vrednost tipa niska
Primeri	• `concat('sun', 'set')` → 'sunset' • `concat('a','b','c','d','e')` → 'abcde' • `concat('Anno ', 1984)` → 'Anno 1984' • `concat('The Wall', NULL)` → 'The Wall'

O spajanju polja

Niske ili vrednosti polja možete spajati i pomoću operatora `||` ili +, uz neke posebnosti:

- Operator + znači i sabiranje izraza, pa ako imate celobrojni operand (polje ili numeričku vrednost), ovo može biti podložno greškama i bolje je da koristite ostale:

```
'My feature id is: ' + "gid" => triggers an error as gid returns an integer
```

- Kada je bilo koji od argumenata NULL vrednost, i `||` i + vratiće NULL vrednost. Da biste vratili ostale argumente bez obzira na NULL vrednost, možete koristiti funkciju `concat` ili `array_to_string`:

```
'My feature id is: ' + NULL ==> NULL
'My feature id is: ' || NULL => NULL
concat('My feature id is: ', NULL) => 'My feature id is: '
array_to_string( array('My feature id is: ', NULL) ) => 'My feature id is: '
```

više informacija: *array_to_string*, ||, +

format

Formatira nisku pomoću zadatih argumenata.

Sintaksa	format(string, arg1, arg2, …)
Argumenti	• **string** - Niska sa čuvarima mesta %1, %2 itd. za argumente. Čuvari mesta se mogu ponavljati. Čuvar mesta sa najmanjim brojem zamenjuje se sa arg1, sledeći sa arg2 itd. • **arg** - bilo koji tip. Bilo koji broj argumenata.
Primeri	• `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

Formatira tip datuma ili nisku u prilagođeni format niske. Koristi Qt formate datuma/vremena. Pogledajte QDateTime::toString.

Sintaksa	format_date(datetime, format, [language]) [] označava opcione argumente
Argumenti	• **datetime** - vrednost datuma, vremena ili datetime • **format** - Šablon niske koji se koristi za formatiranje niske.

Expression	Izlaz
d	dan kao broj bez vodeće nule (1 do 31)
dd	dan kao broj sa vodećom nulom (01 do 31)
ddd	skraćeni lokalizovani naziv dana (npr. 'Mon' do 'Sun')
dddd	dugi lokalizovani naziv dana (npr. 'Monday' do 'Sunday')
M	mesec kao broj bez vodeće nule (1-12)
MM	mesec kao broj sa vodećom nulom (01-12)
MMM	skraćeni lokalizovani naziv meseca (npr. 'Jan' do 'Dec')
MMMM	dugi lokalizovani naziv meseca (npr. 'January' do 'December')
yy	godina kao dvocifren broj (00-99)
yyyy	godina kao četvorocifren broj

Ovi izrazi se mogu koristiti za vremenski deo niske formata:

Expression	Izlaz
h	sat bez vodeće nule (0 do 23 ili 1 do 12 ako je prikaz AM/PM)
hh	sat sa vodećom nulom (00 do 23 ili 01 do 12 ako je prikaz AM/PM)
H	sat bez vodeće nule (0 do 23, čak i uz AM/PM prikaz)
HH	sat sa vodećom nulom (00 do 23, čak i uz AM/PM prikaz)
m	minut bez vodeće nule (0 do 59)
mm	minut sa vodećom nulom (00 do 59)
s	sekunda bez vodeće nule (0 do 59)
ss	sekunda sa vodećom nulom (00 do 59)
z	milisekunde bez pratećih nula (0 do 999)
zzz	milisekunde sa pratećim nulama (000 do 999)
AP ili A	tumači se kao AM/PM vreme. *AP* mora biti ili 'AM' ili 'PM'.
ap ili a	Tumači se kao AM/PM vreme. *ap* mora biti ili 'am' ili 'pm'.

- **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za formatiranje datuma u prilagođenu nisku. Podrazumevano se koristi trenutni QGIS korisnički lokalitet.

Primeri	• `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', ako je trenutni lokalitet neka engleska varijanta • `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

Vraća broj formatiran sa graničnikom hiljada prema lokalitetu. Podrazumevano se koristi trenutni QGIS korisnički lokalitet. Takođe skraćuje decimalna mesta na zadati broj mesta.

Sintaksa	format_number(number, [places:=0], [language], [omit_group_separators:=false], [trim_trailing_zeroes:=false]) [] označava opcione argumente
Argumenti	• **number** - broj koji se formatira • **places** - ceo broj koji predstavlja broj decimalnih mesta na koji se niska skraćuje. • **language** - jezik (mala slova, dvoslovni ili troslovni ISO 639 kod jezika) koji se koristi za formatiranje broja u nisku. Podrazumevano se koristi trenutni QGIS korisnički lokalitet. • **omit_group_separators** - ako je postavljeno na true, graničnici grupa neće biti uključeni u nisku • **trim_trailing_zeroes** - ako je postavljeno na true, prateće nule iza decimalne tačke biće uklonjene iz niske
Primeri	• `format_number(10000000.332,2)` → '10,000,000.33' ako je npr. trenutni lokalitet neka engleska varijanta • `format_number(10000000.332,2,'fr')` → '10 000 000,33'

left

Vraća podnisku koja sadrži *n* krajnjih levih znakova niske.

Sintaksa	left(string, length)
Argumenti	• **string** - niska • **length** - ceo broj. Broj znakova sa leve strane niske koji se vraćaju.
Primeri	• `left('Hello World',5)` → 'Hello'

Više informacija: *right*

length

Vraća broj znakova u niski ili dužinu linijske geometrije.

Varijanta sa niskom

Vraća broj znakova u niski.

Sintaksa	length(string)
Argumenti	• **string** - niska čija se dužina broji
Primeri	• `length('hello')` → 5

Varijanta sa geometrijom

Računa dužinu objekta linijske geometrije. Izračunavanja su uvek planimetrijska u prostornom referentnom sistemu (SRS) te geometrije, a jedinice vraćene dužine odgovaraće jedinicama tog SRS-a. Ovo se razlikuje od izračunavanja koja vrši funkcija $length, koja izvodi elipsoidna izračunavanja na osnovu podešavanja elipsoida i jedinice rastojanja projekta.

Sintaksa	length(geometry)
Argumenti	• **geometry** - objekat linijske geometrije
Primeri	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

lower

Pretvara nisku u mala slova.

Sintaksa	lower(string)
Argumenti	• **string** - niska koja se pretvara u mala slova
Primeri	• `lower('HELLO World')` → 'hello world'

Više informacija: *upper*, *title*

lpad

Vraća nisku dopunjenu sa leve strane do zadate širine, pomoću znaka za popunu. Ako je ciljna širina manja od dužine niske, niska se skraćuje.

Sintaksa	lpad(string, width, fill)
Argumenti	• **string** - niska koja se dopunjuje • **width** - dužina nove niske • **fill** - znak kojim se popunjava preostali prostor
Primeri	• `lpad('Hello', 10, 'x')` → 'xxxxxHello' • `lpad('Hello', 3, 'x')` → 'Hel'

Više informacija: *rpad*

ltrim

Uklanja najdužu nisku koja sadrži samo zadate znakove (podrazumevano razmak) sa početka niske.

Sintaksa	ltrim(string, [characters:=' ']) [] označava opcione argumente
Argumenti	• **string** - niska koja se skraćuje • **characters** - znakovi koji se skraćuju
Primeri	• `ltrim(' hello world ')` → 'hello world ' • `ltrim('zzzytest', 'xyz')` → 'test'

Više informacija: *rtrim*, *trim*

regexp_match

Vraća prvu poziciju podudaranja sa regularnim izrazom unutar unicode niske, ili 0 ako podniska nije pronađena.

Sintaksa	regexp_match(input_string, regex)
Argumenti	• **input_string** - niska koja se testira regularnim izrazom • **regex** - Regularni izraz sa kojim se testira. Znakovi obrnute kose crte moraju biti dvostruko izbegnuti (npr. „\\s" za podudaranje sa belim znakom ili „\\b" za podudaranje sa granicom reči).
Primeri	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

Više informacija: *strpos*

regexp_replace

Vraća nisku sa zamenjenim zadatim regularnim izrazom.

Sintaksa	regexp_replace(input_string, regex, replacement)
Argumenti	• **input_string** - niska u kojoj se zamenjuju podudaranja • **regex** - Regularni izraz koji se zamenjuje. Znakovi obrnute kose crte moraju biti dvostruko izbegnuti (npr. „\\s" za podudaranje sa belim znakom). • **replacement** - Niska koja će zameniti sva podudaranja zadatog regularnog izraza. Uhvaćene grupe mogu se umetnuti u nisku zamene pomoću \\1, \\2 itd.
Primeri	• `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'

Više informacija: *replace*, *array_replace*

regexp_substr

Vraća deo niske koji se podudara sa zadatim regularnim izrazom.

Sintaksa	regexp_substr(input_string, regex)
Argumenti	• **input_string** - niska u kojoj se traže podudaranja • **regex** - Regularni izraz sa kojim se traži podudaranje. Znakovi obrnute kose crte moraju biti dvostruko izbegnuti (npr. „\\s" za podudaranje sa belim znakom).
Primeri	• `regexp_substr('abc123','(\\d+)')` → '123'

Više informacija: *substr*, *regexp_matches*

repeat

Ponavlja nisku zadati broj puta.

Sintaksa	repeat(string, number)
Argumenti	• **string** - niska koja se ponavlja • **number** - broj ponavljanja niske
Primeri	• `repeat('Hello', 3)` → 'HelloHelloHello'

replace

Vraća nisku sa zamenjenom zadatom niskom, nizom ili mapom niski.

Varijanta sa niskom i nizom

Vraća nisku u kojoj je zadata niska ili niz niski zamenjen niskom ili nizom niski.

Sintaksa	replace(string, before, after)
Argumenti	• **string** - ulazna niska • **before** - niska ili niz niski koje se zamenjuju • **after** - niska ili niz niski koje se koriste kao zamena
Primeri	• `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'

Varijanta sa mapom

Vraća nisku u kojoj su zadati ključevi mape zamenjeni uparenim vrednostima. Duži ključevi mape izračunavaju se prvi.

Sintaksa	replace(string, map)
Argumenti	• **string** - ulazna niska • **map** - mapa koja sadrži ključeve i vrednosti
Primeri	• `replace('APP SHOULD ROCK',map('APP','QGIS','SHOULD','DOES'))` → 'QGIS DOES ROCK' • `replace('forty two',map('for','4','two','2','forty two','42'))` → '42'

Više informacija: *regexp_replace*, *array_replace*

reverse

Obrće smer linije ili obrće tekstualnu nisku.

Varijanta sa niskom

Obrće redosled znakova u niski.

Sintaksa	reverse(string)
Argumenti	• **string** - niska koja se obrće
Primeri	• `reverse('hello')` → 'olleh'

Varijanta sa geometrijom

Obrće smer linije obrtanjem redosleda njenih temena.

Sintaksa	reverse(geometry)
Argumenti	• **geometry** - geometrija
Primeri	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → 'LINESTRING(2 2, 1 1, 0 0)'

right

Vraća podnisku koja sadrži *n* krajnjih desnih znakova niske.

Sintaksa	right(string, length)
Argumenti	• **string** - niska • **length** - ceo broj. Broj znakova sa desne strane niske koji se vraćaju.
Primeri	• `right('Hello World',5)` → 'World'

Više informacija: *left*

rpad

Vraća nisku dopunjenu sa desne strane do zadate širine, pomoću znaka za popunu. Ako je ciljna širina manja od dužine niske, niska se skraćuje.

Sintaksa	rpad(string, width, fill)
Argumenti	• **string** - niska koja se dopunjuje • **width** - dužina nove niske • **fill** - znak kojim se popunjava preostali prostor
Primeri	• `rpad('Hello', 10, 'x')` → 'Helloxxxxx' • `rpad('Hello', 3, 'x')` → 'Hel'

Više informacija: *lpad*

rtrim

Uklanja najdužu nisku koja sadrži samo zadate znakove (podrazumevano razmak) sa kraja niske.

Sintaksa	rtrim(string, [characters=' ']) [] označava opcione argumente
Argumenti	• **string** - niska koja se skraćuje • **characters** - znakovi koji se skraćuju
Primeri	• `rtrim(' hello world ')` → ' hello world' • `rtrim('testxxzx', 'xyz')` → 'test'

Više informacija: *ltrim*, *trim*

strpos

Vraća prvu poziciju podudaranja podniske unutar druge niske, ili 0 ako podniska nije pronađena.

Sintaksa	strpos(haystack, needle)
Argumenti	• **haystack** - niska koja se pretražuje • **needle** - niska koja se traži
Primeri	• `strpos('HELLO WORLD','WORLD')` → 7 • `strpos('HELLO WORLD','GOODBYE')` → 0

Više informacija: *regexp_match*

substr

Vraća deo niske.

Sintaksa	substr(string, start, [length]) [] označava opcione argumente
Argumenti	• **string** - puna ulazna niska • **start** - ceo broj koji predstavlja početnu poziciju izdvajanja, počev od 1; ako je start negativan, vraćena niska počinje od kraja niske umanjenog za vrednost start • **length** - ceo broj koji predstavlja dužinu niske koja se izdvaja; ako je length negativan, vraćena niska izostavlja zadatu dužinu znakova sa kraja niske
Primeri	• `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'

Više informacija: *regexp_substr*, *regexp_matches*

title

Pretvara sve reči niske u naslovna slova (sve reči malim slovima sa velikim početnim slovom).

Sintaksa	title(string)
Argumenti	• **string** - niska koja se pretvara u naslovna slova
Primeri	• `title('hello WOrld')` → 'Hello World'

Više informacija: *lower*, *upper*

to_string

Pretvara broj u nisku. Pretvaranje ne uzima u obzir lokalitet; za alternativu koja uzima lokalitet u obzir pogledajte 'format_number'.

Sintaksa	to_string(number)
Argumenti	• **number** - Ceo ili realan broj. Broj koji se pretvara u nisku.
Primeri	• `to_string(1.23)` → '1.23'

Više informacija: *format_number*

trim

Uklanja sve vodeće i prateće beline (razmake, tabulatore itd.) iz niske.

Sintaksa	trim(string)
Argumenti	• **string** - niska koja se skraćuje
Primeri	• `trim(' hello world ')` → 'hello world'

Više informacija: *ltrim*, *rtrim*

upper

Pretvara nisku u velika slova.

Sintaksa	upper(string)
Argumenti	• **string** - niska koja se pretvara u velika slova
Primeri	• `upper('hello WOrld')` → 'HELLO WORLD'

Više informacija: *lower*, *title*

wordwrap

Vraća nisku prelomljenu na maksimalan/minimalan broj znakova.

Sintaksa	wordwrap(string, wrap_length, [delimiter_string]) [] označava opcione argumente
Argumenti	• **string** - niska koja se prelama • **wrap_length** - ceo broj. Ako je wrap_length pozitivan, broj predstavlja idealan maksimalan broj znakova za prelamanje; ako je negativan, broj predstavlja minimalan broj znakova za prelamanje. • **delimiter_string** - Opciona niska graničnika za prelamanje u novi red.
Primeri	• `wordwrap('UNIVERSITY OF QGIS',13)` → 'UNIVERSITY OF QGIS' • `wordwrap('UNIVERSITY OF QGIS',-3)` → 'UNIVERSITY OF QGIS'

9.2.27 Korisnički izrazi

Ova grupa sadrži izraze sačuvane kao *korisnički izrazi*.

9.2.28 Promenljive

Ova grupa sadrži dinamičke promenljive vezane za aplikaciju, projektnu datoteku i druga podešavanja. Dostupnost promenljivih zavisi od konteksta:

- iz dijaloga  Izaberi izrazom
- iz dijaloga  Kalkulator polja
- iz dijaloga svojstava sloja
- iz štampanog prikaza

Da biste ove promenljive koristili u izrazu, ispred njih treba staviti znak `@` (npr. `@row_number`).

Variable	Opis
algorithm_id	Jedinstveni ID algoritma
animation_end_time	Kraj ukupnog vremenskog opsega animacije (kao datetime vrednost)
animation_interval	Trajanje ukupnog vremenskog opsega animacije (kao interval vrednost)
animation_start_time	Početak ukupnog vremenskog opsega animacije (kao datetime vrednost)
atlas_feature	Trenutni objekat atlasa (kao feature objekat)
atlas_featureid	ID trenutnog objekta atlasa
atlas_featurenumber	Broj trenutnog objekta atlasa u prikazu
atlas_filename	Naziv trenutne atlas datoteke
atlas_geometry	Geometrija trenutnog objekta atlasa
atlas_layerid	ID sloja pokrivača atlasa
atlas_layername	Naziv sloja pokrivača atlasa
atlas_pagename	Naziv trenutne strane atlasa
atlas_totalfeatures	Ukupan broj objekata u atlasu
band	Broj kanala u rasterskom sloju
band_description	Opis kanala u rasterskom sloju
band_name	Naziv kanala u rasterskom sloju
canvas_cursor_point	Poslednji položaj kursora na platnu u geografskim koordinatama projekta
cluster_color	Boja simbola unutar klastera, ili NULL ako simboli imaju mešane boje
cluster_size	Broj simbola sadržanih u klasteru
current_feature	Objekat koji se trenutno uređuje u atributskoj formi ili redu tabele
current_geometry	Geometrija objekta koji se trenutno uređuje u formi ili redu tabele
current_parent_feature	predstavlja objekat koji se trenutno uređuje u nadređenoj formi. Upotrebljivo samo u kontekstu ugrađene forme.
current_parent_geometry	predstavlja geometriju objekta koji se trenutno uređuje u nadređenoj formi. Upotrebljivo samo u kontekstu ugrađene forme.
form_mode	Za šta se forma koristi, kao AddFeatureMode, SingleEditMode, MultiEditMode, SearchMode, AggregateSearchMode ili IdentifyMode, kao niska.
feature	Trenutni objekat koji se izračunava. Ovo se može koristiti sa funkcijom 'attribute' za izračunavanje vrednosti atributa trenutnog objekta.
frame_duration	Vremensko trajanje svakog kadra animacije (kao interval vrednost)
frame_number	Broj trenutnog kadra tokom reprodukcije animacije
frame_rate	Broj kadrova u sekundi tokom reprodukcije animacije
fullextent_maxx	Maksimalna x vrednost punog obuhvata platna (uključujući sve slojeve)
fullextent_maxy	Maksimalna y vrednost punog obuhvata platna (uključujući sve slojeve)
fullextent_minx	Minimalna x vrednost punog obuhvata platna (uključujući sve slojeve)
fullextent_miny	Minimalna y vrednost punog obuhvata platna (uključujući sve slojeve)
geometry	Geometrija trenutnog objekta koji se izračunava
geometry_part_count	Broj delova u geometriji iscrtanog objekta

наставак на следећој страни

Табела 9.1 – настављено са претходне стране

Variable	Opis
geometry_part_num	Broj trenutnog dela geometrije objekta koji se iscrtava
geometry_point_count	Broj tačaka u delu iscrtane geometrije
geometry_point_num	Broj trenutne tačke u delu iscrtane geometrije
geometry_ring_num	Broj trenutnog prstena geometrije za objekat koji se iscrtava (samo za poligonske objekte). Spoljni prsten ima vrednost 0.
grid_axis	Trenutna osa napomene mreže (npr. 'x' za geografsku dužinu, 'y' za geografsku širinu)
grid_number	Trenutna vrednost napomene mreže
id	ID trenutnog objekta koji se izračunava
item_id	Korisnički ID stavke prikaza (nije nužno jedinstven)
item_uuid	Jedinstveni ID stavke prikaza
layer	Trenutni sloj
layer_crs	ID autoriteta koordinatnog referentnog sistema trenutnog sloja
layer_crs_ellipsoid	ID autoriteta elipsoida CRS-a trenutnog sloja
layer_cursor_point	Geometrija tačke ispod položaja miša na platnu karte (ili GetFeatureInfo položaj u kontekstu QGIS Servera), u CRS-u aktivnog sloja
layer_id	ID trenutnog sloja
layer_ids	ID-jevi svih slojeva karte u trenutnom projektu kao lista
layer_name	Naziv trenutnog sloja
layer_vertical_crs	Identifikator vertikalnog koordinatnog referentnog sistema sloja (npr. 'EPSG:5703')
layer_vertical_crs_definition	Puna definicija vertikalnog koordinatnog referentnog sistema sloja
layer_vertical_crs_description	Naziv vertikalnog koordinatnog referentnog sistema sloja
layer_vertical_crs_wkt	WKT definicija vertikalnog koordinatnog referentnog sistema trenutnog sloja
layers	Svi slojevi karte u trenutnom projektu kao lista
layout_dpi	Rezolucija kompozicije (DPI)
layout_name	Naziv prikaza
layout_numpages	Broj strana u prikazu
layout_page	Broj strane trenutne stavke u prikazu
layout_pageheight	Visina aktivne strane u prikazu (u mm za standardne veličine papira, ili u jedinici korišćenoj za prilagođenu veličinu papira)
layout_pageoffsets	Niz Y koordinata vrha svake strane. Omogućava dinamičko postavljanje stavki na strane u kontekstu u kojem se veličine strana mogu menjati
layout_pagewidth	Širina aktivne strane u prikazu (u mm za standardne veličine papira, ili u jedinici korišćenoj za prilagođenu veličinu papira)
legend_column_count	Broj kolona u legendi
legend_filter_by_map	Označava da li je sadržaj legende filtriran kartom
legend_filter_out_atlas	Označava da li je atlas izostavljen iz legende
legend_split_layers	Označava da li se slojevi mogu deliti u legendi
legend_title	Naslov legende
legend_wrap_string	Znak(ovi) koji se koriste za prelamanje teksta legende
map_crs	Koordinatni referentni sistem trenutne karte
map_crs_acronym	Akronim koordinatnog referentnog sistema trenutne karte
map_crs_definition	Puna definicija koordinatnog referentnog sistema trenutne karte
map_crs_description	Naziv koordinatnog referentnog sistema trenutne karte
map_crs_ellipsoid	Akronim elipsoida koordinatnog referentnog sistema trenutne karte
map_crs_proj4	Proj4 definicija koordinatnog referentnog sistema trenutne karte
map_crs_projection	Opisni naziv metode projekcije koju koristi koordinatni referentni sistem karte (npr. 'Albers Equal Area')
map_crs_wkt	WKT definicija koordinatnog referentnog sistema trenutne karte
map_end_time	Kraj vremenskog opsega karte (kao datetime vrednost)
map_extent	Geometrija koja predstavlja trenutni obuhvat karte
map_extent_center	Tačkasti objekat u centru karte
map_extent_height	Trenutna visina karte

наставак на следећој страни

Табела 9.1 – настављено са претходне стране

Variable	Opis
map_extent_width	Trenutna širina karte
map_id	ID trenutnog odredišta karte. Biće 'canvas' za iscrtavanje na platnu, a ID stavke za iscrtavanje karte u prikazu
map_interval	Trajanje vremenskog opsega karte (kao interval vrednost)
map_layer_ids	Lista ID-jeva slojeva karte vidljivih na karti
map_layers	Lista slojeva karte vidljivih na karti
map_rotation	Trenutna rotacija karte
map_scale	Trenutna razmera karte
map_start_time	Početak vremenskog opsega karte (kao datetime vrednost)
map_units	Jedinice mera karte
map_z_range_lower	Donja visina visinskog opsega karte
map_z_range_upper	Gornja visina visinskog opsega karte
model_path	Puna putanja (uključujući naziv datoteke) trenutnog modela (ili putanja projekta ako je model ugrađen u projekat).
model_folder	Folder koji sadrži trenutni model (ili folder projekta ako je model ugrađen u projekat).
model_name	Naziv trenutnog modela
model_group	Grupa trenutnog modela
notification_message	Sadržaj poruke obaveštenja koju šalje provajder (dostupno samo za radnje pokrenute obaveštenjima provajdera).
parent	Odnosi se na trenutni objekat u nadređenom sloju, omogućavajući pristup njegovim atributima i geometriji pri filtriranju *agregatne* funkcije
plot_axis	Pripadajuća osa grafika, npr. 'x' ili 'y'.
plot_axis_value	Trenutna vrednost za osu grafika.
project_abstract	Apstrakt projekta, preuzet iz metapodataka projekta
project_area_units	Jedinica površine za trenutni projekat, koja se koristi pri računanju površina geometrija
project_author	Autor projekta, preuzet iz metapodataka projekta
project_basename	Osnovni naziv datoteke trenutnog projekta (bez putanje i ekstenzije)
project_creation_date	Datum kreiranja projekta, preuzet iz metapodataka projekta
project_crs	Identifikator koordinatnog referentnog sistema projekta (npr. 'EPSG:4326')
project_crs_arconym	Akronim koordinatnog referentnog sistema projekta
project_crs_definition	Puna definicija koordinatnog referentnog sistema projekta
project_crs_description	Opis koordinatnog referentnog sistema projekta
project_crs_ellipsoid	Elipsoid koordinatnog referentnog sistema projekta
project_crs_proj4	Proj4 predstavljanje koordinatnog referentnog sistema projekta
project_crs_wkt	WKT (well known text) predstavljanje koordinatnog referentnog sistema projekta
project_distance_units	Jedinica rastojanja za trenutni projekat, koja se koristi pri računanju dužina geometrija i rastojanja
project_ellipsoid	Naziv elipsoida trenutnog projekta, koji se koristi pri računanju geodetskih površina ili dužina geometrija
project_filename	Naziv datoteke trenutnog projekta
project_folder	Folder trenutnog projekta
project_home	Matična putanja trenutnog projekta
project_identifier	Identifikator projekta, preuzet iz metapodataka projekta
project_keywords	Ključne reči projekta, preuzete iz metapodataka projekta
project_last_saved	Datum/vreme poslednjeg čuvanja projekta.
project_path	Puna putanja (uključujući naziv datoteke) trenutnog projekta
project_title	Naslov trenutnog projekta
project_units	Jedinice CRS-a projekta
project_vertical_crs	Identifikator vertikalnog koordinatnog referentnog sistema projekta (npr. 'EPSG:5703')
project_vertical_crs_definition	Puna definicija vertikalnog koordinatnog referentnog sistema projekta

наставак на следећој страни

Табела 9.1 – настављено са претходне стране

Variable	Opis
project_vertical_crs_description	Opis vertikalnog koordinatnog referentnog sistema projekta
project_vertical_crs_wkt	WKT (well known text) predstavljanje vertikalnog koordinatnog referentnog sistema projekta
qgis_locale	Trenutni jezik QGIS-a
qgis_os_name	Naziv trenutnog operativnog sistema, npr. 'windows', 'linux' ili 'osx'
qgis_platform	QGIS platforma, npr. 'desktop' ili 'server'
qgis_release_name	Naziv trenutnog QGIS izdanja
qgis_short_version	Kratka niska trenutne verzije QGIS-a
qgis_version	Niska trenutne verzije QGIS-a
qgis_version_no	Broj trenutne verzije QGIS-a
row_number	Čuva broj trenutnog reda
snapping_results	Omogućava pristup rezultatima lepljenja tokom digitalizacije objekta (dostupno samo pri dodavanju objekta)
scale_value	Trenutna vrednost rastojanja razmernika
selected_file_path	Izabrana putanja datoteke iz birača datoteka pri otpremanju datoteke sistemom spoljnog skladištenja
symbol_angle	Ugao simbola koji se koristi za iscrtavanje objekta (važi samo za markerske simbole)
symbol_color	Boja simbola koji se koristi za iscrtavanje objekta
symbol_count	Broj objekata koje predstavlja simbol (u legendi prikaza)
symbol_frame	Broj kadra (samo za animirane simbole)
symbol_id	Interni ID simbola (u legendi prikaza)
symbol_label	Natpis simbola (bilo korisnički definisan natpis ili podrazumevani automatski generisan natpis - u legendi prikaza)
symbol_layer_count	Ukupan broj slojeva simbola u simbolu
symbol_layer_index	Indeks trenutnog sloja simbola
symbol_marker_column	Broj kolone za marker (važi samo za tačkaste popune šarom).
symbol_marker_row	Broj reda za marker (važi samo za tačkaste popune šarom).
user_account_name	Naziv naloga trenutnog korisnika na operativnom sistemu
user_full_name	Korisničko ime trenutnog korisnika na operativnom sistemu
value	Trenutna vrednost
vector_tile_zoom	Tačan nivo uvećanja vektorskih pločica karte koja se iscrtava (izveden iz trenutne razmere karte). Obično u intervalu [0, 20]. Za razliku od @zoom_level, ova promenljiva je vrednost u pokretnom zarezu koja se može koristiti za interpolaciju vrednosti između dva celobrojna nivoa uvećanja.
with_variable	Omogućava postavljanje promenljive za upotrebu unutar izraza i izbegavanje ponovnog računanja iste vrednosti
zoom_level	Nivo uvećanja vektorskih pločica karte koja se iscrtava (izveden iz trenutne razmere karte). Obično u intervalu [0, 20].

Nekoliko primera:

- Vraća X koordinatu centra stavke karte u prikazu:

```
x( map_get( item_variables( 'map1'), 'map_extent_center' ) )
```

- Vraća, za svaki objekat na trenutnom sloju, broj preklapajućih objekata aerodroma:

```
aggregate( layer:='airport', aggregate:='count', expression:="code",
              filter:=intersects( $geometry, geometry( @parent ) ) )
```

- Preuzmi object_id prve zalepljene tačke linije:

```
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 Nedavne funkcije

Ova grupa sadrži nedavno korišćene funkcije. U zavisnosti od konteksta upotrebe (izbor objekata, kalkulator polja, opšte), nedavno primenjeni izrazi dodaju se u odgovarajuću listu (do deset izraza), poređani od novijih ka starijim. Time se omogućava brzo preuzimanje i ponovna primena prethodno korišćenih izraza.

ГЛАВА 10

Biblioteka stilova

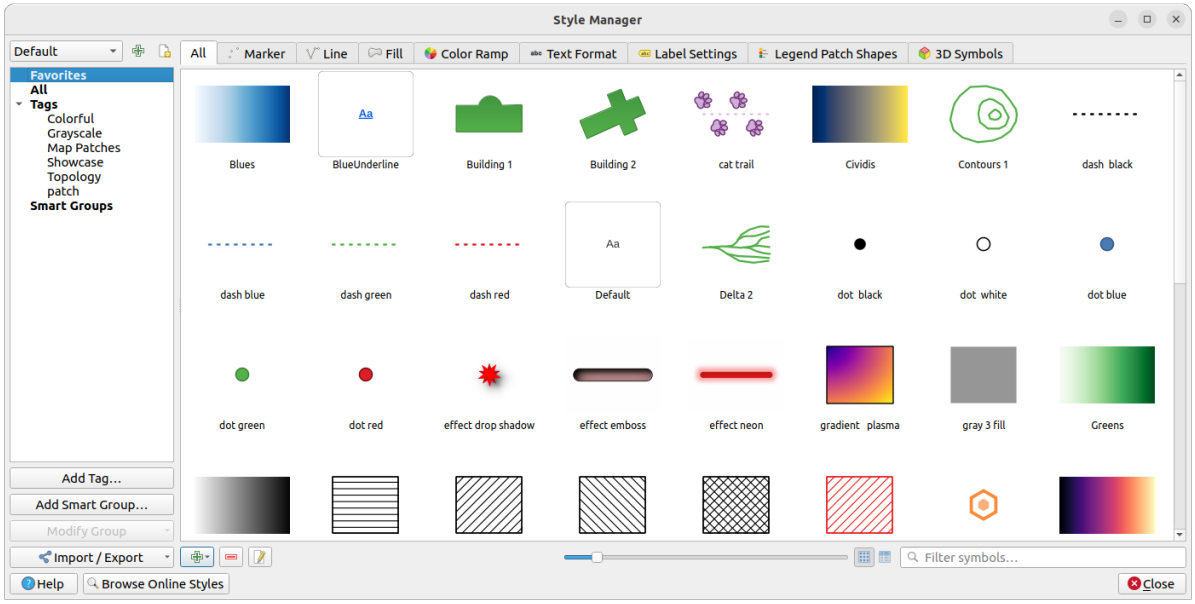
10.1 The Style Manager

10.1.1 The Style Manager dialog

The *Style Manager* is the place where you can manage and create generic style items. These are symbols, color ramps, text formats or label settings that can be used to symbolize features, layers or print layouts. They are stored in the `symbology-style.db` database under the active *user profile* and shared with all the project files opened with that profile. Style items can also be shared with others thanks to the export/import capabilities of the *Style Manager* dialog.

You can open that modeless dialog either:

- from the *Settings* ►  *Style Manager…* menu
- with the  Style Manager button from the Project toolbar
- or with the  Style Manager button from a vector *Layer Properties* ► menu (while *configuring a symbol* or *formatting a text*).



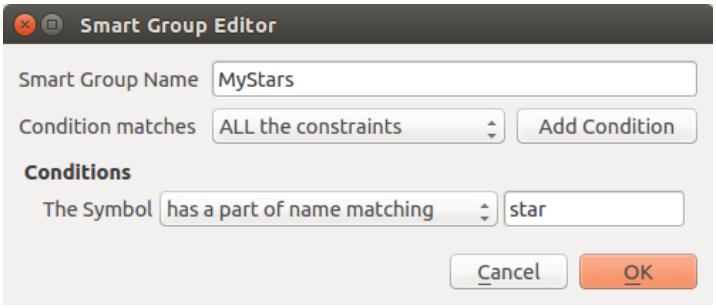
Слика10.1: The Style Manager

Organizing style items

In the upper left corner of the *Style Manager* dialog, within the drop-down menu, you can choose the style database you want to connect to. If you choose *Default* you will be connected to the default style database where you can find all available default style items and the ones that you saved in this database. Choosing *Project Styles* will connect you to the Project Style database where you can find only symbols that you saved in this database. If you created more style databases, they will be listed in the drop-down menu. There are also options to Add existing style database to project and Create new style database (see more *Style database*).

For each style database, you can organize the elements into different categories, listed in the panel on the left:

- **Favorites**: displayed by default when configuring an item, it shows an extensible set of items;
- **All**: lists all the available items for the active type;
- **Tags**: shows a list of labels you can use to identify the items. An item can be tagged more than once. Select a tag in the list and the tabs are updated to show only their items that belong to it. To create a new tag you could later attach to a set of items, use the *Add Tag...* button or select the *Add Tag...* from any tag contextual menu;
- **Smart Group**: a smart group dynamically fetches its symbols according to conditions set (see eg, Сл.10.2). Click the *Add Smart Group...* button to create smart groups. The dialog box allows you to enter an expression to filter the items to select (has a particular tag, have a string in its name, etc.). Any symbol, color ramp, text format or label setting that satisfies the entered condition(s) is automatically added to the smart group.



Слика10.2: Creating a Smart Group

Tags and smart groups are not mutually exclusive: they are simply two different ways to organize your style elements. Unlike the smart groups that automatically fetch their belonged items based on the input constraints, tags are filled by the user. To edit any of those categories, you can either:

- select the items, right-click and choose *Add to Tag* ► and then select the tag name or create a new tag;
- select the tag and press *Modify group…* ► *Attach Selected Tag to Symbols*. A checkbox appears next to each item to help you select or deselect it. When selection is finished, press *Modify group…* ► *Finish Tagging*.
- select the smart group, press *Modify group…* ► *Edit smart group…* and configure a new set of constraints in the *Smart Group Editor* dialog. This option is also available in the contextual menu of the smart group.

To remove a tag or a smart group, right-click on it and select the *Remove* button. Note that this does not delete the items grouped in the category.

The *Style Manager* dialog displays in its center a frame with previewed items organized into tabs:

- *All* for a complete collection of point, linear and surface symbols and label settings as well as predefined color ramps and text formats;
- *Marker* for point symbols only;
- *Line* for linear symbols only;
- *Fill* for surface symbols only;
- *Color ramp*;
- abc *Text format* to manage *text formats*, which store the font, color, buffers, shadows, and backgrounds of texts (i.e. all the formatting parts of the label settings, which for instance can be used in layouts);
- *Label settings* to manage *label settings*, which include the text formats and some layer-type specific settings such as label placement, priority, callouts, rendering…
- *Legend Patch Shapes* to manage custom legend patch shapes, which include *Marker*, *Line* and *Fill* geometries.
- *3D Symbols* to configure symbols with *3D properties* (extrusion, shading, altitude, …) for the features to render in a *3D Map view*

You can arrange the Styles in *Icon View* or in *List View* on the bottom right side. In both views the tooltip shows a larger instance of the style. The thumbnail size slider at the left of the icons helps you adjust the actual thumbnail sizes in the dialog, for a better preview of the symbols!

Adding, editing or removing an item

As seen earlier, style elements are listed under different tabs whose contents depend on the active category (tag, smart group, favorites…). When a tab is enabled, you can:

- Add new items: press the  Add item button and configure the item following *symbols*, *color ramps* or *text format and label* builder description.
- Modify an existing item: select an item and press  Edit item button and configure as mentioned above.
- Delete existing items: to delete an element you no longer need, select it and click  Remove item (also available through right-click). The item will be deleted from the local database.

Note that the *All* tab provides access to these options for every type of item.

Right-clicking over a selection of items also allows you to:

- *Add to Favorites*;
- *Remove from Favorites*;
- *Add to Tag* ► and select the appropriate tag or create a new one to use; the currently assigned tags are checked;
- *Clear Tags*: detaching the symbols from any tag;
- *Remove Item(s)*;
- *Edit Item*: applies to the item you right-click over;
- *Copy Item*;
- *Paste Item* …: pasting to one of the categories of the style manager or elsewhere in QGIS (symbol or color buttons)
- *Export Selected Symbol(s) as PNG…* (only available with symbols);
- *Export Selected Symbol(s) as SVG…* (only available with symbols);

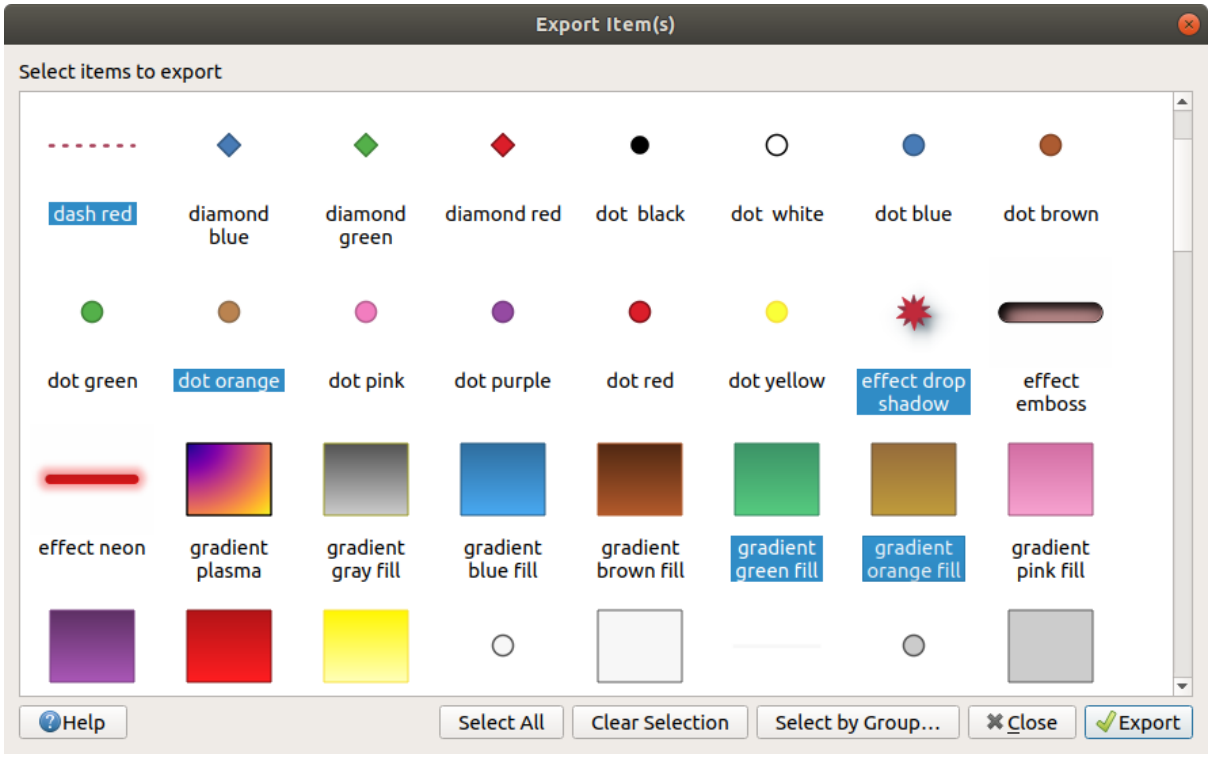
Sharing style items

The  *Import/Export* tool, at the left bottom of the Style Manager dialog, offers options to easily share symbols, color ramps, text formats and label settings with others. These options are also available through right-click over the items.

Exporting items

You can export a set of items to an `.XML` file:

1. Expand the  *Import/Export* drop-down menu and select  *Export Item(s)…*
2. Choose the items you'd like to integrate. Selection can be done with the mouse or using a tag or a group previously set.
3. Press *Export* when ready. You'll be prompted to indicate the destination of the saved file. The XML format generates a single file containing all the selected items. This file can then be imported in another user's style library.



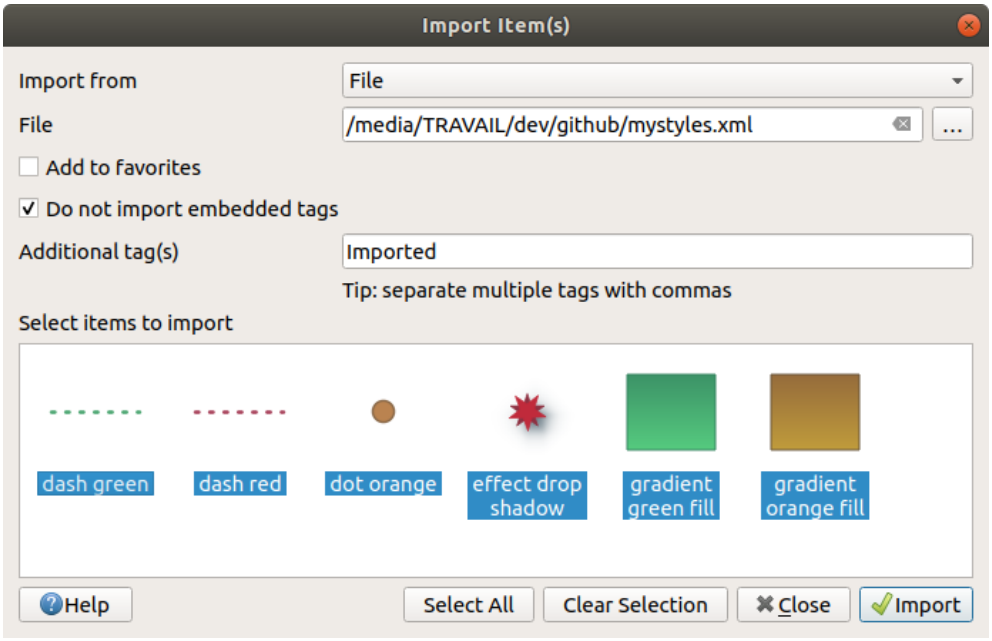
Слика10.3: Exporting style items

When symbols are selected, you can also export them to `.PNG` or `.SVG`. Exporting to `.PNG` or `.SVG` (both not available for other style item types) creates a file for each selected symbol in a given folder. The SVG folder can be added to the *SVG paths* in *Settings* ► *Options* ► *System* menu of another user, allowing him direct access to all these symbols.

Importing items

You can extend your style library by importing new items:

1. Expand the *Import/Export* drop-down menu and select *Import Item(s)* at the left bottom of the dialog.
2. In the new dialog, indicate the source of the style items (it can be an `.xml` file on the disk or a url).
3. Set whether to *Add to favorites* the items to import.
4. Check *Do not import embedded tags* to avoid the import of tags associated to the items being imported.
5. Give the name of any *Additional tag(s)* to apply to the new items.
6. Select from the preview the symbols you want to add to your library.
7. And press *Import*.



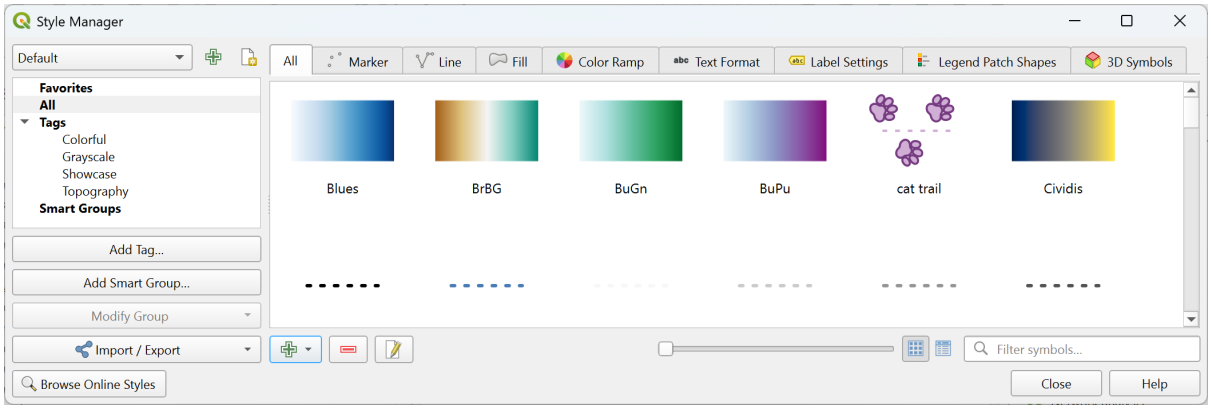
Слика10.4: Importing style items

Using the Browser panel

It's also possible to import style items into the active user profile style database directly from the *Browser* panel:

1. Select the style `.xml` file in the browser
2. Drag-and-drop it over the map canvas or right-click and select *Import Style…*
3. Fill the *Import Items* dialog following *Importing items*
4. Press *Import* and the selected style items are added to the style database

Double-clicking the style file in the browser opens the *Style Manager* dialog showing the items in the file. You can select them and press *Copy to Default Style…* to import them into the active style database. Tags can be assigned to items. Also available through right-click, *Open Style…* command.



Слика10.5: Opening a style items file

The dialog also allows to export single symbols as `.PNG` or `.SVG` files.

Using the online repository

The QGIS project maintains a repository with a collection of styles shared by QGIS users. This is available at https://hub.qgis.org/styles and can be accessed from the *Style Manager* dialog, pressing the  *Browse Online Styles* button at the bottom.

From that repository, you can:

1. Browse and search for any style items, based on their type or name
2. Download the style file and unzip it
3. Load the `.xml` based file into your style database in QGIS, using any of the aforementioned import methods.

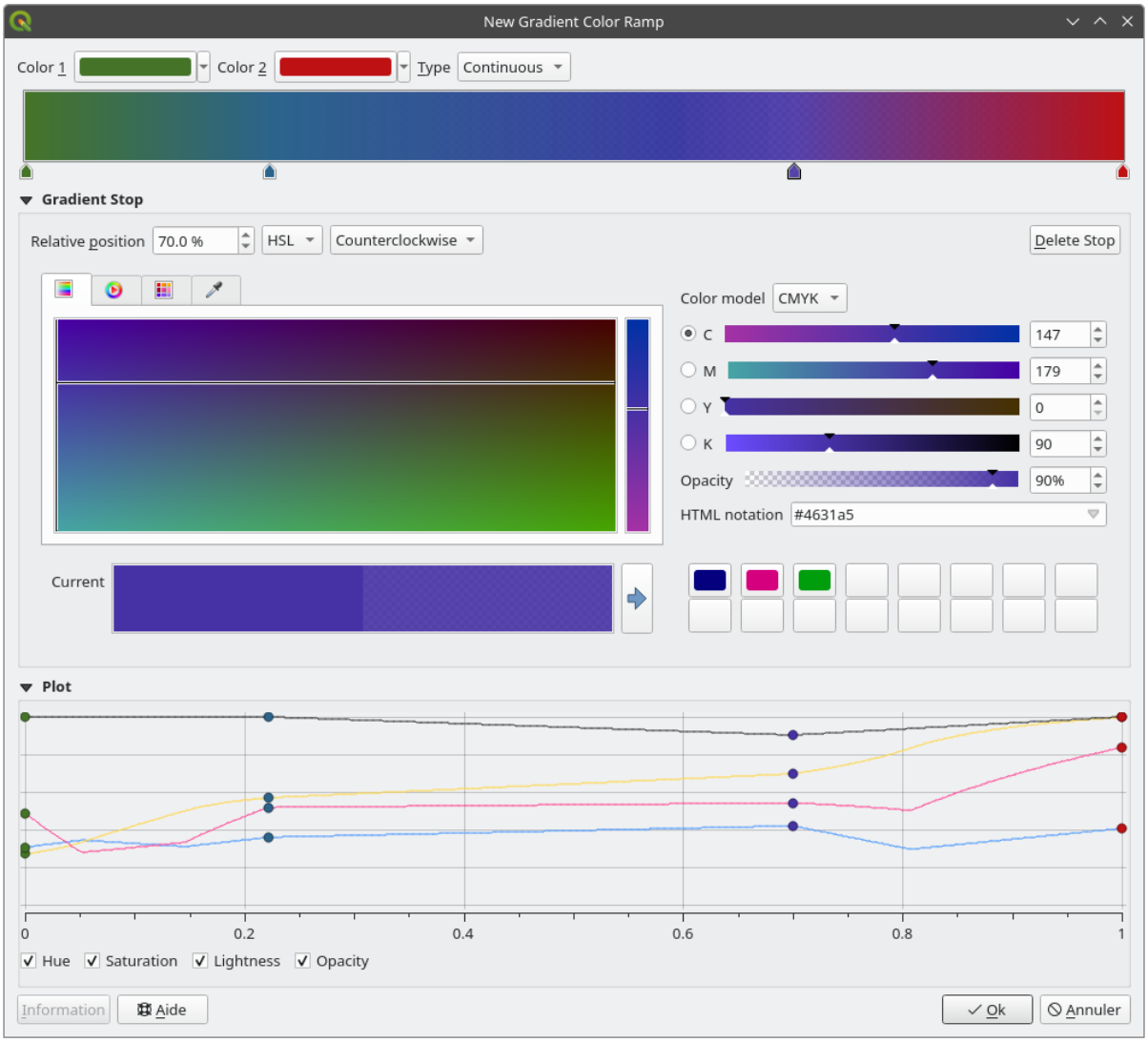
10.1.2 Setting a Color Ramp

The Color ramp tab in the *Style Manager* dialog helps you preview different color ramps based on the category selected in the left panel.

To create a custom color ramp, activate the Color ramp tab and click the  Add item button. The button reveals a drop-down list to choose the ramp type:

- *Gradient*: given a start and end colors, generates a color ramp which can be **continuous** or **discrete**. With double-clicking the ramp preview, you can add as many intermediate color stops as you want. Click on the color stop indicator and under *Gradient stop* you can:
  - adjust its *Relative position* from the color ramp start. Also possible dragging the indicator with the mouse, or pressing the arrow keys (combine with `Shift` key for a larger move)
  - specify the color model to use when interpolating between colors: it can be *RGB*, *HSL* or *HSV*. In some circumstances, this option can help avoid desaturated mid tones, resulting in more visually pleasing gradients.
  - set the direction which the interpolation should follow for the **Hue** component of a *HSL* or *HSV* color specification. It can be *Clockwise* or *Counterclockwise*.
  - set the *color properties*
  - remove the color stop pressing *Delete stop* or `DEL`

  The *Plots* group provides another graphical way to design the color ramp, changing the position or the opacity and HSL components of the color stops.

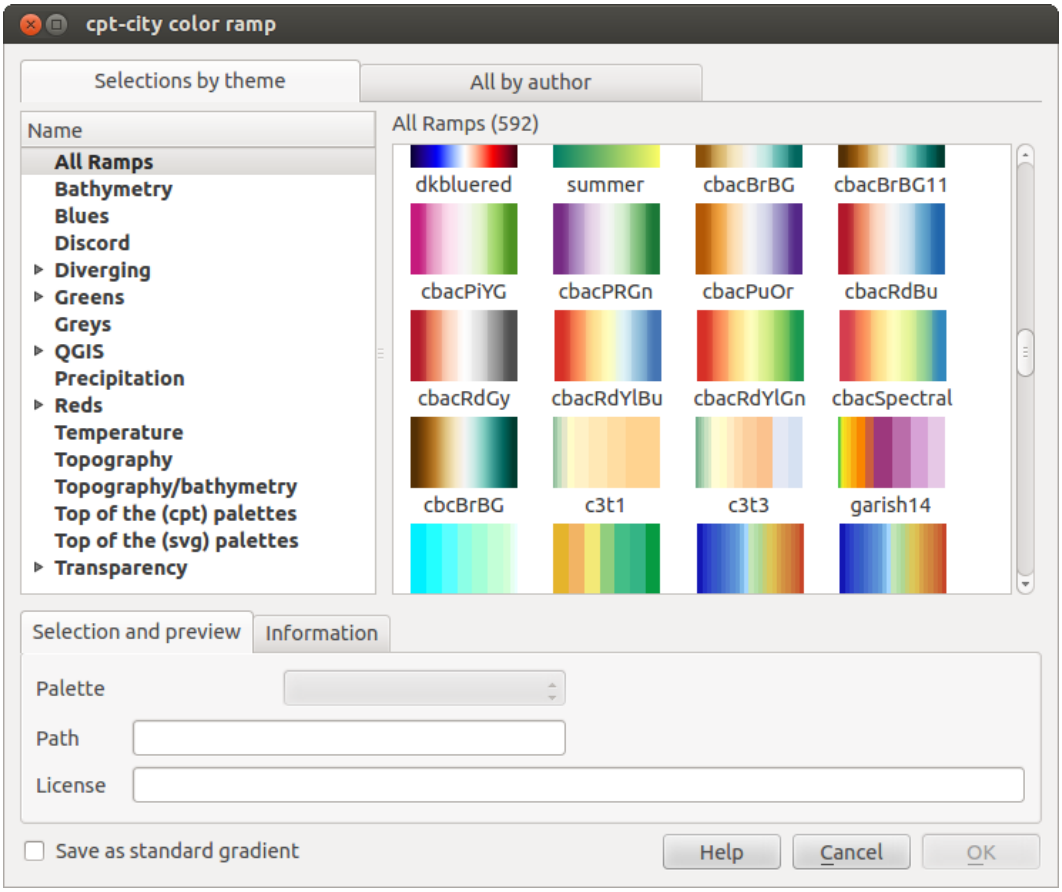


Слика10.6: Example of custom gradient color ramp with multiple stops

Путоказ

Drag-and-drop a color from a color spot onto the gradient ramp preview adds a new color stop.

- *Color presets*: allows to create a color ramp consisting of a list of colors selected by the user;
- *Random*: creates a random set of colors based on range of values for *Hue*, *Saturation*, *Value* and *Opacity* and a number of colors (*Classes*);
- *Catalog: ColorBrewer*: a set of predefined discrete color gradients you can customize the number of colors in the ramp;
- or *Catalog: cpt-city*: an access to a whole catalog of color gradients to locally *save as standard gradient*. The cpt-city option opens a new dialog with hundreds of themes included 'out of the box'.



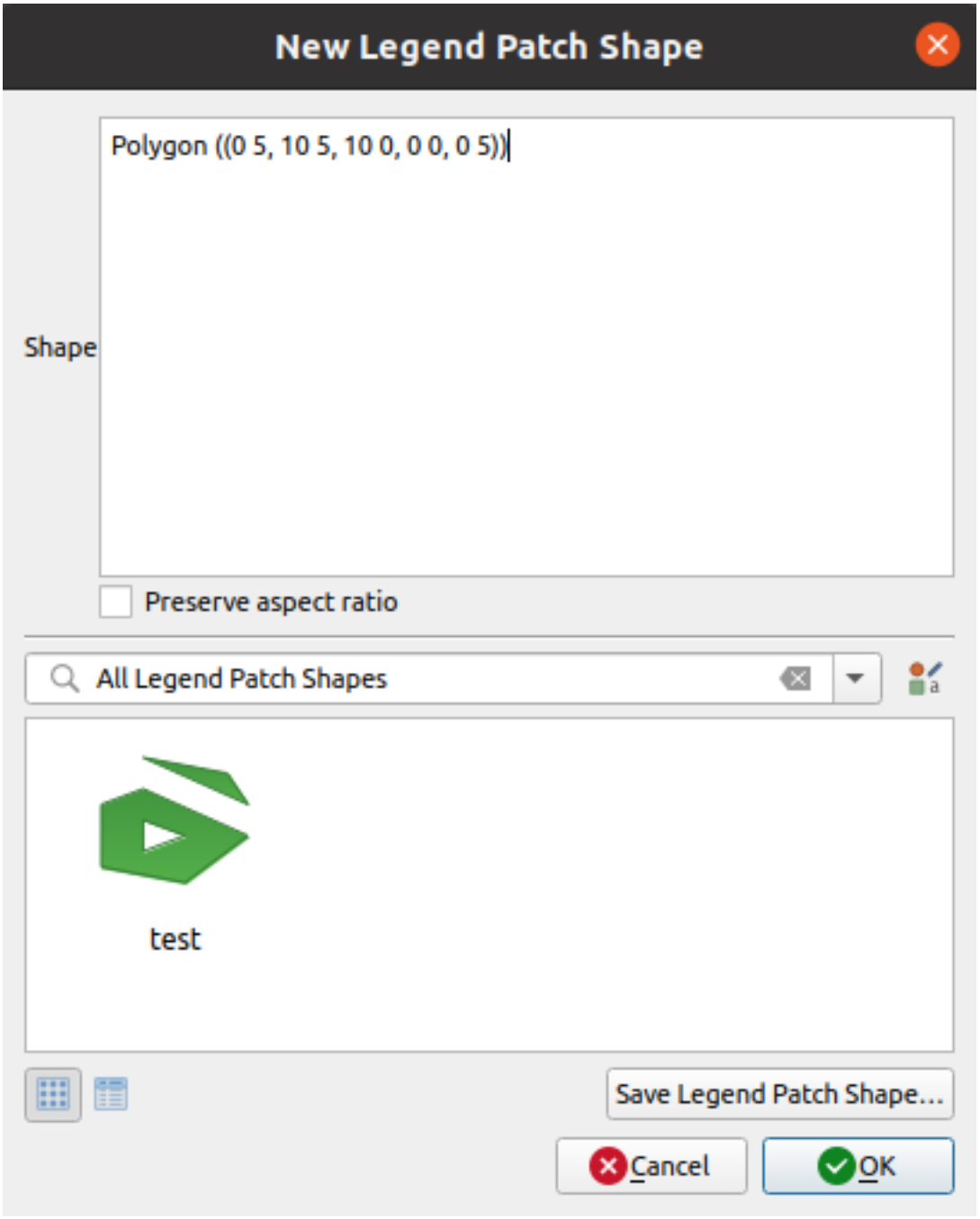
Слика10.7: cpt-city dialog with hundreds of color ramps

10.1.3 Creating a Legend Patch Shape

To create a new Legend Patch Shape, activate the *Legend Patch Shapes* tab and click the Add item button. The button reveals a drop-down list to choose the geometry type:

- *Marker Legend Patch Shape…*: to use with point geometries.
- *Line Legend Patch Shape…*: to use with line geometries.
- *Fill Legend Patch Shape…*: to use with polygon geometries.

All three options will show the same dialog.

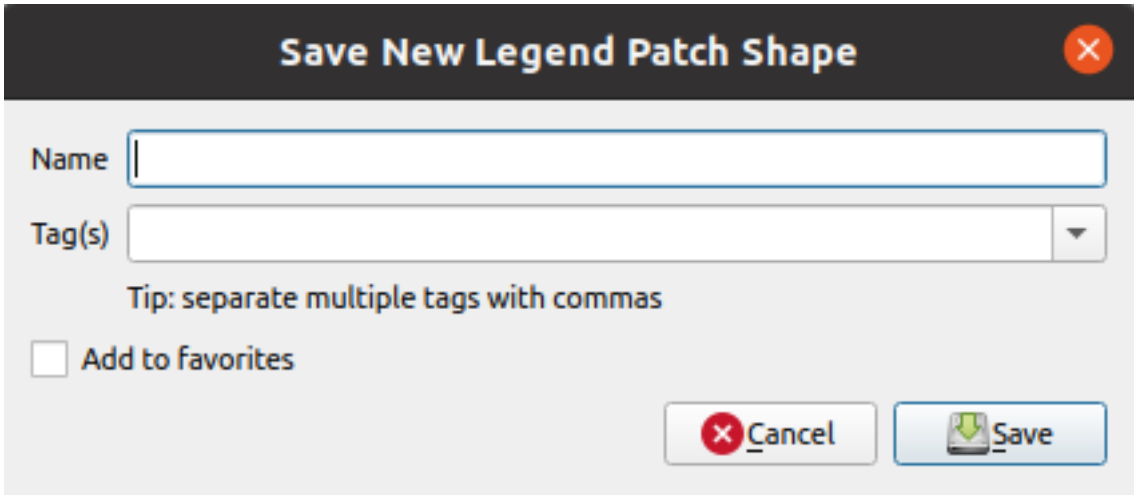


Слика10.8: Create a new Legend Patch Shape

Only the shape type and displayed legend patch shapes will differ regarding to the chosen geometry type. The following options will be available:

- *Shape*: define the shape of the legend patch shape as a WKT string. Single and multipart geometries may be used, but no GeometryCollection.
- *Preserve aspect ratio*
- *Icon View* or *List View* of available legend patch shapes, filtered by tags.

When the new Shape is defined you can *Save Legend Patch Shape…* or press *OK*, which will both lead to the same dialog.



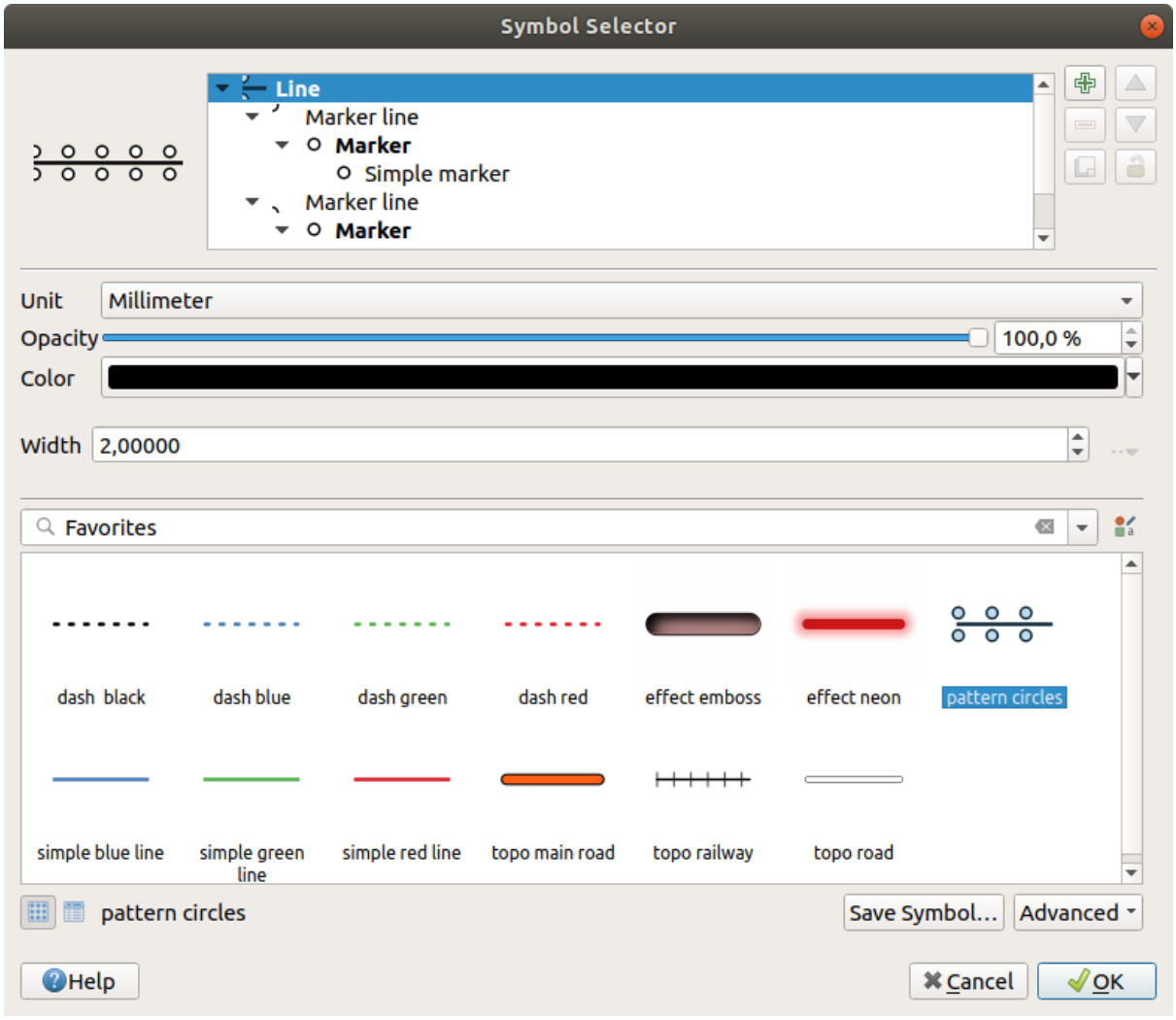
Слика10.9: Save a new Legend Patch Shape

Here you have to choose a name, tags to describe the shape and if it should be added to favorites.

If you press *Save…*, the shape is added to the list and you are directed back to the *New Legend Patch Shape* dialog to keep creating new shapes.

10.2 The Symbol Selector

The Symbol selector is the main dialog to design a symbol. You can create or edit Marker, Line or Fill Symbols.



Слика10.10: Designing a Line symbol

Two main components structure the symbol selector dialog:

- the symbol tree, showing symbol layers that are combined afterwards to shape a new global symbol
- and settings to configure the selected symbol layer in the tree.

10.2.1 The symbol layer tree

A symbol can consist of several *Symbol layers*. The symbol tree shows the overlay of these symbol layers that are combined afterwards to shape a new global symbol. Besides, a dynamic symbol representation is updated as soon as symbol properties change.

Depending on the level selected in the symbol tree items, various tools are made available to help you manage the tree:

- add new symbol layer: you can stack as many symbols as you want
- remove the selected symbol layer
- lock colors of symbol layer: a locked color stays unchanged when user changes the color at the global (or upper) symbol level
- duplicate a (group of) symbol layer(s)

- move up or down the symbol layer

10.2.2 Configuring a symbol

In QGIS, configuring a symbol is done in two steps: the symbol and then the symbol layer.

The symbol

At the top level of the tree, it depends on the layer geometry and can be of **Marker**, **Line** or **Fill** type. Each symbol can embed one or more symbols (including, of any other type) or symbol layers.

You can setup some parameters that apply to the global symbol:

- *Unit*: it can be **Millimeters**, **Points**, **Pixels**, **Meters at Scale**, **Map units** or **Inches** (see *Unit Selector* for more details)
- *Opacity*
- *Color*: when this parameter is changed by the user, its value is echoed to all unlocked sub-symbols color
- *Size* and *Rotation* for marker symbols
- *Width* for line symbols

> **Савет**
>
> Use the *Size* (for marker symbols) or the *Width* (for line symbols) properties at the symbol level to proportionally resize all of its embedded *symbol layers* dimensions.

> **Белешка**
>
> The *Data-defined override* button next to the width, size or rotation parameters is inactive when setting the symbol from the Style manager dialog. When the symbol is connected to a map layer, this button helps you create *proportional or multivariate analysis* rendering.

- A preview of the *symbols library*: Symbols of the same type are shown and, through the editable drop-down list just above, can be filtered by free-form text or by *categories*. You can also update the list of symbols using the  Style Manager button and open the eponym dialog. There, you can use any capabilities as exposed in *The Style Manager* section.

  The symbols are displayed either:

  - in an icon list (with thumbnail, name and associated tags) using the  List View button below the frame;
  - or as icon preview using the  Icon View button.

- Press the *Save Symbol* button to open the *Save New Symbol* dialog. Here, you can choose the *Destination* where you want to add the symbol being edited, give it a *Name* and add *Tag(s)*. You also have the option to  Add to favorites your new symbol.

- With the *Advanced*  option, you can:

  - for line and fill symbols, *Clip features to canvas extent*.
  - for fill symbols, *Force right-hand rule orientation*: allows forcing rendered fill symbols to follow the standard „right hand rule” for ring orientation (i.e, polygons where the exterior ring is clockwise, and the interior rings are all counter-clockwise).

    The orientation fix is applied while rendering only, and the original feature geometry is unchanged. This allows for creation of fill symbols with consistent appearance, regardless of the dataset being rendered and the ring orientation of individual features.

- for marker symbols, the *Buffer settings…* enables addition of a halo effect around the marker symbol in order to make it more readable against different backgrounds. The buffer is calculated and drawn using the shape of ALL the symbol layers in the marker. You can adjust the *Size*, *Join style* and *Symbol* of display.
- 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…* to define the order of symbols rendering
  * *Data-defined Size Legend*
  * *Match to Saved Symbols…* and *Match to Symbols from File…* to automatically *assign symbols to classes*

The symbol layer

At a lower level of the tree, you can customize the symbol layers. The available symbol layer types depend on the upper symbol type. You can apply on the symbol layer *paint effects* to enhance its rendering.

Because describing all the options of all the symbol layer types would not be possible, only particular and significant ones are mentioned below.

Common parameters

Some common options and widgets are available to build a symbol layer, regardless it's of marker, line or fill sub--type:

- the *color selector* widget to ease color manipulation
- *Units*: it can be **Millimeters**, **Points**, **Pixels**, **Meters at Scale**, **Map units** or **Inches** (see *Unit Selector* for more details)
- the Data-defined override widget near almost all options, extending capabilities of customizing each symbol (see *Data defined override setup* for more information)
- the *Enable symbol layer* option controls the symbol layer's visibility. Disabled symbol layers are not drawn when rendering the symbol but are saved in the symbol. Being able to hide symbol layers is convenient when looking for the best design of your symbol as you don't need to remove any for the testing. The data-defined override then makes it possible to hide or display different symbol layers based on expressions (using, for instance, feature attributes).
- the *Draw effects* button for *effects rendering*.

Белешка

While the description below assumes that the symbol layer type is bound to the feature geometry, keep in mind that you can embed symbol layers in each others. In that case, the lower level symbol layer parameter (placement, offset…) might be bound to the upper-level symbol, and not to the feature geometry itself.

Marker Symbols

Appropriate for point geometry features, marker symbols have several *Symbol layer types*:

Табела10.1: List of marker symbol layer types

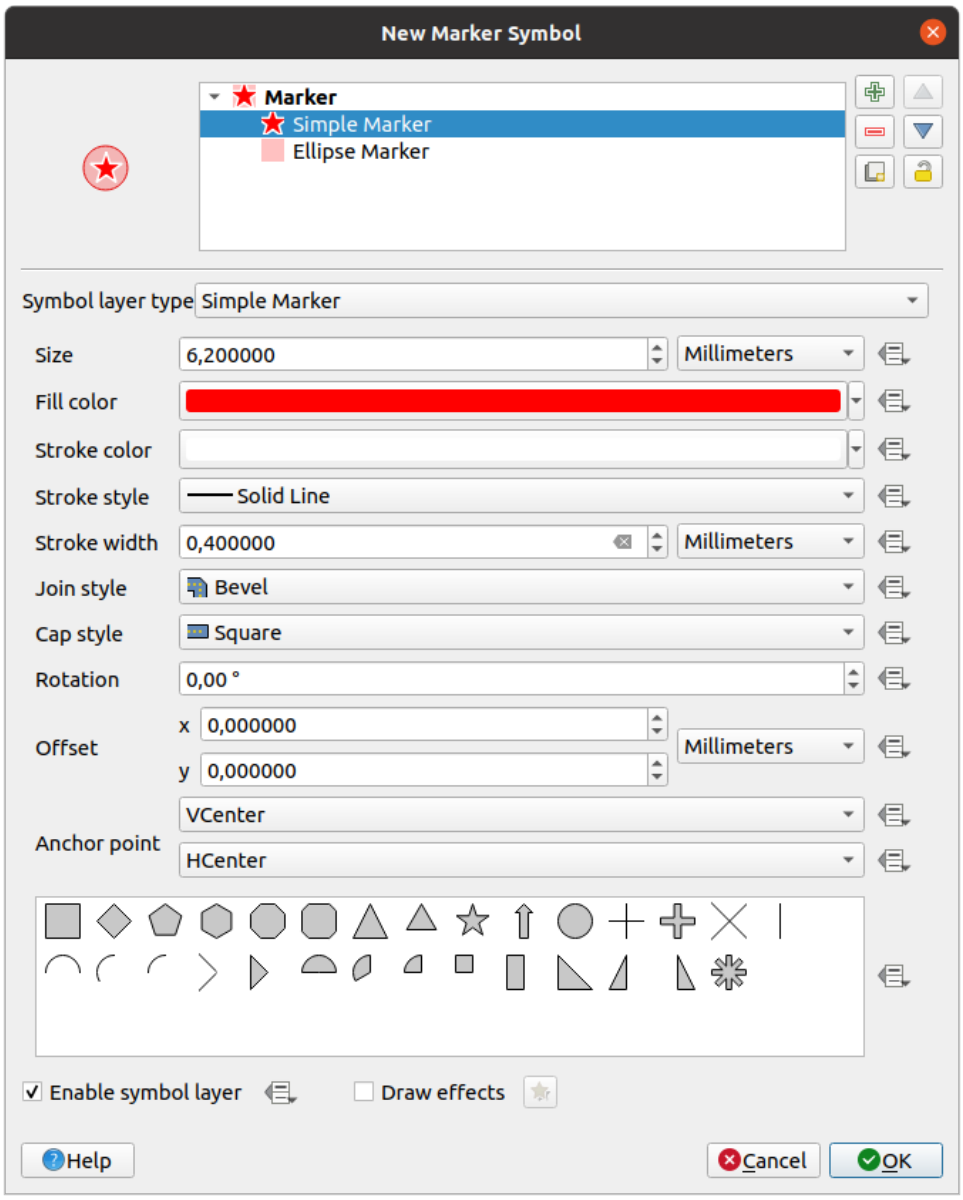
Simple marker	*Animated marker*	*Ellipse marker*

наставак на следећој страни

Табела 10.1 – настављено са претходне стране

Filled marker	*Font marker*	*Geometry generator*
Mask	*Raster image marker*	*Vector Field marker*
SVG marker		

- **Simple marker** (default)



Слика10.11: Designing a Simple Marker Symbol

The simple marker symbol layer type has the following properties:

- *Size* in various supported units
- *Fill color*
- *Stroke color*, *Stroke style* from a predefined list and *Stroke size*
- *Join style*: it can be **Bevel**, **Miter** or **Round**
- *Cap style*: it can be **Square**, **Flat** or **Round**

  - *Rotation*
  - *Offset* in *X* and *Y* directions from the feature
  - *Anchor point*: defining the quadrant point on the symbol to settle as placement origin. This is the point the *Offset* is applied on.
- **Animated marker** (see *Animated marker*)
- **Ellipse marker**: a simple marker symbol layer, with customizable width and height
- **Filled marker**: similar to the simple marker symbol layer, except that it uses a *fill sub symbol* to render the marker. This allows use of all the existing QGIS fill (and stroke) styles for rendering markers, e.g. gradient or shapeburst fills.
- **Font marker**: similar to the simple marker symbol layer, except that it uses installed fonts to render the marker. Its additional properties are:
  - *Font family*
  - *Font style*
  - *Character(s)*, representing the text to display as symbol. They can be typed in or selected from the font characters collection widget and you can live *Preview* them with the selected settings.
- **Geometry generator** (see *The Geometry Generator*)
- **Mask**: its sub-symbol defines a mask shape whose color property will be ignored and only the opacity will be used. This is convenient when the marker symbol overlaps with labels or other symbols whose colors are close, making it hard to decipher. More details at *Masks Properties*.
- **Raster image marker**: use an image (`PNG`, `JPG`, `BMP` …) as marker symbol. The image can be a file on the disk, a remote URL, embedded in the style database (*more details*) or it can be encoded as a base64 string. Width and height of the image can be set independently or using the  Lock aspect ratio. The size can be set using any of the *common units* or as a percentage of the image's original size (scaled by the width).
- **Vector Field marker** (see *The Vector Field Marker*)
- **SVG marker**: provides you with images from your SVG paths (set in *Settings* ► *Options...* ► *System* menu) to render as marker symbol. Width and height of the symbol can be set independently or using the  Lock aspect ratio. Each SVG file colors and stroke can also be adapted. The image can be a file on the disk, a remote URL, embedded in the style database (*more details*) or it can be encoded as a base64 string.

  The symbol can also be set with *Dynamic SVG parameters*. See *Parametrizable SVG* section to parametrize an SVG symbol.

> **Белешка**
>
> SVG version requirements
>
> QGIS renders SVG files that follow the SVG Tiny 1.2 profile, intended for implementation on a range of devices, from cellphones and PDAs to laptop and desktop computers, and thus includes a subset of the features included in SVG 1.1 Full, along with new features to extend the capabilities of SVG.
>
> Some features not included in these specifications might not be rendered correctly in QGIS.

Line Symbols

Appropriate for line geometry features, line symbols have the following symbol layer types:

Табела10.2: List of line symbol layer types

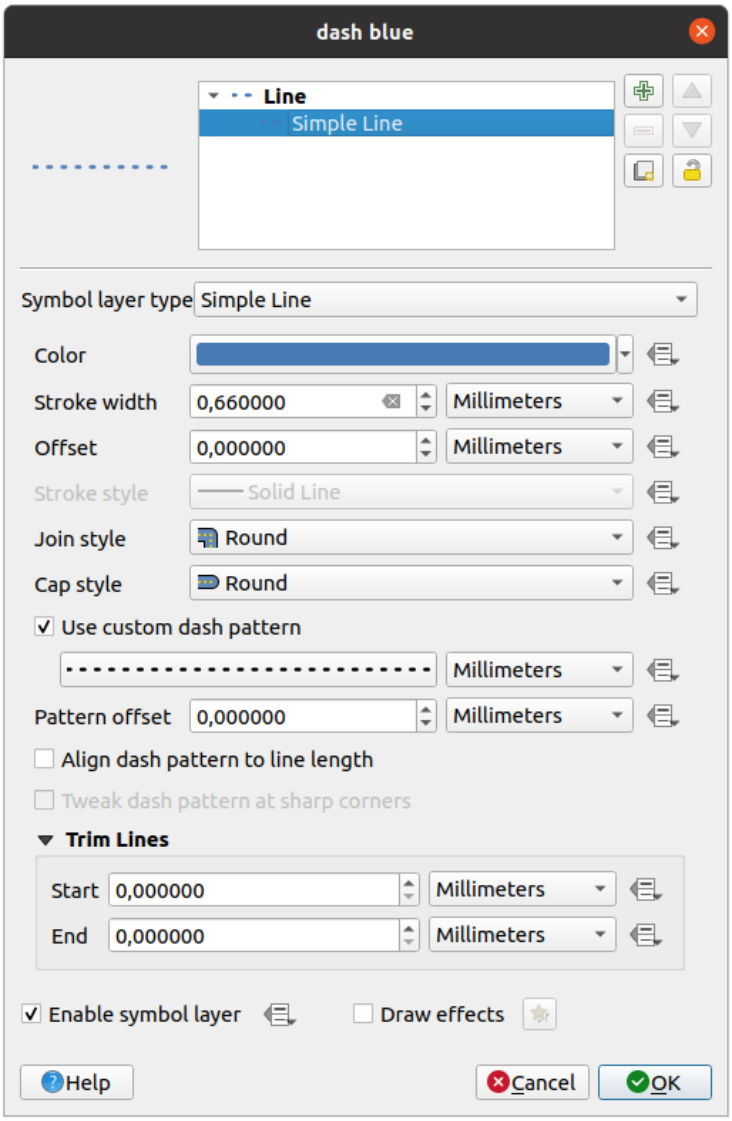
Simple line	*Arrow*	*Filled line*
Geometry generator	*Hashed line*	*Interpolated line*

наставак на следећој страни

Табела 10.2 – настављено са претходне стране

Linear referencing	*Lineburst*	*Marker line*
Raster line		

- **Simple line** (default)



Слика10.12: Designing a Simple Line Symbol

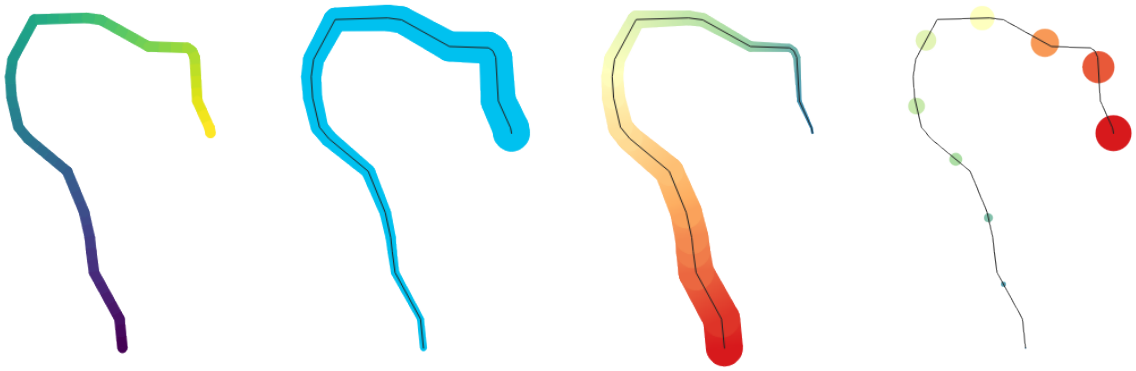
The simple line symbol layer type has many of the same properties as the *simple marker symbol*, and in addition:

- *Use custom dash pattern*: overrides the *Stroke style* setting with a custom dash. You would need to define length of consecutive dashes and spaces shaping the model, in the chosen unit. The total length of the pattern is displayed at the bottom of the dialog.
- *Pattern offset*: the positioning of the dashes/spaces in the line can be tweaked, so that they can be placed at nicer positions to account for corners in the line (also can be used potentially to „align" adjacent dash pattern borders)
- *Align dash pattern to line length*: the dash pattern length will be adjusted so that the line will end with a complete dash element, instead of a gap.

  - *Tweak dash pattern at sharp corners*: dynamically adjusts the dash pattern placement so that sharp corners are represented by a full dash element coming into and out of the sharp corner. Dependent on *Align dash pattern to line length*.
  - *Trim lines* from *Start* and/or *End*: allows for the line rendering to trim off the first x mm and last y mm from the actual line string when drawing the line. It supports a range of *units*, including percentage of the overall line length, and can be data defined for extra control. The start/end trim distance can be used e.g. when creating complex symbols where a line layer should not overlap marker symbol layers placed at the start and end of the line.

- **Arrow**: draws lines as curved (or not) arrows with a single or a double head with configurable (and data-defined):
  - *Head type*
  - *Arrow type*
  - *Arrow width*
  - *Arrow width at start*
  - *Head length*
  - *Head thickness*
  - *Offset*

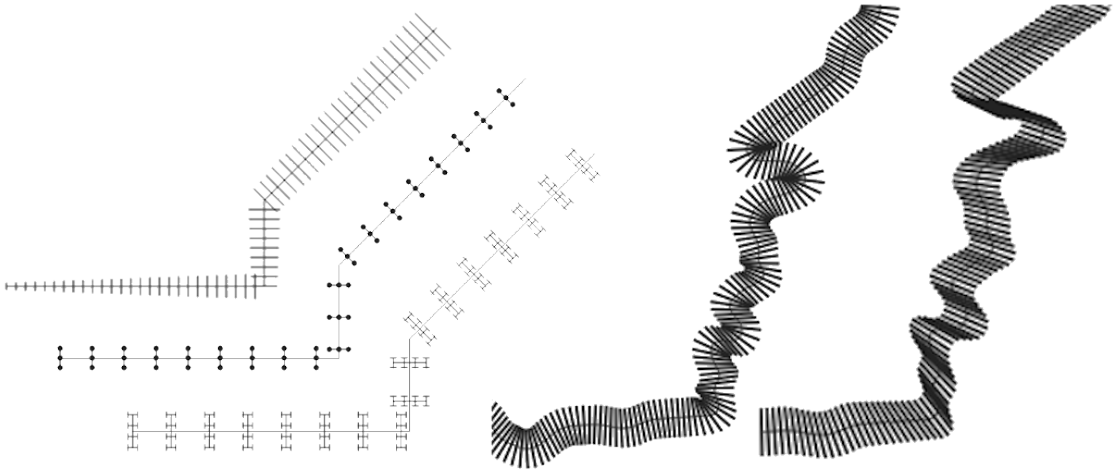
  It is possible to create *Curved arrows* (the line feature must have at least three vertices) and *Repeat arrow on each segment*. It also uses a *fill symbol* such as gradients or shapeburst to render the arrow body. Combined with the geometry generator, this type of layer symbol helps you representing flow maps.
- **Geometry generator** (see *The Geometry Generator*)
- **Interpolated line**: allows to render a line whose *Stroke width* and/or *Color* may be constant (given a *Fixed width* and *Single color* parameters) or vary along the geometry. When varying, necessary inputs are:
  - *Start value* and *End value*: Values that will be used for interpolation at the extremities of the features geometry. They can be fixed values, feature's attributes or based on an expression.
  - *Min. value* and *Max. value*: Values between which the interpolation is performed. Press the Load button to automatically fill them based on the minimum and maximum start/end values applied to the layer.
  - Only available for the stroke option:
    * *Min. width* and *Max. width*: define the range of the varying width. *Min. width* is assigned to the *Min. value* and *Max. width* to the *Max. value*. A *unit* can be associated.
    * *Use absolute value*: only consider absolute value for interpolation (negative values are used as positive).
    * *Ignore out of range*: by default, when the `[start value - end value]` range of a feature is not included in the `[min. value - max. value]` range, the out-of-bounds parts of the feature's geometry are rendered with the min or max width. Check this option to not render them at all.
  - For varying color, you can use any of the interpolation methods of *color ramp classification*



Слика10.13: Examples of interpolated lines

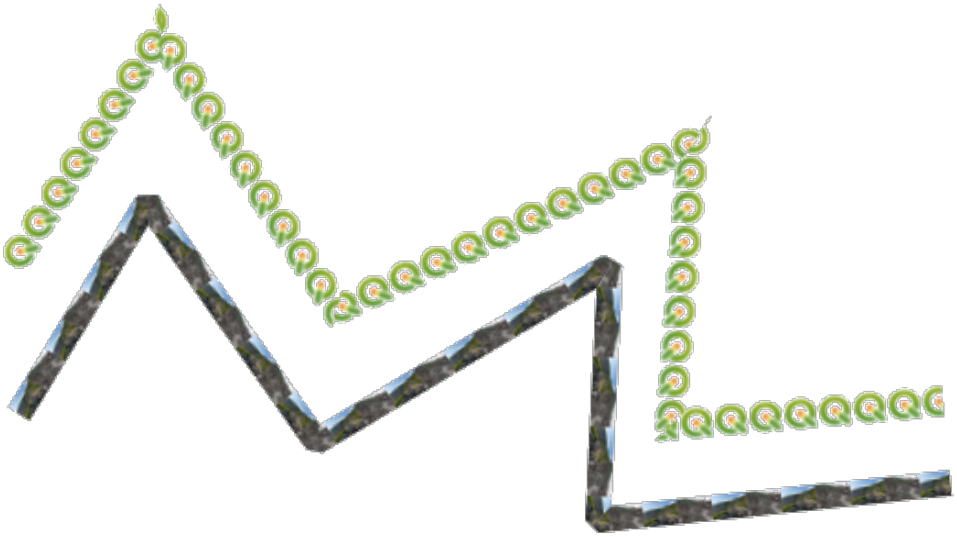
- **Marker line**: repeats a *marker symbol* over the length of a line.
  - The *Marker placement* can be set using a regular interval setting (starting from the first vertex) and/or the line geometry property (on first or last vertex, inner vertices, the central point of the line or of each segment, or on every curve point).
  - When first or last vertex placement is enabled, the *Place on every part extremity* option will make the markers render also at the first or last vertex for every part of multipart geometries.
  - *Offset along line*: the markers placement can also be given an offset along the line, in the ref:*unit <unit_selector>* of your choice (millimeters, points, map unit, meters at scale, percentage, …):
    - A positive value offsets the markers symbols in the line direction (with *On first vertex* and *With interval* placements) and backwards (with *On last vertex* placement).
    - A negative value on a not closed line will result in no offset (for *On first vertex* and *On last vertex* placements) or backwards offset of the symbols (from the last vertex).
    - With a closed ring, QGIS treats the offset as continuing to loop around the ring (forward or backward). E.g. setting the offset to 150% (resp. -10% or -110%) results in the offset being treated as 50% (resp. 90%) of the length of the closed ring.
  - The *Rotate marker to follow line direction* option sets whether each marker symbol should be oriented relative to the line direction or not.

    Because a line is often a succession of segments of different directions, the rotation of the marker is calculated by averaging over a specified distance along the line. For example, setting the *Average angle over* property to `4mm` means that the two points along the line that are `2mm` before and after the symbol placement are used to calculate the line angle for that marker symbol. This has the effect of smoothing (or removing) any tiny local deviations from the overall line direction, resulting in much nicer visual orientations of the marker line symbols.
  - *Line offset*: the marker symbols can also be offset from the line feature.
- **Hashed line**: repeats a line segment (a hash) over the length of a line symbol, with a line sub-symbol used to render each individual segment. In other words, a hashed line is like a marker line in which marker symbols are replaced with segments. As such, the hashed lines have the *same properties* as marker line symbols, along with:
  - *Hash length*
  - *Hash rotation*



Слика10.14: Examples of hashed lines

- **Raster line**: renders and repeats a raster image following the length of a line feature shape. The *Stroke width*, *Offset*, *Join style*, *Cap style* and *Opacity* can be adjusted.



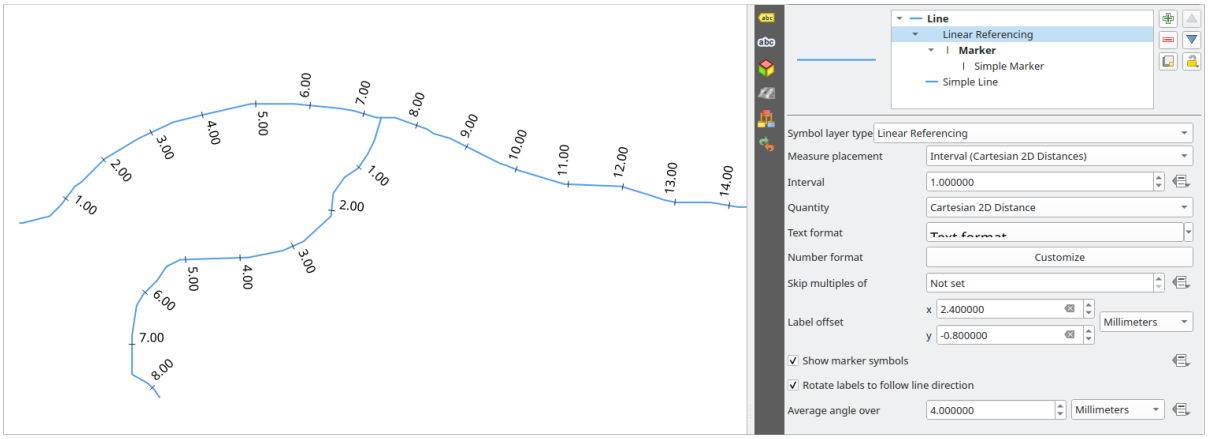
Слика10.15: Examples of raster lines

- **Linear referencing**: allows placing text labels at regular intervals along a line, or at positions corresponding to existing vertices. Positions can be calculated using Cartesian distances, or interpolated from z/m values. Parameters include:
  - *Measure placement*: Labels can be placed at regular cartesian 2d distances, at regular linearly interpolated spacing calculated using the Z or M values in geometries, or at existing vertices. A fixed or data-defined *Interval* is given.
  - *Quantity*: Label displayed at each position represent the running distance along the line, or the linearly interpolated Z or M value. Labels are rendered using the full range of *text* and *number* formatting functionalities available in QGIS.

Белешка

When using the distance-based placement or labels, the distances are calculated using 2D only. Cartesian calculations are based on the original layer CRS.

- *Skip multiples of*: If set, labels which are a multiple of this value will be skipped over. This allows construction of complex referencing labels, e.g. where a symbol has two linear referencing symbol layers, one set to label every 100m in a small font, skipping multiples of 1000, and a second set to label every 1000m in a big bold font.
- *Average angle over*: Labels are rendered using an angle calculated by averaging the linestring, so sharp tiny jaggies don't result in unslightly label rotation.
- *Show marker symbols*, at referenced points in the line feature, using a full QGIS marker symbol. This allows e.g. showing a cross-hatch at the labeled point, for a „ruler" style line.



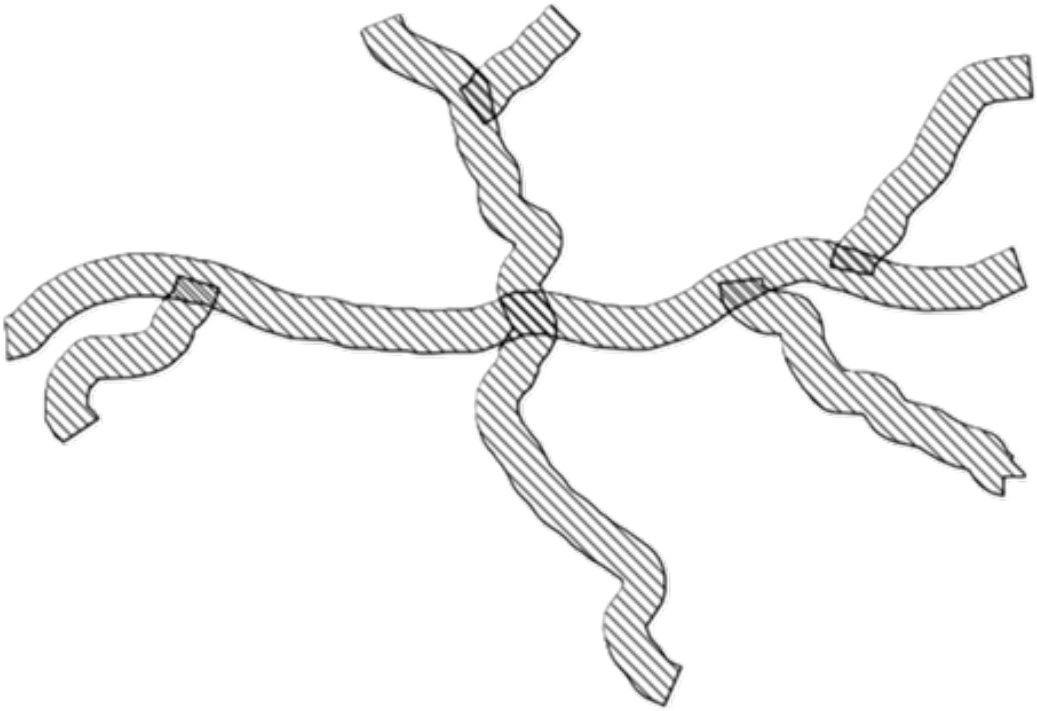
Слика10.16: Drawing linear labels along line feature

- **Lineburst**: renders a gradient along the width of a line. You can choose between *Two color* or *Color ramp* and the *Stroke width*, *Offset*, *Join style*, *Cap style* can be adjusted.



Слика10.17: Examples of lineburst lines

- **Filled line**: renders the interior of the lines using a fill symbol allowing for lines filled with gradients, line hatches, etc. The *Stroke width*, *Offset*, *Join style*, *Cap style* can be adjusted.



Слика10.18: Examples of filled lines

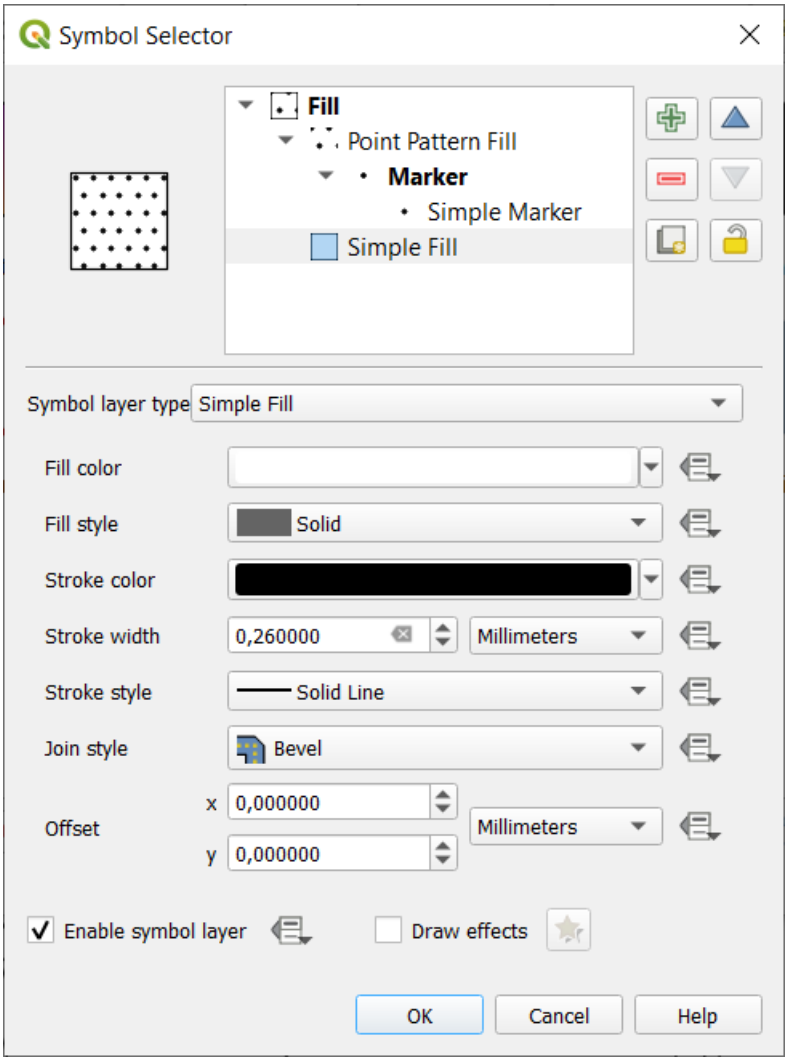
Fill Symbols

Appropriate for polygon geometry features, fill symbols have also several symbol layer types:

Табела10.3: 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	

- **Simple fill** (default): fills a polygon with a uniform color



Слика10.19: Designing a Simple Fill Symbol

- **Centroid fill**: places a *marker symbol* at the centroid of the visible feature. The position of the marker may

not be the real centroid of the feature, because calculation takes into account the polygon(s) clipped to area visible in map canvas for rendering and ignores holes. Use the *geometry generator symbol* if you want the exact centroid.

You can:

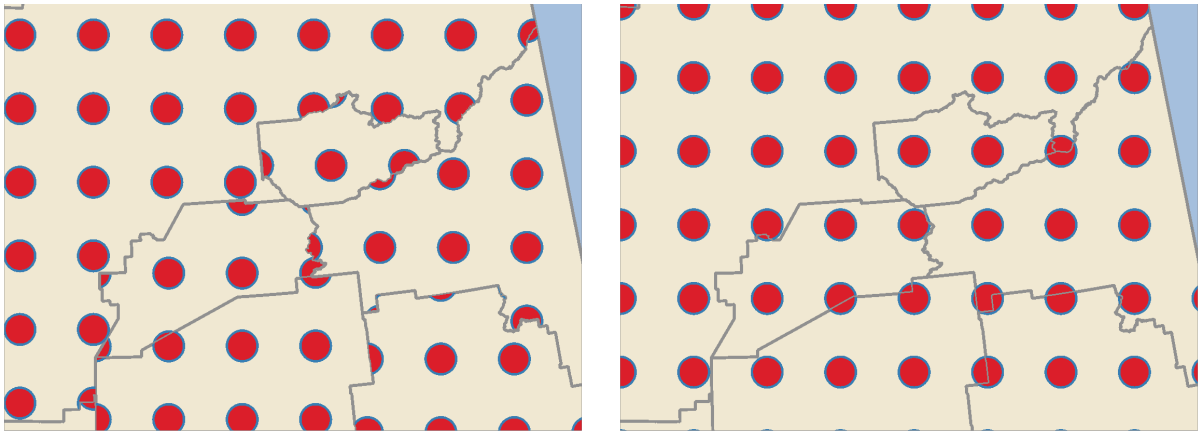
- *Force placement of markers inside polygons*
- *Draw markers on every part of multi-part features* or place the point only on its biggest part
- display the marker symbol(s) in whole or in part, keeping parts overlapping the current feature geometry (*Clip markers to polygon boundary*) or the geometry part the symbol belongs to (*Clip markers to current part boundary only*)

- **Geometry generator** (see *The Geometry Generator*)
- **Gradient fill**: uses a radial, linear or conical gradient, based on either simple two color gradients or a predefined *gradient color ramp* to fill polygons. The gradient can be rotated and applied on a single feature basis or across the whole map extent. Also start and end points can be set via coordinates or using the centroid (of feature or map). A data-defined offset can be defined.
- **Line pattern fill**: fills the polygon with a hatching pattern of *line symbol layer*. You can set:
  - *Alignment*: defines how the pattern is positioned relative to the feature(s):
    - *Align pattern to feature*: lines are rendered within each feature
    - *Align pattern to map extent*: a pattern is rendered over the whole map extent, allowing lines to align nicely across features
  - *Rotation* of the lines, counter-clockwise
  - *Spacing*: distance between consecutive lines
  - *Offset* distance of the lines from the feature boundary
  - *Clipping*: allows to control how lines in the fill should be clipped to the polygon shape. Options are:
    - *Clip During Render Only*: lines are created covering the whole bounding box of the feature and then clipped while drawing. Line extremities (beginning and end) will not be visible.
    - *Clip Lines Before Render*: lines are clipped to the exact shape of the polygon prior to rendering. Line extremities (including cap styles, start/end marker line objects, …) will be visible, and may sometimes extend outside of the polygon (depending on the line symbol settings).
    - *No Clipping*: no clipping at all is done - lines will cover the whole bounding box of the feature

  Белешка

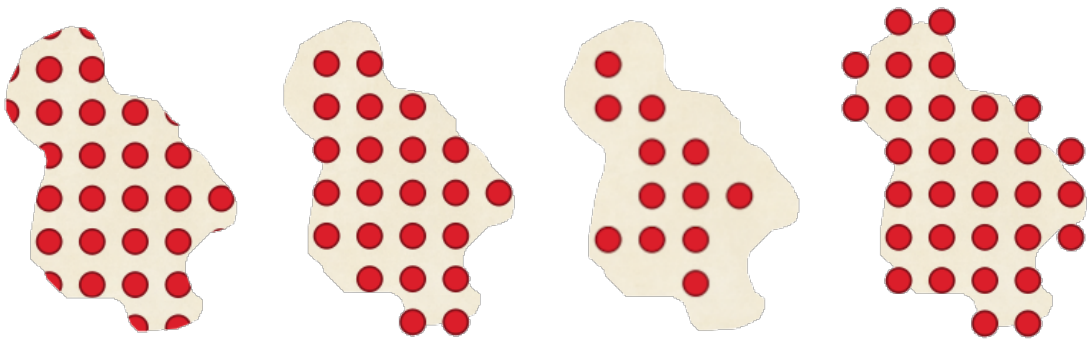
  When exporting maps to vector formats such as PDF or SVG, or when the line symbol in the pattern has data-defined properties, QGIS renders the line pattern fill using a line-by-line vector-based approach. This results in smaller output file sizes, higher quality exports, and improved editability of the pattern lines in external vector graphics applications.

- **Point pattern fill**: fills the polygon with a grid pattern of *marker symbol*. You can set:
  - *Alignment*: defines how the pattern is positioned relative to the feature(s):
    - *Align pattern to feature*: marker lines are rendered within each feature
    - *Align pattern to map extent*: a pattern is rendered over the whole map extent, allowing markers to align nicely across features



Слика10.20: Aligning point pattern to feature (left) and to map extent (right)

- *Distance*: *Horizontal* and *Vertical* distances between consecutive markers
- *Displacement*: a *Horizontal* (resp. *Vertical*) offset of alignment between consecutive markers in a column (resp. in a row)
- *Offset*: *Horizontal* and *Vertical* distances from the feature boundary
- *Clipping*: allows to control how markers in the fill should be clipped to the polygon shape. Options are:
  - *Clip to shape*: markers are clipped so that only the portions inside the polygon are visible
  - *Marker centroid within shape*: only markers where the center of the marker falls inside the polygon are drawn, but these markers won't be clipped to the outside of the polygon
  - *Marker completely within shape*: only markers which fall completely within the polygon are shown
  - *No clipping*: any marker which intersects at all with the polygon will be completely rendered (strictly speaking its the „intersects with the bounding box of the marker")



Слика10.21: Clipping markers in fill - From left to right: Clip to shape, Marker centroid within shape, Marker completely within shape, No clipping

- *Rotation* of the whole pattern, clockwise
- The *Randomize pattern* group setting allows each point in a point pattern fill to be randomly shifted up to the specified maximum distance *Horizontally* or *Vertically*. You can specify the maximum offset in any supported units, such as millimeters, points, map units, or even „percentage" (where percentage is relative to the pattern width or height).

  You can set an optional random number seed to avoid the symbol patterns „jumping" around between map refreshes. Data defined overrides are also supported.

Белешка

The main difference between the *Randomize pattern* and the *random marker fill* symbol type is that the random offset with a point pattern allows for quasi-„regular" placement of markers – because the points in the pattern are effectively constrained to a grid, this allows creation of semi-random fills which don't have empty areas or overlapping markers. (As opposed to the random marker fill, which will always place points completely randomly… sometimes resulting in visual clusters of points or unwanted empty areas).

- **Random marker fill**: fills the polygon with a *marker symbol* placed at random locations within the polygon boundary. You can set:
  - *Count method*: whether the number of marker symbols to render is considered as an absolute count or density-based
  - *Point count*: the number of marker symbols to render,
  - an optional random number *seed*, to give consistent placement
  - *Density area*: in case of density-based count method, ensures the fill density of markers remains the same on different scale / zoom levels of markers whenever maps are refreshed (also allows random placement to play nice with QGIS server and tile-based rendering)
  - *Clip markers to polygon boundary*: whether markers rendered near the edges of polygons should be clipped to the polygon boundary or not
- **Raster image fill**: fills the polygon with tiles from a raster image (`PNG` `JPG`, `BMP` …). The image can be a file on the disk, a remote URL or an embedded file encoded as a string (*more details*). Options include (data defined) opacity, size, coordinate mode (object or viewport), rotation and offset. Under the option Size you can independently adjust the width and height of the fill pattern, enabling stretched raster fills in either the horizontal or vertical directions. The image width and height can be set using any of the *common units* or as a percentage of the original size.
- **SVG fill**: fills the polygon using *SVG markers* of a given size (*Texture width*).
- **Shapeburst fill**: buffers a gradient fill, where a gradient is drawn from the boundary of a polygon towards the polygon's centre. Configurable parameters include distance from the boundary to shade, use of color ramps or simple two color gradients, optional blurring of the fill and offsets.
- **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.

Белешка

When geometry type is polygon, you can choose to disable the automatic clipping of lines/polygons to the canvas extent. In some cases this clipping results in unfavourable symbology (e.g. centroid fills where the centroid must always be the actual feature's centroid).

Parametrizable SVG

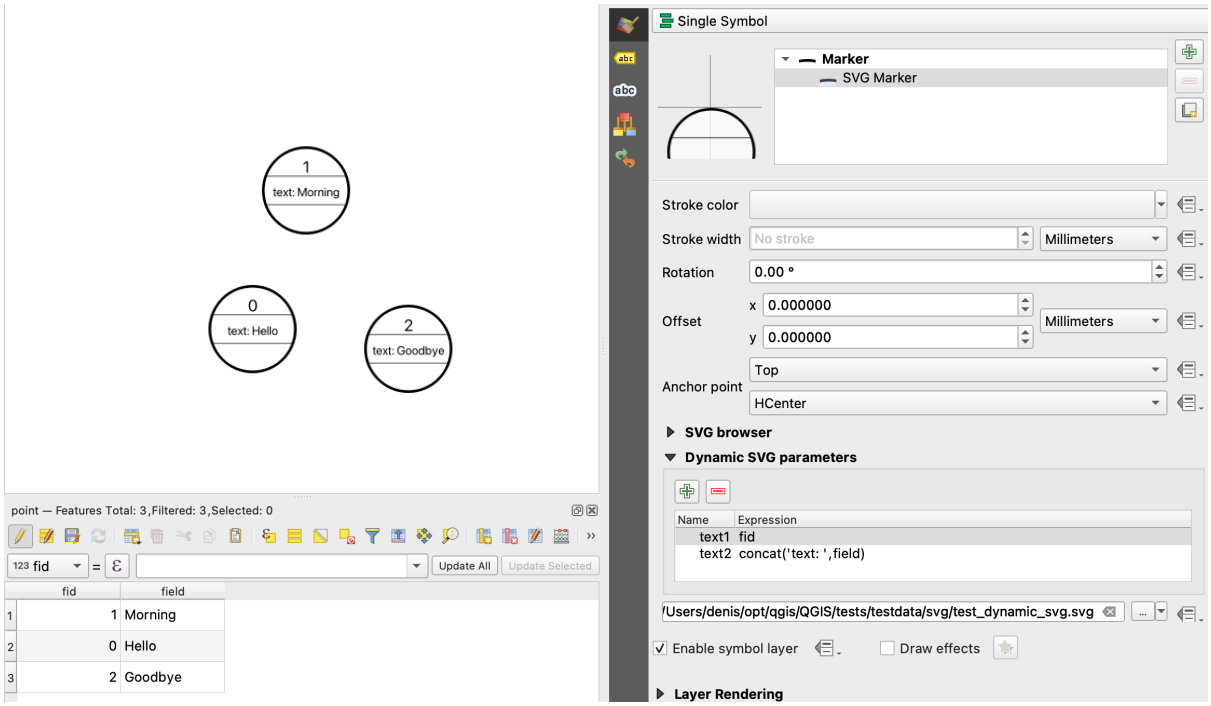
You have the possibility to change the colors of a *SVG marker*. You have to add the placeholders `param(fill)` for fill color, `param(fill-opacity)` for fill opacity, `param(outline)` and `param(outline-opacity)` for stroke color and opacity respectively, and `param(outline-width)` for stroke width. These placeholders can optionally be followed by a default value, e.g.:

```
<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>
```

More generally, SVG can be freely parametrized using `param(param_name)`. This param can either be used as an attribute value or a node text:

```
<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>
```

The parameters can then be defined as expressions in the *Dynamic SVG parameters* table.



Слика10.22: Dynamic SVG parameters table

Белешка

QGIS is looking for a complete SVG node. So if your parameter is within a more complex node, you need to inject the complete node with the expression. For instance, `transform="rotate(param(angle)"` will not work. Instead, you need to do `transform="param(rotation)"` and `rotation` parameter will be defined with the expression `'rotate(' || coalesce(my_field, 0) || ')'`.

The Geometry Generator

Available with all types of symbols, the *geometry generator* symbol layer allows to use *expression syntax* to generate a geometry on the fly during the rendering process. The resulting geometry does not have to match with the original *Geometry type* and you can add several differently modified symbol layers on top of each other.

A *Units* property can be set: when the geometry generator symbol is not applied to a layer (e.g., it is used on a layout item), this allows more control over the generated output.

Some examples:

- Render symbol as the centroid of a feature

```
centroid( $geometry )
```

- Visually overlap features within a 100 map units distance from a point feature, i.e generate a 100m buffer around the point

```
buffer( $geometry, 100 )
```

- Create a radial effect of points surrounding the central feature point when used as a MultiPoint geometry generator

```
collect_geometries(
  array_foreach(
```

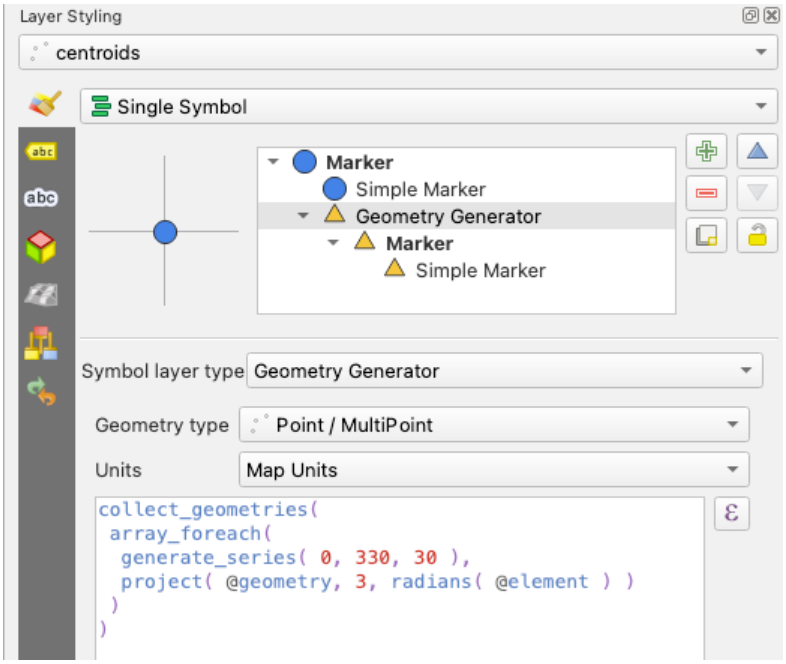
(наставак на следећој страни)

(настављено са претходне стране)

```
    generate_series( 0, 330, 30 ),
    project( $geometry, 3, radians( @element ) )
  )
)
```

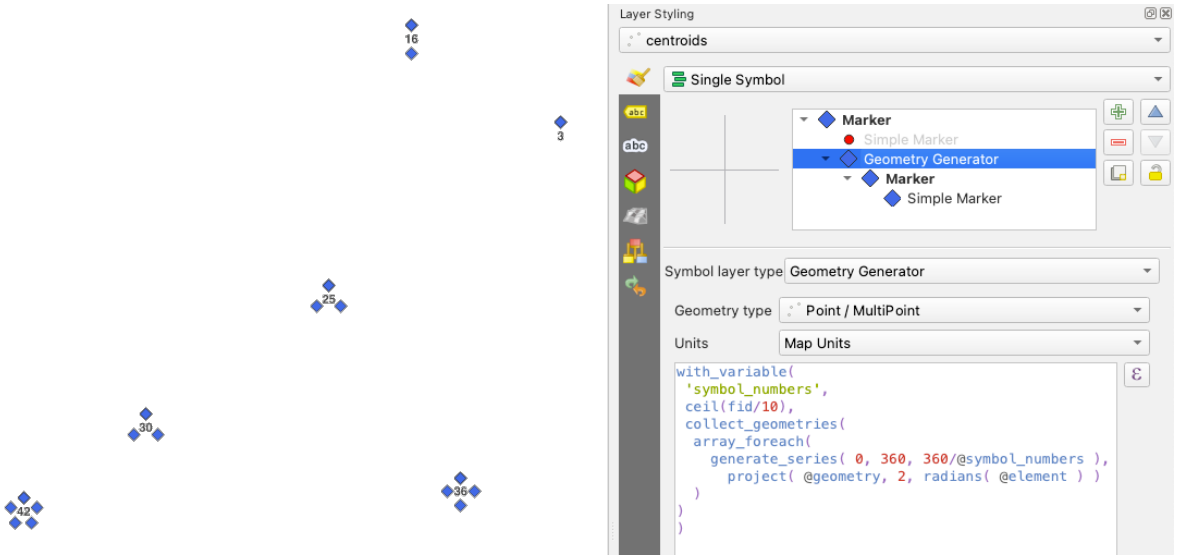






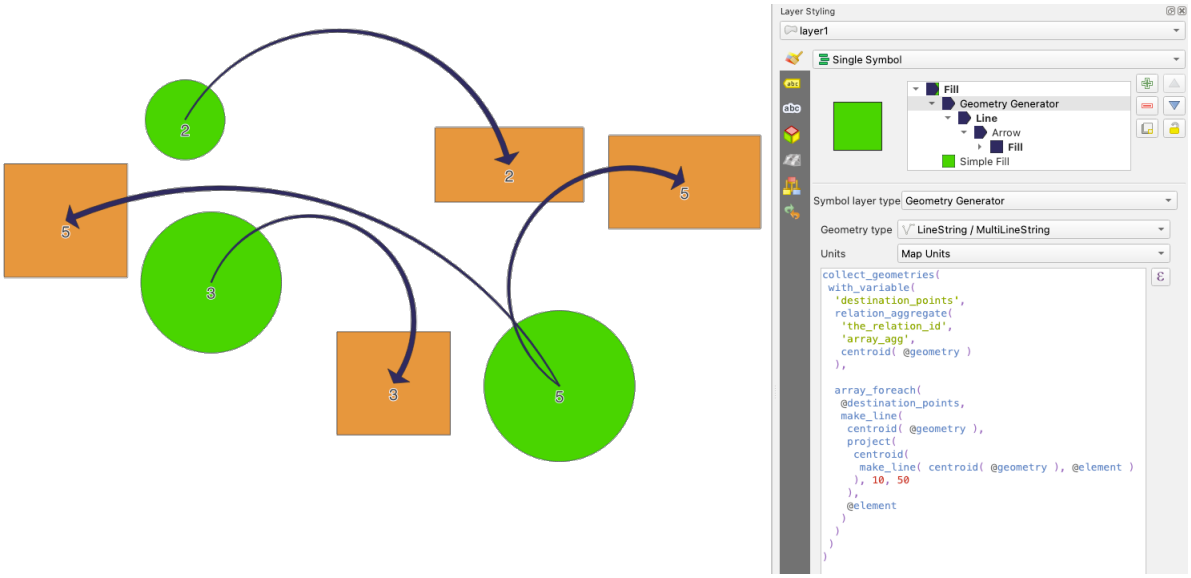
- Create a radial effect of points surrounding the central feature point. The number of points varies based on a field.

```
with_variable(
  'symbol_numbers',
  ceil(fid/10),
  collect_geometries(
    array_foreach(
      generate_series( 0, 360, 360/@symbol_numbers ),
      project( $geometry, 2, radians( @element ) )
    )
  )
)
```



- Create a curved arrow line connecting features of two layers based on their *relation*

```
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
      )
    )
  )
)
```



The Vector Field Marker

The vector field marker is used to display vector field data such as earth deformation, tidal flows, and the like. It displays the vectors as lines (preferably arrows) that are scaled and oriented according to selected attributes of data points. It can only be used to render point data; line and polygon layers are not drawn by this symbology.

The vector field is defined by attributes in the data, which can represent the field either by:

- **cartesian** components (`x` and `y` components of the field)
- or **polar** coordinates: in this case, attributes define `Length` and `Angle`. The angle may be measured either clockwise from north, or Counterclockwise from east, and may be either in degrees or radians.
- or as **height only** data, which displays a vertical arrow scaled using an attribute of the data. This is appropriate for displaying the vertical component of deformation, for example.

The magnitude of field can be scaled up or down to an appropriate size for viewing the field.

Animated marker

Animated marker symbol type allows you to use a `.GIF`, `.WebP`, `.MNG`, etc. animation file to represent points on your map. You can specify:

- *File* path,
- *Frame rate*: number of steps that are shown per second, indicating how fast the animation is played,
- *Size* in any *supported unit*,
- *Opacity*,
- *Rotation*,
- *Offset* in *x* and *y* directions from the marker position,
- *Anchor point*

There are two ways to handle animated symbols:

- **When your map is not configured as an animation** (i.e. it's a standard QGIS project without animations), the frame for the animated markers will be determined solely by the current timestamp.
- **When your map is** *configured as an animation*, the animated markers will sync with the animation's timeline. This means that animated markers will pause when the animation is paused, progress with the animation, and so forth. The map will also be redrawn according to the frame rate established for temporal animation. This mode is also applied when exporting an animation using the temporal controller.

10.3 Setting a label

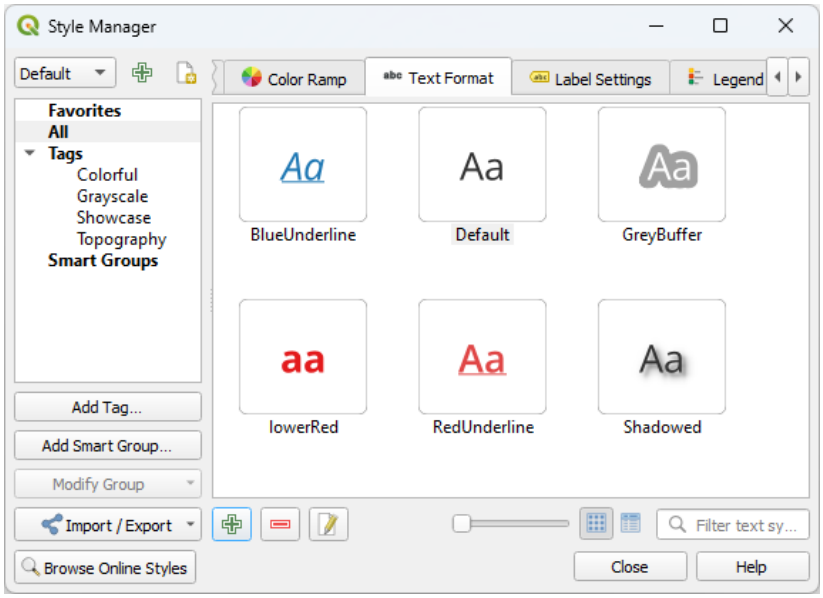
Labels are textual information you can display on maps. They add details you could not necessarily represent using symbols, and may refer to vector features, raster cells, mesh elements, or simple annotations on the map… Two types of text-related items are available in QGIS:

- *Text Format*: defines the appearance of the text, including *font, size, colors*, *shadow*, *background*, *buffer*, …

  They can be used to render texts over the map (layout/map title, decorations, scale bar, …), usually through the *font* widget.

  To create a *Text Format* item:

  1. Open the *Style Manager* dialog
  2. Activate the *Text format* tab



Слика10.23: Text formats in Style Manager dialog

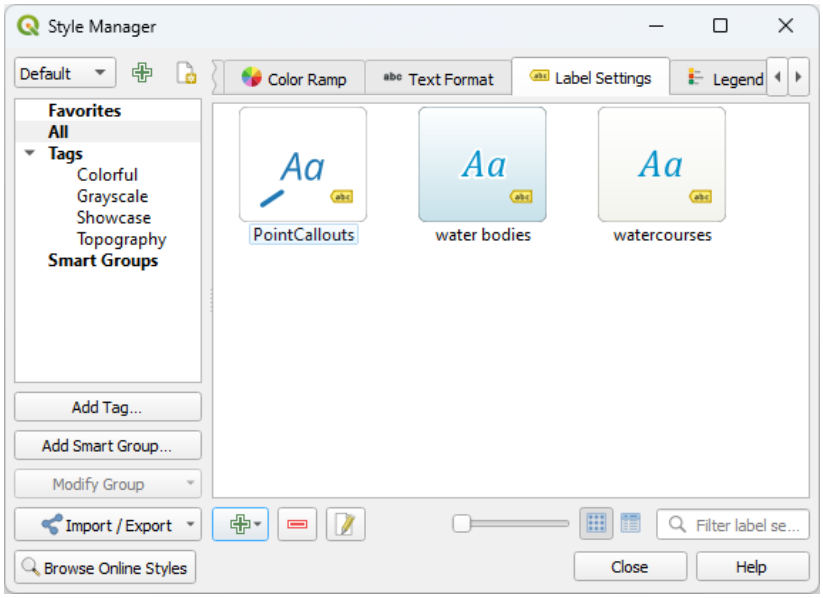
  3. Press the Add item button. The *Text Format* dialog opens for *configuration*. As usual, these properties are *data-definable*.

- *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* …).

  They are used to configure smart labelling for vector and mesh layers through the *Labels* tab of the vector or mesh *Layer Properties* dialog or *Layer Styling* panel or using the Layer Labeling Options button of the *Label toolbar*.

  To create a *Label Settings* item:

  1. Open the *Style Manager* dialog
  2. Activate the *Label Settings* tab



Слика10.24: Label Settings in Style Manager dialog

3. Press the Add item menu and select the entry corresponding to the geometry type of the features you want to label.

The *Label Settings* dialog opens with the following properties. As usual, these properties are *data-definable*.

10.3.1 Formatting the label text

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:

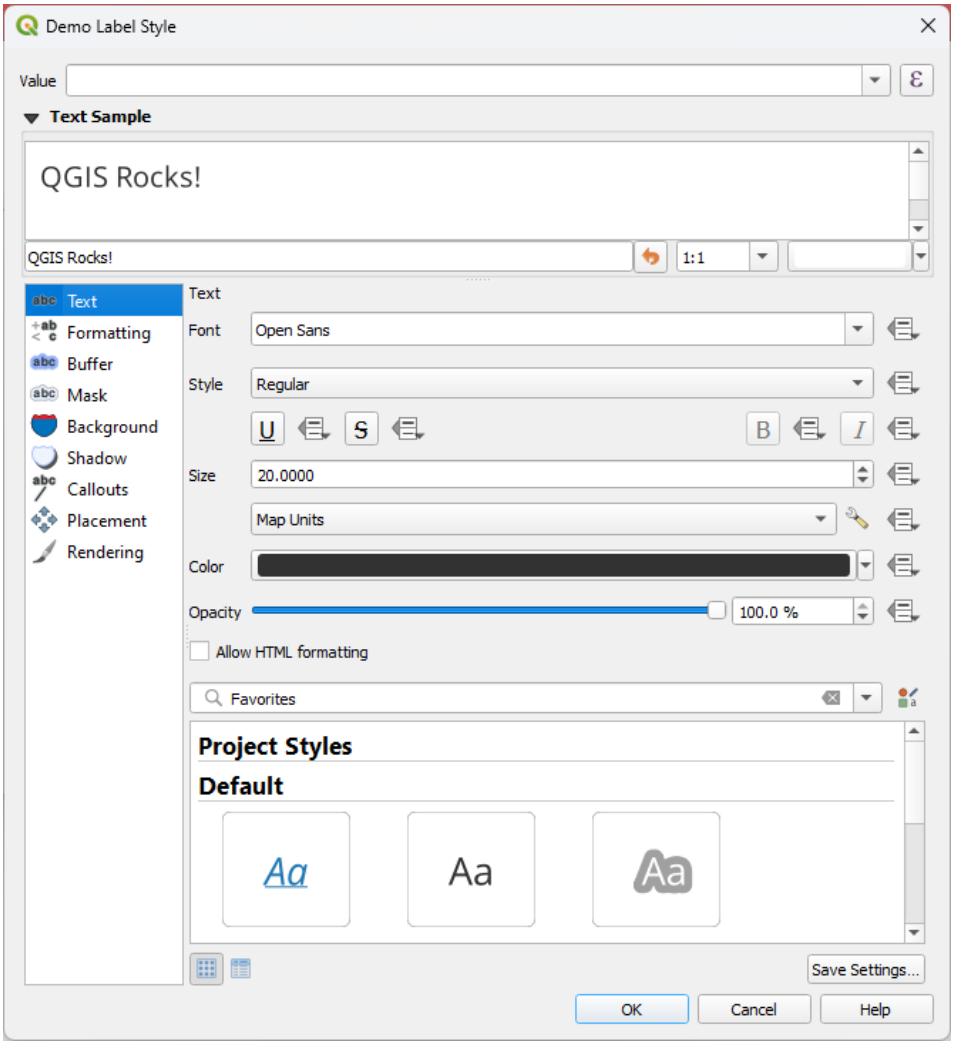
Табела10.4: Label properties by data type

Properties tab	Text format	Label settings	Vector or mesh layer	raster layer
Text	☑	☑	☑	☑
Formatting	☑	☑	☑	☑
Buffer	☑	☑	☑	☑
Mask		☑	☑	
Background	☑	☑	☑	☑
Shadow	☑	☑	☑	☑
Callout		☑	☑	
Placement		☑	☑	☑
Rendering		☑	☑	☑

Пажња

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 tab



Слика10.25: Labels settings - Text tab

In the abc *Text* tab, you can set:

- the *Font*, from the ones available on your machine
- the *Style*: along with the common styles of the font, you can set whether the text should be underlined or striked through
- the *Size* in any *supported unit*
- the *Color*
- the *Opacity*
- and *Allow HTML Formatting* enables the use of a subset of HTML tags and CSS rules to customize the label.

At the bottom of the tab, a widget shows a filterable list of compatible items stored in your *style manager database*. This allows you to easily configure the current text format or label setting based on an existing one, and also save a

new item to the style database: Press the *Save format…* or *Save settings…* button and provide a name and tag(s).

Белешка

When configuring a *Label Settings* item, text format items are also available in this widget. Select one to quickly overwrite the current *textual properties* of the label. Likewise, you can create/overwrite a text format from there.

Allow HTML Formatting

With *Allow HTML Formatting* enabled, you need to provide the HTML code in the *Value* field. Use whitespaces instead of tabs for any kind of indentation. The expression is parsed and any supported HTML tag overrides its corresponding setting in the labels properties. Because it is impossible to list and detail every HTML tag and CSS property that QGIS currently supports, we invite you to explore and test in your labels the ones supported by the underlying Qt library.

Examples of supported HTML tags:

- Text formatting, such as italic or bold, e.g.:

```
<i>QGIS</i> <b>rocks!</b>
```

- Superscript and subscript, where the text will be vertically $^{\text{super}}$ or $_{\text{sub}}$ aligned and automatically sized to 2/3 of the parent font size. You can also set a fixed font size for the superscript/subscript by including css rules, e.g.:

```
<sup style="font-size:33pt">my superscript text</sup>
```

- Text horizontal alignment, using either HTML the `align="xxx"` attribute or `center` tag. For HTML5 compatibility, prefer using the CSS property `text-align`.

Пажња

Horizontal alignment can not be used for curved labels.

- Structuring a block of text using header tags (such as `h1`, `h2`, …), or paragraphs (with `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>
```

- Image insertion: any image format readable by QGIS can be used in HTML label content. It can be served from local file paths, HTTP links, or base64 embedded content, using the `src="xxx"` attribute. Image sizes can be specified via the `width="##"` and `height="##"` attributes, in `points` unit. If width or height is not specified it will automatically be calculated from the original image size. Images are placed inline only, not as floating images, and not on curved text labels.

```
<img src="qgis.png" width=40 height=60>
```

Examples of supported CSS properties:

- Font properties (`color`, `font-family`, `font-size`, `font-weight`, `font-style`, `word-spacing`). Note that `word-spacing` will always use unit points.
- Text decorations such as underline, overline and line-through (`text-decoration`)
- Text alignment (`vertical-align`, `text-align`). Horizontal alignment can not be used for curved labels.
- 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.

  Пажња

  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.

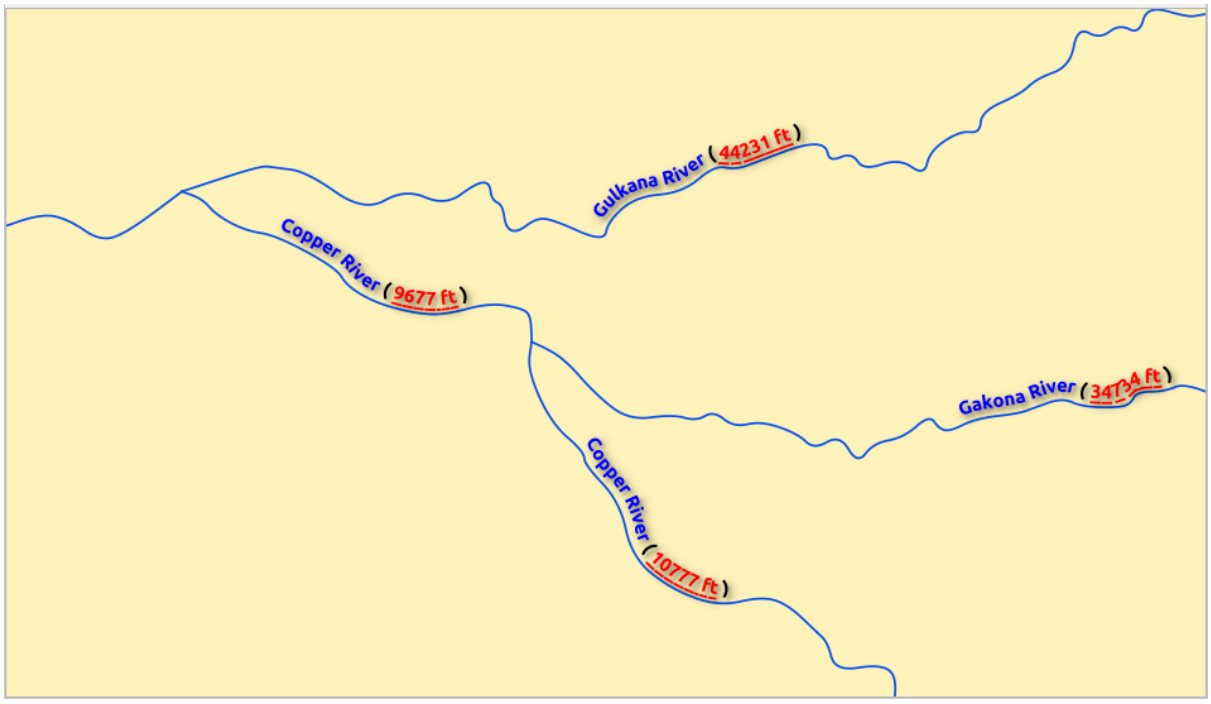
```
<div class="myDiv">
  <h2 style="margin-left: 5pt; margin-right: 10pt">QGIS still rocks...</h2>
  <p style="margin: 5pt 0pt -10pt 0pt">Thanks to you!!</p>
</div>
```

CSS properties can be set on HTML tags with the `style` attribute. The HTML tag `span` does not apply any formatting to text by itself and is ideal if you just want to apply CSS styling. A CSS property name and its value are separated by a colon (`:`). Multiple CSS properties are separated by semicolon (`;`), e.g.:

```
<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>
```

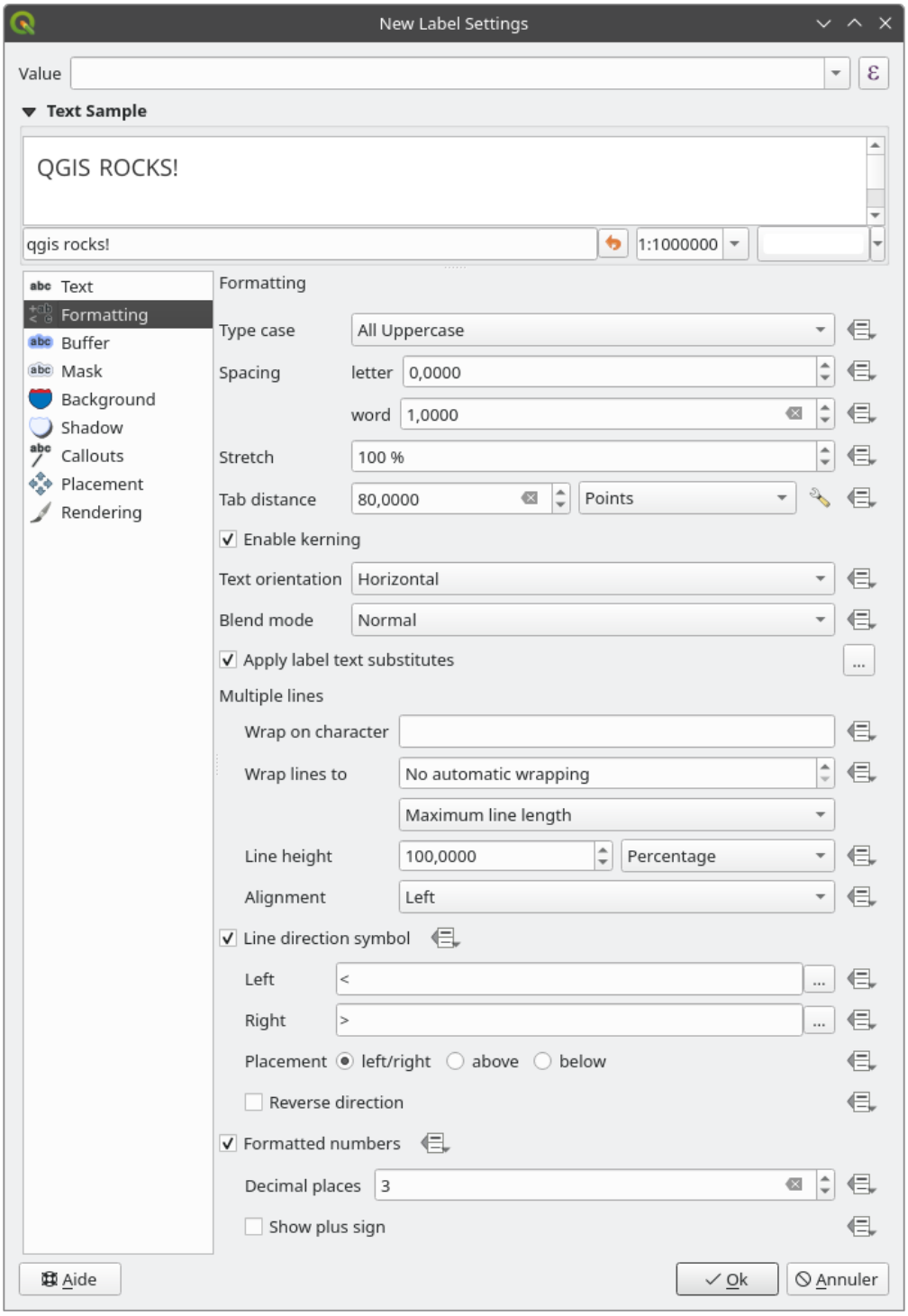
Below an example of a HTML-based expression and rendering (applies different colors and underline to the same label):

```
format(
  '<span style="color:blue">%1</span> ( <span style="color:red"><u>%2 ft</u></span>
↪ )',
  title( lower( "Name" ) ),
  round($length)
)
```



Слика10.26: Labeling with HTML formatting enabled

Formatting tab



Слика10.27: Label settings - Formatting tab

In the *Formatting* tab, you can:

- Use the *Type case* option to change the capitalization style of the text. You have the possibility to render the text as:
  - *No change*

  - *All uppercase*
  - *All lowercase*
  - *Small Caps*: renders lowercase characters as small caps.
  - *All Small Caps*: renders all characters as small caps regardless of their original case.
  - *Title case*: modifies the first letter of each word into capital, and turns the other letters into lower case if the original text is using a single type case. In case of mixed type cases in the text, the other letters are left untouched.
  - *Force first letter to capital*: modifies the first letter of each word into capital and leaves the other letters in the text untouched.
- Under *Spacing*, change the space between words and between individual letters.
- With *Tab distance* you can adjust the replacement spacing for `Tab` characters in the label, in any supported unit. This allows e.g. to properly separate or vertically align elements of a layout legend text.

  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.
- *Stretch* ratio: allows text to be horizontally stretched or condensed by a factor. Handy for tweaking the widths of fonts to fit a bit of extra text into labels.
- *Enable kerning* of the text font
- Set the *Text orientation* which can be *Horizontal* or *Vertical*. It can also be *Rotation-based* when setting a label (e.g., to properly label line features in *parallel* placement mode).
- Use the *Blend mode* option to determine how your labels will mix with the map features below them (more details at *Blending Modes*).
- The *Apply label text substitutes* option allows you to specify a list of texts to substitute to texts in feature labels (e.g., abbreviating street types). Replacement texts are used when displaying labels on the map. Users can also export and import lists of substitutes to make reuse and sharing easier.
- Configure *Multiple lines*:
  - Set a character that will force a line break in the text with the *Wrap on character* option
  - Set an ideal line size for auto-wrapping using the *Wrap lines to* option. The size can represent either the *Maximum line length* or the *Minimum line length*.
  - Decide the *Line Height*: values can be set to be in *Millimeters*, *Points*, *Pixels*, *Percentage*, or *Inches*. When line height is set to percentage it is the percentage of the default text line spacing of that font family. Typically 1.2 to 1.5 times the text size.
  - Format the *Alignment*: typical values available are *Left*, *Right*, *Justify* and *Center*.

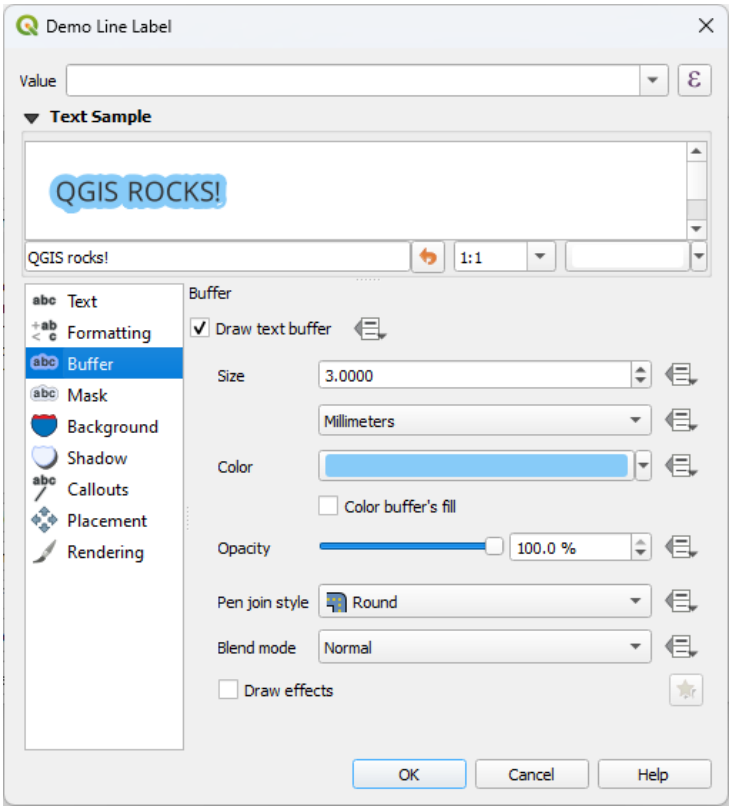
    When setting point labels properties, the text alignment can also be *Follow label placement*. In that case, the alignment will depend on the final placement of the label relative to the point. E.g., if the label is placed to the left of the point, then the label will be right aligned, while if it is placed to the right, it will be left aligned.

  Белешка

  The *Multiple lines* formatting is not yet supported by curve based *label placement*. The options will then be deactivated.
- For line labels you can include *Line direction symbol* to help determine the line directions, with symbols to use to indicate the *Left* or *Right*. They work particularly well when used with the *curved* or *Parallel* placement options from the *Placement* tab. There are options to set the symbols position, and to *Reverse direction*.

- Use the *Formatted numbers* option to format numeric texts. You can set the number of *Decimal places*. By default, 3 decimal places will be used. Use the *Show plus sign* if you want to show the plus sign for positive numbers.

Buffer tab



Слика10.28: Label settings - Buffer tab

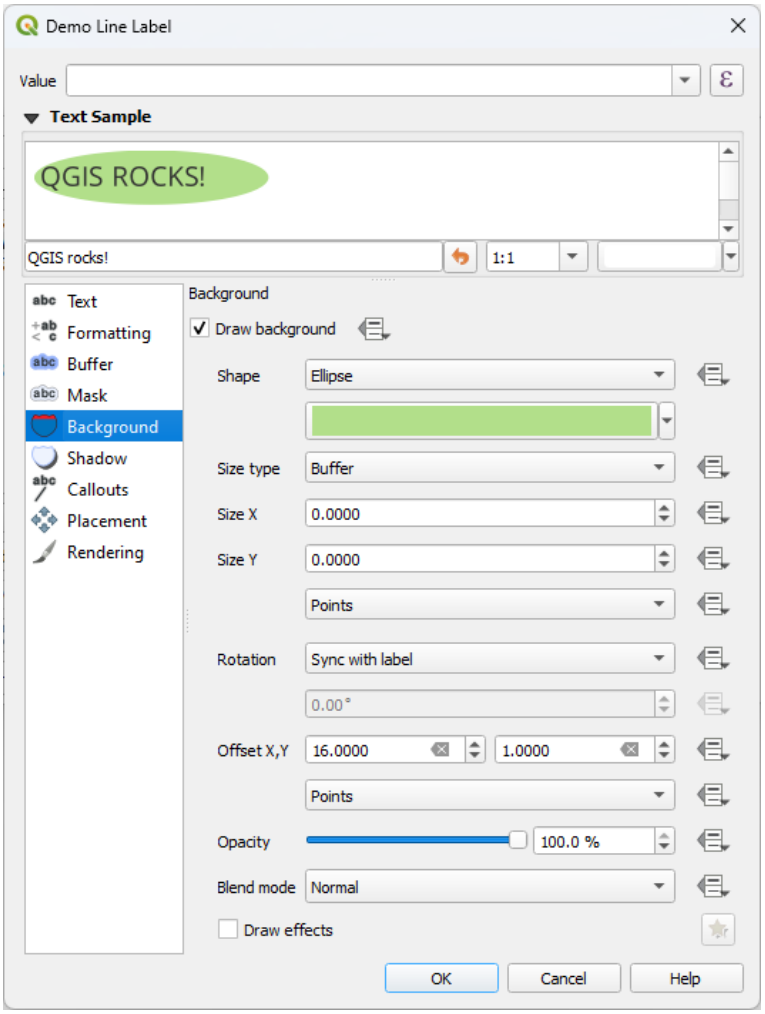
To create a buffer around the label, activate the *Draw text buffer* checkbox in the *Buffer* tab. Then you can:

- Set the buffer's *Size* in any *supported unit*
- Select the buffer's *Color*
- *Color buffer's fill*: The buffer expands from the label's outline, so, if the option is activated, the label's interior is filled. This may be relevant when using partially transparent labels or with non-normal blending modes, which will allow seeing behind the label's text. Unchecking the option (while using totally transparent labels) will allow you to create outlined text labels.
- Define the buffer's *Opacity*
- Apply a *Pen join style*: it can be *Round*, *Miter* or *Bevel*
- Use the *Blend mode* option to determine how your label's buffer will mix with the map components below them (more details at *Blending Modes*).
- Check *Draw effects* to add advanced *paint effects* for improving text readability, eg through outer glows and blurs.

Background tab

The *Background* tab allows you to configure a shape that stays below each label. To add a background, activate the *Draw Background* checkbox and select the *Shape* type. It can be:

- a regular shape such as *Rectangle*, *Square*, *Circle* or *Ellipse* using full properties of a *fill symbol*
- an *SVG* symbol from a file, a URL or embedded in the project or style database (*more details*)
- or a *Marker Symbol* you can create or select from the *symbol library*.



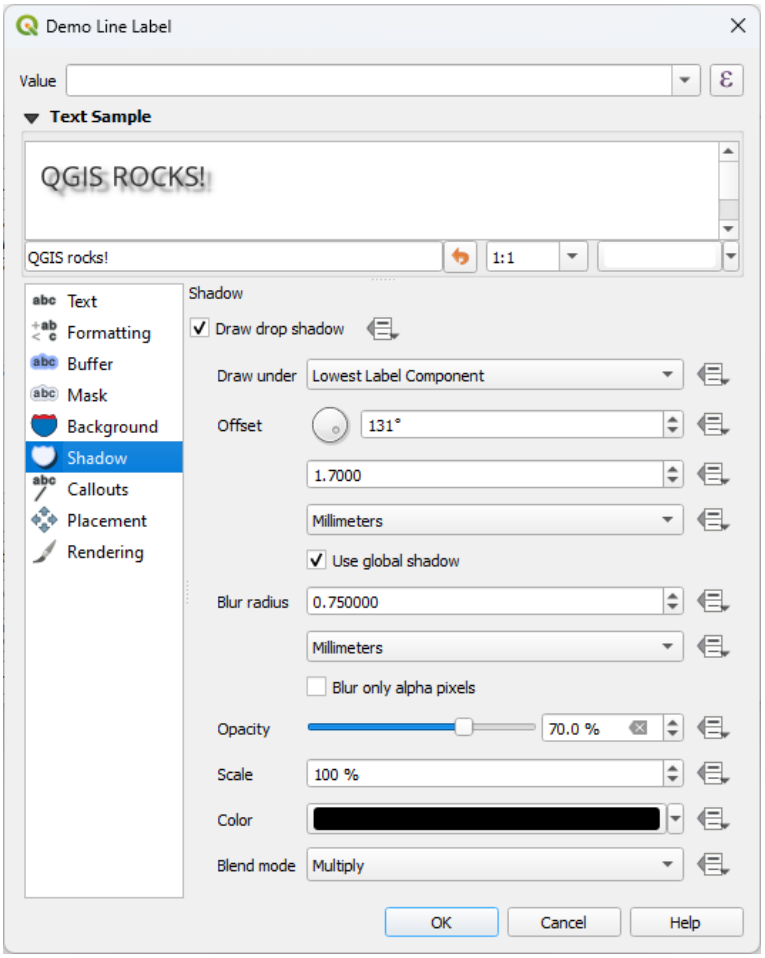
Слика10.29: Label settings - Background tab

Depending on the selected shape, you need to configure some of the following properties:

- The *Size type* of the frame, which can be:
  - *Fixed*: using the same size for all the labels, regardless the size of the text
  - or a *Buffer* over the text's bounding box
- The *Size* of the frame in X and Y directions, using any *supported units*
- A *Rotation* of the background, between *Sync with label*, *Offset of label* and *Fixed*. The last two require an angle in degrees.
- An *Offset X,Y* to shift the background item in the X and/or Y directions
- A *Radius X,Y* to round the corners of the background shape (applies to rectangle and square shapes only)

- An *Opacity* of the background
- A *Blend mode* to mix the background with the other items in the rendering (see *Blending Modes*).
- For SVG symbol, you can use its default properties (*Load symbol parameters*) or set a custom *Fill color*, *Stroke color* and *Stroke width*.
- *Draw effects* to add advanced *paint effects* for improving text readability, eg through outer glows and blurs.

Shadow tab



Слика10.30: Label settings - Shadow tab

To add a shadow to the text, enable the *Shadow* tab and activate the *Draw drop shadow*. Then you can:

- Indicate the item used to generate the shadow with *Draw under*. It can be the *Lowest label component* or a particular component such as the *Text* itself, the *Buffer* or the *Background*.
- Set the shadow's *Offset* from the item being shadowed, ie:
  - The angle: clockwise, it depends on the underlying item orientation
  - The distance of offset from the item being shadowed
  - The units of the offset

  If you tick the *Use global shadow* checkbox, then the zero point of the angle is always oriented to the north and doesn't depend on the orientation of the label's item.

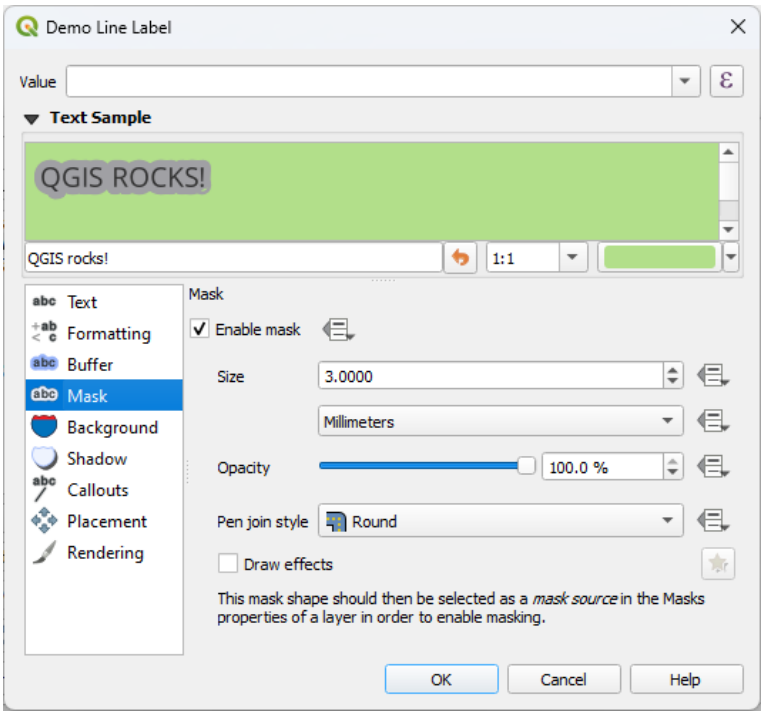
- Influence the appearance of the shadow with the *Blur radius*. The higher the number, the softer the shadows, in the units of your choice.
- Define the shadow's *Opacity*
- Rescale the shadow's size using the *Scale* factor
- Choose the shadow's *Color*
- Use the *Blend mode* option to determine how your label's shadow will mix with the map components below them (more details at *Blending Modes*).

10.3.2 Configuring interaction with labels

Other than the text formatting settings exposed above, you can also set how labels interact with each others or with the features.

Mask tab

The *Mask* tab allows you to define a mask area around the labels. This feature is very useful when you have overlapping symbols and labels with similar colors, and you want to make the labels visible. A label mask prevents specified features from drawing within the boundary set for the mask. For example, you could set a label mask so that a specified layer does not draw within 2mm of the label, but allow features from another layer to still show. Label masks are similar to label buffers in that they allow control of the legibility of labels that cover other features. The label buffer draws on top of any underlying features, while the label mask selectively stops other layers from drawing.



Слика10.31: Labels settings - Mask tab (with the text sample showing a green background representing another layer being excluded)

To create masking effects on labels:

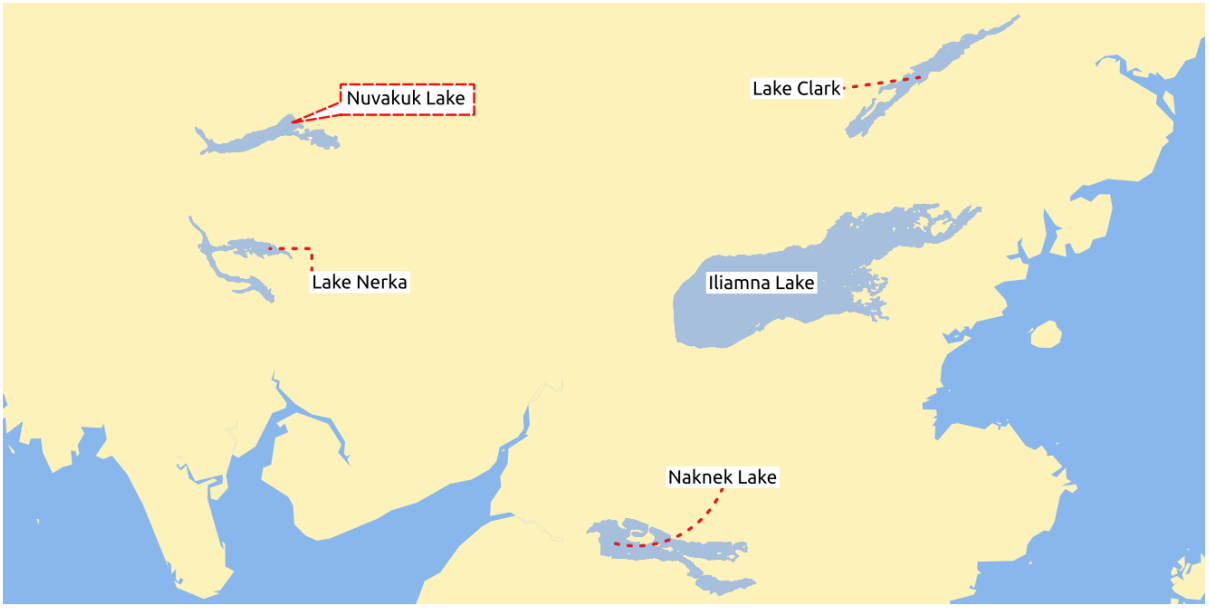
1. Activate the *Enable mask* checkbox in the tab.
2. Then you can set:
   - the mask's *Size* in the *supported units*

- the *Opacity* of the mask area around the label
- a *Pen Join Style*
- *paint effects* through the *Draw effects* checkbox.

3. Select this mask shape as a mask source in the overlapping layer properties *Mask* tab (see *Masks Properties*).

Callouts tab

A common practice when placing labels on a crowded map is to use **callouts** - labels which are placed outside (or displaced from) their associated feature are identified with a dynamic line connecting the label and the feature. If one of the two endings (either the label or the feature) is moved, the shape of the connector is recomputed.



Слика10.32: Labels with various callouts settings

To add a callout to a label, enable the *Callouts* tab and activate the *Draw callouts*. Then you can:

1. Select the *Style* of connector, one of:
   - *Simple lines*: a straight line, the shortest path
   - *Manhattan style*: a 90° broken line
   - *Curved lines*: a curved line
   - *Balloons*: a speech bubble surrounding the label and pointing to the feature. It can have rounded corners.
2. Set the properties of the callout. The following table shows the different properties, with description and compatible connector style.

Табела10.5: Label callout properties

Property	Style of callout	Opis
Fill style	Balloons	A *fill symbol* with full display capabilities, including layer effects, data-defined settings, … for drawing the balloon shape.
Corner radius		Corner radius of the speech bubble

наставак на следећој страни

Табела 10.5 – настављено са претходне стране

Property	Style of callout	Opis
Wedge width		Sets how large the bubble speech connection with feature's pointer should be
Margins		Margins around the label's text, in the unit of your choice
Line style	All but balloons	A *line symbol* with full display capabilities, including layer effects, data-defined settings, … for drawing the connector line.
Curvature	Curved lines	The percentage of curvature of the connection line
Orientation		Orientation, starting from the label to the feature. It can be *Clockwise*, *Counter-clockwise*, or *Automatic* (determining an optimal orientation for each label).
Minimum length	All but balloons	Minimum length of the connector line
Draw lines to all feature parts		In case of a multi-part feature, indicates whether a connector line should be drawn from the label to each geometry part.
Label anchor point		Controls where the connector line should join to the label text. Available options: • *Closest point* • *Centroid* • Fixed position at the edge (*Top left*, *Top center*, *Top right*, *Left middle*, *Right middle*, *Bottom left*, *Bottom center* and *Bottom right*).
Offset from label area		Controls the distance from the label anchor point (where the callout line ends). This avoids drawing lines right up against the text.
Offset from feature	Sve	Controls the distance from the feature (or its anchor point if a polygon) where callout lines end. E.g., this avoids drawing lines right up against the edges of the features.
Feature anchor point		Where the connector line ends on the (polygon) feature. Available options: • *Pole of inaccessibility* • *Point on exterior* • *Point on surface* • *Centroid*
End point marker	Balloons	A *marker symbol* with full display capabilities including layer effects, data-defined, … for rendering a marker symbol below the endpoint of the balloon callout.
Blend mode	Sve	Controls the *blending* of the callout.

Under the *Data defined placement* group, coordinates of the *Origin* (on the label side) and/or *Destination* (on the feature side) points of the callout can be controlled. Callouts can also be controlled manually by using the  Move Label, Diagram or Callout tool in the *Labeling Toolbar*. The start and end points of each callout can be moved this way. The nodes should be highlighted when the mouse pointer is nearby. If needed the `Shift` Key can be held during the movement. This will snap the point in a way that the angle between the two callout points increments by 15 degrees.

Placement tab

Choose the  *Placement* tab for configuring label placement and labeling priority. Note that the placement options differ according to the type of vector or mesh layer, namely point, line or polygon, and are affected by the global *PAL setting*.

Placement for point layers

Point labels placement modes available are:

- *Cartographic*: point labels are generated with a better visual relationship with the point feature, following ideal cartographic placement rules. Labels can be placed:
  - at a set *Distance* in *supported units*, either from the point feature itself or from the bounds of the symbol used to represent the feature (set in *Distance offset from*). The latter option is especially useful when the symbol size isn't fixed, e.g. if it's set by a data defined size or when using different symbols in a *categorized* renderer.
  - within a *Maximum Distance* from the feature, which is an optional setting that allows you to control how far a label can be placed from the feature it's labeling. This works alongside the *Distance* setting to create a range for label placement, adding flexibility to position labels more effectively, especially on busy maps, ensuring they fit neatly around their corresponding features.
  - using the *Prioritize Placement* option, which decides what's more important when placing labels. There are two options:
    - *Prefer closer labels*: By default, labels are kept close to the feature.
    - *Prefer position ordering*: The label will try to stay in a specific position (like top left or top right), even if it's a bit farther away from the feature. The label only moves to other positions if there's no room within the maximum distance at your preferred position.
  - following a *Position priority* which dictates placement candidates for anchoring labels around and (centered) over the point feature, and the order in which the positions are tested. The default order, based on guidelines from Krygier and Wood (2011) and other cartographic textbooks, is as follows:
    1. top right
    2. top left
    3. bottom right
    4. bottom left
    5. middle right
    6. middle left
    7. top, slightly right
    8. bottom, slightly left.

    Using the Data-defined override button, you can provide a comma separated list of placements in order of priority. This also allows only certain placements to be used, for certain features only, so e.g., for coastal features you can prevent labels being placed over the land.
- *Around Point*: labels are placed in a circle around the feature with an equal radius set in *Distance*. Additionally you can set *Maximum Distance* from the feature, to control how far a label can be placed from the feature it's labeling. The placement priority is clockwise from the „top right". The position can be constrained using the data-defined *Quadrant* option.
- *Offset from Point*: labels are placed at an *Offset X,Y* distance from the point feature, in various units, or preferably over the feature. You can use a data-defined *Quadrant* to constrain the placement and can assign a *Rotation* to the label.

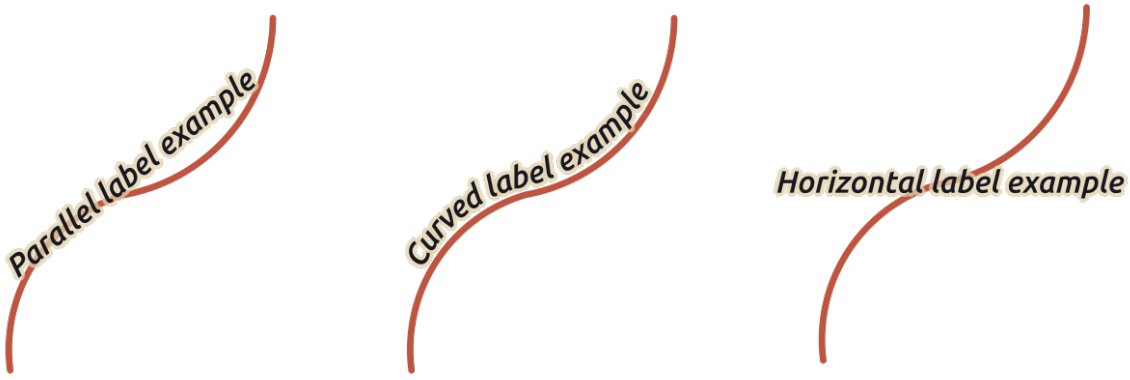
Placement for line layers

Label modes for line layers include:

- *Parallel*: draws the label parallel to a generalised line representing the feature, with preference for placement over straighter portions of the line. You can define:

  - *Allowed positions*: *Above line*, *On line*, *Below line* and *Line orientation dependent position* (placing the label at the left or the right of the line). It's possible to select several options at once. In that case, QGIS will look for the optimal label position.
  - *Distance* between the label and the line

- *Curved*: draws the label following the curvature of the line feature. In addition to the parameters available with the *Parallel* mode, you can set the *Maximum angle between curved characters*, either inside or outside.
- *Horizontal*: draws labels horizontally along the length of the line feature.



Слика10.33: Label placement examples for lines

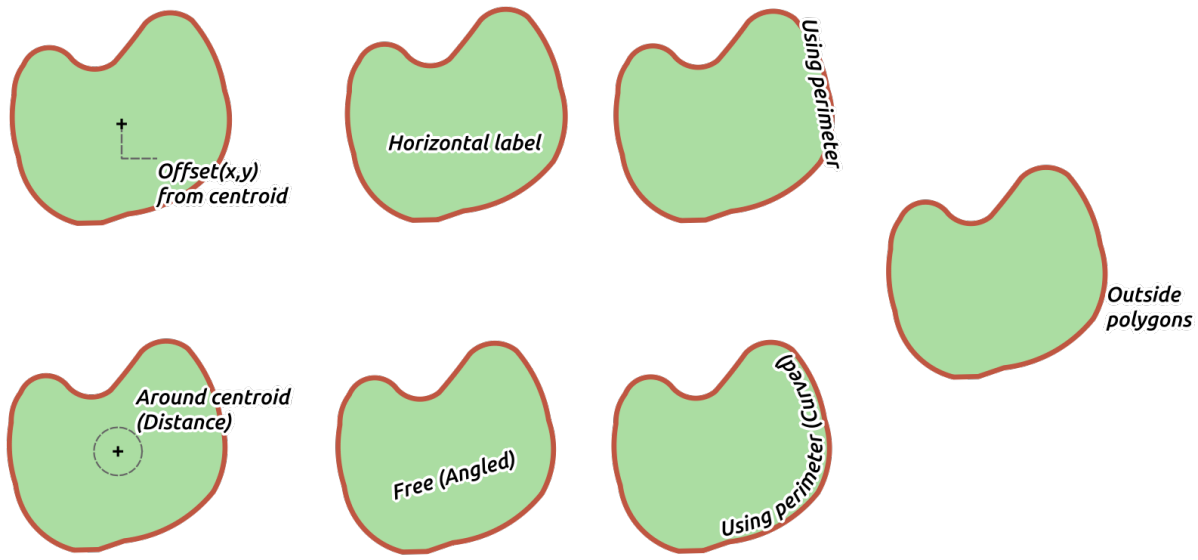
Next to placement modes, you can set:

- *Repeating Labels Distance* to display multiple times the label over the length of the feature. The distance can be in `Millimeters`, `Points`, `Pixels`, `Meters at scale`, `Map Units` and `Inches`.
- A *Label Overrun Distance* (not available for horizontal mode): specifies the maximal allowable distance a label may run past the end (or start) of line features. Increasing this value can allow for labels to be shown for shorter line features.
- *Label Anchoring*: controls the placement of the labels along the line feature they refer to. Click on *Settings …* to choose:
  - the position along the line (as a ratio) which labels will be placed close to. It can be data-defined and possible values are:
    - *Center of Line*
    - *Start of Line*
    - *End of Line*
    - or *Custom….*
  - *Clipping*: Determines how the label placement on a line is calculated. By default only the visible extent of the line is used but the whole extent can be used to have more consistent results.
  - *Anchor text*: controls which part of the text (start, center or end) will line up with the anchor point. Using *Automatic* anchoring means that:
    - For labels anchored near the start of the line (0-25%), the anchor placement will be the **start** of the label text
    - For labels anchored near the end of the line (75-100%), the anchor placement will be the **end** of the label text
    - For labels anchored near the center of the line (25-75%), the anchor placement will be the **center** of the label text

– *Placement Behavior*: use *Preferred Placement Hint* to treat the label anchor only as a hint for the label placement. By choosing *Strict*, labels are placed exactly on the label anchor.

Placement for polygon layers

You can choose one of the following modes for placing labels of polygons:



Слика10.34: Label placement examples for polygons

- *Offset from Centroid*: labels are placed over the feature centroid or at a fixed *Offset X,Y* distance (in *supported units*) from the centroid. The reference centroid can be determined based on the part of the polygon rendered in the map canvas (*visible polygon*) or the *whole polygon*, no matter if you can see it. You can also:
  - force the centroid point to lay inside their polygon
  - place the label within a specific quadrant
  - assign a rotation
  - *Allow placing labels outside of polygons* when it is not possible to place them inside the polygon. Thanks to data-defined properties, this makes possible to either allow outside labels, prevent outside labels, or force outside labels on a feature-by-feature basis.
- *Around Centroid*: places the label within a preset distance around the centroid, with a preference for the placement directly over the centroid. Again, you can define whether the centroid is the one of the *visible polygon* or the *whole polygon*, and whether to force the centroid point inside the polygon.
- *Horizontal*: places at the best position a horizontal label inside the polygon. The preferred placement is further from the edges of the polygon. It's possible to *Allow placing labels outside of polygons*.
- *Free (Angled)*: places at the best position a rotated label inside the polygon. The rotation respects the polygon's orientation and the preferred placement is further from the edges of the polygon. It's possible to *Allow placing labels outside of polygons*.
- *Using Perimeter*: draws the label parallel to a generalised line representing the polygon boundary, with preference for straighter portions of the perimeter. You can define:
  - *Allowed positions*: *Above line*, *On line*, *Below line* and *Line orientation dependent position* (placing the label at the left or the right of the polygon's boundary). It's possible to select several options at once. In that case, QGIS will look for the optimal label position.
  - *Distance* between the label and the polygon's outline
  - the *Repeating Labels Distance* to display multiple times the label over the length of the perimeter.

- *Using Perimeter (Curved)*: draws the label following the curvature of the polygon's boundary. In addition to the parameters available with the *Using Perimeter* mode, you can set the *Maximum angle between curved characters polygon*, either inside or outside.
- *Outside Polygons*: always places labels outside the polygons, at a set *Distance*

Common placement settings

Some label placement settings are available for all layer geometry types:

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*.

Пажња

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.

Пажња

This option is only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

Geometry Generator

The *Geometry Generator* section allows a user to alter the underlying geometry used to place and render the label, by using *expressions*. This can be useful to perform displacement of the geometry dynamically or to convert it to another geometry (type).

In order to use the geometry generator:

1. Check the *Geometry generator* option
2. Enter the expression generating the geometry to rely on
3. If relevant, select the geometry type of the expression output: the label geometry-based settings such as placement or rendering are updated to match the new geometry type capabilities.

Some use cases include:

- Use a geometry which is saved in another field „label_position”
- Use the *generated geometry* from the symbology also for labeling
- Use the @map_scale variable to calculate distances / sizes be zoom level independent.
- Combined with the curved placement mode, creates a circular label around a point feature:

```
exterior_ring(make_circle($geometry, 20))
```

- Add a label at the start and the end of a line feature:

```
collect_geometries( start_point($geometry), end_point($geometry) )
```

- Rely on a smoothed line of a river to get more room for label placement:

```
smooth( $geometry, iterations:=30, offset:=0.25, min_length:=10 )
```

Data Defined

The *Data Defined* group provides direct control on labels placement, on a feature-by-feature basis. It relies on their attributes or an expression to set:

- the *X* and *Y* coordinate
- the text alignment over the custom position set above:
  - *Horizontal*: it can be **Left**, **Center** or **Right**
  - the text *Vertical*: it can be **Bottom**, **Base**, **Half**, **Cap** or **Top**
- the text *Rotation*. Rotation is defined as clockwise angle with 0° pointing in the direction of East for `Horizontal` oriented text and with 0° pointing in North direction for `Vertical` oriented text. Different units can be defined for the labeling rotation (e.g. `degrees`, `minutes of arc`, `turns`). QGIS prioritizes screen display rotation angles by default and always rotates text to be most readable on screen, so „upside-down" rotation angles are overridden while rendering. To change this behavior and force QGIS to render exact data-defined rotation angles there is an option in  *Rendering* tab, under *Show upside-down labels*. Check the *Preserve data rotation values* entry if you want to keep the rotation value in the associated field and apply it to the label, whether the label is pinned or not. If unchecked, unpinning the label rotation is reset and its value cleared from the attribute table.

  Белешка

  Data-defined rotation with polygon features is currently supported only with the *Around centroid* placement mode.

Белешка

Expressions can not be used in combination with the labels map tools (ie the *Rotate label* and *Move label* tools) to *data-define* labels placement. The widget will be reset to the corresponding *auxiliary storage field*.

Priority

In the *Priority* section you can define the placement priority rank of each label, ie if there are different diagrams or labels candidates for the same location, the item with the higher priority will be displayed and the others could be left out.

The priority rank is also used to evaluate whether a label could be omitted due to a greater weighted *obstacle feature*.

Obstacles

In some contexts (eg, high density labels, overlapping features…), the labels placement can result in labels being placed over unrelated features.

An obstacle is a feature over which QGIS avoids placing other features' labels or diagrams. This can be controlled from the *Obstacles* section:

1. Activate the  *Features act as obstacles* option to decide that features of the layer should act as obstacles for any label and diagram (including items from other features in the same layer).

Instead of the whole layer, you can select a subset of features to use as obstacles, using the Data-defined override control next to the option.

2. Use the *Settings* button to tweak the obstacle's weighting.
   - For every potential obstacle feature you can assign an *Obstacle weight*: any *label* or *diagram* whose placement priority rank is greater than this value can be placed over. Labels or diagrams with lower rank will be omitted if no other placement is possible.

     This weighting can also be data-defined, so that within the same layer, certain features are more likely to be covered than others.
   - For polygon layers, you can choose the kind of obstacle the feature is:
     - **over the feature's interior**: avoids placing labels over the interior of the polygon (prefers placing labels totally outside or just slightly inside the polygon)
     - or **over the feature's boundary**: avoids placing labels over the boundary of the polygon (prefers placing labels outside or completely inside the polygon). This can be useful for layers where the features cover the whole area (administrative units, categorical coverages, …). In this case, it is impossible to avoid placing labels within these features, and it looks much better when placing them over the boundaries between features is avoided.

Rendering tab

In the *Rendering* tab, you can tune when the labels can be rendered and their interaction with other labels and features.

Label options

Under *Label options*:

- You find the *scale-based* and the *Pixel size-based* visibility settings.
- The *Label z-index* determines the order in which labels are rendered, as well in relation with other feature labels in the layer (using data-defined override expression), as with labels from other layers. Labels with a higher z-index are rendered on top of labels (from any layer) with lower z-index.

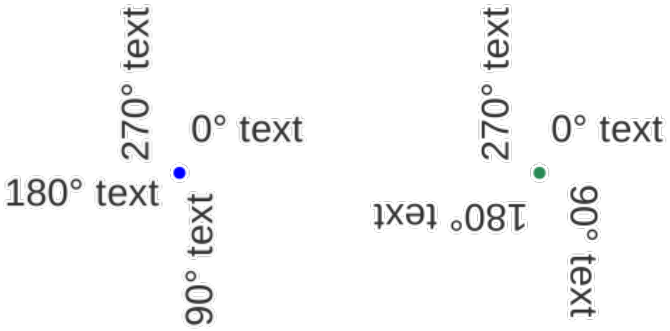
  Additionally, the logic has been tweaked so that if two labels have matching z-indexes, then:
  - if they are from the same layer, the smaller label will be drawn above the larger label
  - if they are from different layers, the labels will be drawn in the same order as their layers themselves (ie respecting the order set in the map legend).

  Белешка

  This setting doesn't make labels to be drawn below the features from other layers, it just controls the order in which labels are drawn on top of all the layers' features.

- *Allow inferior fallback placements*: By default QGIS tries to render labels at their best placement, following your settings. Check this mode to allow features to fallback to worse placement options when there's no other choice, e.g. when a line is too short to fit a curved label text then the label may be placed horizontally just over the feature's center point.
- With data-defined expressions in *Show label* and *Always Show* you can fine tune which labels should be rendered.
- Allow to *Show upside-down labels*: alternatives are **never**, **when rotation defined** or **always**.
  - **never** - default setting, screen readability is prioritized,

- **when rotation defined** - label rotation should be defined under *Placement* tab, within the *Data Defined* group
- **always** - upside-down labels are allowed



Слика10.35: Data defined label rotation with show upside-down labels option set to: „never” (left), „when rotation defined” (right)

- The *Overlapping labels* group allows you to control whether overlapping labels are permitted for features in the layer and how each of them should be handled:
  - *Never overlap*: never ever place overlapping labels for the layer, even if it means some labels will be missing
  - *Allow overlaps if required*: if the label can't otherwise be placed, draw an overlapping label. This mode will cause the label to be moved to a less ideal placement if possible, e.g. moving the label further from the center of a line or polygon, IF doing so will avoid overlapping labels. But if there's no other positions possible, then draw the label overlapping.
  - *Allow overlaps without penalty*: It doesn't matter at all if the label overlaps other labels or obstacles, that's fine to do and the best placement (e.g most central placement) should always be used even if an alternate further placement is possible which avoids overlaps entirely.

  Allowing both overlapping labels and fallback placements options will guarantee that all features in the layer are labeled… not necessarily at their best rendering!

Feature options

Under *Feature options*:

- You can choose to *Label every part of a multi-part features* and *Limit number of features to be labeled to*.
- Both line and polygon layers offer the option to set a minimum size for the features to be labeled, using *Suppress labeling of features smaller than*.
- For polygon features, you can also filter the labels to show according to whether they completely fit within their feature or not.
- For line features, you can choose to *Merge connected lines to avoid duplicate labels*, rendering a quite airy map in conjunction with the *Distance* or *Repeat* options in the *Placement* tab.

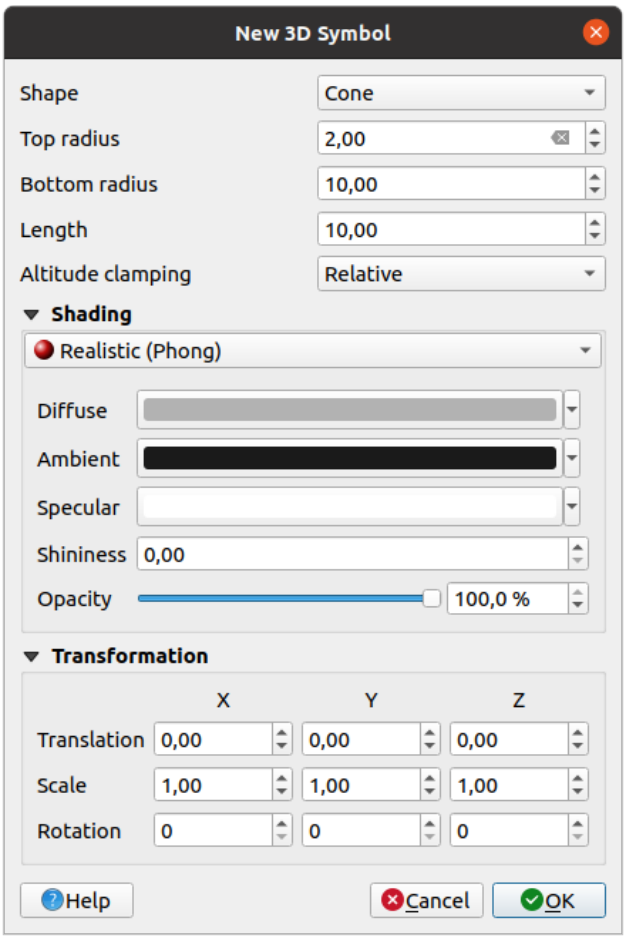
10.4 Creating 3D Symbols

The *Style Manager* helps you create and store 3D symbols for every geometry type to render in the *3D map view*.

As of the other items, enable the *3D Symbols* tab and expand the button menu to create:

- *3D point symbols*
- *3D line symbols*
- *3D polygon symbols*

10.4.1 Point Layers

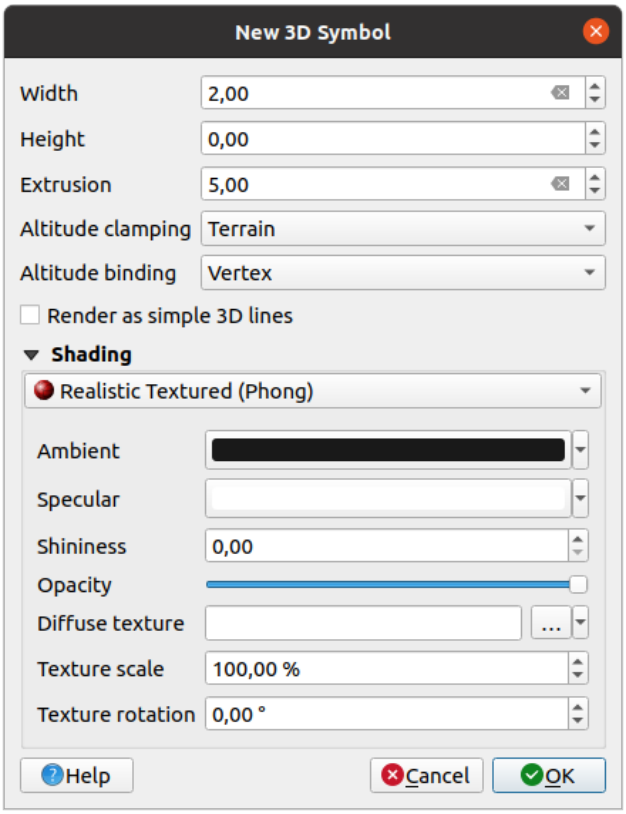


Слика10.36: Properties of a 3D point symbol

- You can define different types of 3D *Shape* to use for point symbols. They are mainly defined by their dimensions whose unit refers to the CRS of the project. Available types are:
  - *Sphere* defined by a *Radius*
  - *Cylinder* defined by a *Radius* and *Length*
  - *Cube* defined by a *Size*
  - *Cone* defined by a *Top radius*, a *Bottom radius* and a *Length*
  - *Plane* defined by a *Size*
  - *Torus* defined by a *Radius* and a *Minor radius*
  - *3D Model*, using a 3D model file: supported formats include wavefront `.obj`, `.glTF` and `.fbx`. Models can be a file on disk, a remote URL or *embedded in the project*. Community-created models are shared on the QGIS Hub.
  - *Billboard*, defined by the *Billboard height* and the *Billboard symbol* (usually based on a *marker symbol*). The symbol will have a stable size. Convenient for visualizing 3D point clouds Shapes.

- The *Altitude clamping* can be set to *Absolute*, *Relative* or *Terrain*. The *Absolute* setting can be used when height values of the 3d vectors are provided as absolute measures from 0. *Relative* and *Terrain* add given elevation values to the underlying terrain elevation.
- The *shading* properties can be defined.
- Under the *Transformations* frame, you can apply affine transformation to the symbol:
  - *Translation* to move objects in x, y and z axis.
  - *Scale* to resize the 3D shapes
  - *Rotation* around the x-, y- and z-axis.

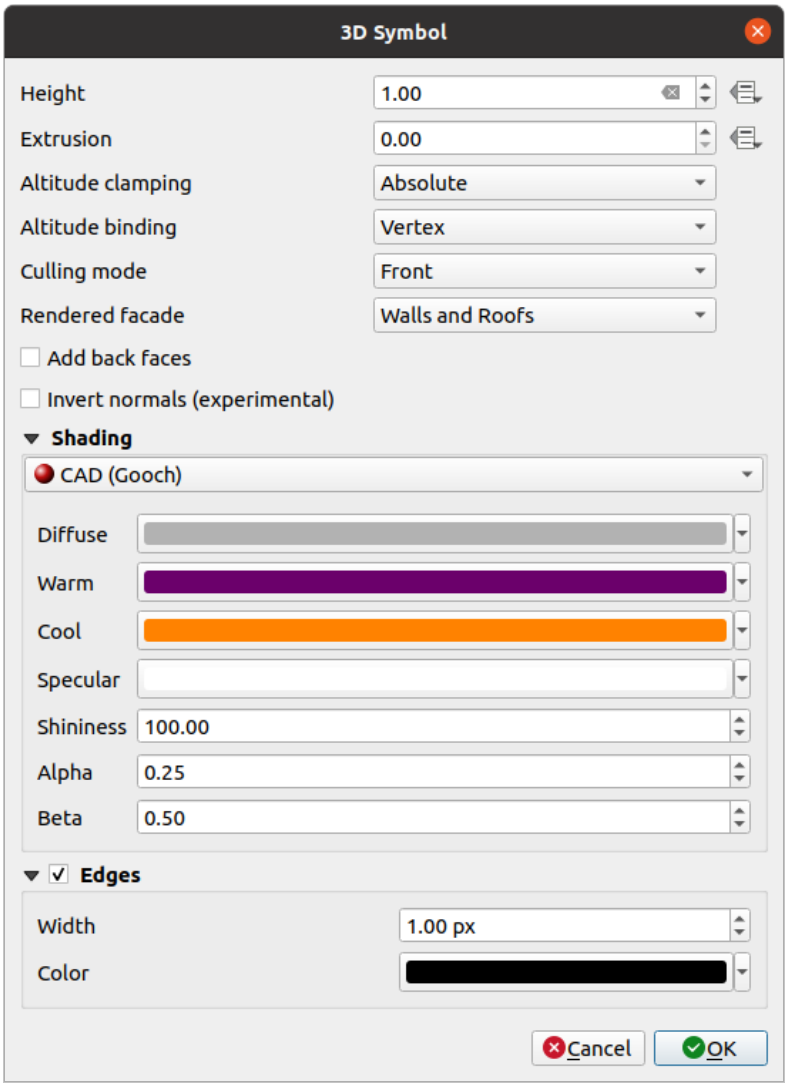
10.4.2 Line layers



Слика10.37: Properties of a 3D line symbol

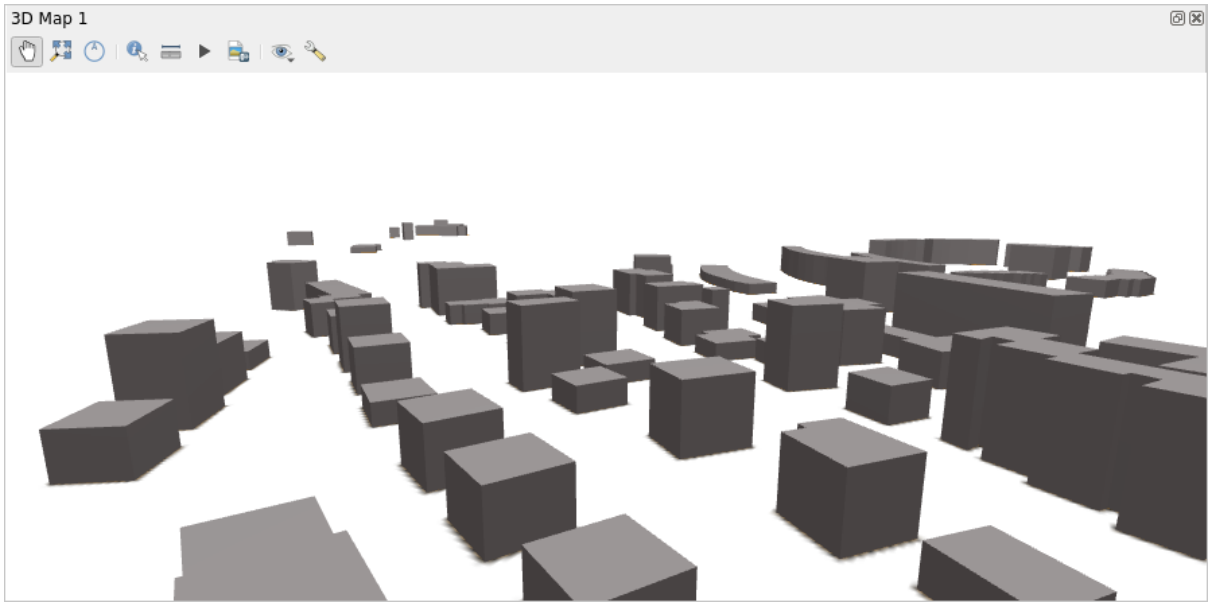
- Beneath the *Width* and *Height* settings you can define the *Extrusion* of the vector lines. If the lines do not have z-values, you can define the 3d volumes with this setting.
- With the *Altitude clamping* you define the position of the 3D lines relative to the underlying terrain surface, if you have included raster elevation data or other 3D vectors.
- The *Altitude binding* defines how the feature is clamped to the terrain. Either every *Vertex* of the feature will be clamped to the terrain or this will be done by the *Centroid*.
- It is possible to *Render as simple 3D lines*.
- The *shading* properties can be defined.

10.4.3 Polygon Layers



Слика10.38: Properties of a 3D polygon symbol

- As for the other ones, *Height* can be defined in CRS units. You can also use the button to overwrite the value with a custom expression, a variable or an entry of the attribute table
- Again, *Extrusion* is possible for missing z-values. Also for the extrusion you can use the button in order to use the values of the vector layer and have different results for each polygon:



Слика10.39: Data Defined Extrusion

- The *Altitude clamping*, *Altitude binding* can be defined as explained above.
- The *Culling mode* to apply to the symbol; it can be:
  - *No Culling*: this can help to avoid seemingly missing surfaces when polygonZ/multipatch data do not have consistent ordering of vertices (e.g. all clock-wise or counter clock-wise)
  - *Front*
  - or *Back*
- The *Rendered facade* determines the faces to display. Possible values are *No facades*, *Walls*, *Roofs*, or *Walls and roofs*
- *Add back faces*: for each triangle, creates both front and back face with correct normals - at the expense of increased number of vertex data. This option can be used to fix shading issues (e.g., due to data with inconsistent order of vertices).
- *Invert normals (experimental)*: can be useful for fixing clockwise/counter-clockwise face vertex orders
- The *shading* properties can be defined.
- Display of the *Edges* of the symbols can be enabled and assigned a *Width* and *Color*.

Путоказ

Combination for best rendering of 3D data

Culling mode, *Add back faces* and *Invert normals* are all meant to fix the look of 3D data if it does not look right. Typically when loading some data, it is best to first try `culling mode=back` and `add back faces=disabled` - it is the most efficient. If the rendering does not look correct, try `add back faces=enabled` and keep `culling mode=no culling`. Other combinations are more advanced and useful only in some scenarios based on how mixed up is the input dataset.

10.4.4 Shading the texture

Shading helps you reveal 3d details of objects which may otherwise be hidden due to the scene's lighting. Ultimately, it's an easier material to work with as you don't need to worry about setting up appropriate scene lighting in order to visualise features.

Various techniques of shading are used in QGIS and their availability depends on the geometry type of the symbol:

- *Realistic (Phong)*: describes the way a surface reflects light as a combination of the *Diffuse* reflection of rough surfaces with the *Specular* reflection of shiny surfaces (*Shininess*). It also includes an *Ambient* option to account for the small amount of light that is scattered about the entire scene. Use the *Opacity* slider to render semi-transparent objects in 3D. Read more at Phong reflection description.
- *Realistic Textured (Phong)*: same as the *Realistic (Phong)* except that an image is used as *Diffuse Texture*. The image can be a file on disk, a remote URL or *embedded in the project*. The *Texture scale* and *Texture rotation* are required. Use the *Opacity* slider to render semi-transparent objects in 3D.
- *CAD (Gooch)*: this technique allows shading to occur only in mid-tones so that edge lines and highlights remain visually prominent. Along with the *Diffuse*, *Specular*, *Shininess* options, you need to provide a *Warm* color (for surface facing toward the light) and a *Cool* color (for the ones facing away). Also, the relative contributions to the cool and warm colors by the diffuse color are controlled by *Alpha* and *Beta* properties respectively. See also Gooch shading.
- *Metal Roughness*: a physically based rendering material that provides an accurate representation of how light interacts with surfaces. Options are available for setting the material *Base color*, *Metalness* and *Roughness*.
- *Embedded Textures* with 3D models shape

10.4.5 Application example

To go through the settings explained above you can have a look at https://app.merginmaps.com/projects/saber/luxembourg/tree.

ГЛАВА 11

Upravljanje izvorom podataka

11.1 Opening Data

As part of an Open Source Software ecosystem, QGIS is built upon different libraries that, combined with its own providers, offer capabilities to read and often write a lot of formats:

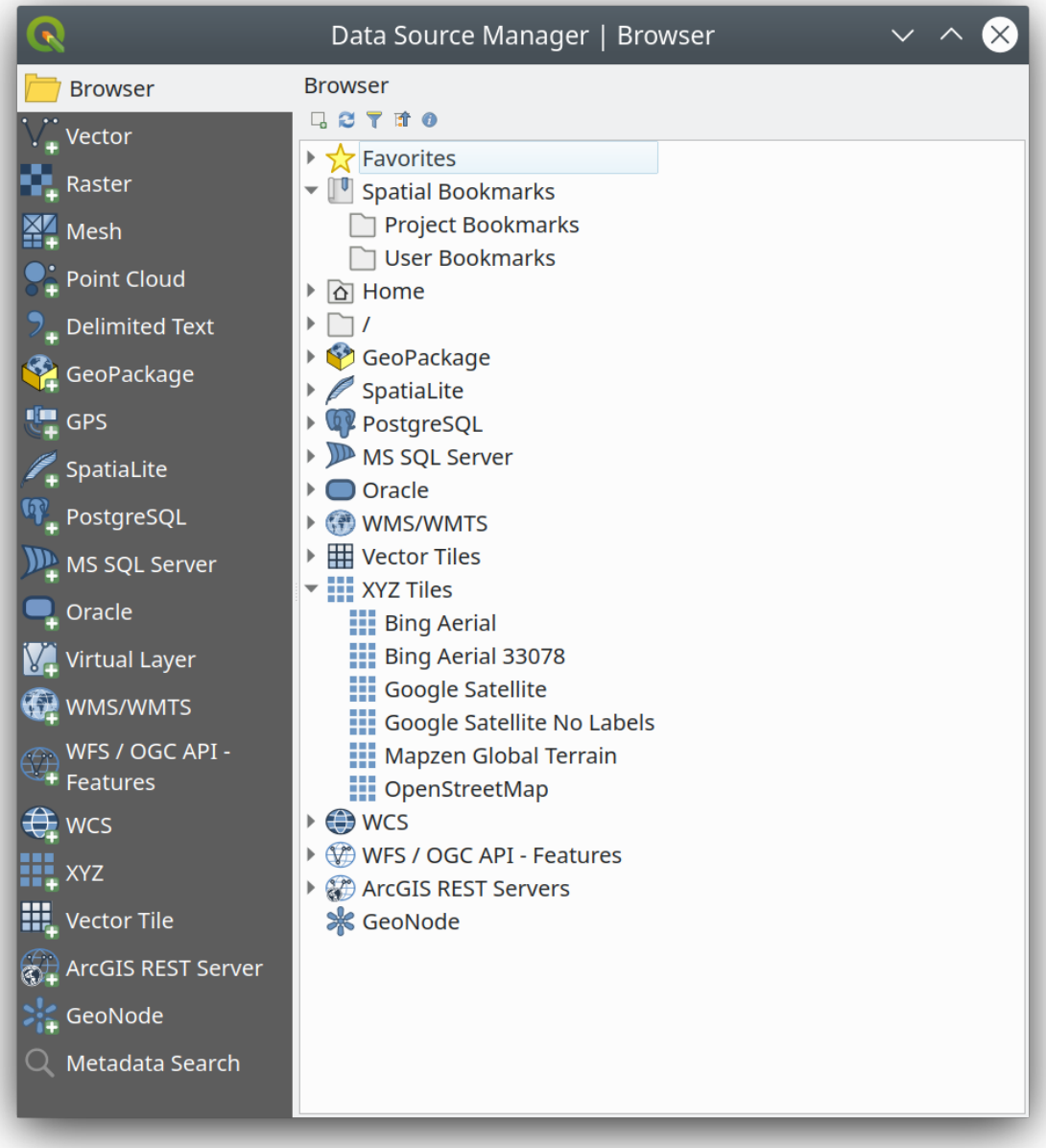
- Vector data formats include GeoPackage, GML, GeoJSON, GPX, KML, Comma Separated Values, ESRI formats (Shapefile, Geodatabase…), MapInfo and MicroStation file formats, AutoCAD DWG/DXF, GRASS and many more… Read the complete list of supported vector formats.
- Raster data formats include GeoTIFF, JPEG, ASCII Gridded XYZ, MBTiles, R or Idrisi rasters, GDAL Virtual, SRTM, Sentinel Data, ERDAS IMAGINE, ArcInfo Binary Grid, ArcInfo ASCII Grid, and many more… Read the complete list of supported raster formats.
- Database formats include PostgreSQL, SQLite/SpatiaLite, Oracle, MS SQL Server, SAP HANA, MySQL…
- Web map and data services (WM(T)S, WFS, WCS, CSW, XYZ tiles, ArcGIS services, …) are also handled by QGIS providers. See *Working with OGC / ISO protocols* for more information about some of these.
- You can read supported files from archived folders and use QGIS native formats such as QML files (*QML - The QGIS Style File Format*) and virtual and memory layers.

More than 80 vector and 140 raster formats are supported by GDAL and QGIS native providers.

Белешка

Not all of the listed formats may work in QGIS for various reasons. For example, some require external proprietary libraries, or the GDAL/OGR installation of your OS may not have been built to support the format you want to use. To see the list of available formats, run the command line `ogrinfo --formats` (for vector) and `gdalinfo --formats` (for raster), or check the *Settings* ► *Options* ► *GDAL* menu in QGIS.

In QGIS, depending on the data format, there are different tools to open a dataset, mainly available in the *Layer* ► *Add Layer* ► menu or from the *Manage Layers* toolbar (enabled through *View* ► *Toolbars* menu). However, all these tools point to a unique dialog, the *Data Source Manager* dialog, that you can open with the  Open Data Source Manager button, available on the *Data Source Manager Toolbar*, or by pressing `Ctrl+L`. The *Data Source Manager* dialog (Сл.11.1) offers a unified interface to open file-based data as well as databases or web services supported by QGIS.



Слика11.1: QGIS Data Source Manager dialog

Beside this main entry point, you also have the *DB Manager* plugin that offers advanced capabilities to analyze and manipulate connected databases. More information on DB Manager capabilities can be found in *DB Manager Plugin*.

There are many other tools, native or third-party plugins, that help you open various data formats.

This chapter will describe only the tools provided by default in QGIS for loading data. It will mainly focus on the *Data Source Manager* dialog but more than describing each tab, it will also explore the tools based on the data provider or format specificities.

11.1.1 The Browser Panel

The *Browser* is one of the main ways to quickly and easily add your data to projects. It's available as:

- a *Data Source Manager* tab, enabled pressing the  Open Data Source Manager button (`Ctrl+L`);
- as a QGIS panel you can open from the menu *View* ► *Panels* (or  *Settings* ► *Panels*) or by pressing `Ctrl+2`.

In both cases, the *Browser* helps you navigate in your file system and manage geodata, regardless the type of layer (raster, vector, table), or the datasource format (plain or compressed files, databases, web services).

Exploring the Interface

At the top of the Browser panel, you find some buttons that help you to:

-  Add Selected Layers: you can also add data to the map canvas by selecting **Add selected layer(s)** from the layer's context menu;
-  Refresh the browser tree;
-  Filter Browser to search for specific data. Enter a search word or wildcard and the browser will filter the tree to only show paths to matching DB tables, filenames or folders – other data or folders won't be displayed. See the Browser Panel(2) example in Сл.11.2. The comparison can be case-sensitive or not. It can also be set to:
  - *Normal*: show items containing the search text
  - *Wildcard(s)*: fine tune the search using the `?` and/or `*` characters to specify the position of the search text
  - *Regular expression*
-  Collapse All the whole tree;
-  Enable/disable properties widget: when toggled on, a new widget is added at the bottom of the panel showing, if applicable, metadata for the selected item.

The entries in the *Browser* panel are organised hierarchically, and there are several top level entries:

1. *Favorites* where you can place shortcuts to often used locations
2. *Spatial Bookmarks* where you can store often used map extents (see *Bookmarking extents on the map*)
3. *Project Home*: for a quick access to the folder in which (most of) the data related to your project are stored. The default value is the directory where your project file resides.
4. *Home* directory in the file system and the filesystem root directory.
5. Connected local or network drives
6. Then comes a number of container / database types and service protocols, depending on your platform and underlying libraries:
   -  *GeoPackage*
   -  *SpatiaLite*
   -  *PostgreSQL*
   -  *SAP HANA*
   -  *MS SQL Server*
   -  *Oracle*

-  *WMS/WMTS*
-  *Vector Tiles*
-  *XYZ Tiles*
-  *WCS*
-  *WFS/OGC API-Features*
-  *ArcGIS REST Server*

Interacting with the Browser items

The browser supports drag and drop within the browser, from the browser to the canvas and *Layers* panel, and from the *Layers* panel to layer containers (e.g. GeoPackage) in the browser.

Project file items inside the browser can be expanded, showing the full layer tree (including groups) contained within that project. Project items are treated the same way as any other item in the browser, so they can be dragged and dropped within the browser (for example to copy a layer item to a geopackage file) or added to the current project through drag and drop or double click.

The context menu for an element in the *Browser* panel is opened by right-clicking on it.

For file system directory entries, the context menu offers the following:

- *New* ► to create in the selected entry a:
  - *Directory…*
  - *GeoPackage…*
  - *ShapeFile…*
- *Add as a Favorite*: favorite folders can be renamed (*Rename favorite…*) or removed (*Remove favorite*) any time.
- *Hide from Browser*: hidden folders can be toggled to visible from the *Settings* ► *Options* ► *Data Sources* ► *Hidden browser paths* setting
- *Fast Scan this Directory*
- *Open Directory*
- *Open in Terminal*
- *Properties…*
- *Directory Properties…*

For leaf entries that can act as layers in the project, the context menu will have supporting entries. For example, for non-database, non-service-based vector, raster and mesh data sources:

- *Export Layer* ► *To File…*
- *Add Layer to Project*
- *Layer Properties*
- *Open with Data Source Manager…*
- *Manage* ► *Rename „<name of file>”…* or *Delete „<name of file>”…*
- *Show in Files*
- *File Properties*

In the *Layer properties* entry, you will find (similar to what you will find in the *vector* and *raster* layer properties once the layers have been added to the project):

- *Metadata* for the layer. Metadata groups: *Information from provider* (if possible, *Path* will be a hyperlink to the source), *Identification*, *Extent*, *Access*, *Fields* (for vector layers), *Bands* (for raster layers), *Contacts*, *Links* (for vector layers), *References* (for raster layers), *History*.
- A *Preview* panel
- The attribute table for vector sources (in the *Attributes* panel).

Use *Open with Data Source Manager…* to directly open and configure the data source in the *Data Source Manager* using the URI of your data source. This simplifies the process of adding a layer from the *Browser* by allowing you to set specific opening options for the data source. It is currently available for vector (including the dedicated GeoPackage entry), raster, and SpatiaLite data sources.

To add a layer to the project using the *Browser*:

1. Enable the *Browser* as described above. A browser tree with your file system, databases and web services is displayed. You may need to connect databases and web services before they appear (see dedicated sections).
2. Find the layer in the list.
3. Use the context menu, double-click its name, or drag-and-drop it into the *map canvas*. Your layer is now added to the *Layers panel* and can be viewed on the map canvas.

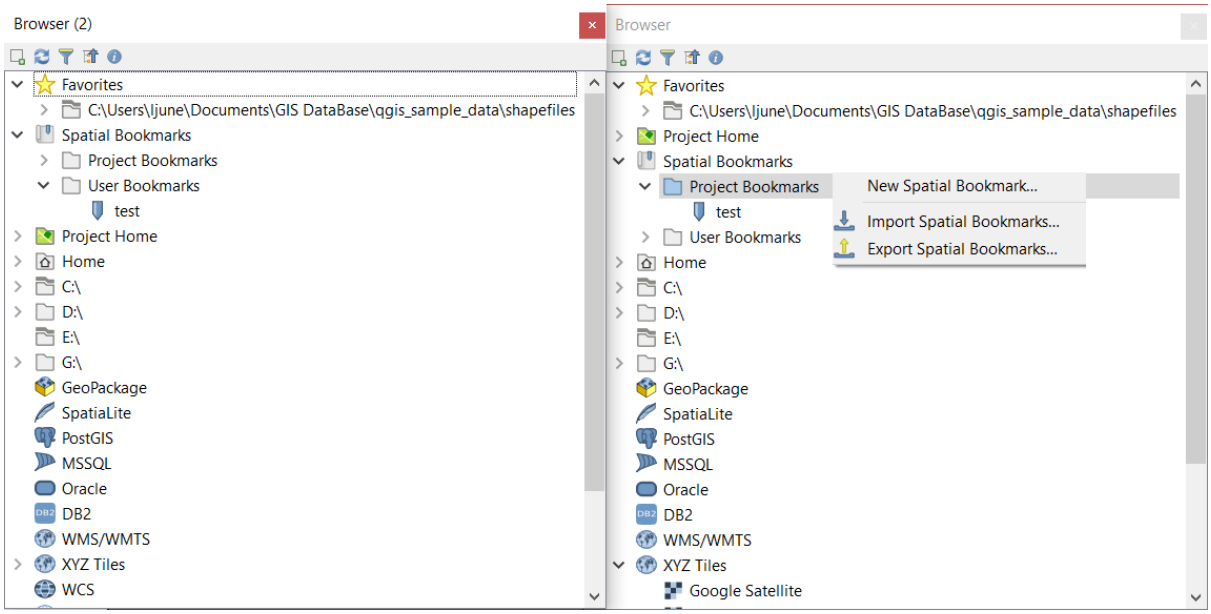
> **Савет**
>
> **Open a QGIS project directly from the browser**
>
> You can also open a QGIS project directly from the Browser panel by double-clicking its name or by drag-and-drop into the map canvas.

Once a file is loaded, you can zoom around it using the map navigation tools. To change the style of a layer, open the *Layer Properties* dialog by double-clicking on the layer name or by right-clicking on the name in the legend and choosing *Properties* from the context menu. See section *Symbology Properties* for more information on setting symbology for vector layers.

Right-clicking an item in the browser tree helps you to:

- for a file or a table, display its metadata or open it in your project. Tables can even be renamed, deleted or truncated.
- for a folder, bookmark it into your favourites or hide it from the browser tree. Hidden folders can be managed from the *Settings* ► *Options* ► *Data Sources* tab.
- manage your *spatial bookmarks*: bookmarks can be created, exported and imported as `XML` files.
- create a connection to a database or a web service.
- refresh, rename or delete a schema.

You can also import files into databases or copy tables from one schema/database to another with a simple drag-and-drop. There is a second browser panel available to avoid long scrolling while dragging. Just select the file and drag-and-drop from one panel to the other.



Слика11.2: QGIS Browser panels side-by-side

Савет

Add layers to QGIS by simple drag-and-drop from your OS file browser

You can also add file(s) to the project by drag-and-dropping them from your operating system file browser to the *Layers Panel* or the map canvas.

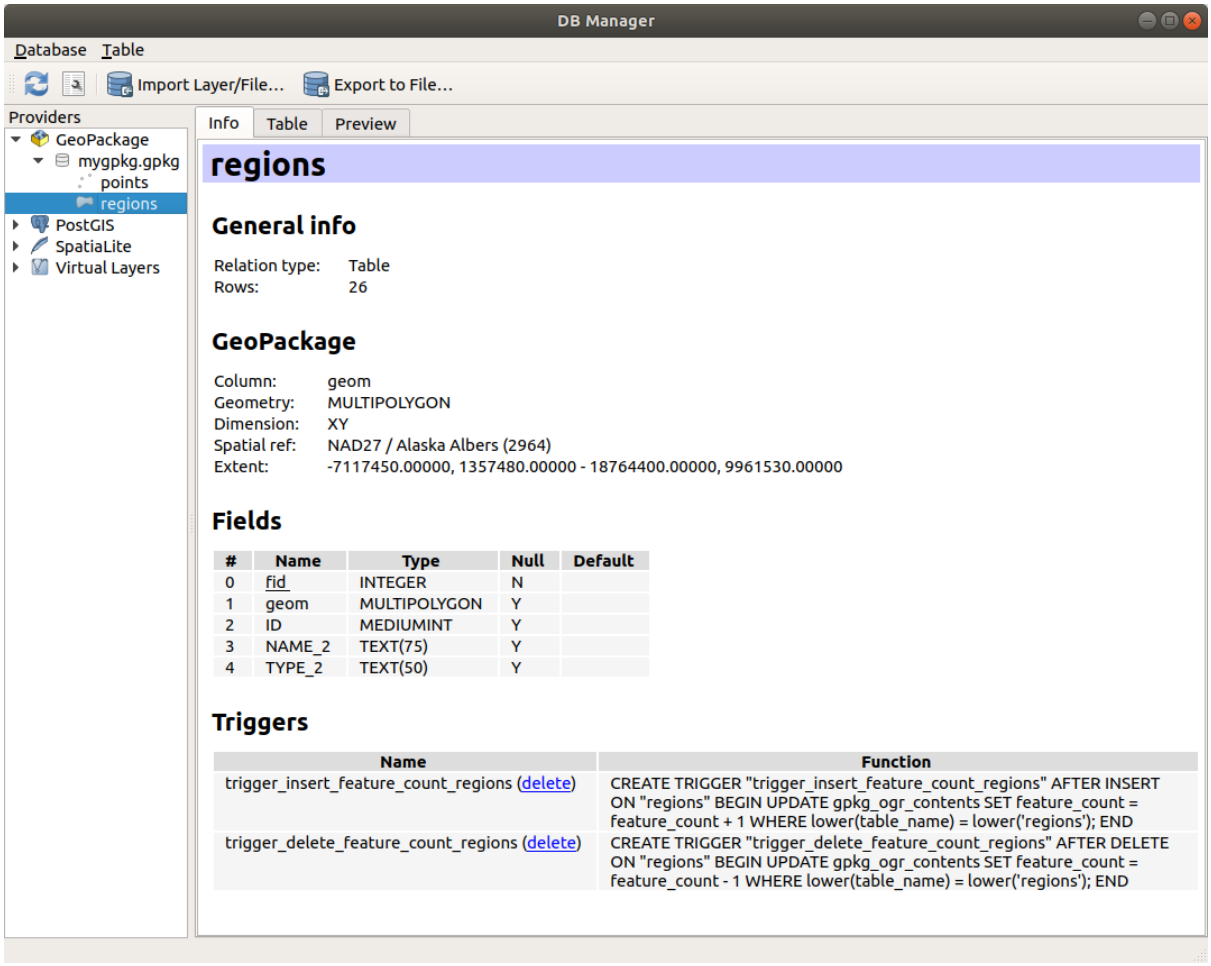
11.1.2 The DB Manager

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.

The DB Manager Plugin provides several features:

- connect to databases and display their structure and contents
- preview tables of databases
- add layers to the map canvas, either by double-clicking or drag-and-drop.
- add layers to a database from the QGIS Browser or from another database
- create SQL queries and add their output to the map canvas
- create *virtual layers*

More information on DB Manager capabilities is found in *DB Manager Plugin*.



Слика11.3: DB Manager dialog

11.1.3 Provider-based loading tools

Beside the Browser Panel and the DB Manager, the main tools provided by QGIS to add layers, you'll also find tools that are specific to data providers.

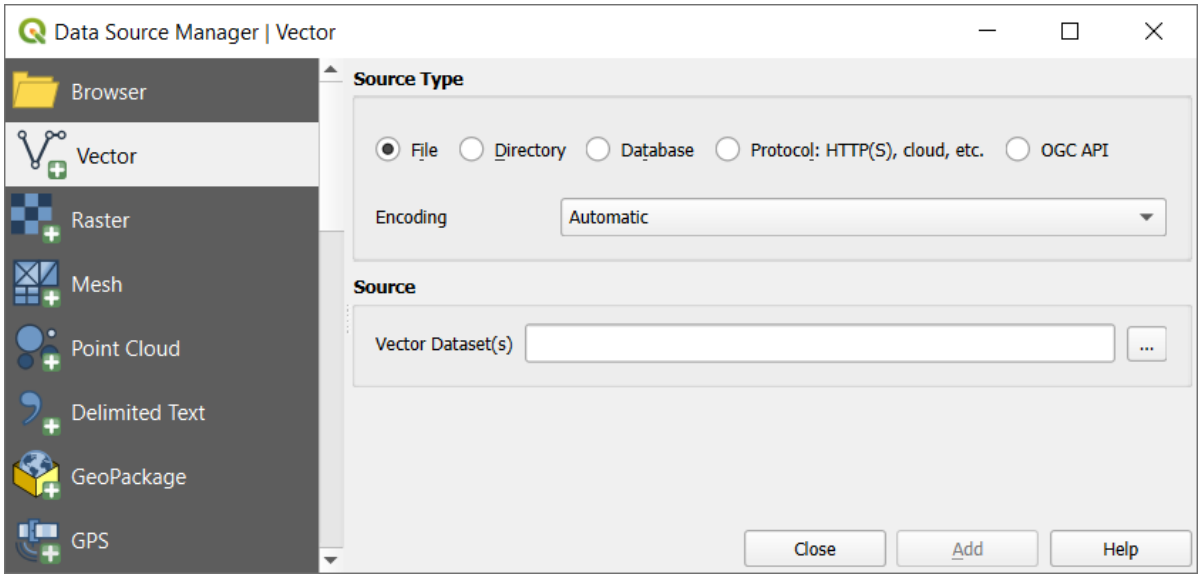
Белешка

Some *external plugins* also provide tools to open specific format files in QGIS.

Loading a layer from a file

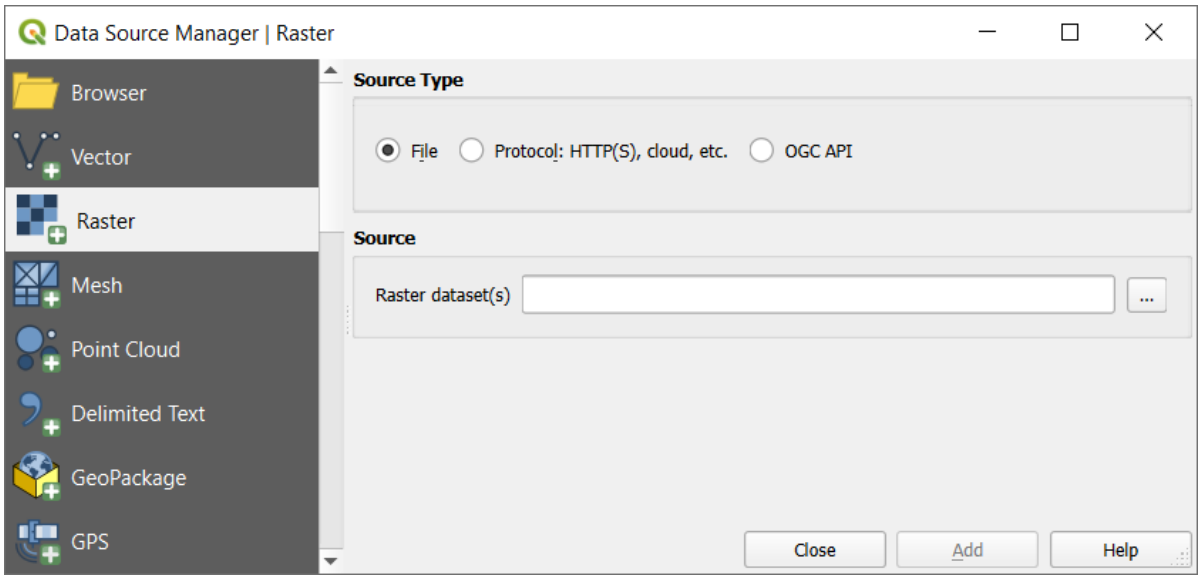
To load a layer from a file:

1. Open the layer type tab in the *Data Source Manager* dialog, ie click the Open Data Source Manager button (or press `Ctrl+L`) and enable the target tab or:
   - for vector data (like GML, ESRI Shapefile, Mapinfo and DXF layers): press `Ctrl+Shift+V`, select the *Layer* ► *Add Layer* ► *Add Vector Layer* menu option or click on the Add Vector Layer toolbar button.



Слика11.4: Add Vector Layer Dialog

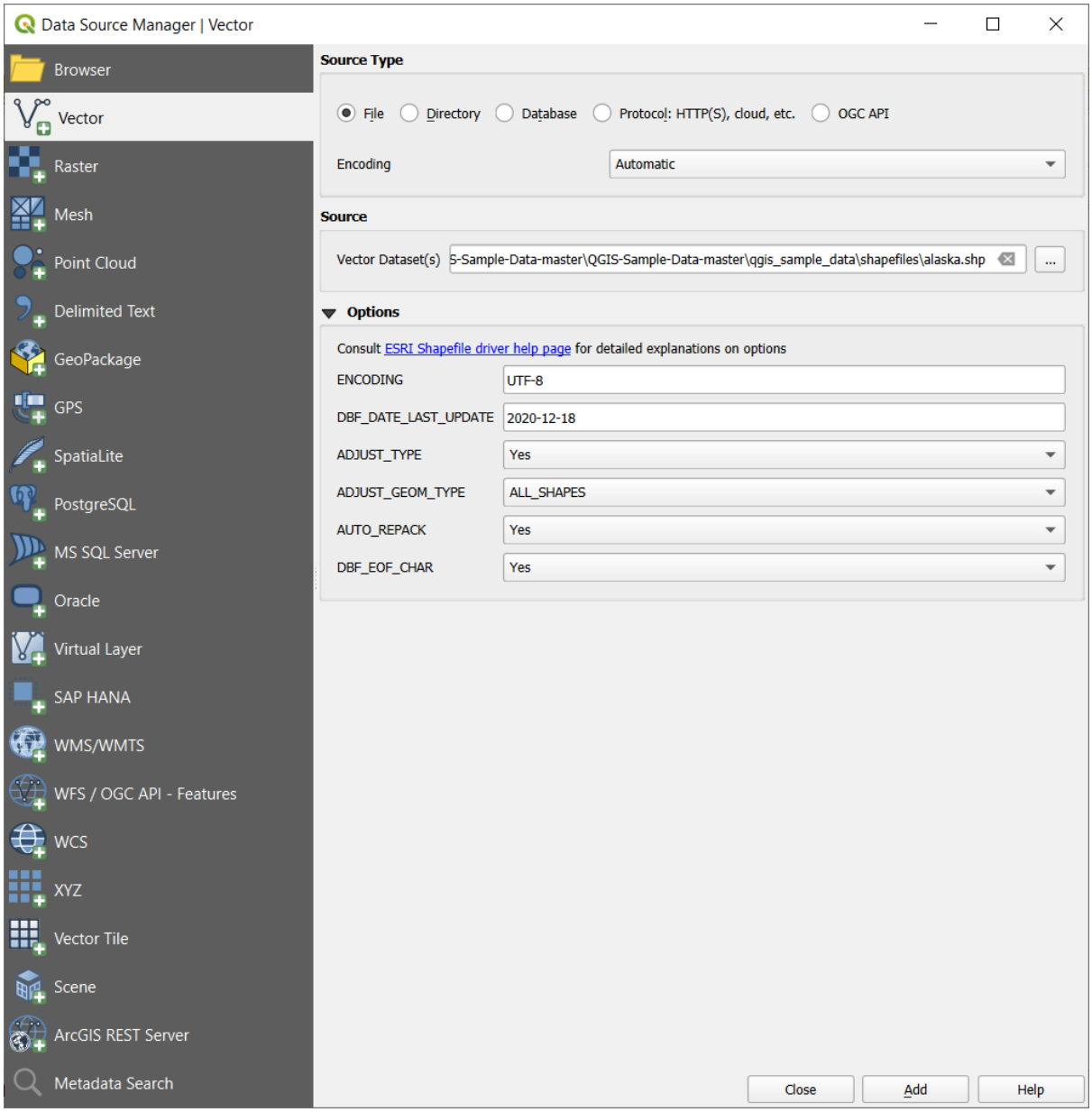
- for raster data (like GeoTiff, MBTiles, GRIdded Binary and DWG layers): press `Ctrl+Shift+R`, select the *Layer* ► *Add Layer* ► *Add Raster Layer* menu option or click on the Add Raster Layer toolbar button.



Слика11.5: Add Raster Layer Dialog

2. Check *File* source type
3. Click on the … Browse button
4. Navigate the file system and load a supported data source. More than one layer can be loaded at the same time by holding down the `Ctrl` key and clicking on multiple items in the dialog or holding down the `Shift` key to select a range of items by clicking on the first and last items in the range. Only formats that have been well tested appear in the formats filter. Other formats can be loaded by selecting `All files` (the top item in the pull-down menu).
5. Press *Open* to load the selected file into *Data Source Manager* dialog.

Depending on the selected layer type, additional *Options* (encoding, geometry type, table filtering, file locking, data formatting …) are available for configuring. These options are described in detail in the specific GDAL vector or raster driver documentation. At the top of the options, a text with hyperlink will directly lead to the documentation of the appropriate driver for the selected file format.

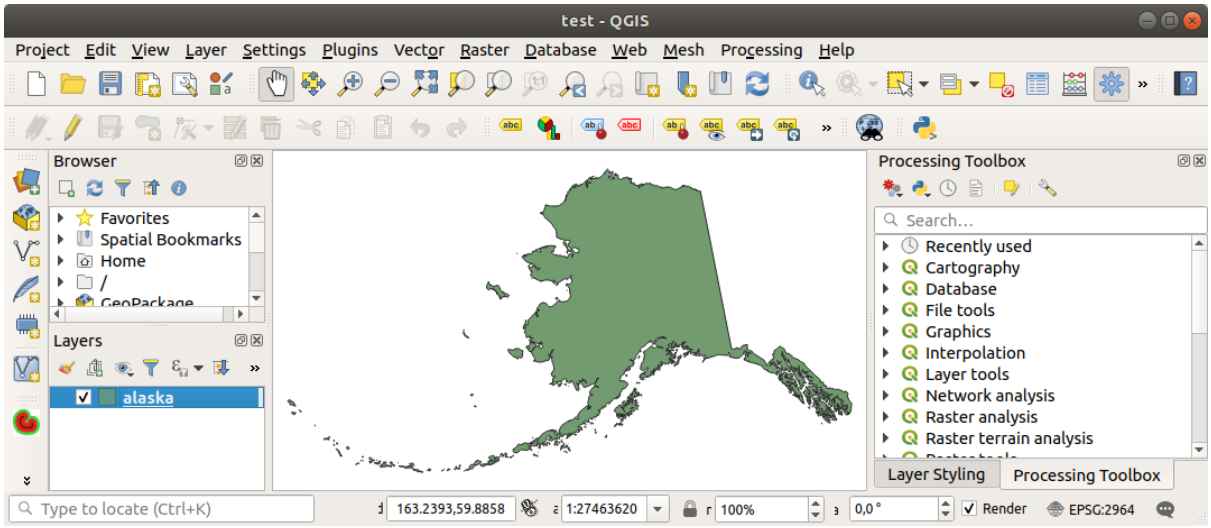


Слика11.6: Loading a Shapefile with open options

6. Press *Add* to load the file in QGIS and display them in the map view. When adding vector datasets containing multiple layers, the *Select Items to Add* dialog will appear. In this dialog, you can choose the specific layers from your dataset that you want to add. Also, under *Options* you can choose to:

   - *Add layers to a group*
   - *Show system and internal tables*
   - *Show empty vector layers.*

   Сл.11.7 shows QGIS after loading the `alaska.shp` file.



Слика11.7: QGIS with Shapefile of Alaska loaded

Белешка

Because some formats like MapInfo (e.g., `.tab`) or Autocad (`.dxf`) allow mixing different types of geometry in a single file, loading such datasets opens a dialog to select geometries to use in order to have one geometry per layer.

The Add Vector Layer and Add Raster Layer tabs allow loading of layers from source types other than *File*:

- You can load specific vector formats like `ArcInfo Binary Coverage`, `UK. National Transfer Format`, as well as the raw TIGER format of the `US Census Bureau` or `OpenfileGDB`. To do that, you select *Directory* as *Source type*. In this case, a directory can be selected in the dialog after pressing … Browse.
- With the *Database* source type you can select an existing database connection or create one to the selected database type. Some possible database types are `ODBC`, `Esri Personal Geodatabase`, `MS SQL Server` as well as `PostgreSQL` or `MySQL` .

  Pressing the *New* button opens the *Create a New OGR Database Connection* dialog whose parameters are among the ones you can find in *Creating a stored Connection*. Pressing *Open* lets you select from the available tables.
- The *Protocol: HTTP(S), cloud, etc.* source type opens data stored locally or on the network, either publicly accessible, or in private buckets of commercial cloud storage services. Supported protocol types are:
  - `HTTP/HTTPS/FTP`, with a *URI* and, if required, an *authentication*.
  - Cloud storage such as `AWS S3`, `Google Cloud Storage`, `Microsoft Azure Blob`, `Microsoft Azure Data Lake Storage`, `Alibaba OSS Cloud`, and `Open Stack Swift Storage` supports direct control over VSI *Credential Options* when adding OGR vector or GDAL raster layers. You need to fill in the *Bucket or container* and the *Object key* first. After that, you can add the necessary *Credential Options*.

    When adding OGR vector or GDAL raster layers from the cloud based protocols, you can also set additional *Credential options* for that specific driver and bucket. When credential options are found in a layer's URI, they will also be automatically set. This allows different layers to use different credentials.
  - service supporting OGC `WFS 3` (still experimental), using `GeoJSON` or `GEOJSON - Newline Delimited` format or based on `CouchDB` database. A *URI* is required, with optional *authentication*.
  - For all vector source types it is possible to define the *Encoding* or to use the *Automatic* ► setting.

- The *OGC API* source type allows you to access vector and raster data from servers that implement the OGC API standards. To use this option:

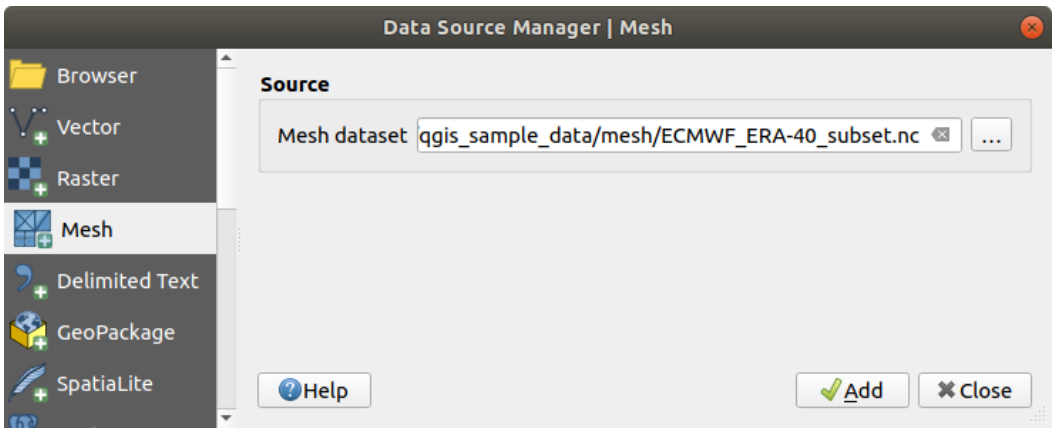
  1. Select *OGC API* from the *Data Source Manager* dialog.
  2. Enter the endpoint of the OGC API service you want to connect to. Note that you don't need to prefix the endpoint with „OGCAPI:".
  3. Click *Connect* to establish a connection to the server.

Loading a mesh layer

A mesh is an unstructured grid usually with temporal and other components. The spatial component contains a collection of vertices, edges and faces in 2D or 3D space. More information on mesh layers at *Working with Mesh Data*.

To add a mesh layer to QGIS:

1. Open the *Data Source Manager* dialog, either by selecting it from the *Layer* ► menu or clicking the Open Data Source Manager button.
2. Enable the *Mesh* tab on the left panel
3. Press the … Browse button to select the file. *Various formats* are supported.
4. Select the file and press *Add*. The layer will be added using the native mesh rendering.
5. If the selected file contains many mesh layers, then you'll be prompted with a dialog to choose the sublayers to load. Do your selection and press *OK* and the layers are loaded with the native mesh rendering. It's also possible to load them within a group.



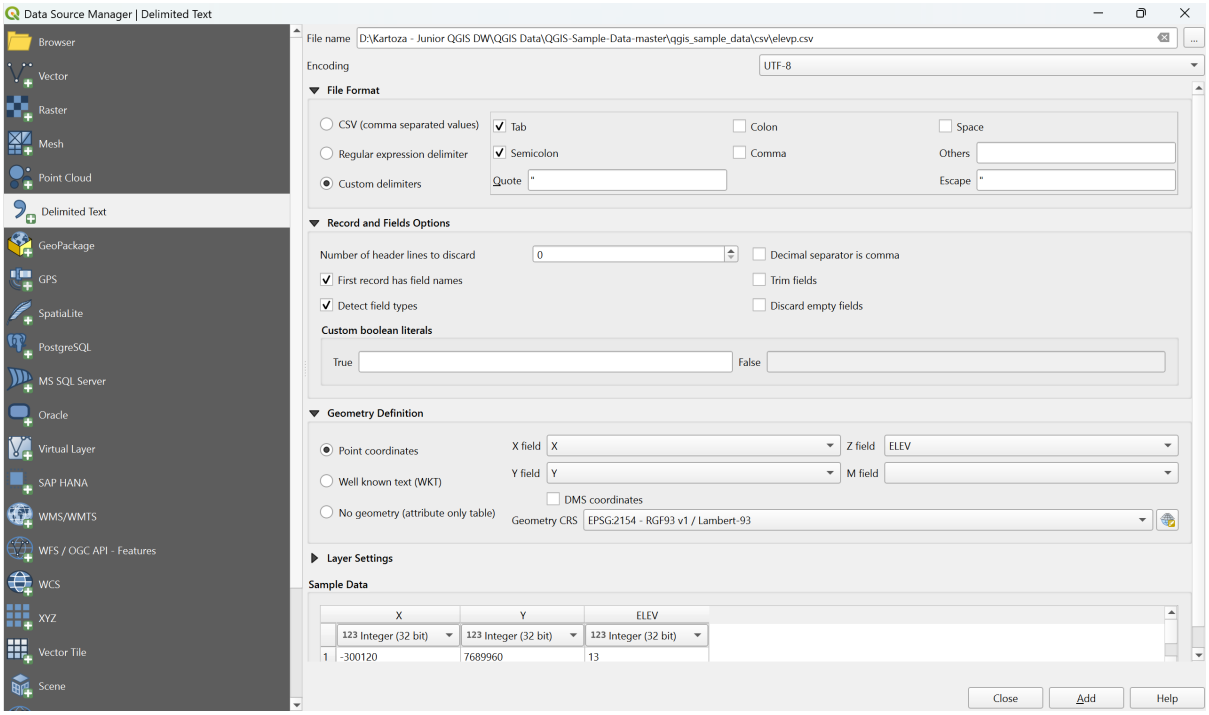
Слика11.8: Mesh tab in Data Source Manager

Importing a delimited text file

Delimited text files (e.g. `.txt`, `.csv`, `.dat`, `.wkt`) can be loaded using the tools described above. This way, they will show up as simple tables. Sometimes, delimited text files can contain coordinates / geometries that you could want to visualize. This is what *Add Delimited Text Layer* is designed for.

1. Click the Open Data Source Manager icon to open the *Data Source Manager* dialog
2. Enable the *Delimited Text* tab
3. Select the delimited text file to import (e.g., `qgis_sample_data/csv/elevp.csv`) by clicking on the … Browse button.

4. Configure the settings to meet your dataset and needs, as explained below.



Слика11.9: Delimited Text Dialog

File format

Once the file is selected, QGIS attempts to parse the file with the most recently used delimiter, identifying fields and rows. To enable QGIS to correctly parse the file, it is important to select the right delimiter. You can specify a delimiter by choosing between:

- *CSV (comma separated values)* to use the comma character.
- *Regular expression delimiter* and enter text into the *Expression* field. For example, to change the delimiter to tab, use `\t` (this is used in regular expressions for the tab character).
- *Custom delimiters*, choosing among some predefined delimiters like `comma`, `space`, `tab`, `semicolon`, ...

Records and fields

Some other convenient options can be used for data recognition:

- *Number of header lines to discard*: convenient when you want to avoid the first lines in the file in the import, either because those are blank lines or with another formatting.
- *First record has field names*: values in the first line are used as field names, otherwise QGIS uses the field names `field_1`, `field_2`...
- *Detect field types*: automatically recognizes the field type. If unchecked then all attributes are treated as text fields.
- *Decimal separator is comma*: you can force decimal separator to be a comma.
- *Trim fields*: allows you to trim leading and trailing spaces from fields.
- *Discard empty fields*.

- *Custom boolean literals*: allows you to add a custom couple of string that will be detected as boolean values.

Field type detection

QGIS tries to detect the field types automatically (unless *Detect field types* is not checked) by examining the content of an optional sidecar CSVT file (see GeoCSV specification) and by scanning the whole file to make sure that all values can actually be converted without errors, the fall-back field type is text.

The detected field type appears under the field name in sample data preview table and can be manually changed if necessary.

The following field types are supported:

- `Boolean` case-insensitive literal couples that are interpreted as boolean values are `1`/`0`, `true`/`false`, `t`/`f`, `yes`/`no`
- `Whole Number (integer)`
- `Whole Number (integer - 64 bit)`
- `Decimal Number`: double precision floating point number
- `Date`
- `Time`
- `Date and Time`
- `Text`

Geometry definition

Once the file is parsed, set *Geometry definition* to

- *Point coordinates* and provide the *X field*, *Y field*, *Z field* (for 3-dimensional data) and *M field* (for the measurement dimension) if the layer is of point geometry type and contains such fields. If the coordinates are defined as degrees/minutes/seconds, activate the *DMS coordinates* checkbox. Provide the appropriate *Geometry CRS* using the Select CRS widget.
- *Well known text (WKT)* option if the spatial information is represented as WKT: select the *Geometry field* containing the WKT geometry and choose the appropriate *Geometry field* or let QGIS auto-detect it. Provide the appropriate *Geometry CRS* using the Select CRS widget.
- If the file contains non-spatial data, activate *No geometry (attribute only table)* and it will be loaded as an ordinary table.

Layer settings

Additionally, you can enable:

- *Use spatial index* to improve the performance of displaying and spatially selecting features.
- *Use subset index* to improve performance of *subset filters* (when defined in the layer properties).
- *Watch file* to watch for changes to the file by other applications while QGIS is running.

At the end, click *Add* to add the layer to the map. In our example, a point layer named `Elevation` is added to the project and behaves like any other map layer in QGIS. This layer is the result of a query on the `.csv` source file (hence, linked to it) and would require *to be saved* in order to get a spatial layer on disk.

Sample Data

As you set the parser properties, the sample data preview updates regarding to the applied settings.

Also in the Sample Data Table it is possible to override the automatically determined column types.

Importing a DXF or DWG file

`DXF` and `DWG` files can be added to QGIS by simple drag-and-drop from the Browser Panel. You will be prompted to select the sublayers you would like to add to the project. Layers are added with random style properties.

Белешка

For DXF files containing several geometry types (point, line and/or polygon), the name of the layers will be generated as <*filename.dxf*> *entities* <*geometry type*>.

To keep the dxf/dwg file structure and its symbology in QGIS, you may want to use the dedicated *Project* ► *Import/Export* ► *Import Layers from DWG/DXF…* tool which allows you to:

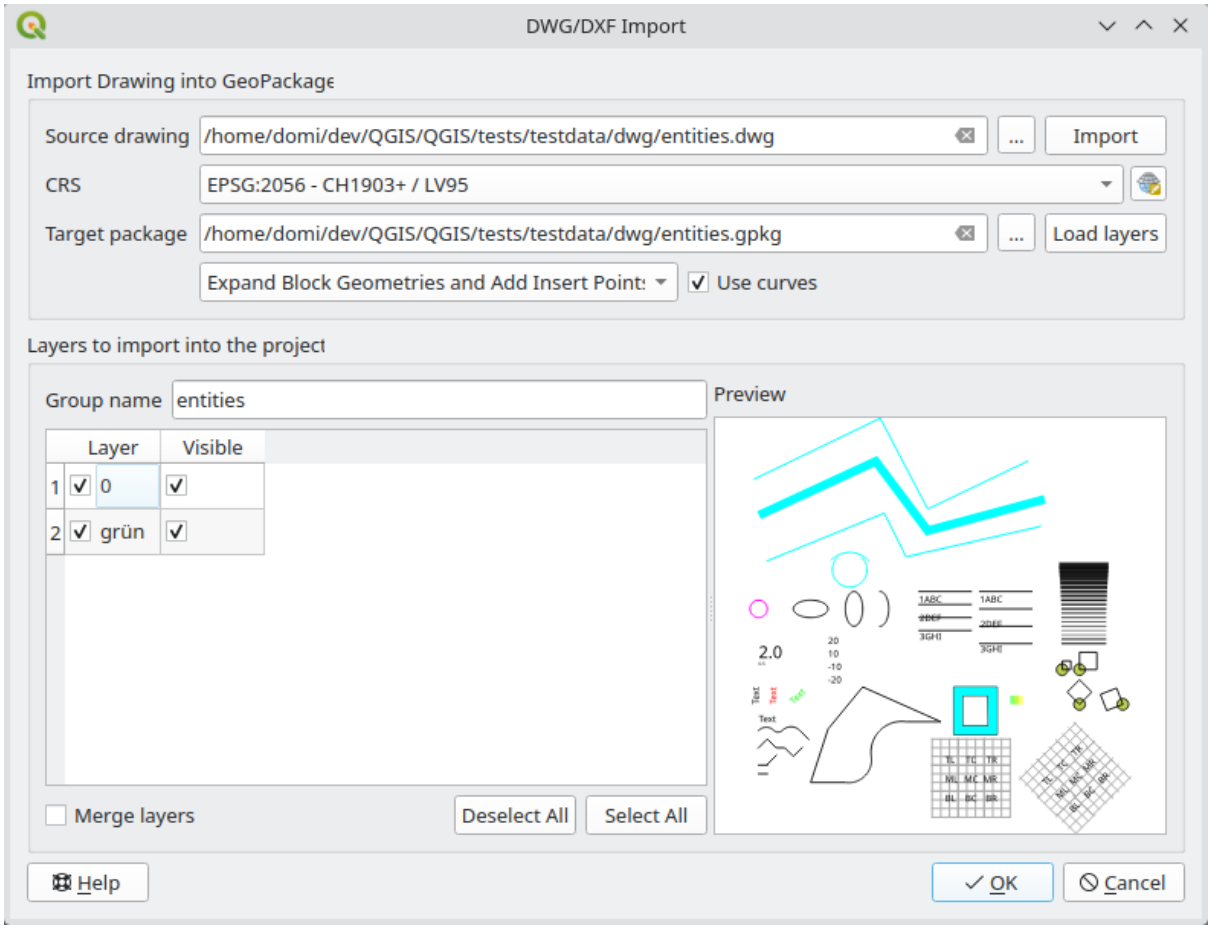
1. import elements from the drawing file into a GeoPackage database.
2. add imported elements to the project.

In the *DWG/DXF Import* dialog, to import the drawing file contents:

1. Input the location of the *Source drawing*, i.e. the DWG/DXF drawing file to import.
2. Specify the coordinate reference system of the data in the drawing file.
3. Input the location of the *Target package*, i.e. the GeoPackage file that will store the data. If an existing file is provided, then it will be overwritten.
4. Choose how to import `blocks` with the dedicated combobox:
   - *Expand Block Geometries*: imports the blocks in the drawing file as normal elements.
   - *Expand Block Geometries and Add Insert Points*: imports the blocks in the drawing file as normal elements and adds the insertion point as a point layer.
   - *Add Only Insert Points*: adds the blocks insertion point as a point layer.
5. Check *Use curves* to promote the imported layers to a `curved` geometry type.
6. Use the *Import* button to import the drawing into the destination GeoPackage file. The GeoPackage database will be automatically populated with the drawing file content. Depending on the size of the file, this can take some time.

After the `.dwg` or `.dxf` data has been imported into the GeoPackage database, the frame in the lower half of the dialog is populated with the list of layers from the imported file. There you can select which layers to add to the QGIS project:

1. At the top, set a *Group name* to group the drawing files in the project. By default this is set to the filename of the source drawing file.
2. Check layers to show: Each selected layer is added to an ad hoc group which contains vector layers for the point, line, label and area features of the drawing layer. The style of the layers will resemble the look they originally had in *CAD.
3. Choose if the layer should be visible at opening.
4. Checking the *Merge layers* option places all layers in a single group.
5. Press *OK* to open the layers in QGIS.



Слика11.10: Import dialog for DWG/DXF files

Importing OpenStreetMap Vectors

The OpenStreetMap project is popular because in many countries no free geodata such as digital road maps are available. The objective of the OSM project is to create a free editable map of the world from GPS data, aerial photography and local knowledge. To support this objective, QGIS provides support for OSM data.

Using the *Browser Panel*, you can load an `.osm` file to the map canvas, in which case you'll get a dialog to select sublayers based on the geometry type. The loaded layers will contain all the data of that geometry type in the `.osm` file, and keep the `osm` file data structure.

SpatiaLite Layers

The first time you load data from a SpatiaLite database, begin by:

- clicking on the Add SpatiaLite Layer toolbar button
- selecting the *Add SpatiaLite Layer…* option from the *Layer* ► *Add Layer* menu
- or by typing `Ctrl+Shift+L`

This will bring up a window that will allow you either to connect to a SpatiaLite database already known to QGIS (which you choose from the drop-down menu) or to define a new connection to a new database. To define a new connection, click on *New* and use the file browser to point to your SpatiaLite database, which is a file with a `.sqlite` extension.

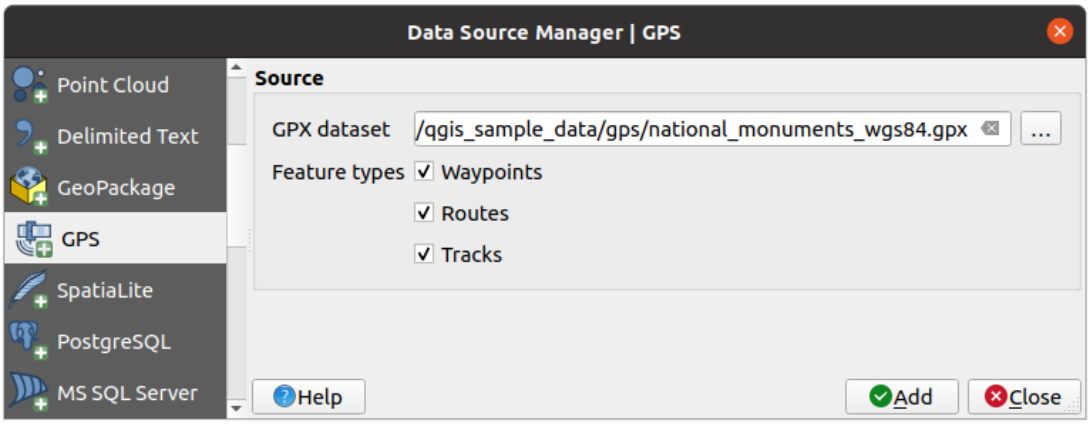
QGIS also supports editable views in SpatiaLite.

GPS

There are dozens of different file formats for storing GPS data. The format that QGIS uses is called GPX (GPS eXchange format), which is a standard interchange format that can contain any number of waypoints, routes and tracks in the same file.

Use the ... Browse button to select the GPX file, then use the check boxes to select the feature types you want to load from that GPX file. Each feature type will be loaded in a separate layer.

More on GPS data manipulation at *Rad sa GPS podacima*.



Слика11.11: Loading GPS Data dialog

GRASS

Working with GRASS vector data is described in section *GRASS GIS Integration*.

Database related tools

Creating a stored Connection

In order to read and write tables from a database format QGIS supports you have to create a connection to that database. While *QGIS Browser Panel* is the simplest and recommended way to connect to and use databases, QGIS provides other tools to connect to each of them and load their tables:

- *Add PostgreSQL Layer...* or by typing `Ctrl+Shift+D`
- *Add Oracle Spatial Layer...* or by typing `Ctrl+Shift+O`
- *Add MS SQL Server Layer*
- *Add SAP HANA Spatial Layer...* or by typing `Ctrl+Shift+G`

These tools are accessible either from the *Manage Layers Toolbar* and the *Layer* ► *Add Layer* ► menu. Connecting to SpatiaLite database is described at *SpatiaLite Layers*.

Савет

Create connection to database from the QGIS Browser Panel

Selecting the corresponding database format in the Browser tree, right-clicking and choosing connect will provide you with the database connection dialog.

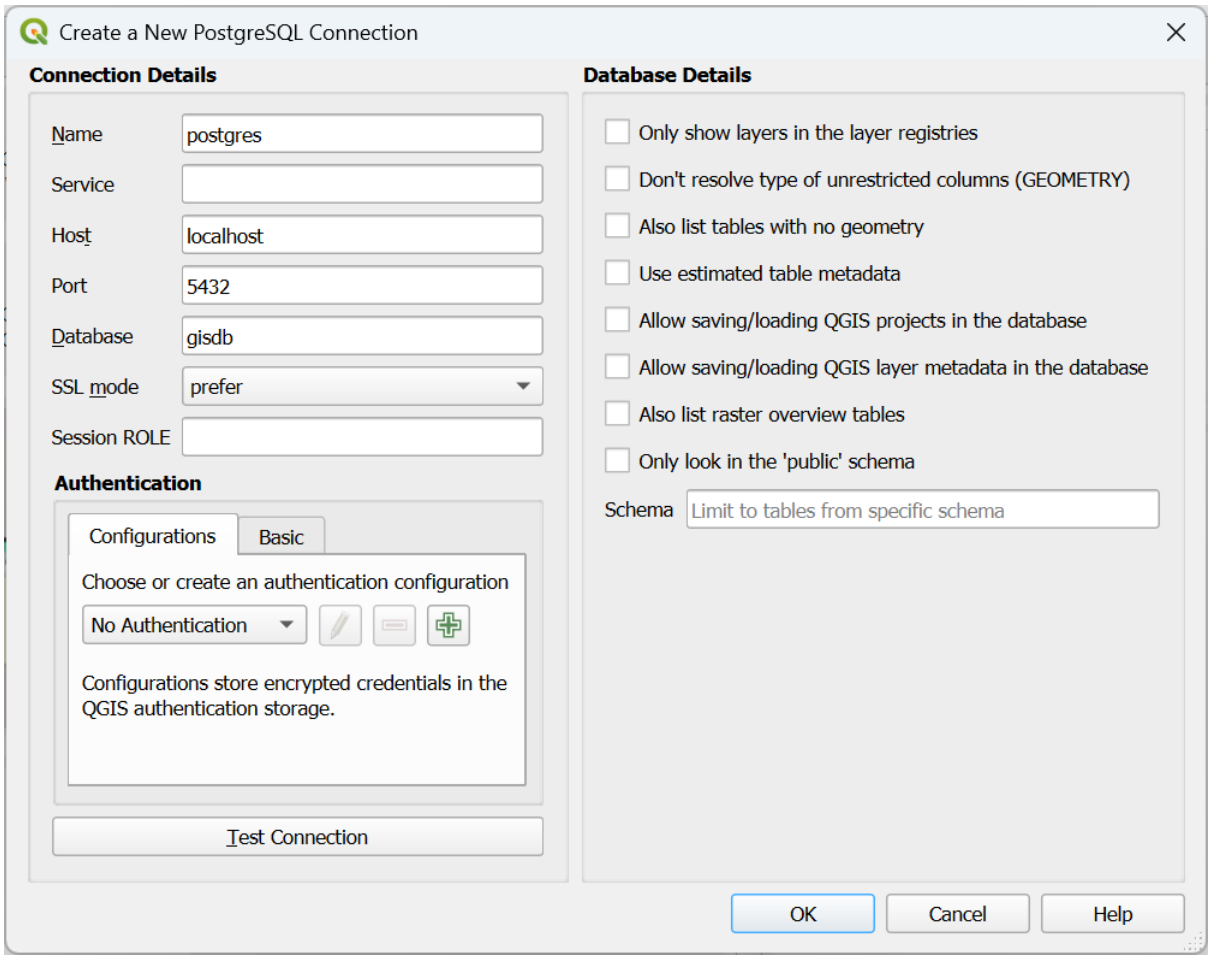
Most of the connection dialogs follow a common structure:

- a section with credentials information to connect to the database

- a section with options to tune which data can be requested in the database

Connecting to PostgreSQL

The first time you use a PostgreSQL data source, you must create a connection to a database that contains the data. Press the appropriate button as exposed above, opening the *PostgreSQL* tab of the *Data Source Manager* dialog. To access the connection manager, click on the *New* button to display the *Create a New PostgreSQL Connection* dialog.



Слика11.12: Create a New PostgreSQL Connection Dialog

- *Name*: A name for this connection. It can be the same as *Database*.
- *Service*: Service parameter to be used alternatively to hostname/port (and potentially database). This can be defined in `pg_service.conf`. Check the *PostgreSQL Service connection file* section for more details.
- *Host*: Name of the database host. This must be a resolvable host name such as would be used to open a TCP/IP connection or ping the host. If the database is on the same computer as QGIS, simply enter *localhost* here.
- *Port*: Port number the PostgreSQL database server listens on. The default port for PostgreSQL is `5432`.
- *Database*: Name of the database.
- *SSL mode*: SSL encryption setup. The following options are available:
  - *Prefer* (the default): I don't care about encryption, but I wish to pay the overhead of encryption if the server supports it.
  - *Require*: I want my data to be encrypted, and I accept the overhead. I trust that the network will make sure I always connect to the server I want.

  - *Verify CA*: I want my data encrypted, and I accept the overhead. I want to be sure that I connect to a server that I trust.
  - *Verify Full*: I want my data encrypted, and I accept the overhead. I want to be sure that I connect to a server I trust, and that it's the one I specify.
  - *Allow*: I don't care about security, but I will pay the overhead of encryption if the server insists on it.
  - *Disable*: I don't care about security, and I don't want to pay the overhead of encryption.
- *Session role*: used to set the current user identifier of the current session. This is useful to automatically give the ownership of a new object (table, view, function) to the session_role group and thus share ownership and associated rights with all members of the session_role group. Read more about session role.
- *Authentication*: For general details about the authentication dialog behavior, see *Authentication*.

Optionally, depending on the type of database, you can activate the following checkboxes:

- ☑ *Only show layers in the layer registries*
- ☑ *Don't resolve type of unrestricted columns (GEOMETRY)*
- ☑ *Also list tables with no geometry*: indicates that tables without geometry should also be listed by default.
- ☑ *Use estimated table metadata*: When initializing layers, various queries may be needed to establish the characteristics of the geometries stored in the database table. When this option is checked, these queries examine only a sample of the rows and use the table statistics, rather than the entire table. This can drastically speed up operations on large datasets, but may result in incorrect characterization of layers (e.g. the feature count of filtered layers will not be accurately determined) and may even cause strange behaviour if columns that are supposed to be unique actually are not.
- ☑ *Allow saving/loading QGIS projects in the database* - more details *here*
- ☑ *Allow saving/loading QGIS layer metadata in the database* - more details *here*
- ☑ *Also list raster overview tables*
- ☑ *Only look in the 'public' schema*
- *Schema*: Allows to specify a single schema to limit a connection to. When set, only tables from the matching schema will be included in the browser panel and data source select for the connection. This can be used to limit the database work required to populate tables for a connection pointing to a large database store.

Once all parameters and options are set, you can test the connection by clicking the *Test Connection* button or apply it by clicking the *OK* button.

PostgreSQL Service connection file

The service connection file allows PostgreSQL connection parameters to be associated with a single service name. That service name can then be specified by a client and the associated settings will be used.

It's called `.pg_service.conf` under *nix systems (GNU/Linux, macOS etc.) and `pg_service.conf` on Windows.

The service file can look like this:

```
[water_service]
host=192.168.0.45
port=5433
dbname=gisdb
user=paul
password=paulspass

[wastewater_service]
host=dbserver.com
```

(наставак на следећој страни)

(настављено са претходне стране)

```
dbname=water
user=waterpass
```

Белешка

There are two services in the above example: `water_service` and `wastewater_service`. You can use these to connect from QGIS, pgAdmin, etc. by specifying only the name of the service you want to connect to (without the enclosing brackets). If you want to use the service with `psql`, you can do `psql service=water_service`.

You can find all the PostgreSQL parameters here

Белешка

If you don't want to save the passwords in the service file you can use the .pg_pass option.

Белешка

QGIS Server and service

When using a service file and QGIS Server, you must configure the service on the server side as well. You can follow the QGIS Server documentation.

On *nix operating systems (GNU/Linux, macOS etc.) you can save the `.pg_service.conf` file in the user's home directory and PostgreSQL clients will automatically be aware of it. For example, if the logged user is `web`, `.pg_service.conf` should be saved in the `/home/web/` directory in order to directly work (without specifying any other environment variables).

You can specify the location of the service file by creating a `PGSERVICEFILE` environment variable (e.g. run the `export PGSERVICEFILE=/home/web/.pg_service.conf` command under your *nix OS to temporarily set the `PGSERVICEFILE` variable).

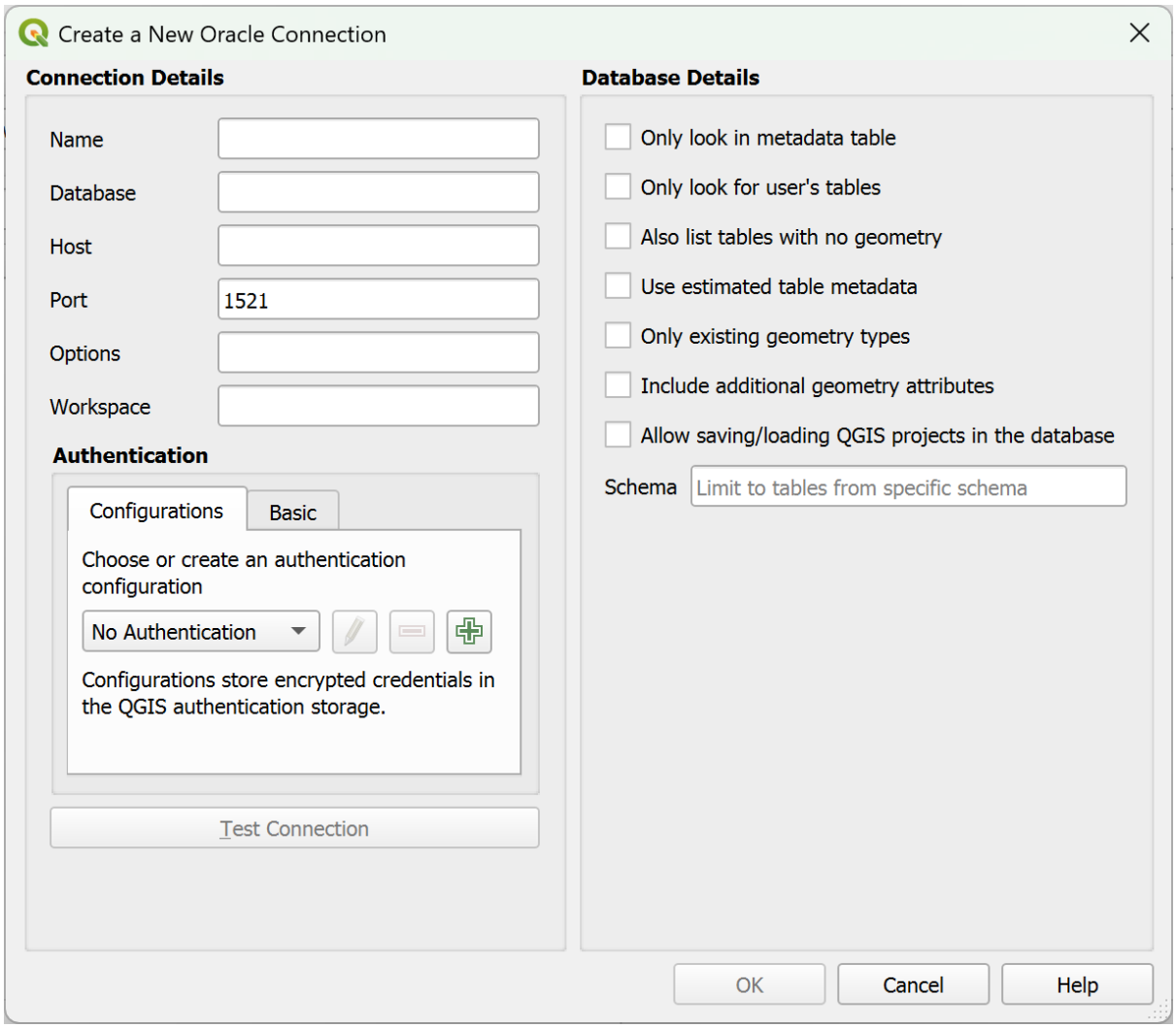
You can also make the service file available system-wide (all users) either by placing the `.pg_service.conf` file in `pg_config --sysconfdir` or by adding the `PGSYSCONFDIR` environment variable to specify the directory containing the service file. If service definitions with the same name exist in the user and the system file, the user file takes precedence.

Упозорење

There are some caveats under Windows:

- The service file should be saved as `pg_service.conf` and not as `.pg_service.conf`.
- The service file should be saved in Unix format in order to work. One way to do it is to open it with Notepad++ and *Edit* ► *EOL Conversion* ► *UNIX Format* ► *File save*.
- You can add environmental variables in various ways; a tested one, known to work reliably, is *Control Panel* ► *System and Security* ► *System* ► *Advanced system settings* ► *Environment Variables* adding `PG-SERVICEFILE` with the path - e.g. `C:\Users\John\pg_service.conf`
- After adding an environment variable you may also need to restart the computer.

Connecting to Oracle Spatial



Слика11.13: Create a New Oracle Connection Dialog

The spatial features in Oracle Spatial aid users in managing geographic and location data in a native type within an Oracle database. The connection dialog proposes:

- *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 *Authentication*.

Optionally, you can activate the following checkboxes:

- ☑ *Only look in metadata table*: restricts the displayed tables to those that are in the `all_sdo_geom_metadata` view. This can speed up the initial display of spatial tables.
- ☑ *Only look for user's tables*: when searching for spatial tables, restricts the search to tables that are owned by the user.
- ☑ *Also list tables with no geometry*: indicates that tables without geometry should also be listed by default.
- ☑ *Use estimated table statistics for the layer metadata*: when the layer is set up, various metadata are required for the Oracle table. This includes information such as the table row count, geometry type and spatial extents of the data in the geometry column. If the table contains a large number of rows, determining this metadata can be time-consuming. By activating this option, the following fast table metadata operations are done: Row count is determined from `all_tables.num_rows`. Table extents are always determined with the SDO_TUNE.EXTENTS_OF function, even if a layer filter is applied. Table geometry is determined from the first 100 non-null geometry rows in the table.
- ☑ *Only existing geometry types*: only lists the existing geometry types and don't offer to add others.
- ☑ *Include additional geometry attributes.*
- ☑ *Allow saving/loading QGIS projects in the database* - more details *here*
- *Schema*: Allows to specify a single schema to limit a connection to. When set, only tables from the matching schema will be included in the browser panel and data source select for the connection. This can be used to limit the database work required to populate tables for a connection pointing to a large database store.

Савет

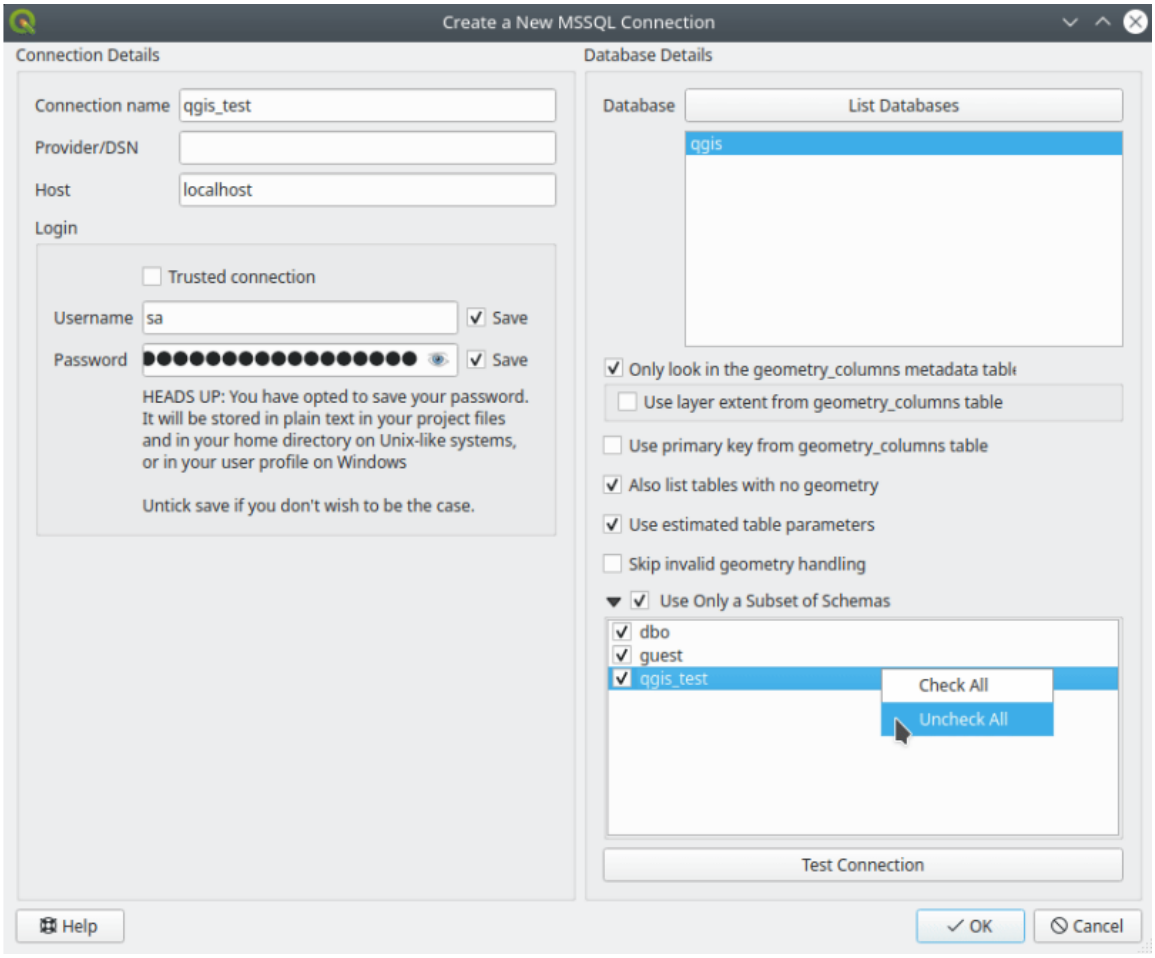
Oracle Spatial Layers

Normally, an Oracle Spatial layer is defined by an entry in the **USER_SDO_METADATA** table.

To ensure that selection tools work correctly, it is recommended that your tables have a **primary key**.

Connecting to MS SQL Server

As mentioned in *Creating a stored Connection* QGIS allows you to create MS SQL Server connection through *Data Source Manager*.



Слика11.14: MS SQL Server Connection

To create a new MS SQL Server connection, you need to provide some of the following information in the *Connection Details* dialog:

- *Connection name*
- *Provider/DSN*
- *Host*
- *Login* information. You can choose to *Save* your credentials.

Navigate to the *Database Details* section and click the *List Databases* button to view the available datasets. Select datasets that you want, then press *OK*. Optionally, you can also perform a *Test Connection*. Once you click *OK* the

Create a New MS SQL Server Connection dialog will close and in the *Data Source Manager* press *Connect*, select a layer and then click *Add*.

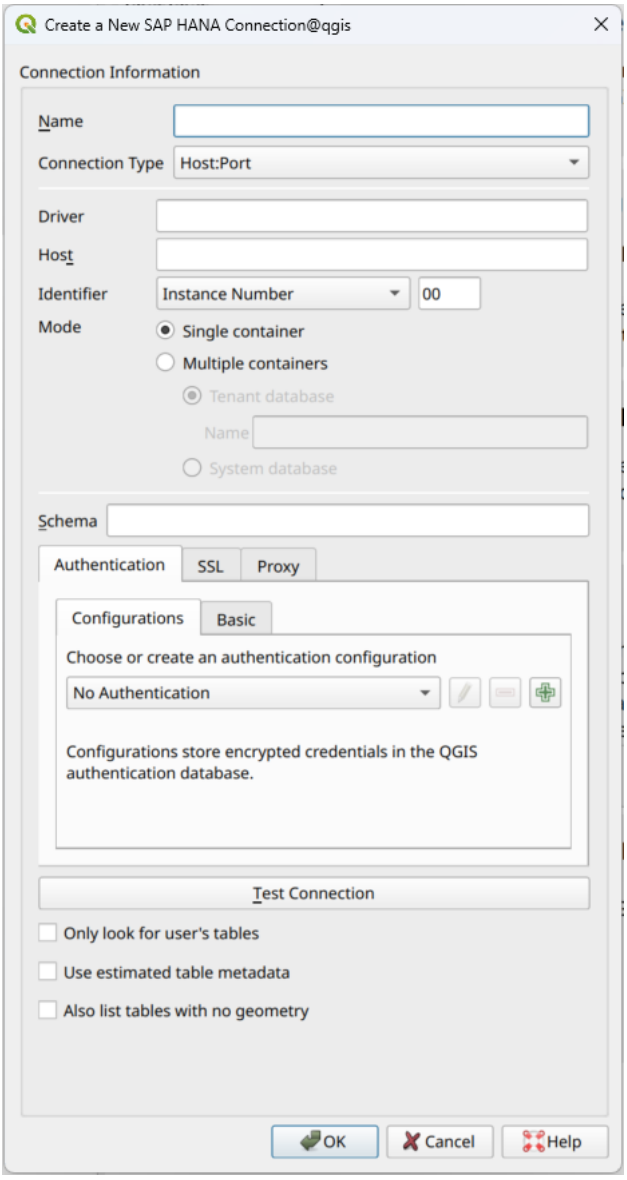
Optionally, you can activate the following options:

- *Only look in the geometry_columns metadata table*: restricts the available tables to the ones in the `geometry_columns` metadata table when scanning for tables. This can speed up the table scanning.
- *Use layer extent from geometry_columns table*: this option, dependent on the previous one, allows QGIS to skip extent calculation when loading layers and thus lowering the amount of time needed to load them. It relies on extent manually specified using additional QGIS-specific columns (`qgis_xmin`, `qgis_xmax`, `qgis_ymin`, `qgis_ymax`) in the `geometry_columns` table.
- *Use primary key from geometry_columns table*: allows QGIS to skip primary key calculation for views when loading them, thus lowering the amount of time needed to load them. It relies on names manually filled in a QGIS-specific `qgis_pkey` column set in the `geometry_columns` table. If more than one column is used for the primary key, they should be filled as comma separated values.
- *Also list table with no geometry*: tables without a geometry column attached will also be shown in the available table list.
- *Use estimated table parameters*: only estimated table metadata will be used. This avoids a slow table scan, but may result in incorrect layer properties such as layer extent.
- *Skip invalid geometry handling*: all handling of records with invalid geometry will be disabled. This speeds up the provider, however, if any invalid geometries are present in a table then the result is unpredictable and may include missing records. Only check this option if you are certain that all geometries present in the database are valid, and any newly added geometries or tables will also be valid.
- *Use only a Subset of Schemas* will allow you to filter schemas for MS SQL connection. If enabled, only checked schemas will be displayed. You can right-click to *Check* or *Uncheck* any schema in the list.

Connecting to SAP HANA

Белешка

You require the SAP HANA Client to connect to a SAP HANA database. You can download the SAP HANA Client for your platform at the SAP Development Tools website.



Слика11.15: Create a New SAP HANA Connection Dialog

The following parameters can be entered:

- *Name*: A name for this connection.
- *Driver* : The name of the HANA ODBC driver. It is `HDBODBC` if you are using 64-bit QGIS, `HDBODBC32` if you are using 32-bit QGIS. The appropriate driver name is entered automatically.
- *Driver* : Either the name under which the SAP HANA ODBC driver has been registered in `/etc/odbcinst.ini` or the full path to the SAP HANA ODBC driver. The SAP HANA Client installer will install the ODBC driver to `/usr/sap/hdbclient/libodbcHDB.so` by default.
- *Host*: The name of the database host.

- *Identifier*: Identifies the instance to connect to on the host. This can be either *Instance Number* or *Port Number*. Instance numbers consist of two digits, port numbers are in the range from 1 to 65,535.
- *Mode*: Specifies the mode in which the SAP HANA instance runs. This setting is only taken into account if *Identifier* is set to *Instance Number*. If the database hosts multiple containers, you can either connect to a tenant with the name given at *Tenant database* or you can connect to the system database.
- *Schema*: This parameter is optional. If a schema name is given, QGIS will only search for data in that schema. If this field is left blank, QGIS will search for data in all schemas.
- *Authentication*: For general details about the authentication dialog behavior, see *Authentication*.
- *SSL Settings*
  - *Enable TLS/SSL encryption*: Enables TLS 1.1 - TLS1.2 encryption. The server will choose the highest available.
  - *Provider*: Specifies the cryptographic library provider used for SSL communication. *sapcrypto* should work on all platforms, *openssl* should work on , *mscrypto* should work on and *commoncrypto* requires CommonCryptoLib to be installed.
  - *Validate SSL certificate*: If checked, the SSL certificate will be validated using the truststore given in *Trust store file with public key*.
  - *Override hostname in certificate*: Specifies the host name used to verify server's identity. The host name specified here verifies the identity of the server instead of the host name with which the connection was established. If you specify `*` as the host name, then the server's host name is not validated. Other wildcards are not permitted.
  - *Keystore file with private key*: Currently ignored. This parameter might allow to authenticate via certificate instead via user and password in future.
  - *Trust store file with public key*: Specifies the path to a trust store file that contains the server's public certificates if using OpenSSL. Typically, the trust store contains the root certificate or the certificate of the certification authority that signed the server's public certificates. If you are using the cryptographic library CommonCryptoLib or msCrypto, then leave this property empty.
- *Only look for user's tables*: If checked, QGIS searches only for tables and views that are owned by the user that connects to the database.
- *Use estimated table metadata*: If checked, estimated table metadata will be used if available. For large tables, this avoids slow table loads and potentially expensive computations, but may result in incorrect layer properties such as layer extent. The fast extent estimation is available starting with QRC1/2024 and SP8 in HANA Cloud and HANA On-Premise respectively.
- *Also list tables with no geometries*: If checked, QGIS searches also for tables and views that do not contain a spatial column.

Савет

Connecting to SAP HANA Cloud

If you'd like to connect to an SAP HANA Cloud instance, you usually must set *Port Number* to `443` and check *Enable TLS/SSL encryption*.

Loading a Database Layer

Once you have one or more connections defined to a database (see section *Creating a stored Connection*), you can load layers from it. Of course, this requires that data are available. See section *Importing Data into PostgreSQL* for a discussion on importing data into a PostgreSQL database.

To load a layer from a database, you can perform the following steps:

1. Open the corresponding tab of the database in the *Data Source Manager* dialog.
2. Choose the connection name from the drop-down list and press *Connect*.
3. The table below will be filled with your data grouped by schema, with a number of metadata information helpful for loading.
4. Select or unselect *Also list tables with no geometry*.
5. Optionally, use some *Search Options* to reduce the list of tables to those matching your search. You can also set this option before you hit the *Connect* button, speeding up the database fetching.
6. Find the layer(s) you wish to add in the list of available layers.
7. Select it by clicking on it. You can select multiple layers by holding down the `Shift` or `Ctrl` key while clicking.

   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:

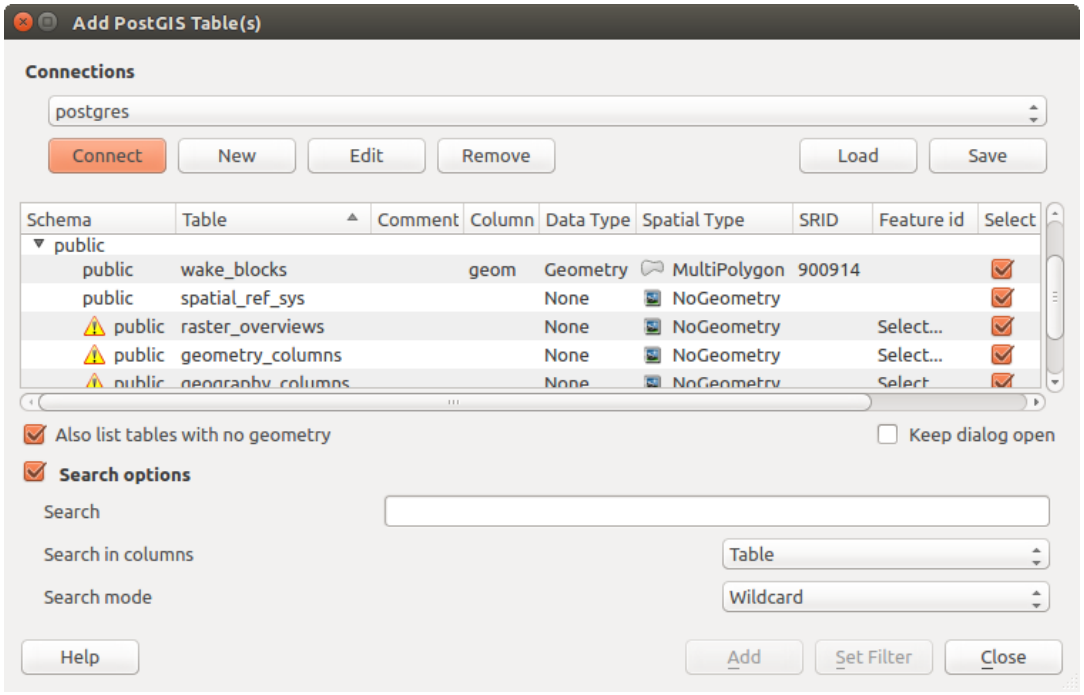
   - features geometry type: in which case you can select the appropriate one in the drop-down list of the *Spatial Type* column
   - layer CRS: you can enter the correct code in the *SRID* column
   - layer's primary key, in order to unequivocally identify each feature: this can be fixed by selecting one or more attributes in the drop-down list at the corresponding *Feature id* column.

   Савет

   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. If applicable, use the *Set Filter* button (or double-click the layer) to start the *Query Builder* dialog (see section *Query Builder*) and define which features to load from the selected layer. The filter expression appears in the `sql` column. This restriction can be removed or edited in the *Layer Properties* ► *General* ► *Provider Feature Filter* frame.
9. The checkbox in the `Select at id` column that is activated by default gets the feature ids without the attributes and generally speeds up the data loading.
10. Click on the *Add* button to add the layer to the map.



Слика11.16: Add PostgreSQL Table(s) Dialog

Савет

Use the Browser Panel to speed up loading of database table(s)

Adding DB tables from the *Data Source Manager* may sometimes be time consuming as QGIS fetches statistics and properties (e.g. geometry type and field, CRS, number of features) for each table beforehand. To avoid this, once *the connection is set*, it is better to use the *Browser Panel* or the *DB Manager* to drag and drop the database tables into the map canvas.

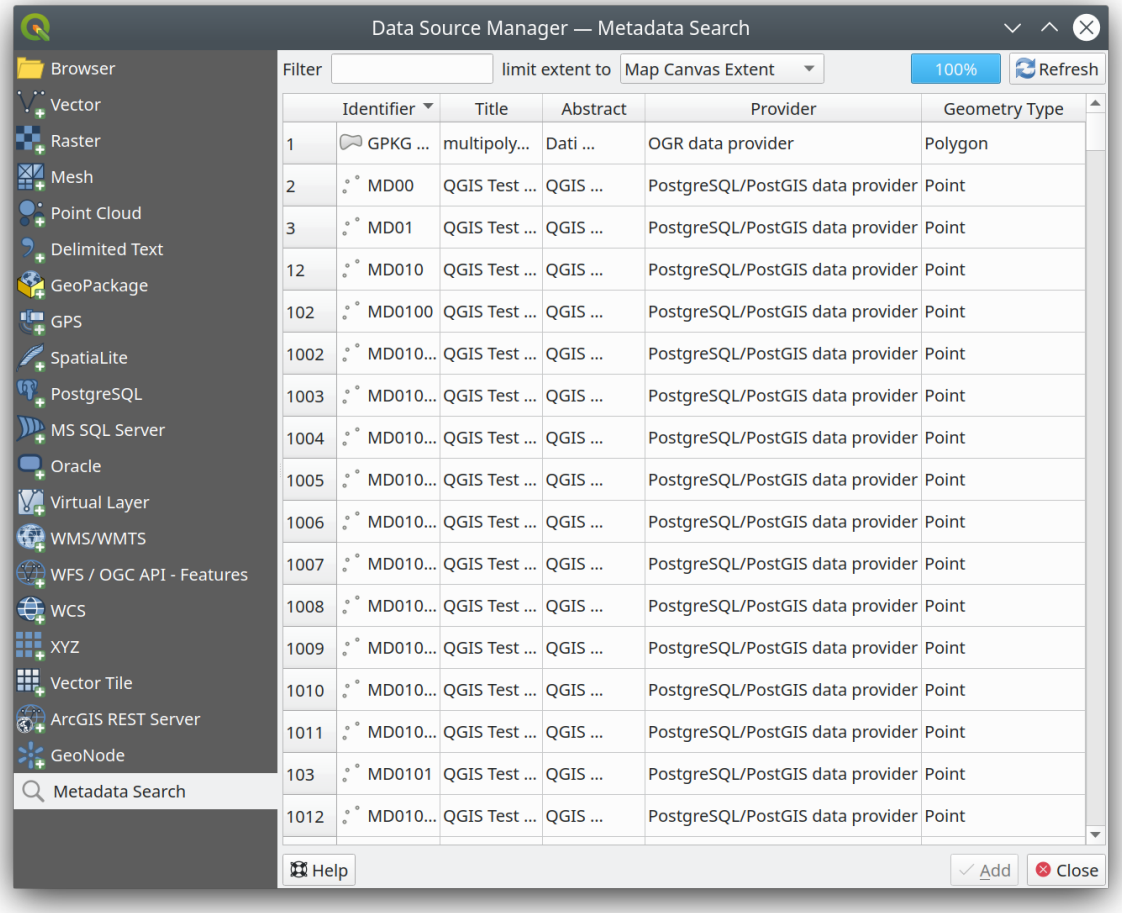
11.1.4 The Layer Metadata Search Panel

By default, QGIS can retrieve layers metadata from the connections or data providers that allow metadata storage (more details on *saving metadata to the database*). The *Metadata search* panel allows to browse the layers by their metadata and add them to the project (either with a double-click or the *Add* button). The list can be filtered:

- by text, watching a set of metadata properties (identifier, title, abstract)
- by spatial extent, using the current *project extent* or the map canvas extent
- by the layer (geometry) type

Белешка

The sources of metadata are implemented through a layer metadata provider system that can be extended by plugins.



Слика11.17: Layer Metadata Search Panel

11.1.5 QGIS Custom formats

QGIS proposes two custom formats:

- Temporary Scratch Layer: a memory layer that is bound to the project (see *Creating a new Temporary Scratch Layer* for more information)
- Virtual Layers: a layer resulting from a query on other layer(s) (see *Creating virtual layers* for more information)

11.1.6 QLR - QGIS Layer Definition File

Layer definitions can be saved as a *Layer Definition File* (QLR - `.qlr`) using *Export* ► *Save As Layer Definition File…* in the layer context menu.

The QLR format makes it possible to share „complete” QGIS layers with other QGIS users. QLR files contain links to the data sources and all the QGIS style information necessary to style the layer.

QLR files are shown in the Browser Panel and can be used to add layers (with their saved styles) to the Layers Panel. You can also drag and drop QLR files from the system file manager into the map canvas.

Available actions for QLR files in the Browser Panel are:

- *Export Layer* ► *To file*
- *Add Layer to Project*
- *Svojstva sloja…*

11.1.7 Connecting to web services

With QGIS you can get access to different types of OGC web services (WM(T)S, WFS(-T), WCS, CSW, …). Thanks to QGIS Server, you can also publish such services. QGIS-Server-manual contains descriptions of these capabilities.

Using Vector Tiles services

Vector Tile services can be added via the  *Vector Tiles* tab of the *Data Source Manager* dialog or the contextual menu of the *Vector Tiles* entry in the *Browser* panel. Services can be either a *New Generic Connection…* or a *New ArcGIS Vector Tile Service Connection…*.

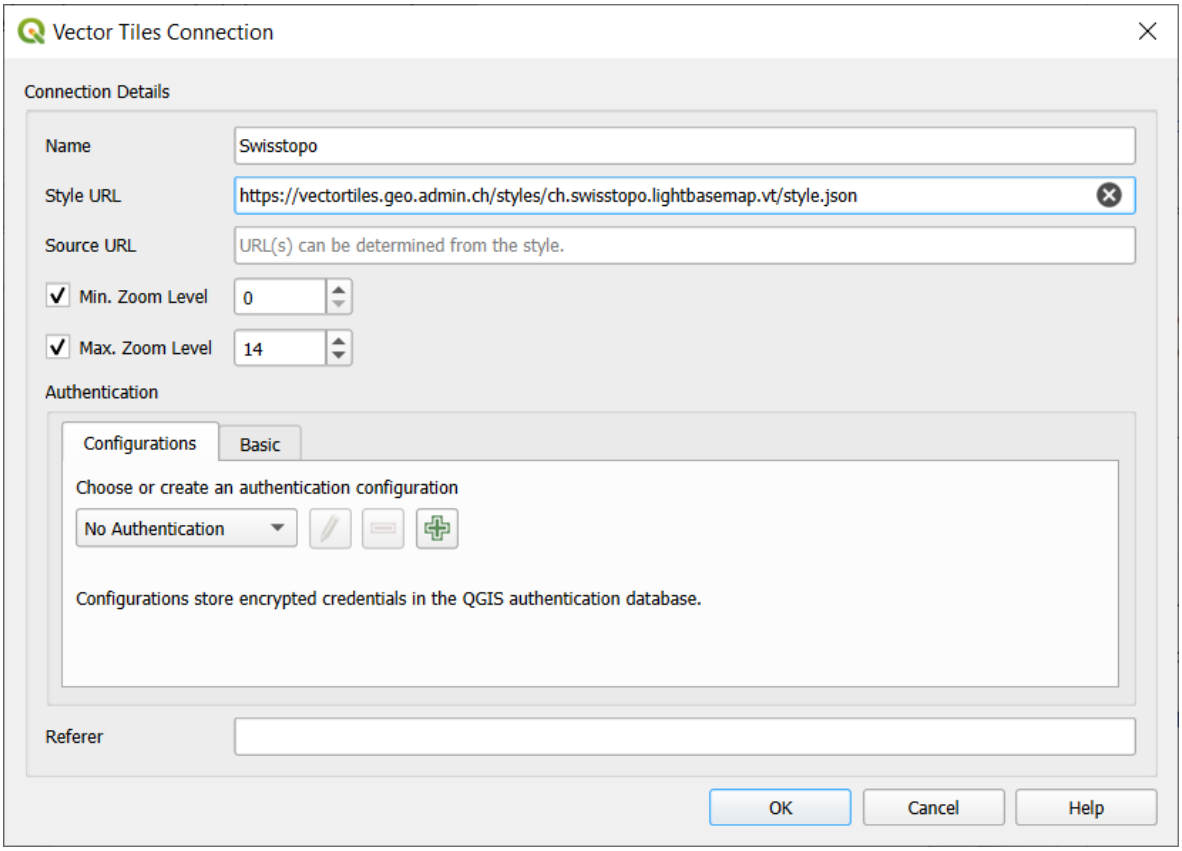
You set up a service by adding:

- a *Name*
- a *Style URL*: a URL to a MapBox GL JSON style configuration. If provided, then that style will be applied whenever the layers from the connection are added to QGIS. In the case of Arcgis vector tile service connections, the URL overrides the default style configuration specified in the server configuration.

  You can load vector tiles directly from a *Style URL*. The data source is automatically parsed from the style, and URLs with multiple sources are supported. That makes *Source URL* optional.
- the *Source URL*: of the type `http://example.com/{z}/{x}/{y}.pbf` for generic services and `http://example.com/arcgis/rest/services/Layer/VectorTileServer` for ArcGIS based services. The service must provide tiles in `.pbf` format.
- the  *Min. Zoom Level* and the  *Max. Zoom Level*. Vector Tiles have a pyramid structure. By using these options you have the opportunity to individually generate layers from the tile pyramid. These layers will then be used to render the Vector Tile in QGIS.

  For Mercator projection (used by OpenStreetMap Vector Tiles) Zoom Level 0 represents the whole world at a scale of 1:500.000.000. Zoom Level 14 represents the scale 1:35.000.
- the *authentication* configuration if necessary
- a *Referer*

Сл.11.18 shows the dialog with the Vector Tiles service configuration.



Слика11.18: Vector Tiles - Service configuration

Configurations can be saved to `.XML` file (*Save Connections*) through the *Vector Tiles* entry in *Data Source Manager* dialog or its context menu in the *Browser* panel. Likewise, they can be added from a file (*Load Connections*).

Once a connection to a vector tile service is set, it's possible to:

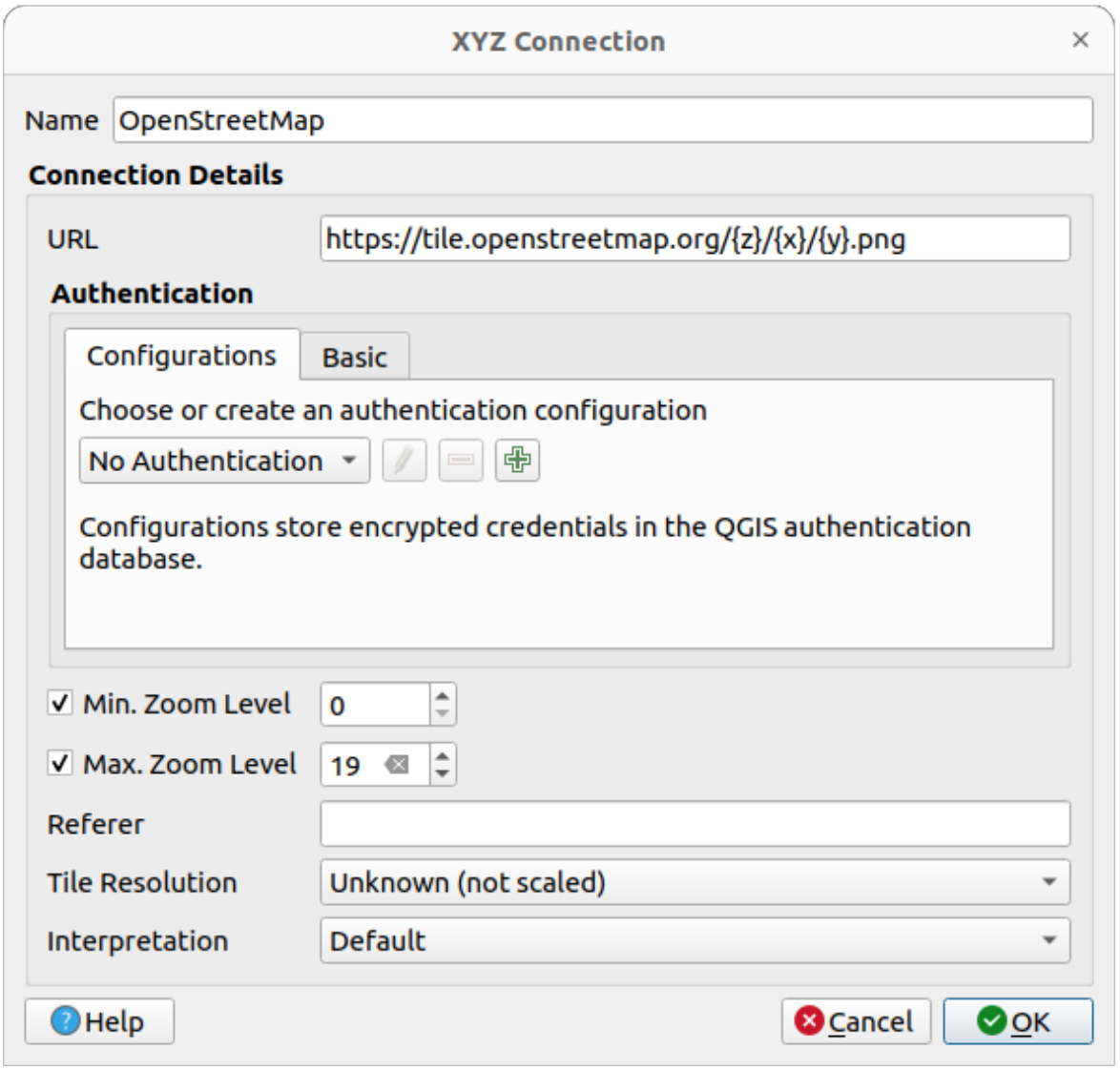
- *Edit* the vector tile connection settings
- *Remove* the connection
- From the *Browser* panel, right-click over the entry and you can also:
  - *Add layer to project*: a double-click also adds the layer
  - View the *Layer Properties…* and get access to metadata and a preview of the data provided by the service. More settings are available when the layer has been loaded into the project.

Using XYZ Tile services

XYZ Tile services can be added via the *XYZ* tab of the *Data Source Manager* dialog or the contextual menu of the *XYZ Tiles* entry in the *Browser* panel. By default, QGIS provides some default and ready-to-use XYZ Tiles services:

- *Mapzen Global Terrain*, allowing an immediate access to global DEM source for the projects. More details and resources at https://registry.opendata.aws/terrain-tiles/
- *OpenStreetMap* to access the world 2D map. Сл.11.19 shows the dialog with the OpenStreetMap XYZ Tile service configuration.

To add a new service, press *New* (respectively *New Connection* from the Browser panel) and provide:



Слика11.19: XYZ Tiles - OpenStreetMap configuration

- a *Name*
- the *URL*, you can add `http://example.com/{z}/{x}/{y}.png` or `file:///local_path/{z}/{x}/{y}.png`
- the *authentication* configuration if necessary
- the *Min. Zoom level* and *Max. Zoom level*
- a *Referer*
- the *Tile Resolution*: possible values are *Unknown (not scaled)*, *Standard (256x256 / 96DPI)* and *High (512x512 / 192DPI)*
- *Interpretation*: converts WMTS/XYZ raster datasets to a raster layer of single band float type following a predefined encoding scheme. Supported schemes are *Default* (no conversion is done), *MapTiler Terrain RGB* and *Terrarium Terrain RGB*. The selected converter will translate the RGB source values to float values for each pixel. Once loaded, the layer will be presented as a single band floating point raster layer, ready for styling using QGIS usual *raster renderers*.

Press *OK* to establish the connection. It will then be possible to:

- *Add* the new layer to the project; it is loaded with the name given in the settings.

- *Edit* the XYZ connection settings
- *Remove* the connection
- From the *Browser* panel, right-click over the entry and you can also:
  - *Export layer…* ► *To File*, *saving it as a raster*
  - *Add layer to project*: a double-click also adds the layer
  - View the *Layer Properties…* and get access to metadata and a preview of the data provided by the service. More settings are available when the layer has been loaded into the project.

Configurations can be saved to `.XML` file (*Save Connections*) through the *XYZ* entry in *Data Source Manager* dialog or its contextual menu in the *Browser* panel. Likewise, they can be added from a file (*Load Connections*).

The XML file for OpenStreetMap looks like this:

```
<!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>
```

Савет

Loading XYZ tiles without creating a connection

It is also possible to add XYZ tiles to a project without necessarily storing its connection settings in you user profile (e.g. for a dataset you may need once). In the *Data Source Manager* ► *XYZ* tab, edit any properties in the *Connection Details* group. The *Name* field above should turn into `Custom`. Press *Add* to load the layer in the project. It will be named by default `XYZ Layer`.

Examples of XYZ Tile services:

- OpenStreetMap Monochrome: *URL*: `http://tiles.wmflabs.org/bw-mapnik/{z}/{x}/{y}.png`, *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.
- Google Maps: *URL*: `https://mt1.google.com/vt/lyrs=m&x={x}&y={y}&z={z}`, *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.
- Open Weather Map Temperature: *URL*: `http://tile.openweathermap.org/map/temp_new/{z}/{x}/{y}.png?appid={api_key}` *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.

Using ArcGIS REST Servers

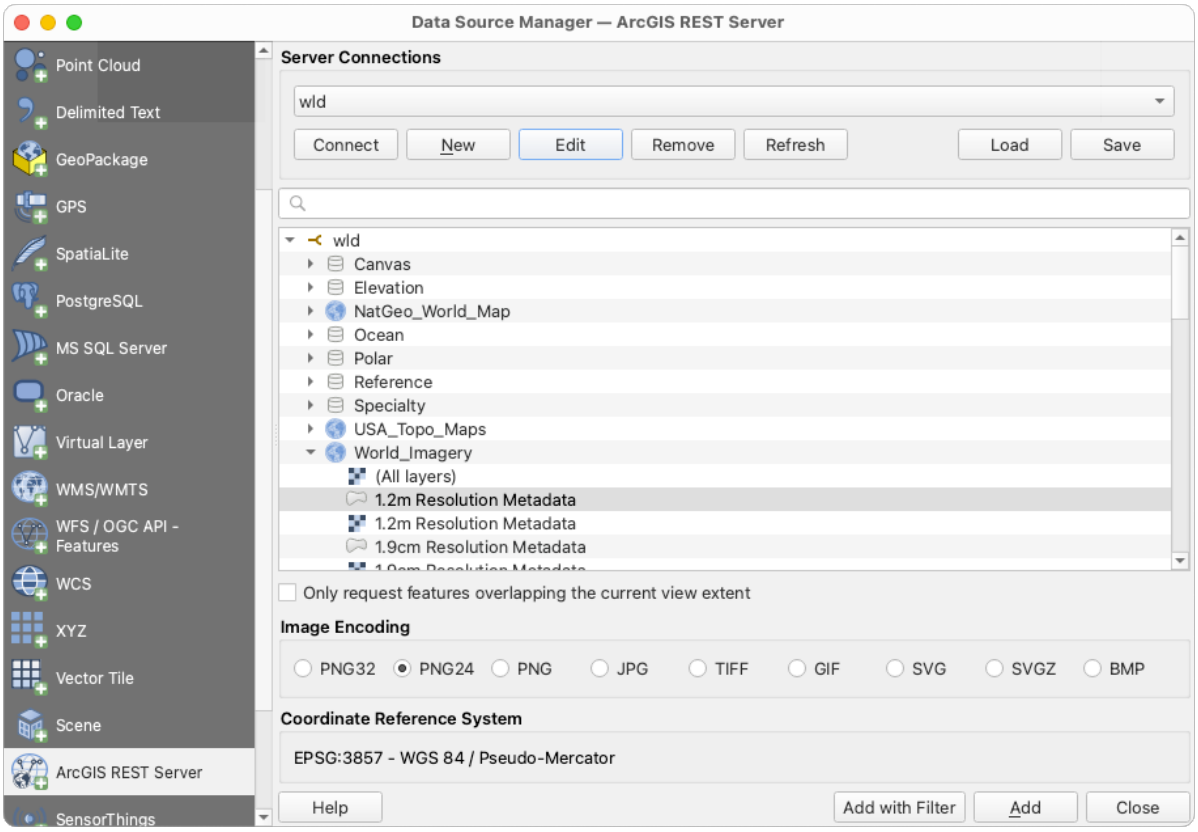
An ArcGIS REST Server can host many different types of web services (feature service, map service, image service, …). ArcGIS REST Servers can be added via the  *ArcGIS REST Server* tab of the *Data Source Manager* dialog or the contextual menu of the *ArcGIS REST Servers* entry in the *Browser* panel:

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.

> **Белешка**
>
> ArcGIS Feature Service connections which have their corresponding Portal endpoint URLS set can be explored by content groups in the browser panel.
>
> If a connection has the Portal endpoints set, then expanding out the connection in the browser will show a “Groups” and “Services” folder, instead of the full list of services usually shown. Expanding out the groups folder will show a list of all content groups that the user is a member of, each of which can be expanded to show the service items belonging to that group.

2. Then press *OK* to validate the configuration settings. These configurations can be saved to `.XML` file (*Save*). Likewise, they can be added from a file (*Load*).
3. Once a connection to an ArcGIS REST Server is set, it is possible to:
   - *Edit* the ArcGIS REST Server connection settings
   - *Remove* the connection
   - *Refresh* the connection
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.



Слика11.20: 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 advertised 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 *The Expression string builder* 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.

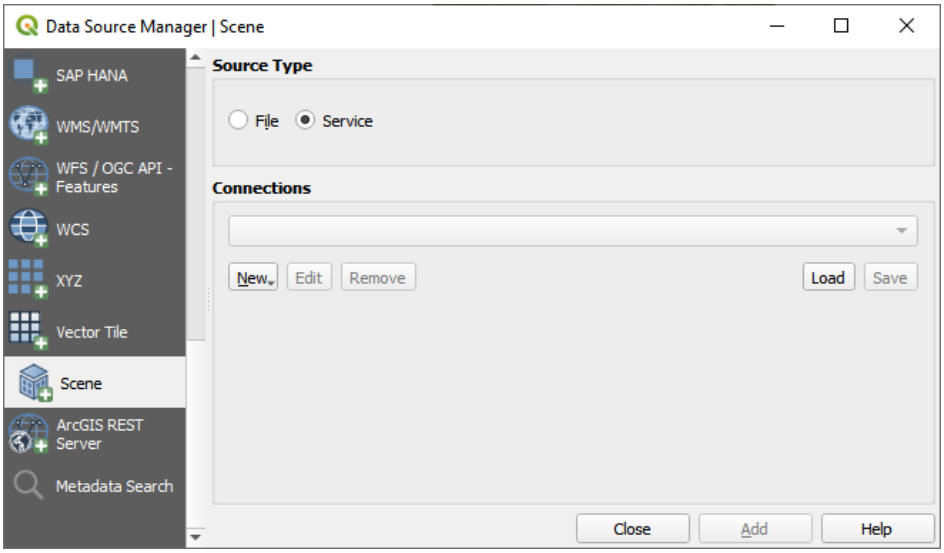
Most of the above tools to connect and access layers on an ArcGIS REST Server are also available from the *Browser* panel, within the contextual menu of the target node item. Depending on the node, you can also:

- *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.

Using 3D tiled scene services

QGIS supports multiple formats of 3D tiled datasets, grouped together as „tiled scenes”. These include Cesium 3D Tiles and Quantized Mesh tiles.

To load a tiled scene dataset into QGIS, use the *Scene* tab in the *Data Source Manager* dialog.



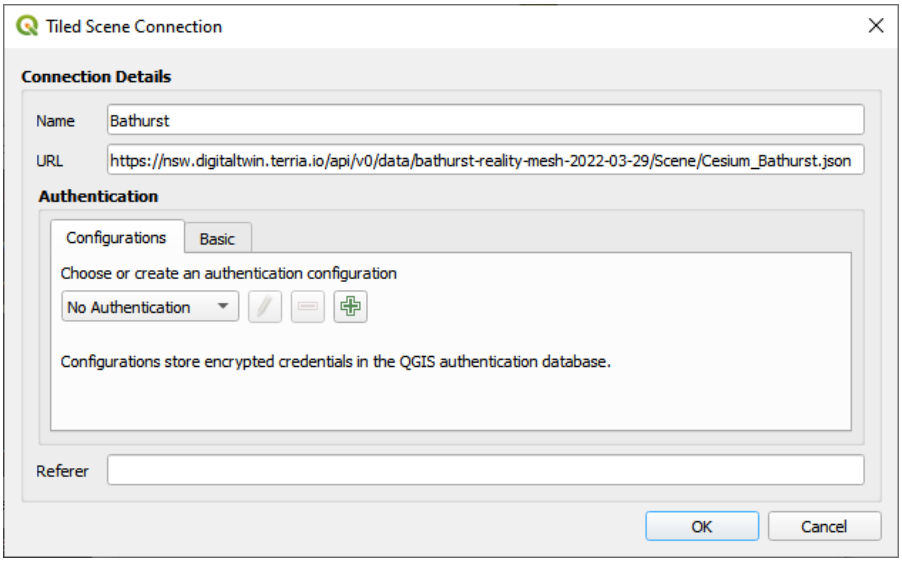
Слика11.21: Data Source Manager - Scene

Create a connection by clicking on *New*. You can add a *New Cesium 3D Tiles Connection* or a *New Quantized Mesh Connection*.

Choose a *Name* and set the *URL* to the URL of a layer description JSON file.

The URL may be remote (e.g. `http://example.com/tileset.json`) or local (e.g. `file:///path/to/tiles/tileset.json`).

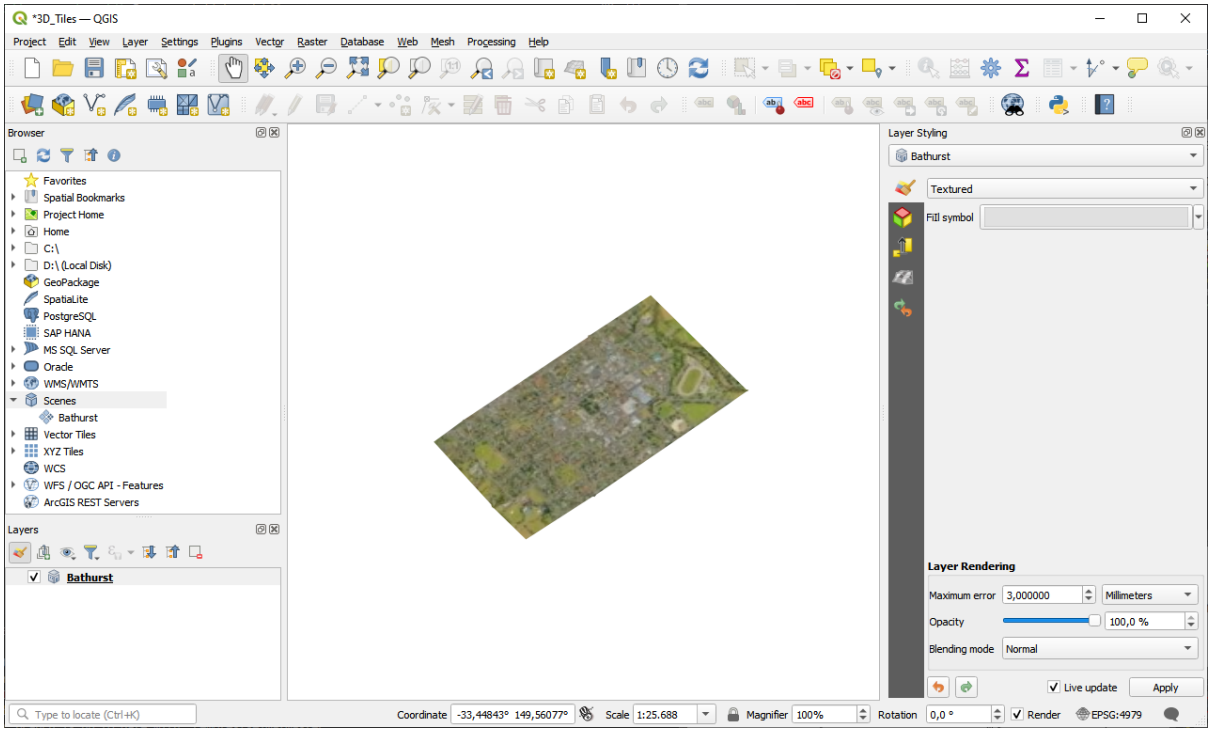
The *authentication* configuration if necessary.



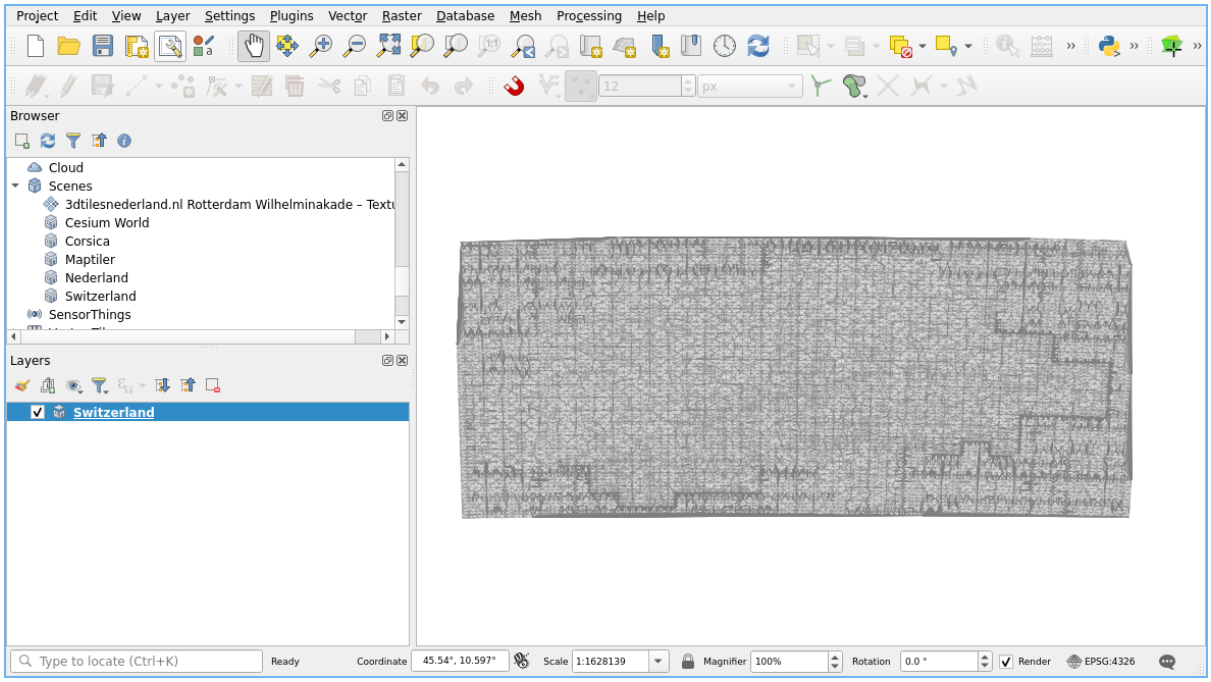
Слика11.22: Tiled Scene Connection

You can also add the service from *Browser Panel*.

After creating new connection you are able to *Add* the new layer to your map.



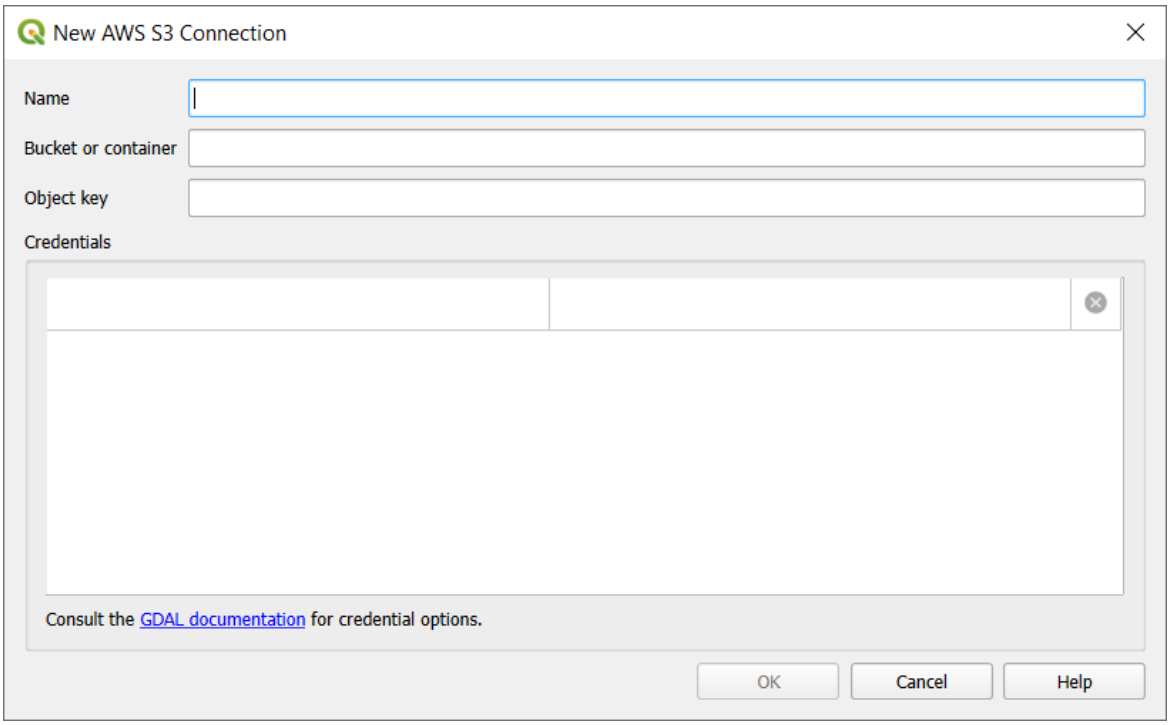
Слика11.23: 3D Tiles Layer - Textured



Слика11.24: Quantized Mesh layer

Using Cloud Connections

QGIS supports connections to cloud services like Alibaba Cloud OSS, Amazon S3, Google Cloud Storage, Microsoft Azure Blob Storage, Microsoft Azure Data Lake Storage, and OpenStack Swift Object Storage. You can load vector and raster data from these services into QGIS. Set up a new *Cloud* connection in the *Browser* panel by right--clicking on the *Cloud* entry and selecting *New Connection*. You will see a drop-down list of available cloud services. Select the service you want to connect to and fill in the required fields:



Слика11.25: Cloud Connection Dialog

- *Name*: A name for the connection.
- *Bucket or Container*: The name of the bucket or container in the cloud service.
- *Object Key* (optional): The key of the object in the bucket or container.
- *Credentials*: The credentials to access the cloud service.

You can also choose to *Save Connection* to an XML file or *Load Connection* from an XML file.

11.2 Creating Layers

Layers can be created in many ways, including:

- empty layers from scratch
- layers from existing layers
- layers from the clipboard
- layers as a result of an SQL-like query based on one or many layers (*virtual layers*)

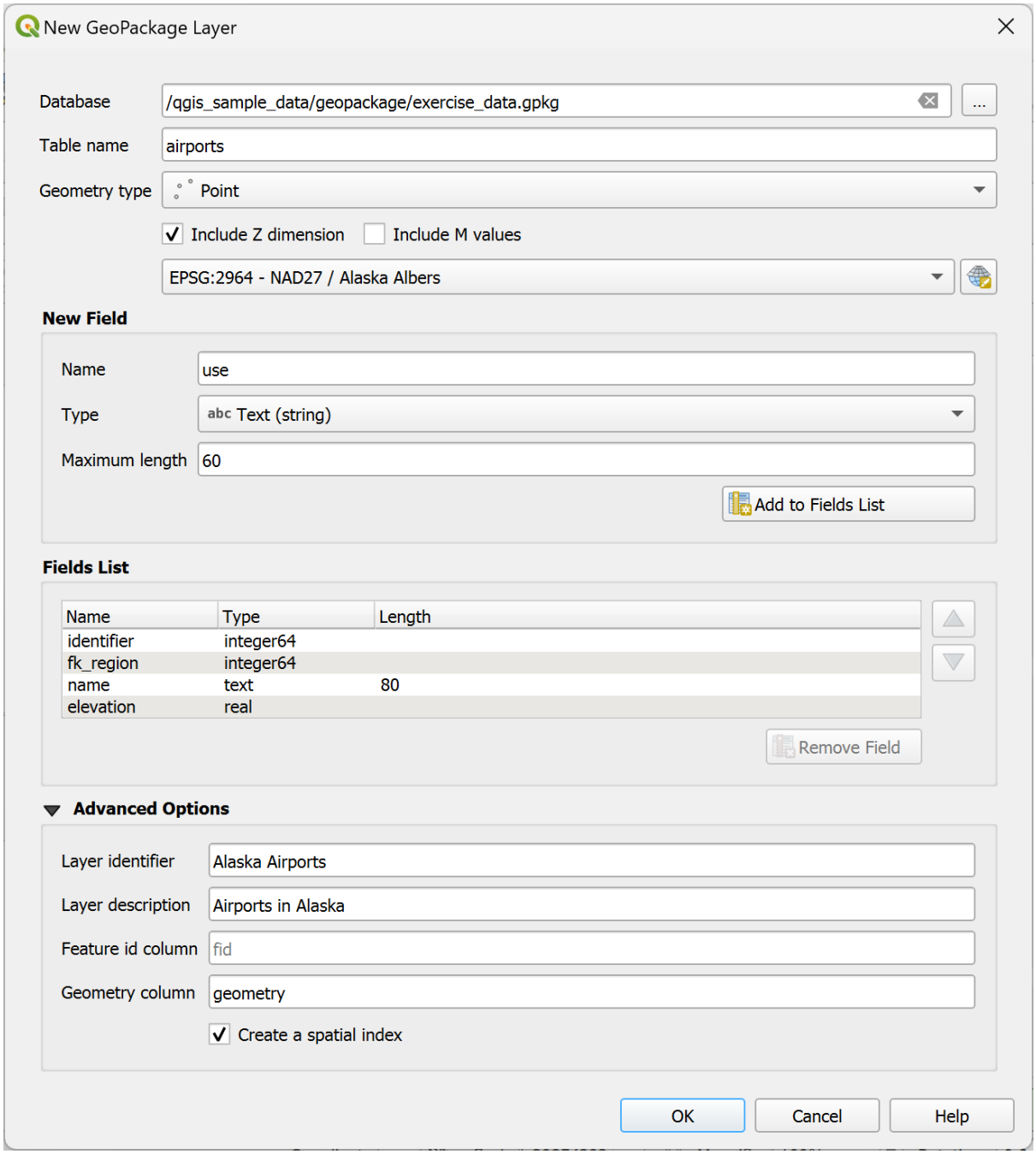
QGIS also provides tools to import/export from/to different formats.

11.2.1 Creating new vector layers

QGIS allows you to create new layers in different formats. It provides tools for creating GeoPackage, Shapefile, SpatiaLite, GPX format and Temporary Scratch layers (aka memory layers). Creation of a *new GRASS layer* is supported within the GRASS plugin.

Creating a new GeoPackage layer

To create a new GeoPackage layer, press the  *New GeoPackage Layer…* button in the *Layer* ► *Create Layer* ► menu or from the *Data Source Manager* toolbar. You can also create a new GeoPackage layer through the *Browser Panel* by selecting the *Create Database and Layer…*. The *New GeoPackage Layer* dialog will be displayed as shown in Сл.11.26.



Слика11.26: Creating a New GeoPackage layer dialog

1. The first step is to indicate the database file location. This can be done by pressing the … button to the right of the *Database* field and select an existing GeoPackage file or create a new one. QGIS will automatically add the right extension to the name you provide.
2. Give the new layer / table a name (*Table name*)
3. Define the *Geometry type*. If not a geometryless layer, you can specify whether it should *Include Z dimension* and/or *Include M values*.
4. Specify the coordinate reference system using the button

To add fields to the layer you are creating:

1. Enter the *Name* of the field

2. Select the data *Type*. Supported types are *Text data*, *Whole number* (both integer and integer64), *Decimal number*, *Date* and *Date and time*, *Binary (BLOB)* and *Boolean*.
3. Depending on the selected data format, enter the *Maximum length* of values.
4. Click on the  *Add to Fields List* button
5. Reproduce the steps above for each field you need to add
6. You can later change the fields order using the  Move Up and  Move Down buttons
7. Once you are happy with the attributes, click *OK*. QGIS will add the new layer to the legend, and you can edit it as described in section *Digitizing an existing layer*.

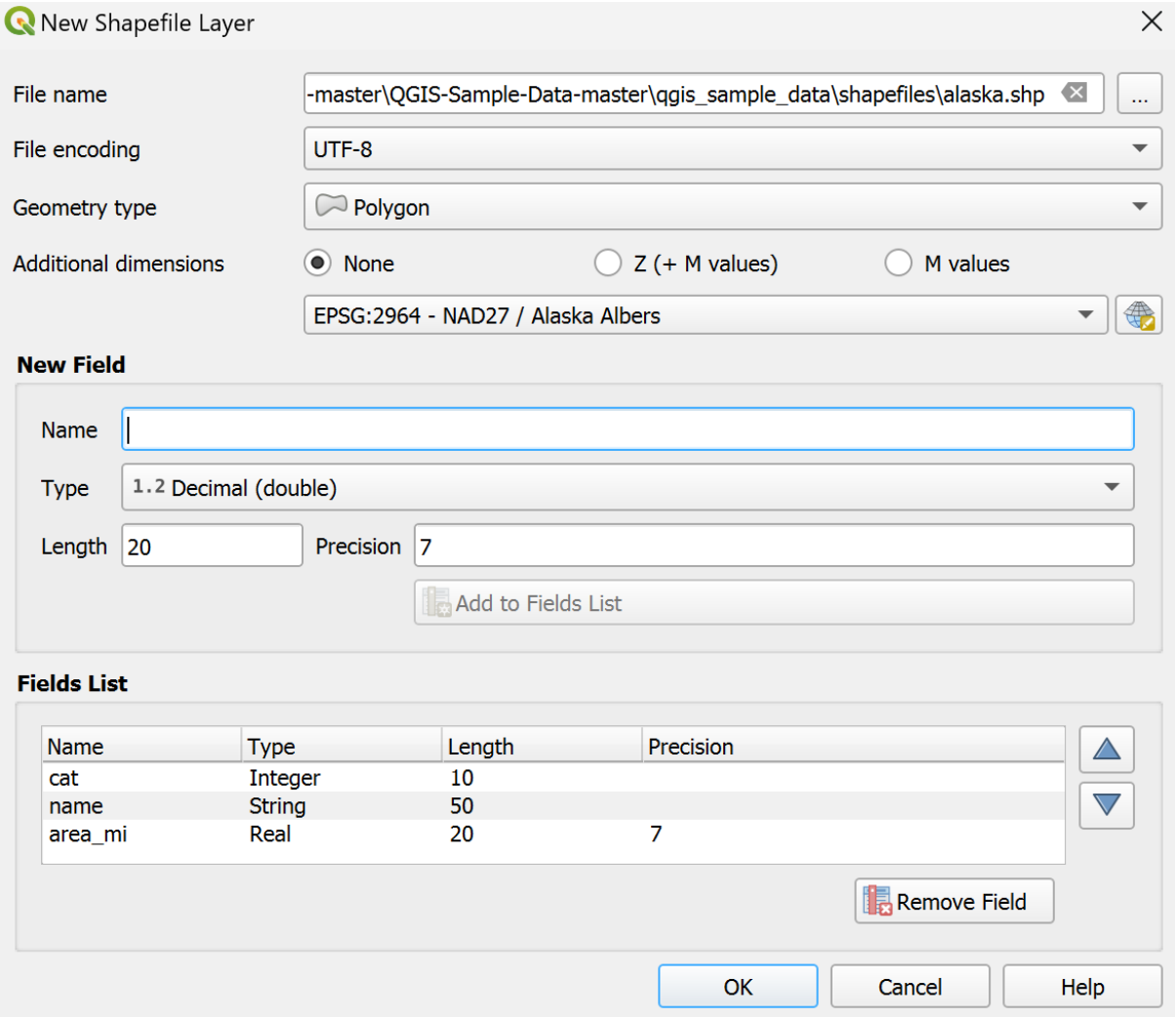
By default, when creating a GeoPackage layer, QGIS generates a *Feature id column* called `fid` which acts as the primary key of the layer. The name can be changed. The geometry field, if available, is named `geometry`, and you can choose to *Create a spatial index* on it. These options can be found under the *Advanced Options* together with the *Layer identifier* (short human readable name of the layer) and the *Layer description*.

Further management of GeoPackage layers can be done with the *DB Manager*.

Creating a new Shapefile layer

To create a new ESRI Shapefile format layer, press the  *New Shapefile Layer…* button in the *Layer* ► *Create Layer* ► menu or from the *Data Source Manager* toolbar. The *New Shapefile Layer* dialog will be displayed as shown in Сл.11.27.

1. Provide a path and file name using the … button next to *File name*. QGIS will automatically add the right extension to the name you provide.
2. Next, indicate the *File encoding* of the data
3. Choose the *Geometry type* of the layer: No Geometry (resulting in a `.DBF` format file), point, multipoint, line or polygon
4. Specify whether the geometry should have additional dimensions: *None*, *Z (+ M values)* or *M values*
5. Specify the coordinate reference system using the  button



Слика11.27: Creating a new Shapefile layer dialog

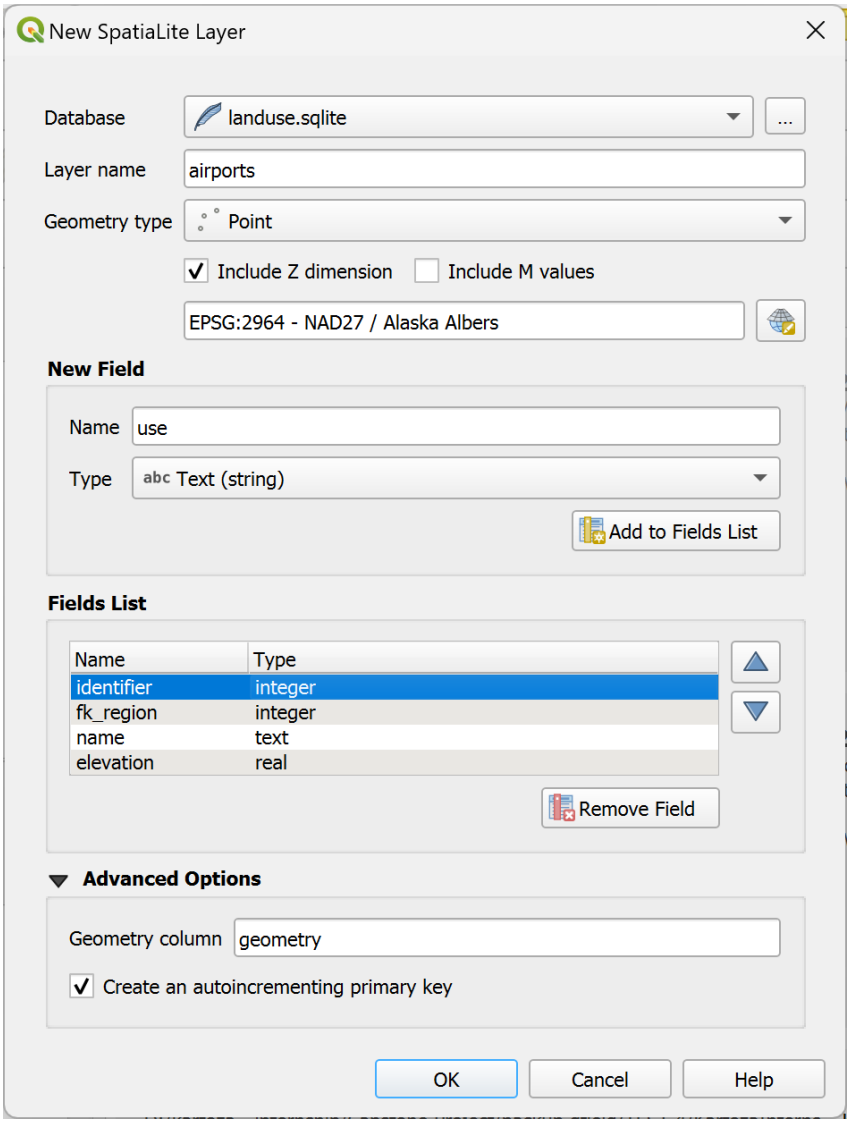
To add fields to the layer you are creating:

1. Enter the *Name* of the field
2. Select the data *Type*. Only *Decimal number*, *Whole number*, *Text data*, *Date*, and *Boolean* data types attributes are supported.
3. Depending on the selected data format, enter the *Length* and *Precision*.
4. Click on the *Add to Fields List* button
5. Reproduce the steps above for each field you need to add
6. You can later change the fields order using the Move Up and Move Down buttons
7. Once you are happy with the attributes, click *OK*. QGIS will add the new layer to the legend, and you can edit it as described in section *Digitizing an existing layer*.

By default, a first integer `id` column is added but can be removed.

Creating a new SpatiaLite layer

To create a new SpatiaLite layer, press the *New SpatiaLite Layer…* button in the *Layer* ► *Create Layer* ► menu or from the *Data Source Manager* toolbar. The *New SpatiaLite Layer* dialog will be displayed as shown in Сл.11.28.



Слика11.28: Creating a New SpatiaLite layer dialog

1. The first step is to indicate the database file location. This can be done by pressing the … button to the right of the *Database* field and select an existing SpatiaLite file or create a new one. QGIS will automatically add the right extension to the name you provide.
2. Provide a name (*Layer name*) for the new layer
3. Define the *Geometry type*. If not a geometryless layer, you can specify whether it should *Include Z dimension* and/or *Include M values*.
4. Specify the coordinate reference system using the button.

To add fields to the layer you are creating:

1. Enter the *Name* of the field
2. Select the data *Type*. Supported types are *Text data*, *Whole number*, *Decimal number*, *Date* and *Date time*.
3. Click on the *Add to Fields List* button

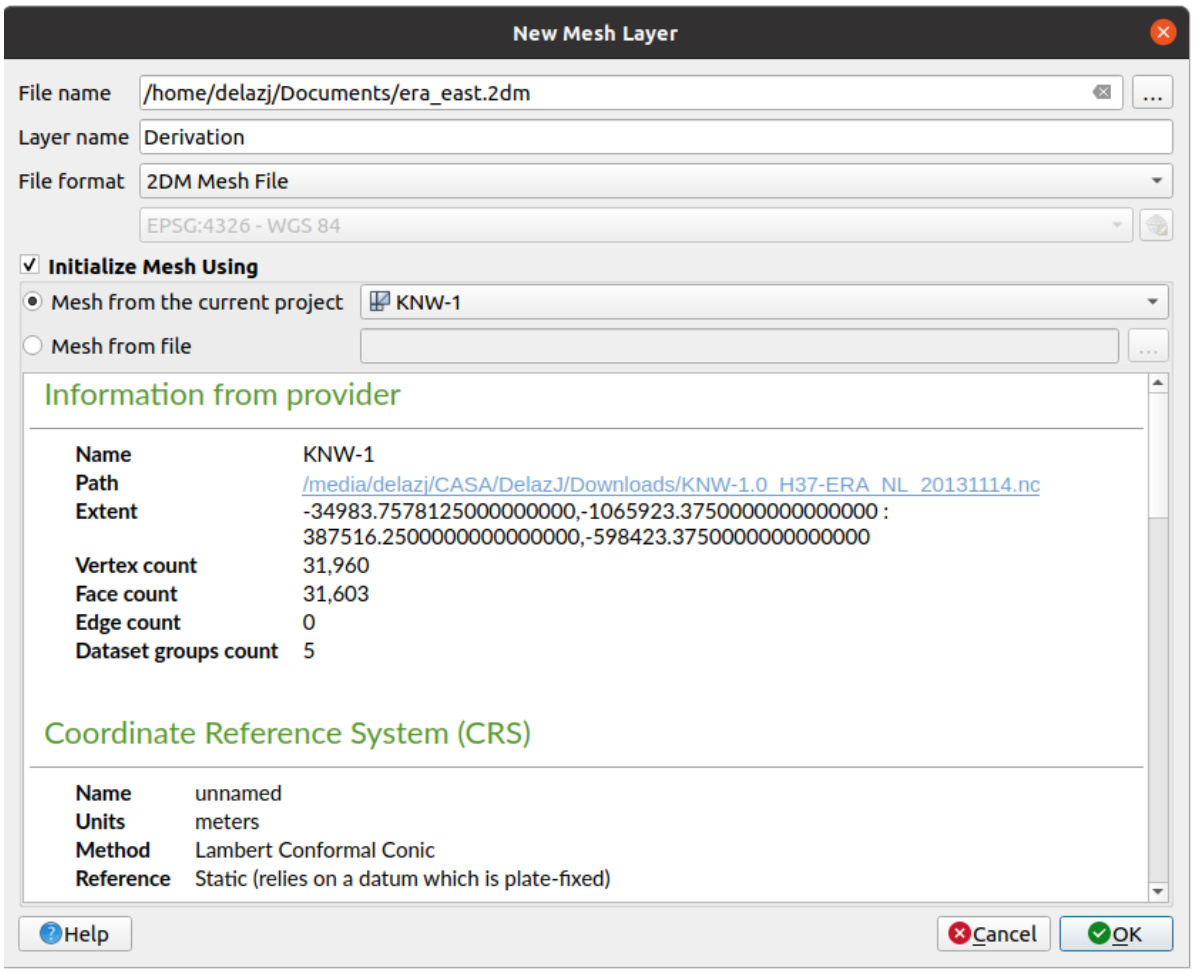
4. Reproduce the steps above for each field you need to add
5. You can later change the fields order using the Move Up and Move Down buttons
6. Once you are happy with the attributes, click *OK*. QGIS will add the new layer to the legend, and you can edit it as described in section *Digitizing an existing layer*.

If desired, you can select *Create an autoincrementing primary key* under the *Advanced Options* section. You can also rename the *Geometry column* (`geometry` by default).

Further management of SpatiaLite layers can be done with *DB Manager*.

Creating a new Mesh layer

To create a new Mesh layer, press the *New Mesh Layer…* button in the *Layer* ► *Create Layer* ► menu or from the *Data Source Manager* toolbar. The *New Mesh Layer* dialog will be displayed as shown in Сл.11.29.



Слика11.29: Creating a New Mesh layer dialog

1. The first step is to indicate the mesh file location. This can be done by pressing the … button to the right of the *File name* field and select an existing mesh file or create a new one.
2. Provide a name (*Layer name*), i.e. the name the layer is displayed with in the *Layers* panel
3. Select the *File format*: currently supported mesh file formats are `2DM Mesh File (*.2dm)`, `Selafin File (*.slf)` and `UGRID (*.nc)`.
4. Indicate the *Coordinate Reference System* to assign to the dataset

5. The above steps will generate an empty layer that you can afterwards digitize vertices and add dataset groups to. It's however also possible to initialize the layer with an existing mesh layer, i.e. populate the new layer with vertices or faces from the other. To do so:

    1. Check *Initialize Mesh using*
    2. and select either a *Mesh from the current project* or *Mesh from a file*. Information on the selected mesh file are displayed for checkup.

    Note that only the frame of the mesh layer is transferred to the new layer; their datasets are not copied.

Creating a new GPX layer

To create a new GPX file:

1. Select *Create Layer* ► *New GPX Layer…* from the *Layer* menu.
2. In the dialog, choose where to save the new file, name it and press *Save*.
3. Three new layers are added to the *Layers Panel*:
    - a point layer to digitize locations (`waypoints`) with fields storing the name, elevation, comment, description, source, url and url name
    - a line layer to digitize sequences of locations that make up a planned route (`routes`) with fields storing the name, symbol, number, comment, description, source, url, url name
    - and a line layer to track the receiver's movement over time (`tracks`) with fields storing the name, symbol, number, comment, description, source, url, url name.
4. You can now edit any of them as described in section *Digitizing an existing layer*.

Creating a new Temporary Scratch Layer

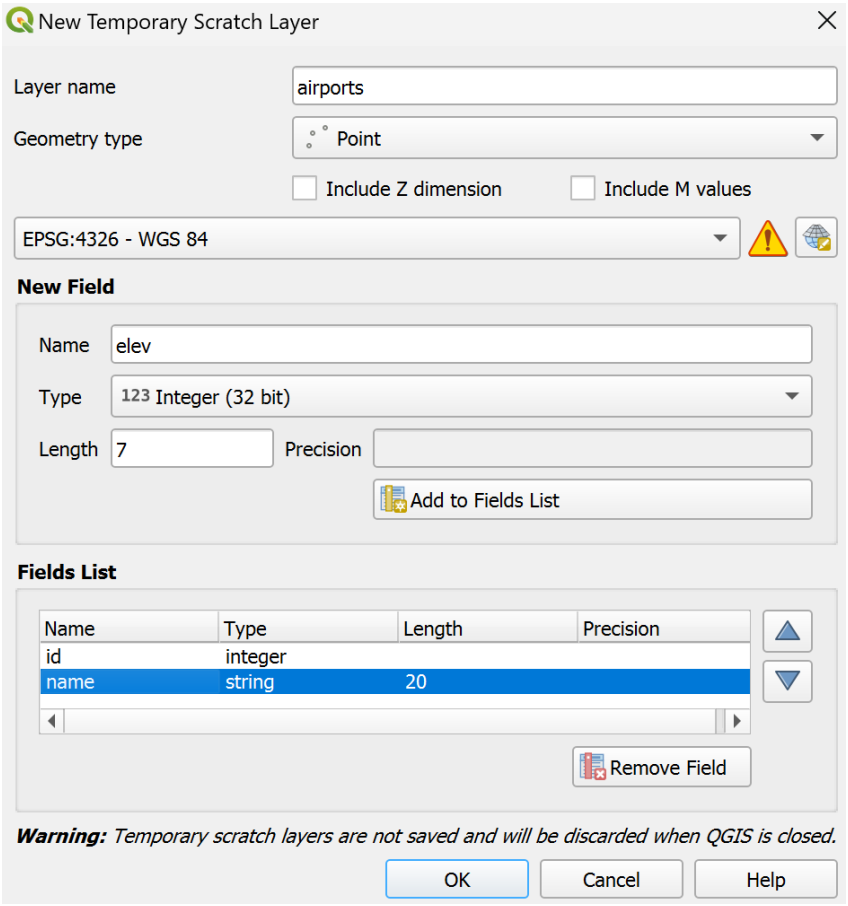
Temporary Scratch Layers are in-memory layers, meaning that they are not saved on disk and will be discarded when QGIS is closed. They can be handy for storing features you temporarily need or as intermediate layers during geoprocessing operations.

To create a new Temporary Scratch layer, choose the *New Temporary Scratch Layer…* entry in the *Layer* ► *Create Layer* ► menu or in the *Data Source Manager* toolbar. The *New Temporary Scratch Layer* dialog will be displayed as shown in Сл.11.30. Then:

1. Provide the *Layer name*
2. Select the *Geometry type*. Here you can create a:
    - `No geometry` type layer, served as simple table,
    - `Point` or `MultiPoint` layer,
    - `LineString/CompoundCurve` or `MultiLineString/MultiCurve` layer,
    - `Polygon/CurvePolygon` or `MultiPolygon/MultiSurface` layer.
3. For geometric types, specify the dimensions of the dataset: check whether it should *Include Z dimension* and/or *Include M values*
4. Specify the coordinate reference system using the button.
5. Add fields to the layer. Note that unlike many formats, temporary layers can be created without any fields. This step is thus optional.
    1. Enter the *Name* of the field
    2. Select the data *Type*: *Text*, *Whole number*, *Decimal number*, *Boolean*, *Date*, *Time*, *Date & Time* and *Binary (BLOB)* are supported.
    3. Depending on the selected data format, enter the *Length* and *Precision*

4. Click on the *Add to Fields List* button
5. Repeat the steps above for each field you need to add
6. You can later change the fields order using the Move Up and Move Down buttons

6. Once you are happy with the settings, click *OK*. QGIS will add the new layer to the *Layers* panel, and you can edit it as described in section *Digitizing an existing layer*.



Слика11.30: Creating a new Temporary Scratch layer dialog

You can also create prepopulated temporary scratch layers using e.g. the clipboard (see *Creating new layers from the clipboard*) or as a result of a *Processing algorithm*.

Савет

Permanently store a memory layer on disk

To avoid data loss when closing a project with temporary scratch layers, you can save these layers to any vector format supported by QGIS:

- clicking the indicator icon next to the layer;
- selecting the *Make permanent* entry in the layer contextual menu;
- using the *Export* ► entry from the contextual menu or the *Layer* ► *Save As…* menu.

Each of these commands opens the *Save Vector Layer as* dialog described in the *Creating new layers from an existing layer* section and the saved file replaces the temporary one in the *Layers* panel.

11.2.2 Creating new layers from an existing layer

Layers (raster, vector and point cloud) can be saved in a different format and/or reprojected to a different coordinate reference system (CRS) using the *Layer* ► *Save As...* menu or right-clicking on the layer in the *Layers panel* and selecting:

- *Export* ► *Save As...* for raster and point cloud layers
- *Export* ► *Save Features As...* or *Export* ► *Save Selected Features As...* for vector layers.
- 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*.

Common parameters

The *Save Layer as...* dialog shows several parameters to change the behavior when saving the layer. Among the common parameters for raster and vector are:

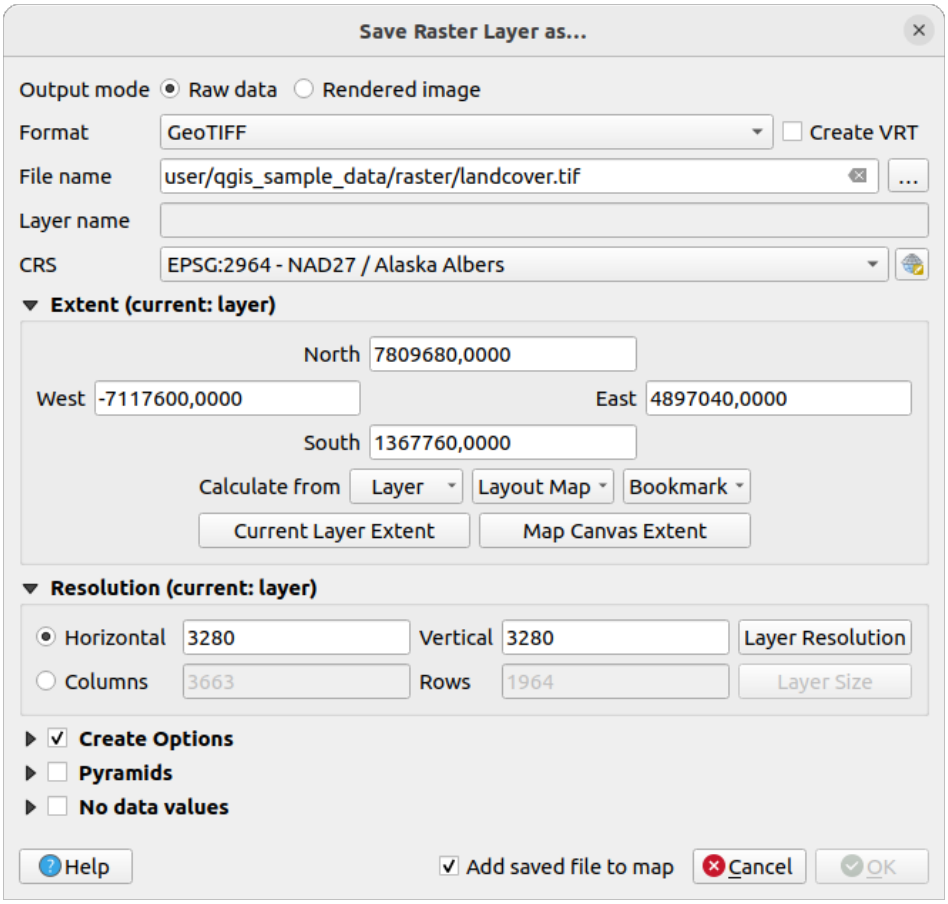
- *File name*: the location of the file on the disk. It can refer to the output layer or to a container that stores the layer (for example database-like formats such as GeoPackage, SpatiaLite or Open Document Spreadsheets).
- *CRS*: can be changed to reproject the data
- *Extent*: restricts the extent of the input that is to be exported using the *extent_selector* widget
- *Add saved file to map*: to add the new layer to the canvas

However, some parameters are specific to certain formats:

Raster specific parameters

Depending on the format of export, some of these options may not be available:

- *Output mode* (it can be **raw data** or **rendered image**)
- *Format*: exports to any raster format GDAL can write to, such as GeoTiff, GeoPackage, MBTiles, Geospatial PDF, SAGA GIS Binary Grid, Intergraph Raster, ESRI .hdr Labelled…
- *Resolution*
- *Create Options*: use advanced options (file compression, block sizes, colorimetry…) when generating files, either from the *predefined create profiles* related to the output format or by setting each parameter.
- *Pyramids* creation
- *VRT Tiles* in case you opted to  *Create VRT*
- *No data values*



Слика11.31: Saving as a new raster layer

Vector specific parameters

Depending on the format of export, some of these options may be available:

- *Format*: exports to any vector format GDAL can write to, such as GeoPackage, GML, ESRI Shapefile, AutoCAD DXF, ESRI FileGDB, Mapinfo TAB or MIF, SpatiaLite, CSV, KML, ODS, …
- *Layer name*: available when the *File name* refers to a container-like format, this entry represents the output layer.
- *Encoding*
- *Save only selected features*
- *Select fields to export and their export options*: provides means to export fields with custom names and *form widget* settings:
  - Check rows under the *Name* column to choose fields to keep in the output layer, or press *Select All* or *Deselect All* buttons
  - Toggle the *Use aliases for exported name* checkbox to populate the *Export name* column with corresponding field aliases or reset to the original field name. Double-clicking a cell will also edit the name.
  - Depending on whether attribute form custom widgets are in use, you can *Replace all selected raw field values by displayed values*. E.g. if a `value map` widget is applied to a field, the output layer will contain the description values instead of the original values. The replacement can also be done on a field by field basis, in the *Replace with displayed values* column.
- *Persist layer metadata*: ensures that any layer *metadata* present in the source layer will be copied and stored:
  - in the newly created layer, if the output is of GeoPackage format

  - as a `.qmd` file along with the output layer, for other formats. Note that file-based formats supporting more than one dataset (e.g. SpatiaLite, DXF,…) may have unintended behavior.

- *Symbology export*: can be used mainly for DXF export and for all file formats who manage OGR feature styles (see note below) as DXF, KML, tab file formats:
  - **No symbology**: default style of the application that reads the data
  - **Feature symbology**: save style with OGR Feature Styles (see note below)
  - **Symbol Layer symbology**: save with OGR Feature Styles (see note below) but export the same geometry multiple times if there are multiple symbology symbol layers used
  - A **Scale** value can be applied to the latest options

Белешка

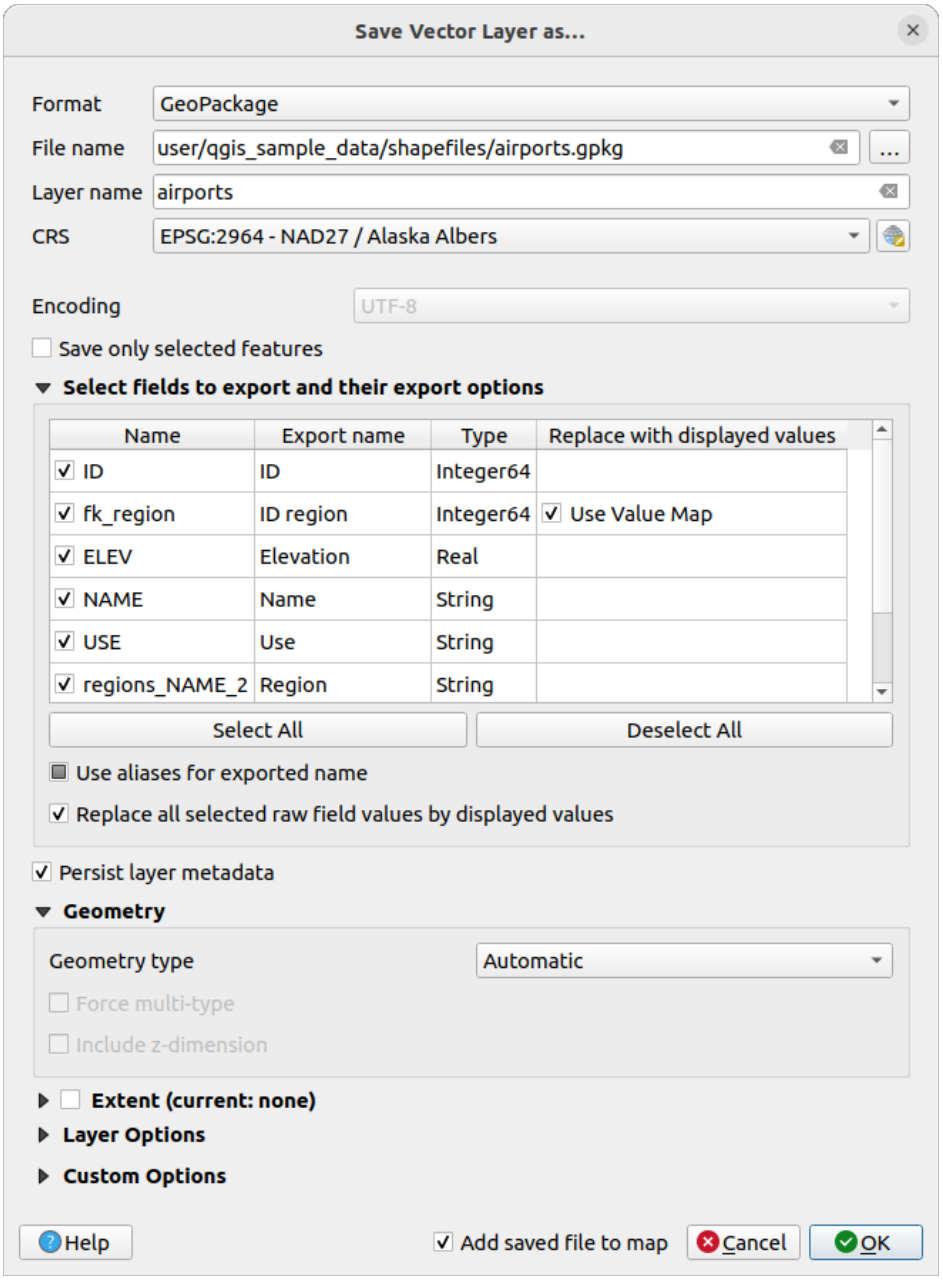
OGR Feature Styles are a way to store style directly in the data as a hidden attribute. Only some formats can handle this kind of information. KML, DXF and TAB file formats are such formats. For advanced details, you can read the OGR Feature Styles specification document.

- *Geometry*: you can configure the geometry capabilities of the output layer
  - *geometry type*: keeps the original geometry of the features when set to **Automatic**, otherwise removes or overrides it with any type. You can add an empty geometry column to an attribute table and remove the geometry column of a spatial layer.
  - *Force multi-type*: forces creation of multi-geometry features in the layer.
  - *Include z-dimension* to geometries.

Савет

Overriding layer geometry type makes it possible to do things like save a geometryless table (e.g. `.csv` file) into a shapefile WITH any type of geometry (point, line, polygon), so that geometries can then be manually added to rows with the  Add Part tool.

- *Datasource Options*, *Layer Options* or *Custom Options* which allow you to configure advanced parameters depending on the output format. Some are described in *Exploring Data Formats and Fields* but for full details, see the GDAL driver documentation. Each file format has its own custom parameters, e.g. for the `GeoJSON` format have a look at the GDAL GeoJSON documentation.



Слика11.32: Saving as a new vector layer

When saving a vector layer into an existing file, depending on the capabilities of the output format (Geopackage, SpatiaLite, FileGDB…), the user can decide whether to:

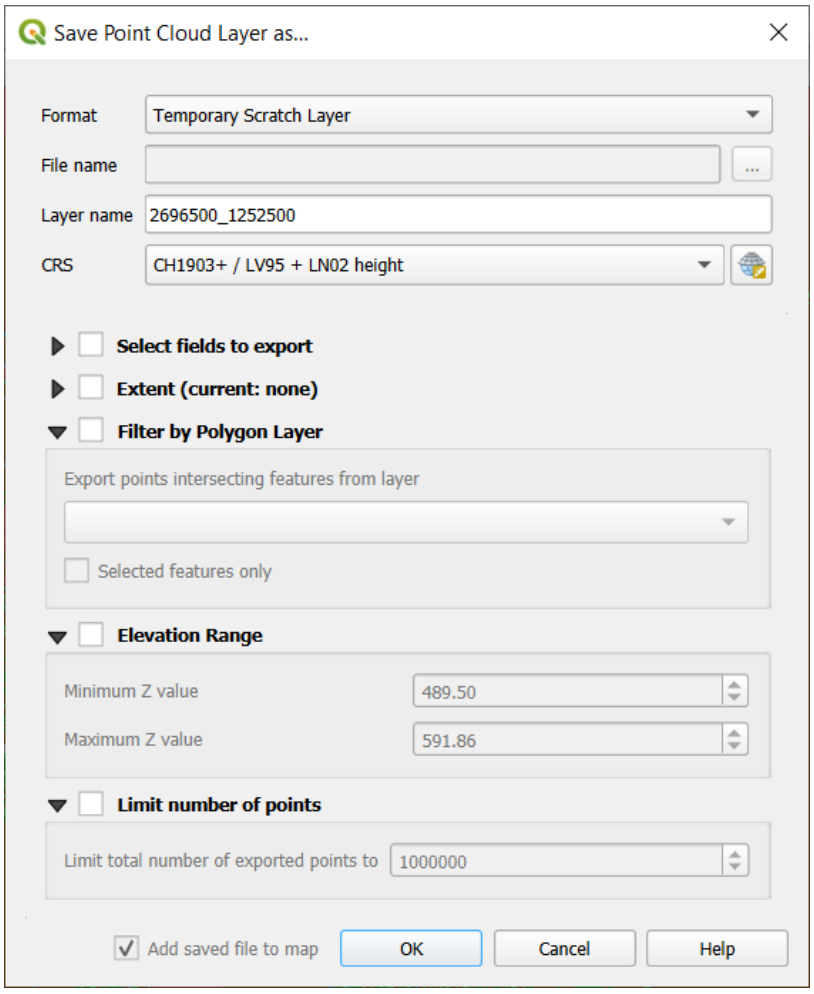
- overwrite the whole file
- overwrite only the target layer (the layer name is configurable)
- append features to the existing target layer
- append features, add new fields if there are any.

For formats like ESRI Shapefile, MapInfo .tab, feature append is also available.

Point Cloud specific parameters

Similar to raster and vector layers, point cloud layers can be saved in a different format and/or reprojected to a different coordinate reference system (CRS). This allows you to export a point cloud layer to vector or point cloud formats. Current supported formats are: Temporary scratch (memory layer), GeoPackage, ESRI Shapefile, DXF and LAS/LAZ point cloud. In addition to the common parameters listed above, exporting point cloud layers includes the following options:

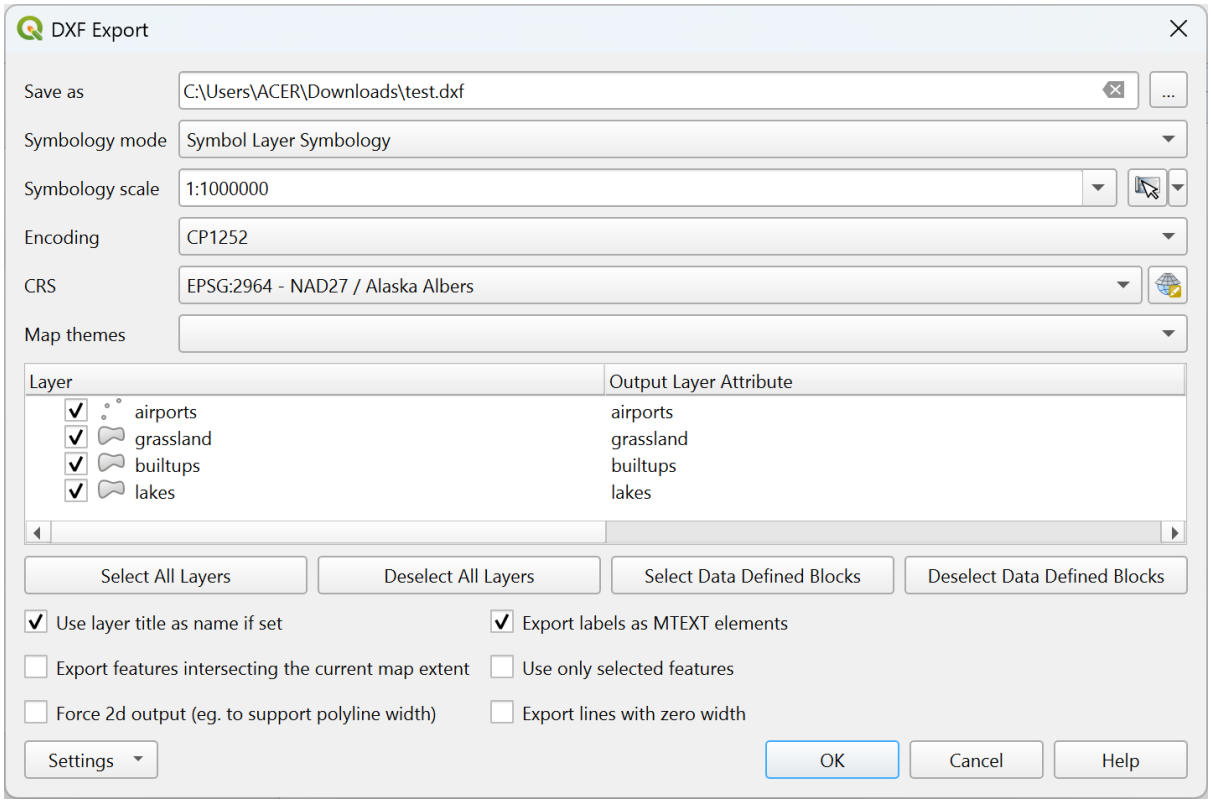
- *Filter by Polygon Layer*: Allows you to filter the point cloud data based on a polygon layer.
- *Elevation Range*: Enables filtering of the point cloud data based on a specified Z range.
- *Limit number of points*: Provides an option to limit the number of points exported from the point cloud layer.



Слика11.33: Saving a point cloud layer as a new layer

11.2.3 Creating new DXF files

Besides the *Save As...* dialog which provides options to export a single layer to another format, including `*.DXF`, QGIS provides another tool to export multiple layers as a single DXF layer. It's accessible in the *Project* ► *Import/Export* ► *Export Project to DXF...* menu.



Слика11.34: Exporting a project to DXF dialog

In the *DXF Export* dialog:

1. Provide the destination file.
2. Choose the symbology mode and scale (see the *OGR Feature Styles* note), if applicable.
3. Select the data *Encoding*.
4. Select the *CRS* to apply: the selected layers will be reprojected to the given CRS.
5. Select the layers to include in the DXF files either by checking them in the table widget or automatically picking them from an existing *map theme*. The *Select All* and *Deselect All* buttons can help to quickly set the data to export.

   For each layer, you can:

   - Override the output layer name without altering the original project layer. For this, click on the *Layer* name in the dialog and write the output name to use.
   - *Output layer attribute*: Choose whether to export all the features in a single DXF layer or rely on a field whose values are used to split the features into layers in the DXF output. In the latter case, each layer will take its name from the corresponding field value.
   - *Allow data defined symbol blocks*:
   - *Maximum number of symbol blocks*: creates symbol blocks up to the specified limit, starting with the ones containing the highest number of references. The other symbols are written as they are. `-1` means no limitation.

Optionally, you can also choose to:

- ☐ *Use the layer title as name if set* instead of the layer name itself: the title is taken from the *metadata* or *server* properties of the layer;
- ☐ *Export features intersecting the current map extent*;

- ☐ *Force 2d output (eg. to support polyline width)*;
- ☑ *Export label as MTEXT elements* or TEXT elements;
- ☐ *Use only selected features*;
- ☐ *Export lines with zero width*: all the lines are exported with minimal width `0` (hairline) if enabled. This helps keep the lines minimal in the file regardless of the zoom level, and can be handy for doing further CAD-editing with the exported dxf, especially if there are many features next to each other on the map.

Белешка

The precedence for defining the output layer name is as follows:

1. The field value from *Output layer attribute*
2. The overridden name in *Layer* column
3. The *Use the layer title as name if set* option
4. The layer name

Current settings defined in the *DXF Export* dialog may be stored in an XML file for reusing them in other sessions. For this, the *Settings* combo box has two options: *Load Settings from File…* and *Save Settings to File…*.

11.2.4 Creating new layers from the clipboard

Features that are on the clipboard can be pasted into a new layer. To do this, Select some features, copy them to the clipboard, and then paste them into a new layer using *Edit* ► *Paste Features as* ► and choosing:

- *New Vector Layer…*: the *Save vector layer as…* dialog appears (see *Creating new layers from an existing layer* for parameters)
- or *Temporary Scratch Layer…*: you need to provide a name for the layer

A new layer, filled with selected features and their attributes is created (and added to map canvas).

Белешка

Creating layers from the clipboard is possible with features selected and copied within QGIS as well as features from another application, as long as their geometries are defined using well-known text (WKT).

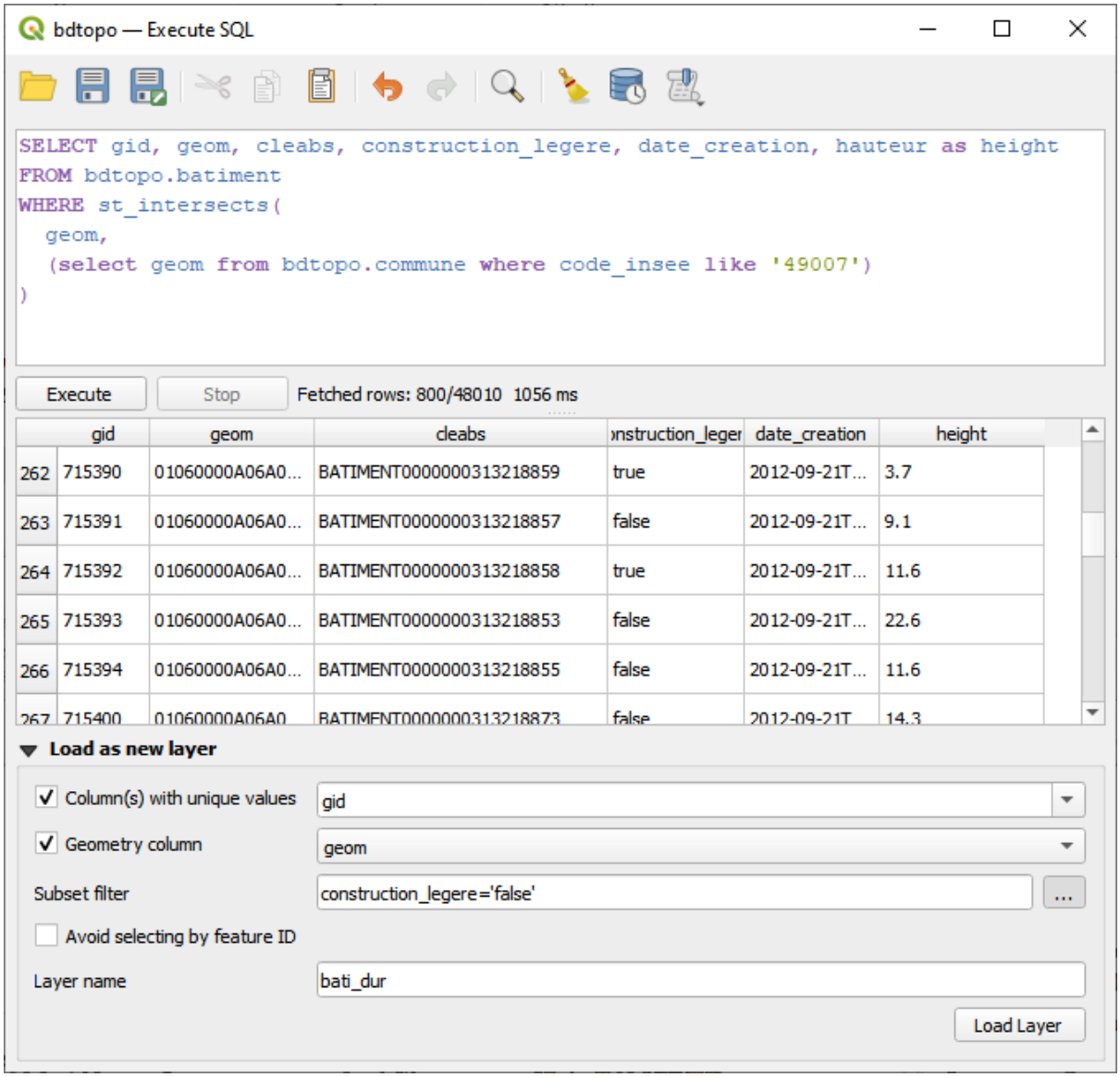
11.2.5 Creating SQL Query Layers

Beside loading an entire layer in a project or creating new layers from scratch or pasted features, you can also load layers generated on the fly from other layer(s). They are the result of a more or less advanced filter using SQL language, applied to stored layers regardless of their data provider or their availability in the active project. This means that temporary layers are not compatible with this feature. Depending on the provider, one or more layers can be used to write the query. The generated layer remains dependent on the layer(s) involved in the query and is loaded with the  Filter icon next to it.

This feature is accessible:

- From the *Browser* panel, right-click on a supported data (plain layer, database connection, schema or table) and select *Execute SQL…* entry in the contextual menu.
- From the *Layers* panel, select a loaded layer, right-click and select *Execute SQL…* entry in the contextual menu.

This opens a window with a central text box widget where you can write SQL queries.



Слика11.35: 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.

Савет

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.

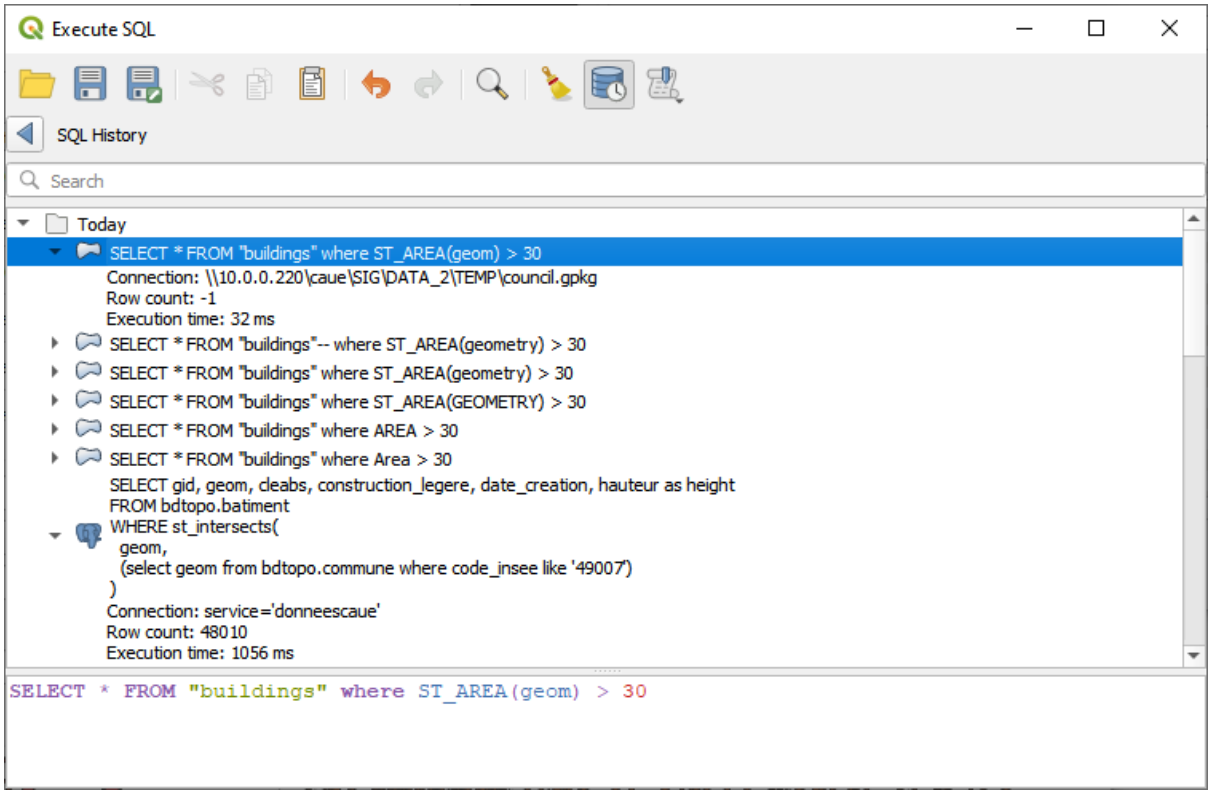
The returned table can be loaded in QGIS expanding the *Load as new layer* group and configuring parameters (their availability depends on the layer provider):

- *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.



Слика11.36: 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 Creating virtual layers

A virtual layer is a special kind of vector layer. It allows you to define a layer as the result of an SQL query involving any number of other vector layers that QGIS is able to open. Virtual layers do not carry data by themselves and can be seen as views.

To create a virtual layer, open the virtual layer creation dialog by:

- choosing the  *Add/Edit Virtual Layer* entry in the *Layer* ► *Add Layer* ► menu;

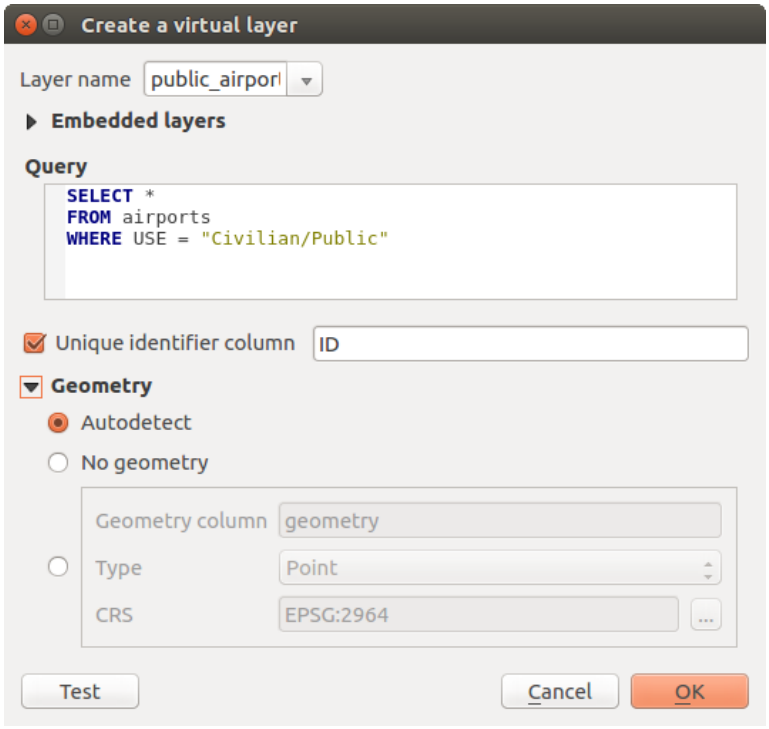
- enabling the  *Add Virtual Layer* tab in the *Data Source Manager* dialog;
- or using the *DB Manager* dialog tree.

The dialog allows you to specify a *Layer name* and an SQL *Query*. The query can use the name (or id) of loaded vector layers as tables, as well as their field names as columns.

For example, if you have a layer called `airports`, you can create a new virtual layer called `public_airports` with an SQL query like:

```
SELECT *
FROM airports
WHERE USE = "Civilian/Public"
```

The SQL query will be executed, regardless of the underlying provider of the `airports` layer, even if this provider does not directly support SQL queries.



Слика11.37: Create virtual layers dialog

Joins and complex queries can also be created, for example, to join airports and country information:

```
SELECT airports.*, country.population
FROM airports
JOIN country
ON airports.country = country.name
```

Белешка

It's also possible to create virtual layers using the SQL window of *DB Manager Plugin*.

Embedding layers for use in queries

Besides the vector layers available in the map canvas, the user can add layers to the *Embedded layers* list, which can be used in queries without the need to have them showing in the map canvas or Layers panel.

To embed a layer, click *Add* and provide the *Local name*, *Provider*, *Encoding* and the path to the *Source*.

The *Import* button allows adding layers in the map canvas into the Embedded layers list. Those layers can then be removed from the Layers panel without breaking existent queries.

Supported query language

The underlying engine uses SQLite and SpatiaLite to operate.

It means you can use all of the SQL your local installation of SQLite understands.

Functions from SQLite and spatial functions from SpatiaLite can also be used in a virtual layer query. For instance, creating a point layer out of an attribute-only layer can be done with a query similar to:

```
SELECT id, MakePoint(x, y, 4326) as geometry
FROM coordinates
```

Functions of QGIS expressions can also be used in a virtual layer query.

To refer the geometry column of a layer, use the name `geometry`.

Contrary to a pure SQL query, all the fields of a virtual layer query must be named. Don't forget to use the `as` keyword to name your columns if they are the result of a computation or a function call.

Performance issues

With default parameters, the virtual layer engine will try its best to detect the type of the different columns of the query, including the type of the geometry column if one is present.

This is done by introspecting the query when possible or by fetching the first row of the query (LIMIT 1) as a last resort. Fetching the first row of the result just to create the layer may be undesirable for performance reasons.

The creation dialog parameters:

- *Unique identifier column*: specifies a field of the query that represents unique integer values that QGIS can use as row identifiers. By default, an autoincrementing integer value is used. Defining a unique identifier column speeds up the selection of rows by id.
- *No geometry*: forces the virtual layer to ignore any geometry field. The resulting layer is an attribute-only layer.
- Geometry *Column*: specifies the name of the geometry column.
- Geometry *Type*: specifies the type of the geometry.
- Geometry *CRS*: specifies the coordinate reference system of the virtual layer.

Special comments

The virtual layer engine tries to determine the type of each column of the query. If it fails, the first row of the query is fetched to determine column types.

The type of a particular column can be specified directly in the query by using some special comments.

The syntax is the following: `/*:type*/`. It has to be placed just after the name of a column. `type` can be either `int` for integers, `real` for floating point numbers or `text`.

For instance:

```
SELECT id+1 as nid /*:int*/
FROM table
```

The type and coordinate reference system of the geometry column can also be set thanks to special comments with the following syntax `/*:gtype:srid*/` where `gtype` is the geometry type (`point`, `linestring`, `polygon`, `multipoint`, `multilinestring` or `multipolygon`) and `srid` an integer representing the EPSG code of a coordinate reference system.

Use of indexes

When requesting a layer through a virtual layer, the source layer indices will be used in the following ways:

- if an = predicate is used on the primary key column of the layer, the underlying data provider will be asked for a particular id (FilterFid)
- for any other predicates (>, <=, !=, etc.) or on a column without a primary key, a request built from an expression will be used to request the underlying vector data provider. It means indexes may be used on database providers if they exist.

A specific syntax exists to handle spatial predicates in requests and triggers the use of a spatial index: a hidden column named `_search_frame_` exists for each virtual layer. This column can be compared for equality to a bounding box. Example:

```
SELECT *
FROM vtab
WHERE _search_frame_=BuildMbr(-2.10,49.38,-1.3,49.99,4326)
```

Spatial binary predicates like `ST_Intersects` are sped up significantly when used in conjunction with this spatial index syntax.

11.3 Georeferencer

 Georeferencer je alat za poravnavanje nereferenciranih rasterskih ili vektorskih slojeva sa poznatim koordinatnim sistemima pomoću kontrolnih tačaka na terenu (GCP). Podržava izvoz transformisanih rastera (GeoTIFF) i vektora (shapefile, geopackage itd.) ili pisanje pratećih world datoteka (samo za raster). Osnovni pristup georeferenciranju sloja jeste pronalaženje tačaka na njemu za koje možete tačno odrediti koordinate.

Mogućnosti

Табела11.1: Alati Georeferencer-a

Ikona	Svrha	Ikona	Svrha
	Otvori raster		Otvori vektor
	Pokreni georeferenciranje		Generiši GDAL skriptu
	Učitaj GCP tačke		Sačuvaj GCP tačke kao
	Podešavanja transformacije		
	Dodaj GCP tačku		Obriši GCP tačku
	Pomeri GCP tačku		Tip lepljenja
	Prikaži dok za napredno digitalizovanje		Pomeri
	Uveličaj		Umanji
	Uveličaj na sloj		Uveličaj poslednje
	Uveličaj sledeće		Poveži Georeferencer sa QGIS-om
	Poveži QGIS sa Georeferencer-om		Puno rastezanje histograma
	Lokalno rastezanje histograma		

11.3.1 Georeferenciranje rasterskog sloja

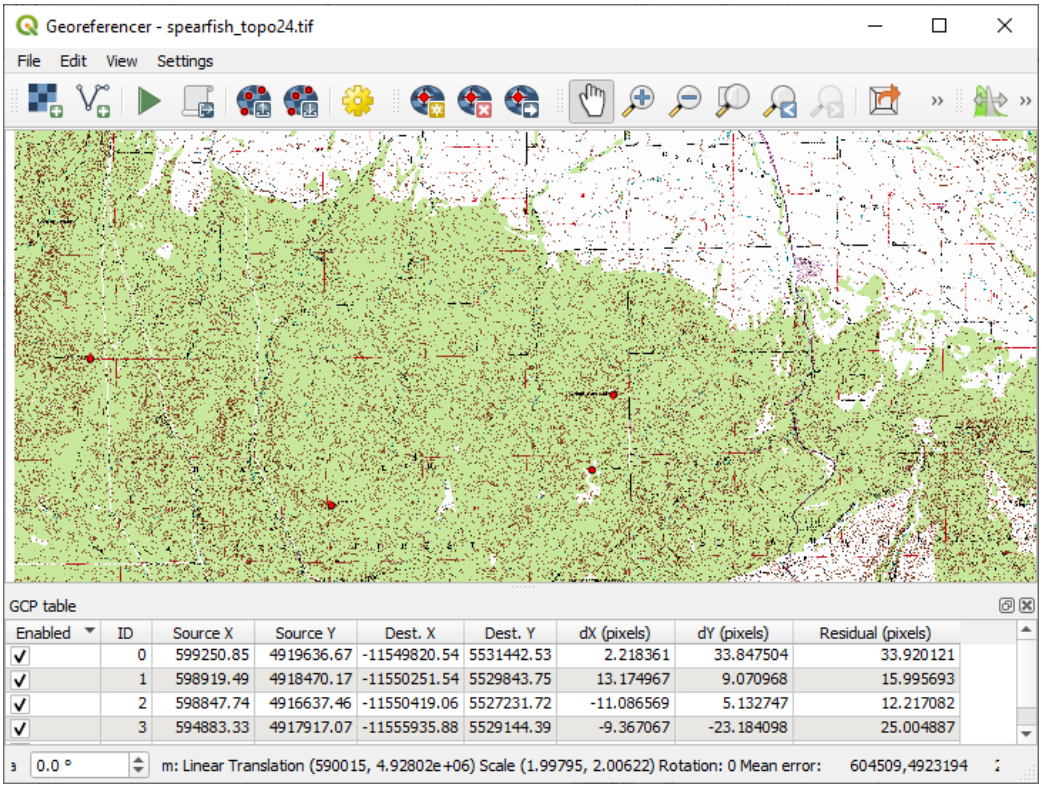
Kao X i Y koordinate (DMS (dd mm ss.ss), DD (dd.dd) ili projektovane koordinate (mmmm.mm)), koje odgovaraju izabranoj tački na slici, mogu se koristiti dva alternativna postupka:

- Sam raster ponekad pruža krstove sa koordinatama „upisanim“ na slici. U tom slučaju možete ručno uneti koordinate.
- Korišćenje već georeferenciranih slojeva. To mogu biti vektorski ili rasterski podaci koji sadrže iste objekte koje imate na slici koju želite da georeferencirate i sa projekcijom koju želite za svoju sliku. U tom slučaju možete uneti koordinate klikom na referentni skup podataka učitan u QGIS platno karte.

Uobičajen postupak za georeferenciranje slike podrazumeva izbor više tačaka na rasteru, zadavanje njihovih koordinata i izbor relevantnog tipa transformacije. Na osnovu ulaznih parametara i podataka, Georeferencer će izračunati parametre world datoteke. Što više koordinata pružite, to će rezultat biti bolji.

Prvi korak je pokretanje QGIS-a i klik na *Sloj* ►  *Georeferencer*, koji se pojavljuje u traci menija QGIS-a. Pojavljuje se dijalog Georeferencer kao što je prikazano na Сл.11.38.

Za ovaj primer koristimo topografski list Južne Dakote iz SDGS-a. Kasnije se može vizuelizovati zajedno sa podacima iz GRASS `spearfish60` lokacije. Topografski list možete preuzeti ovde: https://grass.osgeo.org/sampledata/spearfish_toposheet.tar.gz.



Слика11.38: Dijalog Georeferencer

Unos kontrolnih tačaka na terenu (GCP)

1. Da bismo počeli georeferenciranje nereferenciranog rastera, moramo ga učitati pomoću dugmeta . Raster će se pojaviti u glavnoj radnoj oblasti dijaloga. Kada je raster učitan, možemo početi da unosimo referentne tačke.

2. Pomoću dugmeta Dodaj GCP tačku dodajte tačke u glavnu radnu oblast i unesite njihove koordinate (pogledajte sliku Сл.11.39). Za ovaj postupak imate sledeće opcije:

   - Kliknite na tačku na rasterskoj slici i ručno unesite X i Y koordinate, zajedno sa CRS-om tačke.
   - Kliknite na tačku na rasterskoj slici i izaberite dugme Sa platna karte da biste dodali X i Y koordinate uz pomoć georeferencirane karte već učitane u QGIS platno karte. CRS će biti postavljen automatski.
   - Pri unosu GCP-ova sa glavnog platna karte, imate mogućnost da sakrijete prozor georeferencer-a dok birate tačke sa glavnog platna. Ako je polje *Automatski sakrij prozor georeferencer-a* označeno, nakon klika na *Sa platna karte*, glavni prozor georeferencer-a biće sakriven dok se ne doda tačka na platnu karte. Dijalog *Unesite koordinate karte* ostaće otvoren. Ako polje nije označeno, oba prozora ostaće otvorena dok birate tačku na platnu karte. Ova opcija deluje samo kada prozor georeferencer-a nije usidren u glavnom interfejsu.

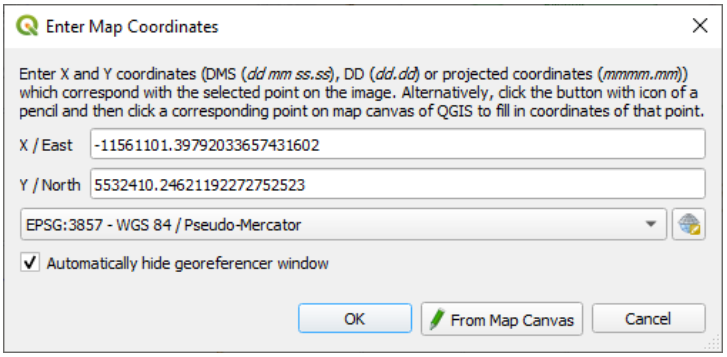
3. Nastavite sa unosom tačaka. Trebalo bi da imate bar četiri tačke, a što više koordinata možete pružiti, to će rezultat biti bolji. Postoje dodatne alatke za uveličavanje i pomeranje radne oblasti radi pronalaženja relevantnog skupa GCP tačaka.

Савет

Da biste izbegli stalno prebacivanje između dugmadi Pomeri, Dodaj GCP tačku i Pomeri GCP tačku, možete zgodno koristiti strelice na tastaturi za pomeranje i točkić miša za skaliranje georeferencirane

karte.

4. Nakon što pružite nekoliko tačaka, možete koristiti dugmad  Poveži QGIS sa Georeferencer-om i/ili  Poveži Georeferencer sa QGIS-om koja će prilagoditi, redom, obuhvat karte glavnog QGIS prozora trenutnom prikazu u Georeferencer-u i/ili obrnuto.
5. Alatkom  možete pomerati GCP-ove i na platnu i u prozoru za georeferenciranje, ako je potrebno da ih ispravite.

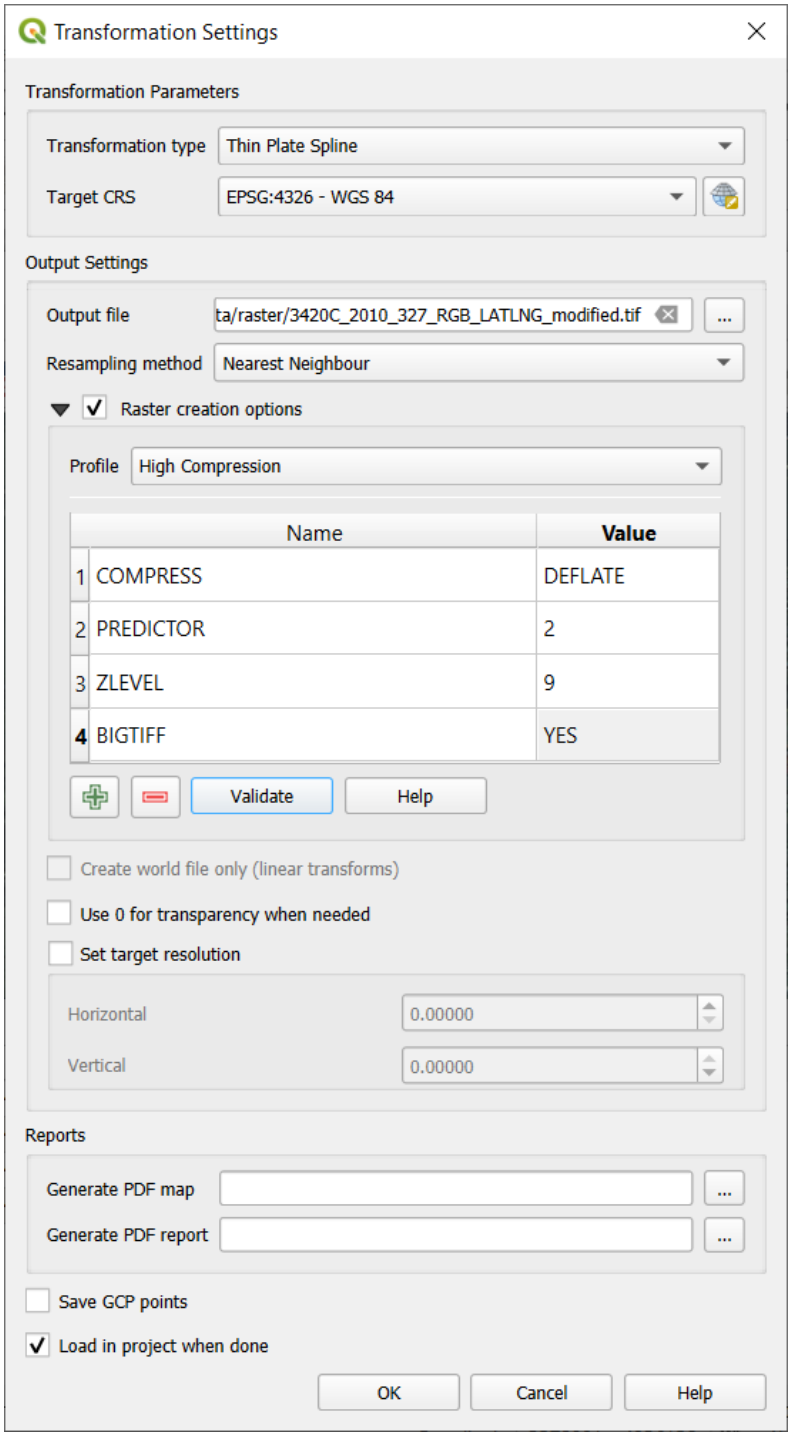


Слика11.39: Dodavanje tačaka na rastersku sliku

Tačke koje se dodaju na kartu čuvaju se u zasebnoj tekstualnoj datoteci (`[filename].points`) obično zajedno sa rasterskom slikom. To nam omogućava da ponovo otvorimo Georeferencer kasnije i dodamo nove tačke ili obrišemo postojeće radi optimizacije rezultata. Datoteka tačaka sadrži vrednosti oblika: `mapX, mapY, pixelX, pixelY`. Za upravljanje datotekama možete koristiti dugmad  Učitaj GCP tačke i  Sačuvaj GCP tačke kao.

Definisanje podešavanja transformacije

Nakon što dodate GCP-ove na rastersku sliku, potrebno je da definišete podešavanja transformacije za proces georeferenciranja.



Слика11.40: Definisanje podešavanja transformacije georeferencer-a

Dostupni algoritmi transformacije

Dostupan je niz algoritama transformacije, u zavisnosti od tipa i kvaliteta ulaznih podataka, prirode i količine geometrijske distorzije koju ste spremni da unesete u konačni rezultat i broja kontrolnih tačaka na terenu (GCP).

Trenutno su dostupni sledeći *Tipovi transformacije*:

- Algoritam **Linearni** koristi se za pravljenje world datoteke i razlikuje se od drugih algoritama jer zapravo ne transformiše piksele rastera. Omogućava pozicioniranje (translaciju) slike i jednoobrazno skaliranje, ali ne i rotaciju ili druge transformacije. Najpogodniji je ako je vaša slika rasterska karta dobrog kvaliteta, u poznatom CRS-u, ali joj samo nedostaju informacije o georeferenciranju. Potrebne su najmanje 2 GCP

tačke.

- Transformacija **Helmert** takođe omogućava rotaciju. Naročito je korisna ako je vaš raster lokalna karta dobrog kvaliteta ili ortorektifikovana vazdušna slika, ali nije poravnata sa smerom mreže u vašem CRS-u. Potrebne su najmanje 2 GCP tačke.
- Algoritam **Polinomni 1** omogućava opštiju afinu transformaciju, naročito i jednoobrazno smicanje. Prave linije ostaju prave (tj. kolinearne tačke ostaju kolinearne), a paralelne linije ostaju paralelne. Ovo je naročito korisno za georeferenciranje kartograma podataka, koji su možda iscrtani (ili podaci prikupljeni) sa različitim veličinama piksela na terenu u različitim pravcima. Potrebne su najmanje 3 GCP tačke.
- Algoritmi **Polinomni** 2-3 koriste opštije polinome 2. ili 3. stepena umesto samo afine transformacije. To im omogućava da uzmu u obzir zakrivljenost ili drugo sistematsko iskrivljenje slike, na primer fotografisane karte sa zakrivljenim ivicama. Potrebno je najmanje 6 (odnosno 10) GCP tačaka. Uglovi i lokalna razmera se ne čuvaju niti tretiraju jednoobrazno preko slike. Naročito, prave linije mogu postati zakrivljene, a na ivicama ili daleko od bilo kojih GCP-ova može doći do značajne distorzije usled predalekog ekstrapoliranja polinoma prilagođenih podacima.
- Algoritam **Projektivni** generalizuje Polinomni 1 na drugačiji način, omogućavajući transformacije koje predstavljaju centralnu projekciju između 2 neparalelne ravni, slike i platna karte. Prave linije ostaju prave, ali se paralelnost ne čuva, a razmera preko slike dosledno varira sa promenom perspektive. Ovaj tip transformacije najkorisniji je za georeferenciranje fotografija pod uglom (umesto ravnih skenova) karata dobrog kvaliteta, ili kosih vazdušnih slika. Potrebne su najmanje 4 GCP tačke.
- Najzad, algoritam **Thin Plate Spline** (TPS) „razvlači“ raster poput gume koristeći više lokalnih polinoma za podudaranje sa zadatim GCP-ovima, uz minimizovanu ukupnu zakrivljenost površine. Oblasti udaljene od GCP-ova biće pomerene u izlazu radi prilagođavanja podudaranju GCP-ova, ali će inače biti minimalno lokalno deformisane. TPS je najkorisniji za georeferenciranje oštećenih, deformisanih ili na drugi način blago netačnih karata, ili loše ortorektifikovanih vazdušnih snimaka. Koristan je i za približno georeferenciranje i implicitno reprojektovanje karata sa nepoznatim tipom ili parametrima projekcije, ali gde se pravilna mreža ili gust skup ad-hoc GCP-ova može podudariti sa referentnim slojem karte. Tehnički zahteva najmanje 10 GCP-ova, ali obično više da bi bio uspešan.

U svim algoritmima osim TPS-a, ako se zada više od minimalnog broja GCP-ova, parametri će biti prilagođeni tako da se minimizuje ukupna rezidualna greška. Ovo je korisno za minimizovanje uticaja grešaka registracije, tj. blagih nepreciznosti u klikovima pokazivača ili unetim koordinatama, ili drugih malih lokalnih deformacija slike. Bez drugih GCP-ova koji bi ih kompenzovali, takve greške ili deformacije mogle bi se pretvoriti u značajne distorzije, naročito blizu ivica georeferencirane slike. Međutim, ako se zada više od minimalnog broja GCP-ova, oni će se u izlazu podudarati samo približno. Nasuprot tome, TPS će se precizno podudariti sa svim zadatim GCP-ovima, ali može uneti značajne deformacije između bliskih GCP-ova sa greškama registracije.

Definisanje metoda preuzorkovanja

Tip preuzorkovanja koji izaberete verovatno će zavisiti od vaših ulaznih podataka i krajnjeg cilja vežbe. Ako ne želite da promenite statistiku rastera (osim onako kako to podrazumeva neuniformno geometrijsko skaliranje ako koristite nešto drugo osim Linearne, Helmert ili Polinomne 1 transformacije), možda ćete želeti da izaberete „Najbliži sused“. Nasuprot tome, „kubno preuzorkovanje“ će, na primer, obično dati vizuelno glađi rezultat.

Moguće je birati između pet različitih metoda preuzorkovanja:

1. Najbliži sused
2. Bilinearno (jezgro 2x2)
3. Kubno (jezgro 4x4)
4. Kubni B-Spline (jezgro 4x4)
5. Lanczos (jezgro 6x6)

Definisanje opcija pravljenja rastera

Pri izvozu rastera, *Opcije pravljenja rastera* omogućava vam da definišete dodatne opcije koje kontrolišu kako je izlazna datoteka strukturirana i komprimovana. Više pogledajte na *Opcije rasterskog drajvera*.

Савет

Izaberite prazan unos ako želite da napravite sopstvenu prilagođenu kombinaciju parametara.

Definisanje podešavanja transformacije

Postoji nekoliko opcija koje je potrebno definisati za georeferencirani izlazni raster.

- Polje *Napravi world datoteku* dostupno je samo ako odlučite da koristite tip transformacije Linearni ili Helmert, jer to znači da rasterska slika zapravo neće biti transformisana. U tom slučaju polje *Izlazni raster* nije aktivirano, jer će biti napravljena samo nova world datoteka.
- Za sve druge tipove transformacije, morate definisati *Izlazni raster*. Podrazumevano se nova datoteka ([file-name]_modified) pravi u istom folderu zajedno sa originalnom rasterskom slikom.
- Kao sledeći korak, morate definisati *Ciljni CRS* (koordinatni referentni sistem) za georeferencirani raster (pogledajte *Working with Projections*).
- Ako želite, možete **generisati pdf kartu** i takođe **pdf izveštaj**. Izveštaj uključuje informacije o korišćenim parametrima transformacije, sliku reziduala i listu sa svim GCP-ovima i njihovim RMS greškama.
- Osim toga, možete aktivirati polje *Postavi ciljnu rezoluciju* i definisati rezoluciju piksela izlaznog rastera. Podrazumevana horizontalna i vertikalna rezolucija je 1.
- Polje *Koristi 0 za prozirnost kada je potrebno* može se aktivirati ako pikseli sa vrednošću 0 treba da budu vizuelizovani prozirno. U našem primeru topografskog lista, sve bele oblasti bile bi prozirne.
- Polje *Sačuvaj GCP tačke* čuvaće GCP tačke u datoteci pored izlaznog rastera.
- Najzad, *Učitaj u projekat po završetku* automatski učitava izlazni raster u QGIS platno karte kada je transformacija završena.

Pokretanje transformacije

Nakon što su svi GCP-ovi prikupljeni i sva podešavanja transformacije definisana, samo pritisnite dugme Pokreni georeferenciranje da biste napravili novi georeferencirani raster.

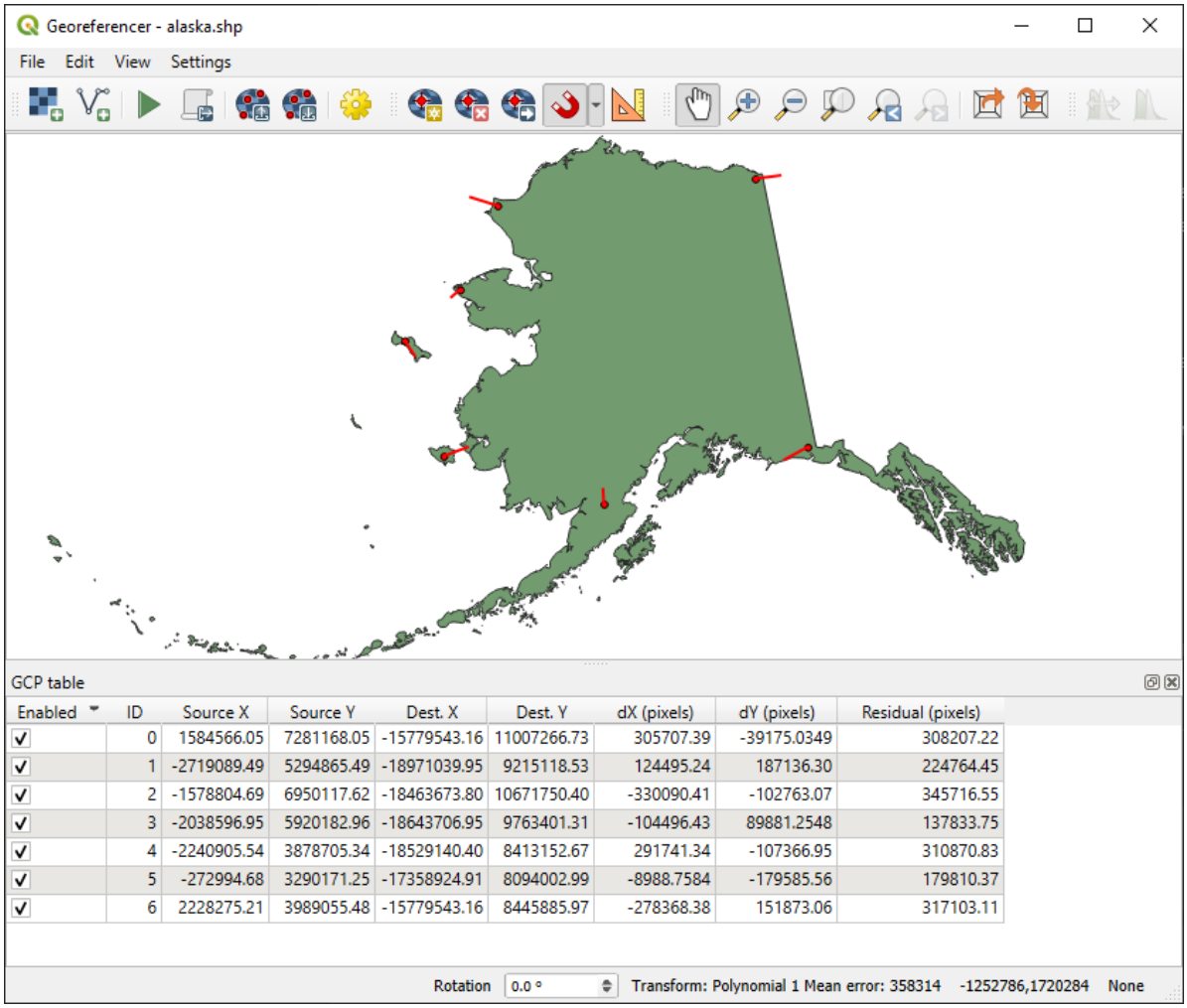
11.3.2 Georeferenciranje vektorskog sloja

Georeferenciranje vektorskih slojeva radi slično georeferenciranju rastera, ali umesto podudaranja piksela slike, podudarate vektorske geometrije (tačke, linije ili poligone) sa poznatim prostornim referencama.

Standardni postupak počinje isto kao za georeferenciranje rastera: otvorite QGIS i dodajte sloj na platno karte da biste ga koristili kao referencu. To može biti georeferencirani rasterski ili vektorski sloj, ili WMS sloj.

Otvorite dijalog Georeferencer iz *Sloj* ► *Georeferencer*.

Pokrenite georeferenciranje prateći ove korake (u ovom primeru koristimo nereferencirani `alaska.shp`):



Слика11.41: Dijalog Vektorski georeferencer

1. Učitajte nereferencirani vektorski sloj pomoću dugmeta . Vektorski sloj će se pojaviti u glavnoj radnoj oblasti dijaloga.

2. Aktivirajte lepljenje klikom na dugme i izborom željenog tipa(ova) lepljenja. Time se omogućava lepljenje za referentni sloj pri postavljanju GCP-ova.

   Takođe možete koristiti dok Napredno digitalizovanje da biste obezbedili izbor tačaka visoke preciznosti. Za više informacija pogledajte odeljak *Napredno digitalizovanje*.

3. Koristite dugme Dodaj GCP tačku da biste dodali tačku u radnu oblast. Ručno unesite njene koordinate i postavite CRS, ili kliknite na dugme Sa platna karte da biste izabrali koordinate iz georeferenciranog sloja na glavnom QGIS platnu karte. U tom slučaju CRS će biti postavljen automatski.

4. Definišite podešavanja transformacije:

   - Izaberite *Tip transformacije*. Algoritmi transformacije su isti kao oni za georeferenciranje rastera. Za više detalja pogledajte *Definisanje podešavanja transformacije*.
   - Definišite *Ciljni CRS* (koordinatni referentni sistem) za georeferencirani vektor (pogledajte *Working with Projections*).
   - Postavite format i putanju izlazne datoteke (npr. GeoPackage, Shapefile). Podrazumevano se nova datoteka sa sufiksom `_modified` pravi u istom folderu kao originalna vektorska datoteka.

- Opciono, omogućite **Generiši PDF kartu** i **Generiši PDF izveštaj**. Izveštaj uključuje parametre transformacije, GCP reziduale i rezime RMS grešaka.
- Omogućite *Sačuvaj GCP tačke* da biste sačuvali GCP-ove u datoteci uz izlazni vektorski sloj.
- Omogućite *Učitaj u projekat po završetku* da biste dodali rezultat direktno na platno karte.

5. Kliknite na Pokreni georeferenciranje da biste pokrenuli transformaciju i generisali georeferencirani vektorski sloj.

11.3.3 Prikaz i prilagođavanje svojstava sloja

Klik na opciju *Svojstva izvora* u meniju *Podešavanja* otvara *Svojstva rasterskog sloja* ili *Svojstva vektorskog sloja* u zavisnosti od tipa sloja koji georeferencirate.

11.3.4 Konfigurisanje georeferencer-a

Ponašanje georeferencer-a možete prilagoditi u *Podešavanja* ► *Konfiguriši Georeferencer* (ili koristite prečicu na tastaturi `Ctrl+P`).

- Pod *Savet za tačku* možete koristiti polja za potvrdu da biste uključili/isključili prikaz GCP ID-jeva i X/Y koordinata i u prozoru Georeferencer-a i na glavnom platnu karte.
- *Jedinice reziduala* kontroliše da li se jedinice reziduala daju u pikselima ili jedinicama karte
- *PDF izveštaj* omogućava vam da postavite veličinu margine u mm za izvoz izveštaja
- *PDF karta* omogućava vam da izaberete veličinu papira za izvoz karte
- Najzad, možete aktivirati *Prikaži prozor Georeferencer-a usidren*. Time se prozor Georeferencer-a usidruje u glavni QGIS prozor umesto da se prikazuje kao zaseban prozor koji se može minimizovati.

11.4 Exploring Data Formats and Fields

11.4.1 Raster data

GIS raster data are matrices of discrete cells that represent features / phenomena on, above or below the earth's surface. Each cell in the raster grid has the same size, and cells are usually rectangular (in QGIS they will always be rectangular). Typical raster datasets include remote sensing data, such as aerial photography, or satellite imagery and modelled data, such as elevation or temperature.

Unlike vector data, raster data typically do not have an associated database record for each cell. They are geocoded by pixel resolution and the X/Y coordinate of a corner pixel of the raster layer. This allows QGIS to position the data correctly on the map canvas.

The GeoPackage format is convenient for storing raster data when working with QGIS. The popular and powerful GeoTiff format is a good alternative.

QGIS makes use of georeference information inside the raster layer (e.g., GeoTiff) or an associated *world file* to properly display the data.

11.4.2 Vector Data

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.

Белешка

QGIS supports (multi)point, (multi)line, (multi)polygon, CircularString, CompoundCurve, CurvePolygon, MultiCurve, MultiSurface feature types, all optionally with Z and/or M values.

You should also note that some drivers don't support some of these feature types, like CircularString, CompoundCurve, CurvePolygon, MultiCurve, MultiSurface feature type. QGIS will convert them.

GeoPackage

The GeoPackage (GPKG) format is platform-independent, and is implemented as a SQLite database container, and can be used to store both vector and raster data. The format was defined by the Open Geospatial Consortium (OGC), and was published in 2014.

GeoPackage can be used to store the following in a SQLite database:

- **vector** features
- **tile matrix sets of imagery** and **raster** maps
- attributes (non-spatial data)
- extensions

Since QGIS version 3.8, GeoPackage can also store QGIS projects. GeoPackage layers can have JSON fields.

GeoPackage is the default format for vector data in QGIS.

ESRI Shapefile format

The ESRI Shapefile format is still one of the most used vector file formats, even if it has some limitations compared to for instance GeoPackage and SpatiaLite.

An ESRI Shapefile format dataset consists of several files. The following three are required:

1. `.shp` file containing the feature geometries
2. `.dbf` file containing the attributes in dBase format
3. `.shx` index file

An ESRI Shapefile format dataset can also include a file with a `.prj` suffix, which contains projection information. While it is very useful to have a projection file, it is not mandatory. A Shapefile format dataset can contain additional files. For further details, see the the ESRI technical specification.

GDAL has read-write support for compressed ESRI Shapefile format (`shz` and `shp.zip`).

Improving Performance for ESRI Shapefile format datasets

To improve the drawing performance for an ESRI Shapefile format dataset, you can create a spatial index. A spatial index will improve the speed of both zooming and panning. Spatial indexes used by QGIS have a `.qix` extension.

Use these steps to create the index:

1. Load an ESRI Shapefile format dataset (see *The Browser Panel*)
2. Open the *Layer Properties* dialog by double-clicking on the layer name in the legend or by right-clicking and choosing *Properties...* from the context menu
3. In the *Source* tab, click the *Create Spatial Index* button

Problem loading a .prj file

If you load an ESRI Shapefile format dataset with a `.prj` file and QGIS is not able to read the coordinate reference system from that file, you will need to define the proper projection manually in the *Layer Properties* ► *Source* tab of the layer by clicking the  Select CRS button. This is due to the fact that `.prj` files often do not provide the complete projection parameters as used in QGIS and listed in the *CRS* dialog.

For the same reason, if you create a new ESRI Shapefile format dataset with QGIS, two different projection files are created: a `.prj` file with limited projection parameters, compatible with ESRI software, and a `.qpj` file, providing all the parameters of the CRS. Whenever QGIS finds a `.qpj` file, it will be used instead of the `.prj`.

Delimited Text Files

Delimited text files are very common and widely used because of their simplicity and readability – data can be viewed and edited in a plain text editor. A delimited text file is tabular data with columns separated by a defined character and rows separated by line breaks. The first row usually contains the column names. A common type of delimited text file is a CSV (Comma Separated Values), with columns separated by commas. Delimited text files can also contain positional information (see *Storing geometry information in delimited text files*).

QGIS allows you to load a delimited text file as a layer or an ordinary table (see *The Browser Panel* or *Importing a delimited text file*). First check that the file meets the following requirements:

1. The file must have a delimited header row of field names. This must be the first line of the data (ideally the first row in the text file).
2. If geometry should be enabled, the file must contain field(s) that define the geometry. These field(s) can have any name.
3. The X and Y coordinates fields (if geometry is defined by coordinates) must be specified as numbers. The coordinate system is not important.
4. If you have a CSV file with non-string columns, you can have an accompanying CSVT file (see section *Using CSVT file to control field formatting*).

The elevation point data file `elevp.csv` in the QGIS sample dataset (see section *Preuzimanje primernih podataka*) is an example of a valid text file:

```
X;Y;ELEV
-300120;7689960;13
-654360;7562040;52
1640;7512840;3
[...]
```

Some things to note about the text file:

1. The example text file uses `;` (semicolon) as delimiter (any character can be used to delimit the fields).
2. The first row is the header row. It contains the fields `X`, `Y` and `ELEV`.
3. No quotes (`"`) are used to delimit text fields
4. The X coordinates are contained in the `X` field
5. The Y coordinates are contained in the `Y` field

Storing geometry information in delimited text files

Delimited text files can contain geometry information in two main forms:

- As coordinates in separate columns (eg. `Xcol`, `Ycol`...), for point geometry data;
- As well-known text (WKT) representation of geometry in a single column, for any geometry type.

Features with curved geometries (CircularString, CurvePolygon and CompoundCurve) are supported. Here are some examples of geometry types in a delimited text file with geometries coded as WKT:

```
Label;WKT_geom
LineString;LINESTRING(10.0 20.0, 11.0 21.0, 13.0 25.5)
CircularString;CIRCULARSTRING(268 415,227 505,227 406)
CurvePolygon;CURVEPOLYGON(CIRCULARSTRING(1 3, 3 5, 4 7, 7 3, 1 3))
CompoundCurve;COMPOUNDCURVE((5 3, 5 13), CIRCULARSTRING(5 13, 7 15,
  9 13), (9 13, 9 3), CIRCULARSTRING(9 3, 7 1, 5 3))
```

Delimited text files also support Z and M coordinates in geometries:

```
LINESTRINGZ(10.0 20.0 30.0, 11.0 21.0 31.0, 11.0 22.0 30.0)
```

Using CSVT file to control field formatting

When loading CSV files, the GDAL driver assumes all fields are strings (i.e. text) unless it is told otherwise. You can create a CSVT file to tell GDAL (and QGIS) the data type of the different columns:

Tip	Назив	Example
Whole number	Integer	4
Logička vrednost	Integer(Boolean)	true
Decimal number	Real	3.456
Datum	Date (YYYY-MM-DD)	2016-07-28
Vreme	Time (HH:MM:SS+nn)	18:33:12+00
Date & Time	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)

The CSVT file is a **ONE line** plain text file with the data types in quotes and separated by commas, e.g.:

```
"Integer","Real","String"
```

You can even specify width and precision of each column, e.g.:

```
"Integer(6)","Real(5.5)","String(22)"
```

This file is saved in the same folder as the `.csv` file, with the same name, but `.csvt` as the extension.

You can find more information at GDAL CSV Driver.

Савет

Detect Field Types

Instead of using a CSVT file to tell the data types, QGIS provides the possibility to automatically detect the field types and to change the assumed field types.

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.

Савет

PostGIS Layers

Normally, a PostGIS layer is identified by an entry in the geometry_columns table. QGIS can load layers that do not have an entry in the geometry_columns table. This includes both tables and views. Refer to your PostgreSQL manual for information on creating views.

This section contains some details on how QGIS accesses PostgreSQL layers. Most of the time, QGIS should simply provide you with a list of database tables that can be loaded, and it will load them on request. However, if you have trouble loading a PostgreSQL table into QGIS, the information below may help you understand QGIS messages and give you directions for modifying the PostgreSQL table or view definition to allow QGIS to load it.

Белешка

A PostgreSQL database can also store QGIS projects.

Primary key

QGIS requires that PostgreSQL layers contain a column that can be used as a unique key for the layer. For tables, this usually means that the table needs a primary key, or a column with a unique constraint on it. In QGIS, this column needs to be of type int4 (an integer of size 4 bytes). Alternatively, the ctid column can be used as primary key. If a table lacks these items, the oid column will be used instead. Performance will be improved if the column is indexed (note that primary keys are automatically indexed in PostgreSQL).

QGIS offers a checkbox **Select at id** that is activated by default. This option gets the ids without the attributes, which is faster in most cases.

Prikaz

If the PostgreSQL layer is a view, the same requirement exists, but views do not always have primary keys or columns with unique constraints on them. You have to define a primary key field (has to be integer) in the QGIS dialog before you can load the view. If a suitable column does not exist in the view, QGIS will not load the layer. If this occurs, the solution is to alter the view so that it does include a suitable column (a type of integer and either a primary key or with a unique constraint, preferably indexed).

As for table, a checkbox **Select at id** is activated by default (see above for the meaning of the checkbox). It can make sense to disable this option when you use expensive views.

Белешка

PostgreSQL foreign table

PostgreSQL foreign tables are not explicitly supported by the PostgreSQL provider and will be handled like a view.

QGIS layer_style table and database 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. Make a PLAIN backup of the `layer_style` table
2. Open the file within a text editor
3. Change the line `SET xmloption = content;` into `SET XML OPTION DOCUMENT;`
4. Save the file
5. Use psql to restore the table in the new database

Filter database side

QGIS allows to filter features already on server side. Check *Settings* ► *Options* ► *Data Sources* ► *Execute expressions on server-side if possible* to do so. Only supported expressions will be sent to the database. Expressions using unsupported operators or functions will gracefully fallback to local evaluation.

Support of PostgreSQL data types

Data types supported by the PostgreSQL provider include: integer, float, boolean, binary object, varchar, geometry, timestamp, array, hstore and json.

Importing Data into PostgreSQL

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 comes with a core plugin named DB Manager. It can be used to load data, and it includes support for schemas. See section *DB Manager Plugin* for more information.

shp2pgsql

PostGIS includes a utility called **shp2pgsql**, that can be used to import Shapefile format datasets into a PostGIS-enabled database. For example, to import a Shapefile format dataset named `lakes.shp` into a PostgreSQL database named `gis_data`, use the following command:

```
shp2pgsql -s 2964 lakes.shp lakes_new | psql gis_data
```

This creates a new layer named `lakes_new` in the `gis_data` database. The new layer will have a spatial reference identifier (SRID) of 2964. See section *Working with Projections* for more information about spatial reference systems and projections.

Савет

Exporting datasets from PostgreSQL

There is also a tool for exporting PostgreSQL datasets with 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
```

If you prefer to use the PostgreSQL's **COPY** command instead of the default **INSERT INTO** method, you can export the following environment variable (at least available on and):

```
export PG_USE_COPY=YES
```

ogr2ogr does not create spatial indexes like **shp2pgsl** does. You need to create them manually, using the normal SQL command **CREATE INDEX** afterwards, as an extra step (as described in the next section *Improving Performance*).

Improving Performance

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).

Савет

You can use the DBManager to create an index for your layer. You should first select the layer and click on *Table* ► *Edit table*, go to *Indexes* tab and click on *Add Spatial Index*.

The syntax for creating a GiST index is:

```
CREATE INDEX [indexname] ON [tablename]
  USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

Note that for large tables, creating the index can take a long time. Once the index is created, you should perform a `VACUUM ANALYZE`. See the PostGIS documentation (POSTGIS-PROJECT in *Literatura i veb reference*) for more information.

The following example creates a 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 Layers

If you want to save a vector layer using the SpatiaLite format, you can do this by following instructions at *Creating new layers from an existing layer*. You select `SpatiaLite` as *Format* and enter both *File name* and *Layer name*.

Also, you can select `SQLite` as format and then add `SPATIALITE=YES` in the *Custom Options* ► *Data source* field. This tells GDAL to create a SpatiaLite database. See also https://gdal.org/en/latest/drivers/vector/sqlite.html.

QGIS also supports editable views in SpatiaLite. For SpatiaLite data management, you can also use the core plugin *DB Manager*.

If you want to create a new SpatiaLite layer, please refer to section *Creating a new SpatiaLite layer*.

GeoJSON specific parameters

When *exporting layers* to GeoJSON, there are some specific *Layer Options* available. These options come from GDAL which is responsible for the writing of the file:

- *COORDINATE_PRECISION* the maximum number of digits after the decimal separator to write in coordinates. Defaults to 15 (note: for Lat Lon coordinates 6 is considered enough). Truncation will occur to remove trailing zeros.
- *RFC7946* by default GeoJSON 2008 will be used. If set to YES, the updated RFC 7946 standard will be used. Default is NO (thus GeoJSON 2008). See https://gdal.org/en/latest/drivers/vector/geojson.html#rfc-7946-write-support for the main differences, in short: only EPSG:4326 is allowed, other crs's will be transformed, polygons will be written such as to follow the right-hand rule for orientation, values of a „bbox" array are [west, south, east, north], not [minx, miny, maxx, maxy]. Some extension member names are forbidden in FeatureCollection, Feature and Geometry objects, the default coordinate precision is 7 decimal digits
- *WRITE_BBOX* set to YES to include the bounding box of the geometries at the feature and feature collection level

Besides GeoJSON there is also an option to export to „GeoJSON - Newline Delimited" (see https://gdal.org/en/latest/drivers/vector/geojsonseq.html). Instead of a FeatureCollection with Features, you can stream one type (probably only Features) sequentially separated with newlines.

GeoJSON - Newline Delimited has some specific Layer options available too:

- *COORDINATE_PRECISION* see above (same as for GeoJSON)
- *RS* whether to start records with the RS=0x1E character. The difference is how the features are separated: only by a newline (LF) character (Newline Delimited JSON, geojsonl) or by also prepending a record-separator (RS) character (giving GeoJSON Text Sequences, geojsons). Default to NO. Files are given the `.json` extension if extension is not provided.

SAP HANA Spatial Layers

This section contains some details on how QGIS accesses SAP HANA layers. Most of the time, QGIS should simply provide you with a list of database tables and views that can be loaded, and it will load them on request. However, if you have trouble loading an SAP HANA table or view into QGIS, the information below may help you understand the root cause and assist in resolving the issue.

Feature Identification

If you'd like to use all of QGIS' feature editing capabilities, QGIS must be able to unambiguously identify each feature in a layer. Internally, QGIS uses a 64-bit signed integer to identify features, whereas the negative range is reserved for special purposes.

Therefore, the SAP HANA provider requires a unique key that can be mapped to a positive 64-bit integer to fully support QGIS' feature editing capabilities. If it is not possible to create such a mapping, you might still view the features, but editing might not work.

Adding tables

When adding a table as a layer, the SAP HANA provider uses the table's primary key to map it to a unique feature id. Therefore, to have full feature editing support, you need to have a primary key to your table definition.

The SAP HANA provider supports multi-column primary keys, but if you'd like to get the best performance, your primary key should be a single column of type `INTEGER`.

Adding views

When adding a view as a layer, the SAP HANA provider cannot automatically identify columns that unambiguously identify a feature. Furthermore, some views are read-only and cannot be edited.

To have full feature editing support, the view must be updatable (check column `IS_READ_ONLY` in system view `SYS.VIEWS` for the view in question) and you must manually provide QGIS with one or more columns that identify a feature. The columns can be given by using *Layer* ► *Add Layer* ► *Add SAP HANA Spatial Layer* and then selecting the columns in the *Feature id* column. For best performance, the *Feature id* value should be a single `INTEGER` column.

Field domain

Field domains are rules that define acceptable values for a field in a database table. This is applicable when working with data sources that support field domains, such as GeoPackage and ESRI File Geodatabase. Field domains can be managed from the contextual menu of a field in the *Browser* panel. The following types are available:

- *New Range Domain…*: creates a new domain to restrict field values to a specified numeric range.
- *New Coded Values Domain…*: creates a new domain to restrict field values to a predefined list of acceptable values.
- *New Glob Domain…*: creates a new domain to restrict field values to matching a regular expression pattern.

When creating a field domain, following options are provided in the dialog:

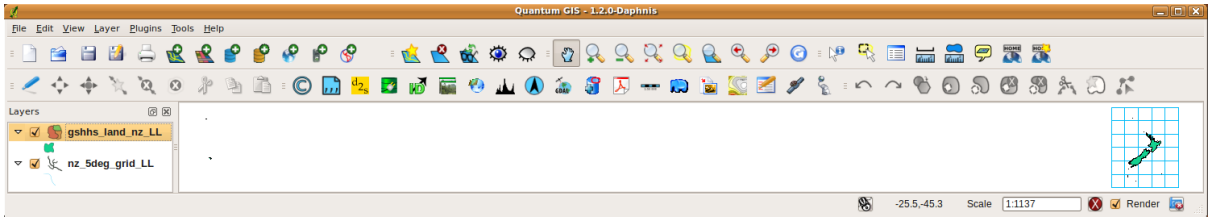
- *Name*: set the name of the new field domain.
- *Description*: provide a description for the new field domain.
- *Field type*: select the data type of the field domain (e.g., Boolean, Text, Integer, Decimal).
- Additional *Policies*, only available for ESRI File Geodatabase format:
  - *Split policy*
  - *Merge policy*
- Under *Range*, you can set the following options:
  - *Minimum*: set the minimum acceptable value for the field.
  - *Maximum*: set the maximum acceptable value for the field.
  - Check the *Inclusive* if you want to include the boundary values in the acceptable range.
- *Values*: click the Add row or Remove row to manage the list of acceptable values for the field.
- *Pattern*: define the *Glob pattern* using wildcard characters.

Белешка

When loading vector layers into QGIS, fields with *Field Domains* (such as those defined in a GeoPackage or ESRI File Geodatabase) are automatically detected. These domains are database-level constraints, meaning they are enforced by the database itself and apply across different applications, not just QGIS.

11.4.3 Layers crossing 180° longitude

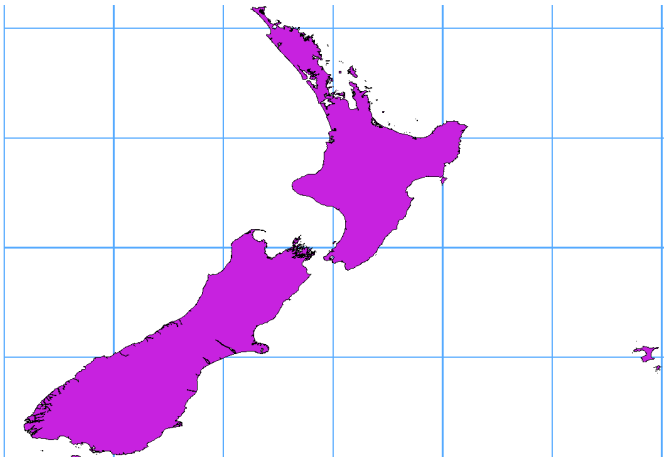
Many GIS packages don't wrap layers with a geographic reference system (lat/lon) crossing the 180 degrees longitude line. As result, if we open such a layer in QGIS, we could see two widely separated locations, that should appear near each other. In Сл.11.42, the tiny point on the far left of the map canvas (Chatham Islands) should be within the grid, to the right of the New Zealand main islands.



Слика11.42: Map in lat/lon crossing the 180° longitude line

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.



Слика11.43: Crossing 180° longitude applying the **ST_ShiftLongitude** function

1. Import data into PostgreSQL (*Importing Data into PostgreSQL*) using, for example, the DB Manager plugin.
2. Use the PostgreSQL command line interface to issue the following command:

```
-- In this example, "TABLE" is the actual name of your PostgreSQL table
update TABLE set geom=ST_ShiftLongitude(geom);
```

3. If everything went well, you should receive a confirmation about the number of features that were updated. Then you'll be able to load the map and see the difference (*Figure_vector_crossing_map*).

ГЛАВА 12

Rad sa vektorskim podacima

12.1 The Vector Properties Dialog

The *Layer Properties* dialog for a vector layer provides general settings to manage appearance of layer features in the map (symbology, labeling, diagrams), interaction with the mouse (actions, map tips, form design). It also provides information about the layer.

To access the *Layer Properties* dialog:

- In the *Layers* panel, double-click the layer or right-click and select *Properties…* from the pop-up menu;
- Go to *Layer* ► *Layer Properties…* menu when the layer is selected.

The vector *Layer Properties* dialog provides the following sections:

Information	*Source*	*Symbology* ([1])
Labels ([1])	*Mask* ([1])	*3D View* ([1])
Diagrams	*Fields*	*Attributes Form*
Joins	*Auxiliary Storage*	*Actions*
Display	*Rendering*	*Temporal*
Variables	*Elevation*	*Metadata*
Dependencies	*Legend*	*QGIS Server*
Digitizing	*External plugins* ([2]) tabs	

Савет

Share full or partial properties of the layer styles

[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.

The *Style* menu at the bottom of the dialog allows you to import or export these or part of these properties from/to several destination (file, clipboard, database). See *Managing Custom Styles*.

Белешка

Because properties (symbology, label, actions, default values, forms…) of embedded layers (see *Embedding layers from external projects*) are pulled from the original project file and to avoid changes that may break this behavior, the layer properties dialog is made unavailable for these layers.

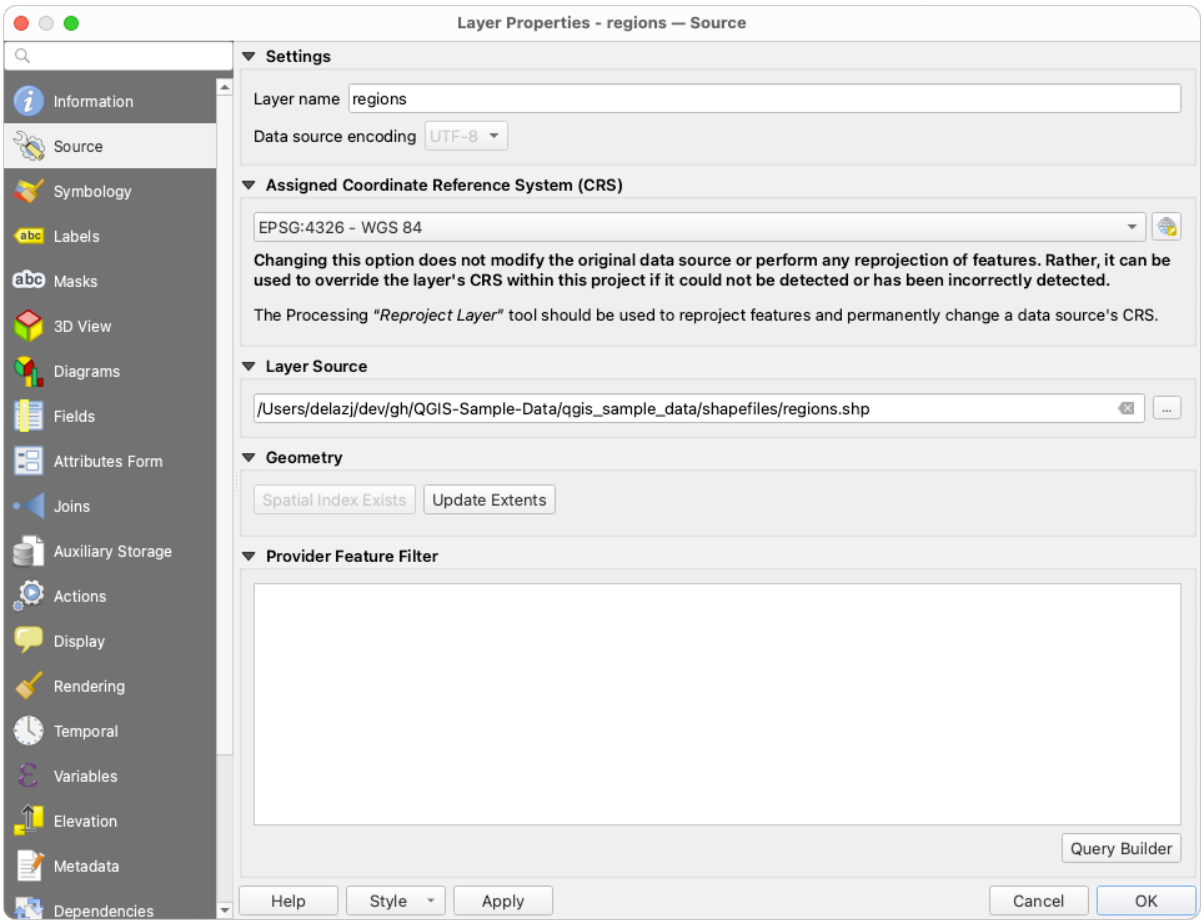
12.1.1 Information Properties

The  *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata on the current layer. Provided information are:

- general such as name in the project, source path, list of auxiliary files, last save time and size, the used provider
- custom properties, used to store in the active project additional information about the layer. Default custom properties may include *layer notes*, *legend widgets*, *layer variables*, *form properties*… More custom properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- based on the provider of the layer: format of storage, geometry type, data source encoding, extent, feature count…
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e. whether it's static or dynamic)
- picked from the *filled metadata*: access, extents, links, contacts, history…
- and related to its geometry (spatial extent, CRS…) or its attributes (number of fields, characteristics of each…).

12.1.2 Source Properties

 Use this tab to define general settings for the vector layer.



Слика12.1: Source tab in vector Layer Properties dialog

Podešavanja

- Set a *Layer name* different from the layer filename that will be used to identify the layer in the project (in the *Layers Panel*, with expressions, in print layout legend, …)
- Depending on the data format, select the *Data source encoding* if not correctly detected by QGIS.

Coordinate Reference System

- Displays the layer's *Assigned Coordinate Reference System (CRS)*. You can change the layer's CRS, selecting a recently used one in the drop-down list or clicking on Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the CRS applied to the layer is a wrong one or if none was applied. If you wish to reproject your data into another CRS, rather use layer reprojection algorithms from Processing or *Save it into another layer*.

Layer source and Geometry

- Depending on the data provider, a *Layer source* group indicates the path to the source of the dataset and allows for replacing the loaded layer:
  - When the layer is stored as file on disk, edit the path shown in the text box or press … Browse to select another file on the disk. Both layers do not need to share attribute fields, geometry type or file formats.
  - When the layer is provided by an ArcGIS Feature service, it is possible to modify its *authentication settings*, while keeping unchanged the details for connecting to the service.
- *Create spatial index* (only for OGR-supported formats): helps speeding layer rendering and features' geometry retrieval.

- *Update extents* information for a layer.

Query Builder

The Query Builder dialog is accessible through the *Query Builder* button at the bottom of the *Source* tab in the Layer Properties dialog, under the *Provider feature filter* group.

The Query Builder provides an interface that allows you to define a subset of the features in the layer using a SQL--like WHERE clause and to display the result in the main window. As long as the query is active, only the features corresponding to its result are available in the project.

You can use one or more layer attributes to define the filter in the `Query Builder`. The use of more than one attribute is shown in Сл.12.2. In the example, the filter combines the attributes

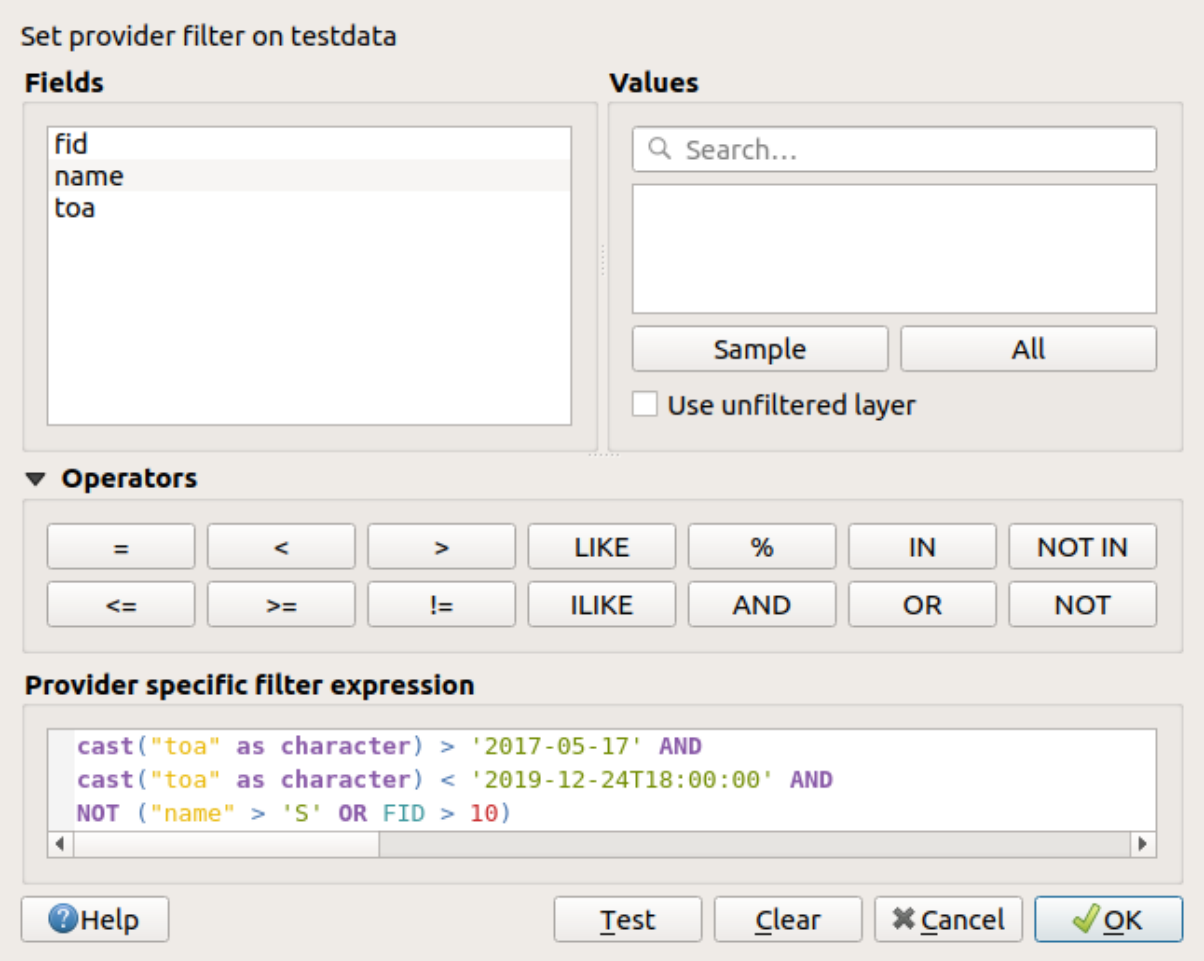
- `toa` (`DateTime` field: `cast("toa" as character) > '2017-05-17'` and `cast("toa" as character) < '2019-12-24T18:00:00'`),
- `name` (`String` field: `"name" > 'S'`) and
- `FID` (`Integer` field: `FID > 10`)

using the AND, OR and NOT operators and parenthesis. This syntax (including the DateTime format for the `toa` field) works for GeoPackage datasets.

The filter is made at the data provider (OGR, PostgreSQL, MS SQL Server…) level. So the syntax depends on the data provider (DateTime is for instance not supported for the ESRI Shapefile format). The complete expression:

```
cast("toa" as character) > '2017-05-17' AND
cast("toa" as character) < '2019-12-24T18:00:00' AND
NOT ("name" > 'S' OR FID > 10)
```

You can also open the *Query Builder* dialog using the *Filter…* option from the *Layer* menu or the layer contextual menu. The *Fields*, *Values* and *Operators* sections in the dialog help you to construct the SQL-like query exposed in the *Provider specific filter expression* box.



Слика12.2: Query Builder

The **Fields** list contains all the fields of the layer. To add an attribute column to the SQL WHERE clause field, double-click its name or just type it into the SQL box.

The **Values** frame lists the values of the currently selected field. To list all unique values of a field, click the *All* button. To instead list the first 25 unique values of the column, click the *Sample* button. To add a value to the SQL WHERE clause field, double click its name in the Values list. You can use the search box at the top of the Values frame to easily browse and find attribute values in the list.

The **Operators** section contains all usable *operators*. To add an operator to the SQL WHERE clause field, click the appropriate button. Relational operators (= , > , …), string comparison operator (`LIKE`), and logical operators (`AND`, `OR`, …) are available.

The *Test* button helps you check your query and displays a message box with the number of features satisfying the current query. Use the *Clear* button to wipe the SQL query and revert the layer to its original state (ie, fully load all the features). It is possible to *Save…* the query as a `.QQF` file, or *Load…* the query from a file into the dialog.

When a filter is applied, QGIS treats the resulting subset acts as if it were the entire layer. For example if you applied the filter above for 'Borough' (`"TYPE_2" = 'Borough'`), you can not display, query, save or edit `Anchorage`, because that is a 'Municipality' and therefore not part of the subset.

Савет

Filtered layers are indicated in the Layers Panel

In the *Layers* panel, filtered layer is listed with a Filter icon next to it indicating the query used when the

mouse hovers over the button. Double-click the icon opens the *Query Builder* dialog for edit. This can also be achieved through the *Layer* ► *Filter…* menu.

12.1.3 Symbology Properties

The  *Symbology* tab provides you with a comprehensive tool for rendering and symbolizing your vector data. You can use tools that are common to all vector data, as well as special symbolizing tools that were designed for the different kinds of vector data. However all types share the following dialog structure: in the upper part, you have a widget that helps you prepare the classification and the symbol to use for features and at the bottom the *Layer rendering* widget.

Савет

Switch quickly between different layer representations

Using the *Styles* ► *Add* menu at the bottom of the *Layer Properties* dialog, you can save as many styles as needed. A style is the combination of all properties of a layer (such as symbology, labeling, diagram, fields form, actions…) as you want. Then, simply switch between styles from the context menu of the layer in *Layers Panel* to automatically get different representations of your data.

Савет

Export vector symbology

You have the option to export vector symbology from QGIS into Google *.kml, *.dxf and MapInfo *.tab files. Just open the right mouse menu of the layer and click on *Save As...* to specify the name of the output file and its format. In the dialog, use the *Symbology export* menu to save the symbology either as *Feature symbology* ► or as *Symbol layer symbology* ►. If you have used symbol layers, it is recommended to use the second setting.

Features rendering

The renderer is responsible for drawing a feature together with the correct symbol. Regardless layer geometry type, there are four common types of renderers: single symbol, categorized, graduated and rule-based. For point layers, there are point displacement, point cluster and heatmap renderers available while polygon layers can also be rendered with the merged features, inverted polygons and 2.5 D renderers.

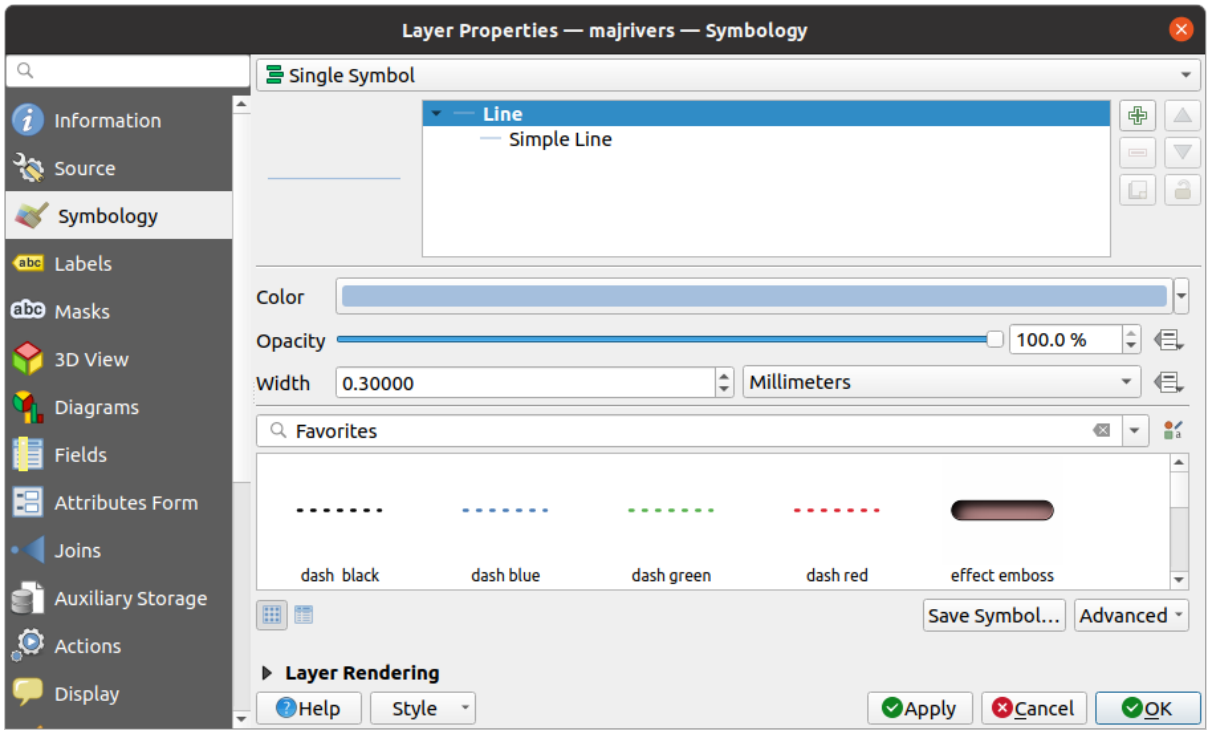
There is no continuous color renderer, because it is in fact only a special case of the graduated renderer. The categorized and graduated renderers can be created by specifying a symbol and a color ramp - they will set the colors for symbols appropriately. For each data type (points, lines and polygons), vector symbol layer types are available. Depending on the chosen renderer, the dialog provides different additional sections.

Белешка

If you change the renderer type when setting the style of a vector layer the settings you made for the symbol will be maintained. Be aware that this procedure only works for one change. If you repeat changing the renderer type the settings for the symbol will get lost.

Single Symbol Renderer

The  *Single Symbol* renderer is used to render all features of the layer using a single user-defined symbol. See *The Symbol Selector* for further information about symbol representation.



Слика12.3: Single symbol line properties

No Symbols Renderer

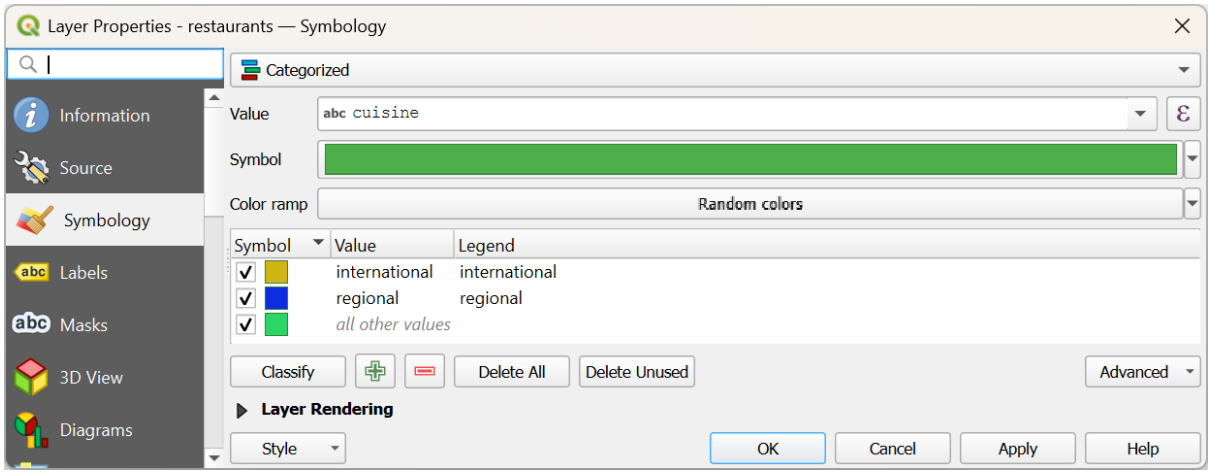
The *No Symbols* renderer is a special use case of the Single Symbol renderer as it applies the same rendering to all features. Using this renderer, no symbol will be drawn for features, but labeling, diagrams and other non-symbol parts will still be shown.

Selections can still be made on the layer in the canvas and selected features will be rendered with a default symbol. Features being edited will also be shown.

This is intended as a handy shortcut for layers which you only want to show labels or diagrams for, and avoids the need to render symbols with totally transparent fill/border to achieve this.

Categorized Renderer

The *Categorized* renderer is used to render the features of a layer, using a user-defined symbol whose aspect reflects the discrete values of a field or an expression.



Слика12.4: Categorized Symbolizing options

To use categorized symbology for a layer:

1. Select the *Value* of classification: it can be an existing field or an *expression* you can type in the box or build using the associated ε button. Using expressions for categorizing avoids the need to create a field for symbology purposes only (eg, if your classification criteria are derived from one or more attributes).

   The expression used to classify features can be of any type; eg, it can:

   - be a comparison. In this case, QGIS returns values `1` (**True**) and `0` (**False**). Some examples:

     ```
     myfield >= 100
     $id = @atlas_featureid
     myfield % 2 = 0
     within( $geometry, @atlas_geometry )
     ```

   - combine different fields:

     ```
     concat( field_1, ' ', field_2 )
     ```

   - be a calculation on fields:

     ```
     myfield % 2
     year( myfield )
     field_1 + field_2
     substr( field_1, -3 )
     ```

   - be used to transform linear values to discrete classes, e.g.:

     ```
     CASE WHEN x > 1000 THEN 'Big' ELSE 'Small' END
     ```

   - combine several discrete values into a single category, e.g.:

     ```
     CASE
     WHEN building IN ('residence', 'mobile home') THEN 'residential'
     WHEN building IN ('commercial', 'industrial') THEN 'Commercial and␣
     ↪Industrial'
     END
     ```

Савет

While you can use any kind of expression to categorize features, for some complex expressions it might be simpler to use *rule-based rendering*.

2. Configure the *Symbol*, which will be used as base symbol for all the classes;
3. Indicate the *Color ramp*, i.e. the range of colors from which the color applied to each symbol is selected.

   Besides the common options of the *color ramp widget*, you can apply a *Random Color Ramp* to the categories. You can click the *Shuffle Random Colors* entry to regenerate a new set of random colors if you are not satisfied.
4. Then click on the *Classify* button to create classes from the distinct values of the provided field or expression.
5. For each class, you can edit the *Legend* column to a more meaningful label (used in the *Layers* panel and the print layout).

   Белешка

   Keep in mind that some values may use widgets that do not display the actual value stored in the field. For example, a checkbox widget may store `1` and `0` for checked and unchecked states, while displaying `True` and `False` labels. In this case, to categorize features based on the checkbox state, you need to use the stored values (`1` and `0`) in the expression. QGIS will automatically use the display value for the legend column.
6. *Apply* the changes if the *live update* is not in use and each feature on the map canvas will be rendered with the symbol of its class.

   By default, QGIS appends an *all other values* class to the list. While empty at the beginning, this class is used as a default class for any feature not falling into the other classes (eg, when you create features with new values for the classification field / expression).

Further tweaks can be done to the default classification:

- You can Add new categories, Remove selected categories, *Delete All* of them or *Delete Unused* categories.
- A class can be disabled by unchecking the checkbox to the left of the class name; the corresponding features are hidden on the map.
- Drag-and-drop the rows to reorder the classes
- To change the symbol, the value or the legend of a class, double click the item.

Right-clicking over selected item(s) shows a contextual menu to:

- *Copy Symbol* and *Paste Symbol*, a convenient way to apply the item's representation to others
- *Change Color…* of the selected symbol(s)
- *Change Opacity…* of the selected symbol(s)
- *Change Output Unit…* of the selected symbol(s)
- *Change Width…* of the selected line symbol(s)
- *Change Size…* of the selected point symbol(s)
- *Change Angle…* of the selected point symbol(s)
- *Merge Categories*: Groups multiple selected categories into a single one. This allows simpler styling of layers with a large number of categories, where it may be possible to group numerous distinct categories into a smaller and more manageable set of categories which apply to multiple values.

Савет

Since the symbol kept for the merged categories is the one of the topmost selected category in the list, you may want to move the category whose symbol you wish to reuse to the top before merging.

- *Unmerge Categories* that were previously merged

The created classes also appear in a tree hierarchy in the *Layers* panel. Double-click an entry in the map legend to edit the assigned symbol. Right-click and you will get some *more options*.

The *Advanced* menu gives access to options to speed classification or fine-tune the symbols rendering:

- *Match to saved symbols*: Using the *symbols library*, assigns to each category a symbol whose name represents the classification value of the category
- *Match to symbols from file…*: Provided a file with symbols, assigns to each category a symbol whose name represents the classification value of the category
- *Symbol levels…* to define the order of symbols rendering.

Graduated Renderer

The  *Graduated* renderer is used to render all the features from a layer, using an user-defined symbol whose color or size reflects the assignment of a selected feature's attribute to a class.

Like the Categorized Renderer, the Graduated Renderer allows you to define rotation and size scale from specified columns.

Also, analogous to the Categorized Renderer, it allows you to select:

- The *Value* of classification: it can be an existing field or an *expression* you can type in the box or build using the associated  button. Using expressions for graduating avoids the need to create a field for symbology purposes only (eg, if your classification criteria are derived from one or more attributes).
- The symbol (using the Symbol selector dialog)
- The legend format and the precision
- The method to use to change the symbol: color or size
- The colors (using the color Ramp list) if the color method is selected
- The size (using the size domain and its unit)

Then you can use the Histogram tab which shows an interactive histogram of the values from the assigned field or expression. Class breaks can be moved or added using the histogram widget.

Белешка

You can use Statistical Summary panel to get more information on your vector layer. See *Statistical Summary Panel*.

Back to the Classes tab, you can specify the number of classes and also the mode for classifying features within the classes (using the Mode list). The available modes are:

- Equal Count (Quantile): each class will have the same number of elements (the idea of a boxplot).
- Equal Interval: each class will have the same size (e.g. with the values from 1 to 16 and four classes, each class will have a size of four).
- Fixed Interval: each class will have a fixed range of values (e.g. with the values from 1 to 16 and an interval size of 4, the classes will be 1-4, 5-8, 9-12 and 13-16).

- Logarithmic scale: suitable for data with a wide range of values. Narrow classes for low values and wide classes for large values (e.g. for decimal numbers with range [0..100] and two classes, the first class will be from 0 to 10 and the second class from 10 to 100).
- Natural Breaks (Jenks): the variance within each class is minimized while the variance between classes is maximized.
- Pretty Breaks: computes a sequence of about n+1 equally spaced nice values which cover the range of the values in x. The values are chosen so that they are 1, 2 or 5 times a power of 10. (based on pretty from the R statistical environment https://www.rdocumentation.org/packages/base/topics/pretty).
- Standard Deviation: classes are built depending on the standard deviation of the values.

The listbox in the center part of the *Symbology* tab lists the classes together with their ranges, labels and symbols that will be rendered.

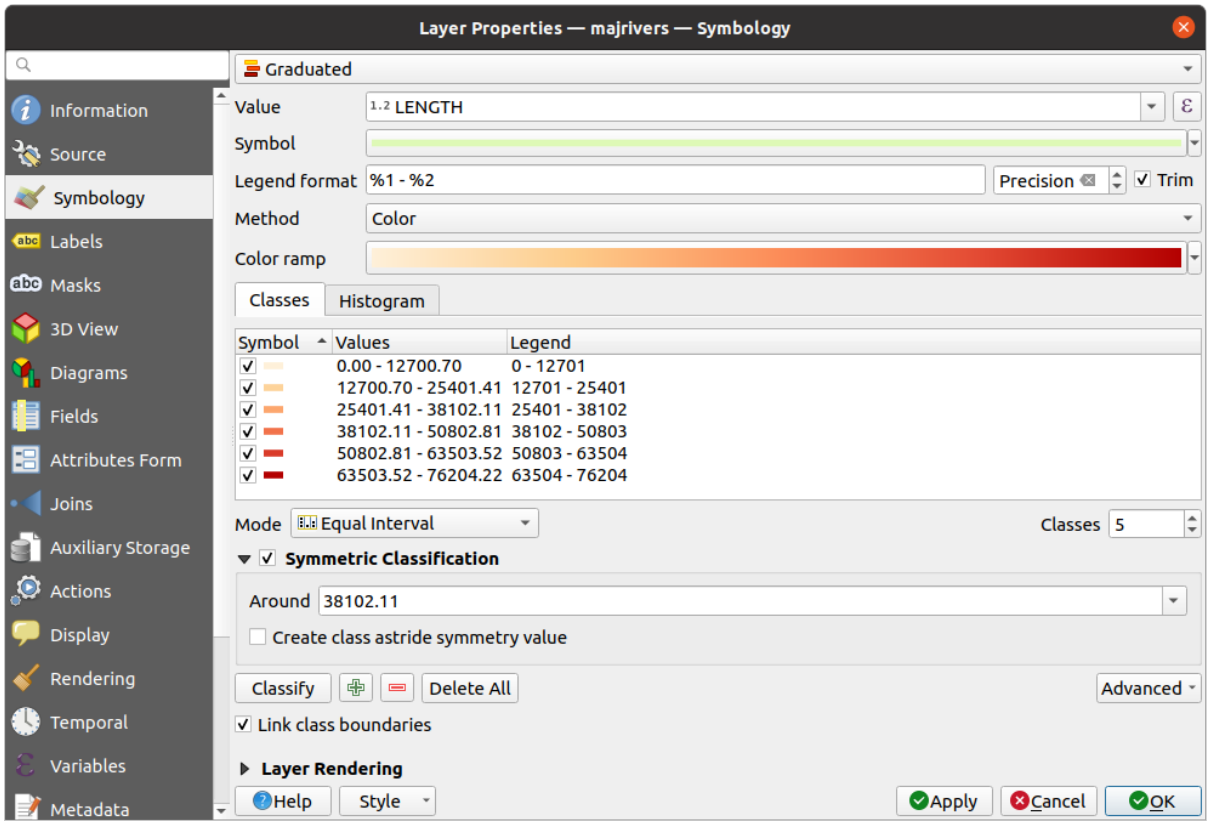
Click on **Classify** button to create classes using the chosen mode. Each classes can be disabled unchecking the checkbox at the left of the class name.

To change symbol, value and/or label of the class, just double click on the item you want to change.

Right-clicking over selected item(s) shows a contextual menu to:

- *Copy Symbol* and *Paste Symbol*, a convenient way to apply the item's representation to others
- *Change Color…* of the selected symbol(s)
- *Change Opacity…* of the selected symbol(s)
- *Change Output Unit…* of the selected symbol(s)
- *Change Width…* of the selected line symbol(s)
- *Change Size…* of the selected point symbol(s)
- *Change Angle…* of the selected point symbol(s)

The example in Сл.12.5 shows the graduated rendering dialog for the major_rivers layer of the QGIS sample dataset.



Слика12.5: Graduated Symbolizing options

The created classes also appear in a tree hierarchy in the *Layers* panel. Double-click an entry in the map legend to edit the assigned symbol. Right-click and you will get some *more options*.

Proportional Symbol and Multivariate Analysis

Proportional Symbol and Multivariate Analysis are not rendering types available from the Symbology rendering drop-down list. However with the *data-defined override* options applied over any of the previous rendering options, QGIS allows you to display your point and line data with such representation.

Creating proportional symbol

To apply a proportional rendering:

1. First apply to the layer the *single symbol renderer*.
2. Then set the symbol to apply to the features.
3. Select the item at the upper level of the symbol tree, and use the Data-defined override *button* next to the *Size* (for point layer) or *Width* (for line layer) option.
4. Select a field or enter an expression, and for each feature, QGIS will apply the output value to the property and proportionally resize the symbol in the map canvas.

   If need be, use the *Size assistant...* option of the menu to apply some transformation (exponential, flannery…) to the symbol size rescaling (see *Using the data-defined assistant interface* for more details).

You can choose to display the proportional symbols in the *Layers panel* and the *print layout legend item*: unfold the *Advanced* drop-down list at the bottom of the main dialog of the *Symbology* tab and select **Data-defined size legend…** to configure the legend items (see *Data-defined size legend* for details).

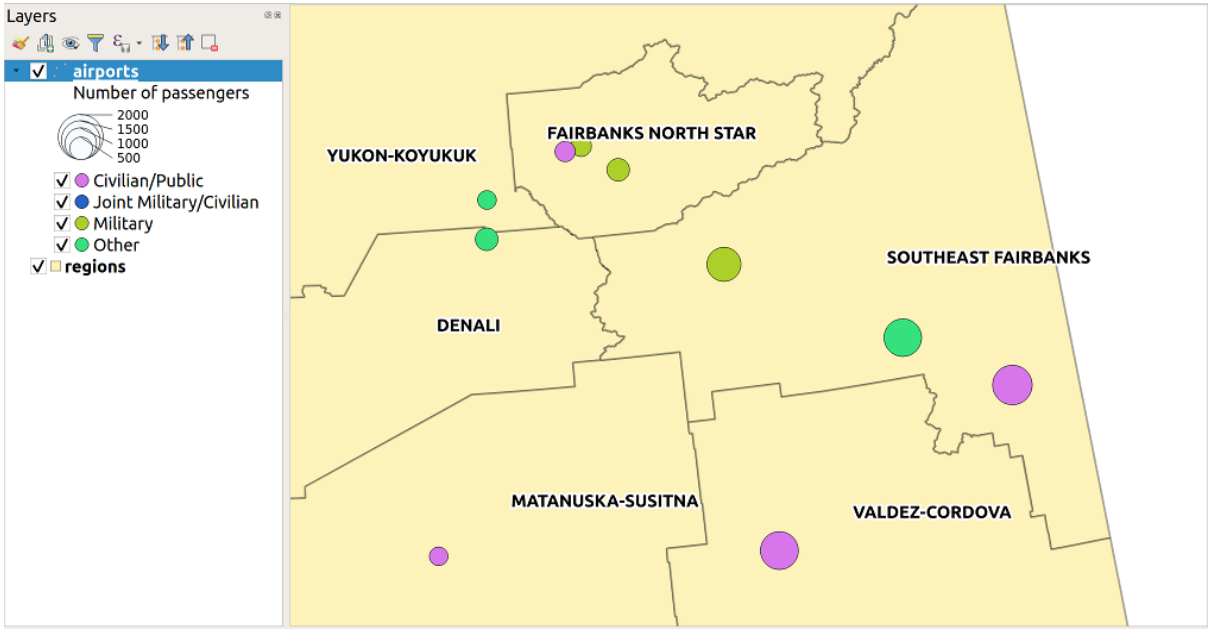
Creating multivariate analysis

A multivariate analysis rendering helps you evaluate the relationship between two or more variables e.g., one can be represented by a color ramp while the other is represented by a size.

The simplest way to create multivariate analysis in QGIS is to:

1. First apply a categorized or graduated rendering on a layer, using the same type of symbol for all the classes.
2. Then, apply a proportional symbology on the classes:
    1. Click on the *Change* button above the classification frame: you get the *The Symbol Selector* dialog.
    2. Rescale the size or width of the symbol layer using the *data defined override* widget as seen above.

Like the proportional symbol, the scaled symbology can be added to the layer tree, on top of the categorized or graduated classes symbols using the *data defined size legend* feature. And both representation are also available in the print layout legend item.



Слика12.6: Multivariate example with scaled size legend

Rule-based Renderer

Rules are QGIS *expressions* used to discriminate features according to their attributes or properties in order to apply specific rendering settings to them. Rules can be nested, and features belong to a class if they belong to all the upper nesting level(s).

The *Rule-based* renderer is thus designed to render all the features from a layer, using symbols whose aspect reflects the assignment of a selected feature to a fine-grained class.

To create a rule:

1. Activate an existing row by double-clicking it (by default, QGIS adds a symbol without a rule when the rendering mode is enabled) or click the Edit rule or Add rule button.
2. In the *Edit Rule* dialog that opens, you can define a label to help you identify each rule. This is the label that will be displayed in the *Layers Panel* and also in the print composer legend.
3. Manually enter an expression in the text box next to the *Filter* option or press the button next to it to open the expression string builder dialog.

4. Use the provided functions and the layer attributes to build an *expression* to filter the features you'd like to retrieve. Press the *Test* button to check the result of the query.
5. You can enter a longer label to complete the rule description.
6. You can use the *Scale Range* option to set scales at which the rule should be applied.
7. Finally, configure the *symbol to use* for these features.
8. And press *OK*.

A new row summarizing the rule is added to the Layer Properties dialog. You can create as many rules as necessary following the steps above or copy pasting an existing rule. Drag-and-drop the rules on top of each other to nest them and refine the upper rule features in subclasses.

The rule-based renderer can be combined with categorized or graduated renderers. Selecting a rule, you can organize its features in subclasses using the *Refine selected rules* drop-down menu. Refined classes appear like sub-items of the rule, in a tree hierarchy and like their parent, you can set the symbology and the rule of each class. Automated rule refinement can be based on:

- **scales**: given a list of scales, this option creates a set of classes to which the different user-defined scale ranges apply. Each new scale-based class can have its own symbology and expression of definition. This can e.g. be a convenient way to display the same features with various symbols at different scales, or display only a set of features depending on the scale (e.g. local airports at large scale vs international airports at small scale).
- **categories**: applies a *categorized renderer* to the features falling in the selected rule.
- or **ranges**: applies a *graduated renderer* to the features falling in the selected rule.

Refined classes appear like sub-items of the rule, in a tree hierarchy and like above, you can set symbology of each class. Symbols of the nested rules are stacked on top of each other so be careful in choosing them. It is also possible to uncheck *Symbols* in the *Edit rule* dialog to avoid rendering a particular symbol in the stack.

In the *Edit rule* dialog, you can avoid writing all the rules and make use of the *Else* option to catch all the features that do not match any of the other rules, at the same level. This can also be achieved by writing `Else` in the *Rule* column of the *Layer Properties* ► *Symbology* ► *Rule-based* dialog.

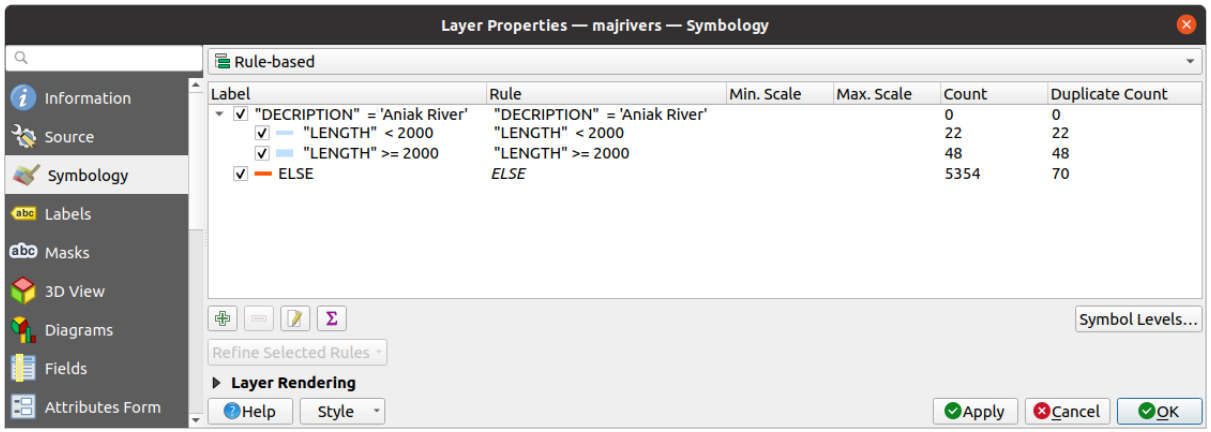
Right-clicking over selected item(s) shows a contextual menu to:

- *Copy* and *Paste*, a convenient way to create new item(s) based on existing item(s)
- *Copy Symbol* and *Paste Symbol*, a convenient way to apply the item's representation to others
- *Change Color…* of the selected symbol(s)
- *Change Opacity…* of the selected symbol(s)
- *Change Output Unit…* of the selected symbol(s)
- *Change Width…* of the selected line symbol(s)
- *Change Size…* of the selected point symbol(s)
- *Change Angle…* of the selected point symbol(s)
- *Refine Current Rule*: open a submenu that allows to refine the current rule with **scales**, **categories** or **Ranges**. Same as selecting the *corresponding menu* at the bottom of the dialog.

Unchecking a row in the rule-based renderer dialog hides in the map canvas the features of the specific rule and the nested ones.

The created rules also appear in a tree hierarchy in the map legend. Double-click an entry in the map legend to edit the assigned symbol. Right-click and you will get some *more options*.

The example in Сл.12.7 shows the rule-based rendering dialog for the rivers layer of the QGIS sample dataset.



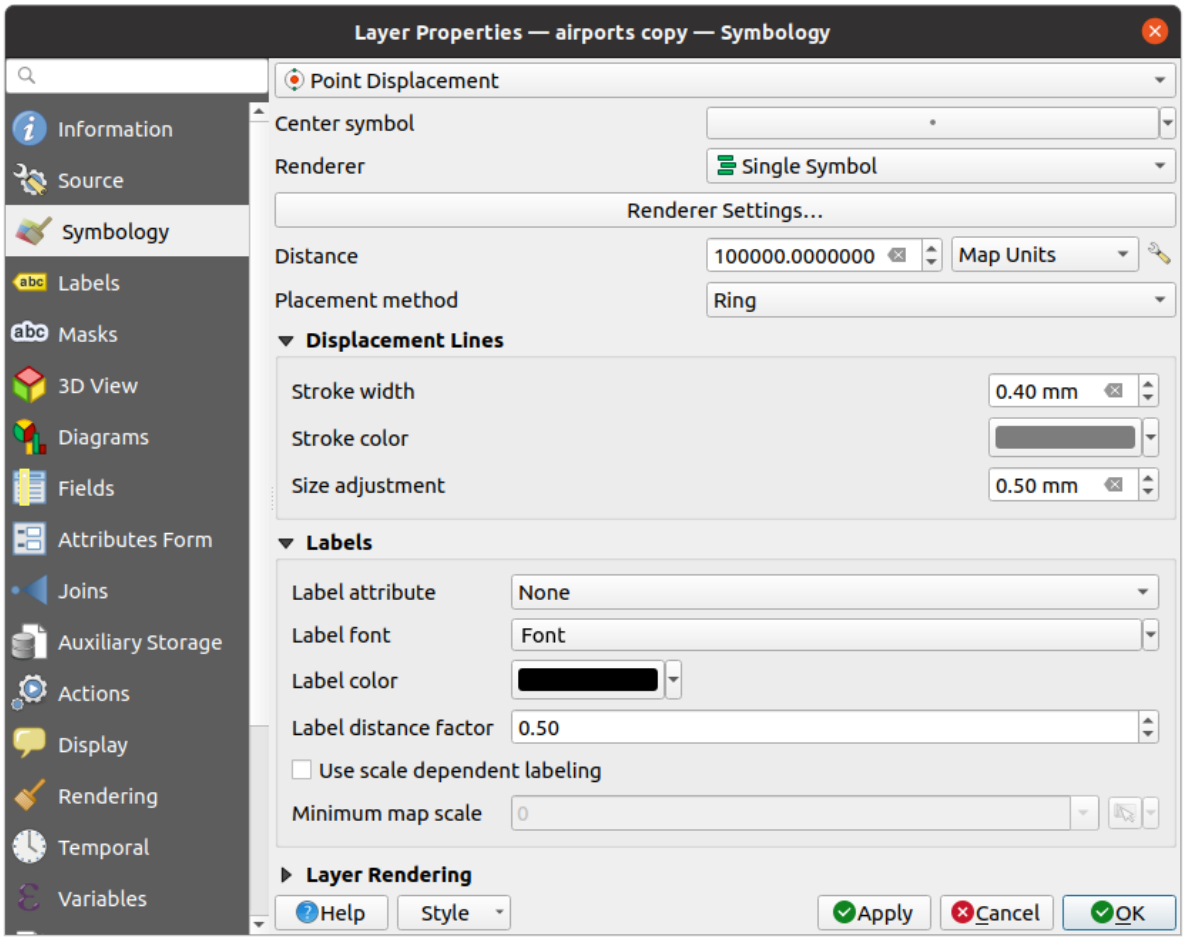
Слика12.7: Rule-based Symbolizing options

Point displacement Renderer

The *Point Displacement* renderer takes the point features falling in a given distance tolerance from each other and places their symbols around their barycenter, following different placement methods. This can be a convenient way to visualize all the features of a point layer, even if they have the same location (e.g. amenities in a building).

To configure a point displacement renderer, you have to:

1. Set the *Center symbol*: how the virtual point at the center will look like
2. Select the *Renderer* type: how you want to classify features in the layer (single, categorized, rule-based…)
3. Press the *Renderer Settings…* button to configure features' symbology according to the selected renderer
4. Indicate the *Distance* tolerance in which close features are considered overlapping and then displaced over the same virtual point. Supports common symbol units.
5. Configure the *Placement methods*:
   - **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. Displaced symbols are placed on the *Displacement lines*. While the minimal spacing of the displacement lines depends on the point symbols renderer, you can still customize some of their settings such as the *Stroke width*, *Stroke color* and *Size adjustment* (e.g., to add more spacing between the rendered points).
7. Use the *Labels* group options to perform points labeling: the labels are placed near the displaced symbol, and not at the feature real position.
   1. Select the *Label attribute*: a field of the layer to use for labeling
   2. Indicate the *Label font* properties and size
   3. Pick a *Label color*
   4. Set a *Label distance factor*: for each point feature, offsets the label from the symbol center proportionally to the symbol's diagonal size.
   5. Turn on *Use scale dependent labeling* if you want to display labels only on scales larger than a given *Minimum map scale*.



Слика12.8: Point displacement dialog

Белешка

Point Displacement renderer does not alter feature geometry, meaning that points are not moved from their position. They are still located at their initial place. Changes are only visual, for rendering purpose. Use instead the Processing *Pomeranje tačaka* algorithm if you want to create displaced features.

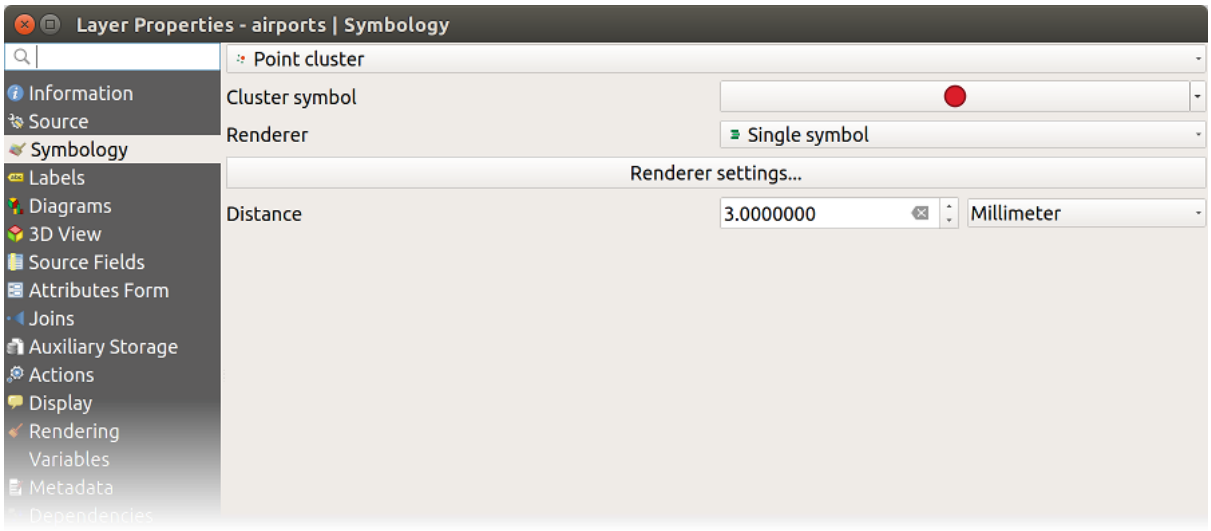
Point Cluster Renderer

Unlike the *Point Displacement* renderer which blows up nearest or overlaid point features placement, the *Point Cluster* renderer groups nearby points into a single rendered marker symbol. Points that fall within a specified distance from each others are merged into a single symbol. Points aggregation is made based on the closest group being formed, rather than just assigning them the first group within the search distance.

From the main dialog, you can:

1. Set the symbol to represent the point cluster in the *Cluster symbol*; the default rendering displays the number of aggregated features thanks to the `@cluster_size` *variable* on Font marker symbol layer.
2. Select the *Renderer* type, i.e. how you want to classify features in the layer (single, categorized, rule-based…)
3. Press the *Renderer Settings…* button to configure features' symbology as usual. Note that this symbology is only visible on features that are not clustered, the *Cluster symbol* being applied otherwise. Also, when all the point features in a cluster belong to the same rendering class, and thus would be applied the same color, that color represents the `@cluster_color` variable of the cluster.

4. Indicate the maximal *Distance* to consider for clustering features. Supports common symbol units.



Слика12.9: Point Cluster dialog

Белешка

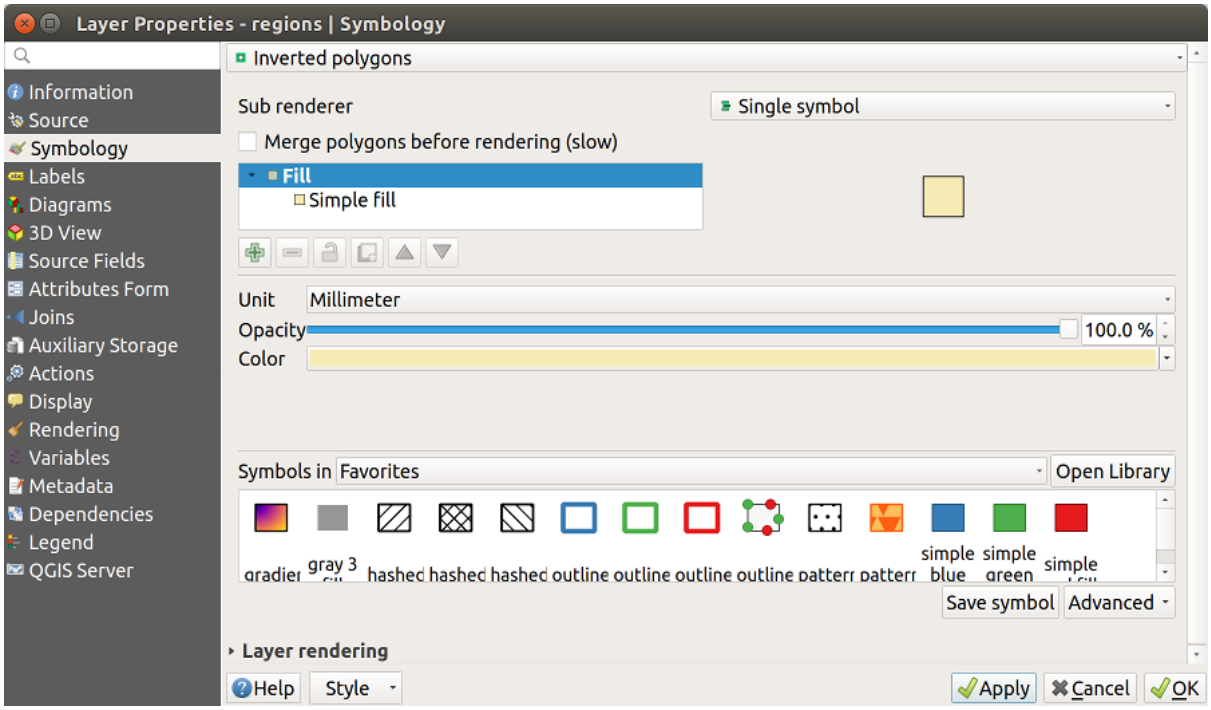
Point Cluster renderer does not alter feature geometry, meaning that points are not moved from their position. They are still located at their initial place. Changes are only visual, for rendering purpose. Use instead the Processing *K-means klasterovanje* or *DBSCAN klasterovanje* algorithm if you want to create cluster-based features.

Merged Features Renderer

The *Merged Features* renderer allows area and line features to be “dissolved” into a single object prior to rendering to ensure that complex symbols or overlapping features are represented by a uniform and contiguous cartographic symbol.

Inverted Polygon Renderer

The *Inverted Polygon* renderer allows user to define a symbol to fill in outside of the layer’s polygons. As above you can select subrenderers, namely Single symbol, Graduated, Categorized, Rule-Based or 2.5D renderer.



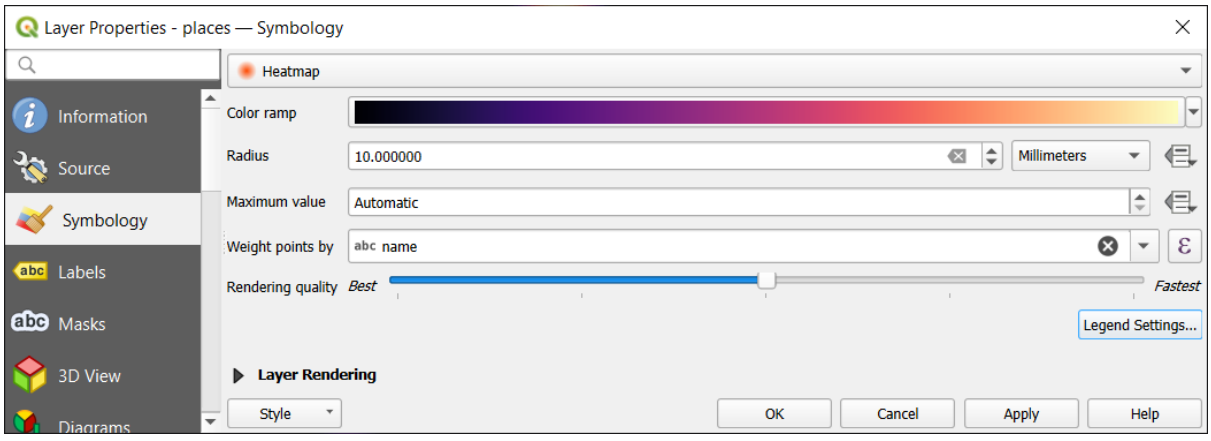
Слика12.10: Inverted Polygon dialog

Heatmap Renderer

With the *Heatmap* renderer you can create live dynamic heatmaps for (multi)point layers. You can specify the heatmap *Radius* in millimeters, points, pixels, map units or inches, choose and edit a *Color ramp* for the heatmap style and use a slider for selecting a trade-off between render speed and quality. You can also define a *Maximum value* limit and *Weight points by* using a field or an expression.

Use Data defined override to dynamically control *Radius* and *Maximum value* based on the attributes of your data. For example, the radius of a heatmap point could be determined by its population attribute, or the maximum value could be based on a temporal range.

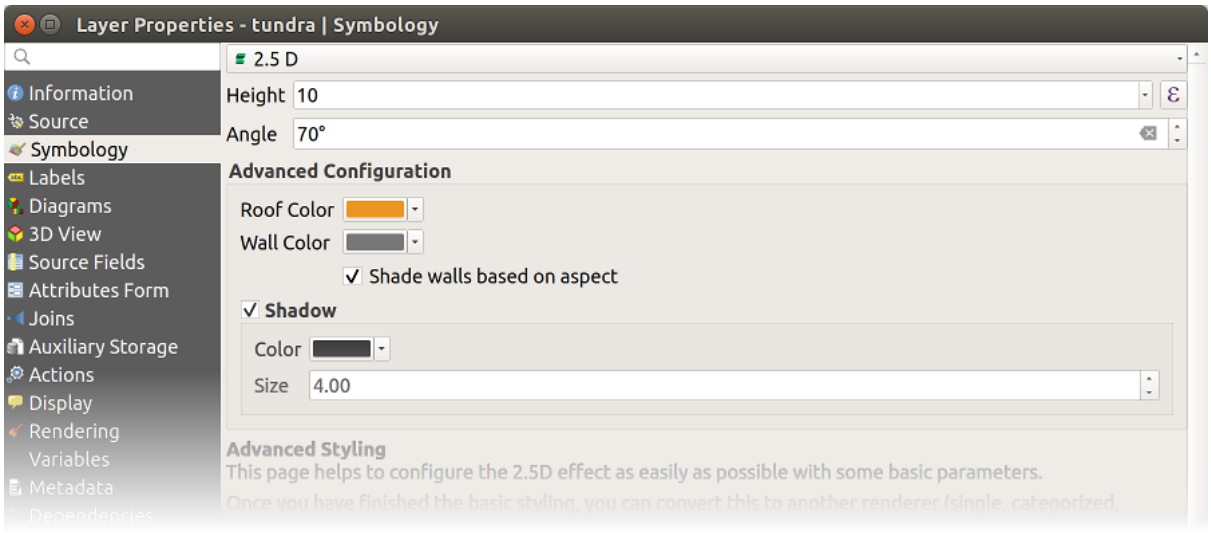
When adding or removing a feature the heatmap renderer updates the heatmap style automatically. The *Color ramp* will be shown as a legend bar and in the *Legend settings* you can set the *Labels* for the *Maximum* and *Minimum* values. You can also change the orientation and direction of the legend in the *Layout*.



Слика12.11: Heatmap dialog

2.5D Renderer

Using the *2.5D* renderer it's possible to create a 2.5D effect on your layer's features. You start by choosing a *Height* value (in map units). For that you can use a fixed value, one of your layer's fields, or an expression. You also need to choose an *Angle* (in degrees) to recreate the viewer position (0° means west, growing in counter clock wise). Use advanced configuration options to set the *Roof Color* and *Wall Color*. If you would like to simulate solar radiation on the features walls, make sure to check the *Shade walls based on aspect* option. You can also simulate a shadow by setting a *Color* and *Size* (in map units).



Слика12.12: 2.5D dialog

Савет

Using 2.5D effect with other renderers

Once you have finished setting the basic style on the 2.5D renderer, you can convert this to another renderer (single, categorized, graduated). The 2.5D effects will be kept and all other renderer specific options will be available for you to fine tune them (this way you can have for example categorized symbols with a nice 2.5D representation or add some extra styling to your 2.5D symbols). To make sure that the shadow and the „building" itself do not interfere with other nearby features, you may need to enable Symbols Levels (*Advanced* ► *Symbol levels…*). The 2.5D height and angle values are saved in the layer's variables, so you can edit it afterwards in the variables tab of the layer's properties dialog.

Embedded Renderer

The *Embedded Symbols* renderer allows to display the 'native' symbology of a provided datasource. This is mostly the case with `KML` and `TAB` datasets that have predefined symbology.

Layer rendering

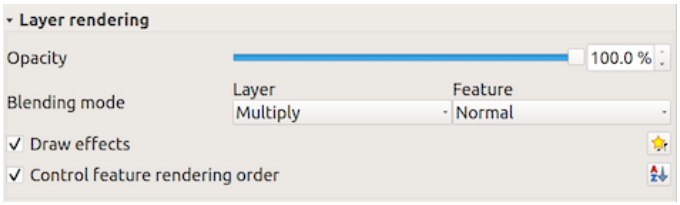
From the Symbology tab, you can also set some options that invariably act on all features of the layer:

- *Opacity*: You can make the underlying layer in the map canvas visible with this tool. Use the slider to adapt the visibility of your vector layer to your needs. You can also make a precise definition of the percentage of visibility in the menu beside the slider.
- *Blending mode* at the *Layer* and *Feature* levels: You can achieve special rendering effects with these tools that you may previously only know from graphics programs. The pixels of your overlaying and underlying layers are mixed through the settings described in *Blending Modes*.

- Apply *paint effects* on all the layer features with the *Draw Effects* button.
- *Control feature rendering order* allows you, using features attributes, to define the z-order in which they shall be rendered. Activate the checkbox and click on the button beside. You then get the *Define Order* dialog in which you:

  1. Choose a field or build an expression to apply to the layer features.
  2. Set in which order the fetched features should be sorted, i.e. if you choose **Ascending** order, the features with lower value are rendered under those with higher value.
  3. Define when features returning NULL value should be rendered: **first** (bottom) or **last** (top).
  4. Repeat the above steps as many times as rules you wish to use.

  The first rule is applied to all the features in the layer, z-ordering them according to their returned value. Then, within each group of features with the same value (including those with NULL value) and thus the same z-level, the next rule is applied to sort them. And so on…



Слика12.13: Layer rendering options

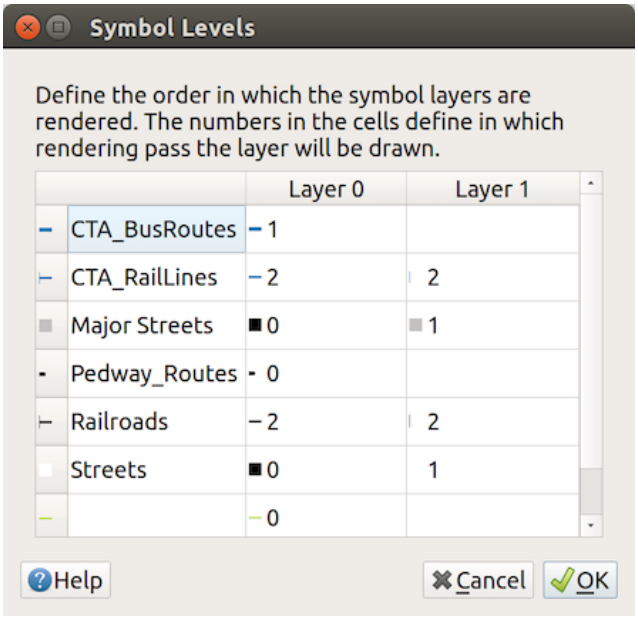
Other Settings

Symbol levels

For renderers that allow stacked symbol layers (only heatmap doesn't) there is an option to control the rendering order of each symbol's levels.

For most of the renderers, you can access the Symbols levels option by clicking the *Advanced* button below the saved symbols list and choosing *Symbol levels*. For the *Rule-based Renderer* the option is directly available through *Symbols Levels…* button, while for *Point displacement Renderer* renderer the same button is inside the *Rendering settings* dialog.

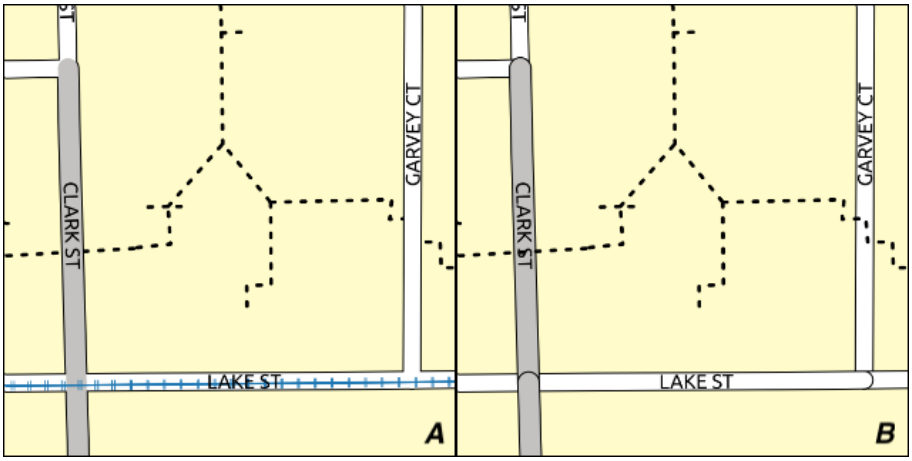
To activate symbols levels, select the *Enable symbol levels*. Each row will show up a small sample of the combined symbol, its label and the individual symbols layer divided into columns with a number next to it. The numbers represent the rendering order level in which the symbol layer will be drawn. Lower values levels are drawn first, staying at the bottom, while higher values are drawn last, on top of the others.



Слика12.14: Symbol levels dialog

Белешка

If symbols levels are deactivated, the complete symbols will be drawn according to their respective features order. Overlapping symbols will simply obfuscate to other below. Besides, similar symbols won't „merge" with each other.



Слика12.15: Symbol levels activated (A) and deactivated (B) difference

Data-defined size legend

When a layer is rendered with the *proportional symbol or the multivariate rendering* or when a *scaled size diagram* is applied to the layer, you can allow the display of the scaled symbols in both the *Layers panel* and the *print layout legend*.

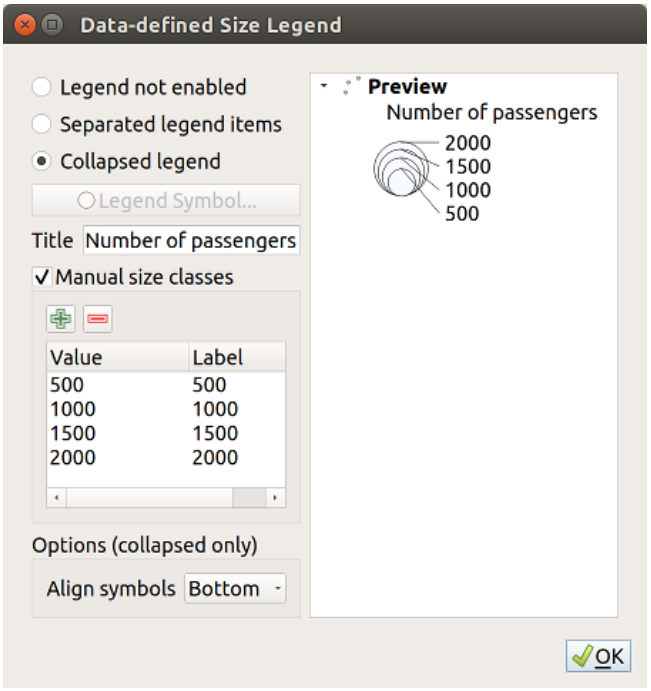
To enable the *Data-defined Size Legend* dialog to render symbology, select the eponym option in the *Advanced* button below the saved symbols list. For diagrams, the option is available under the *Legend* tab. The dialog provides the following options to:

- select the type of legend: *Legend not enabled*, *Separated legend items* and *Collapsed legend*. For

the latter option, you can select whether the legend items are aligned at the **Bottom** or at the **Center**;

- preview the *symbol to use* for legend representation;
- insert the title in the legend;
- resize the classes to use: by default, QGIS provides you with a legend of five classes (based on natural pretty breaks) but you can apply your own classification using the *Manual size classes* option. Use the and buttons to set your custom classes values and labels.
- For collapsed legend, it's possible to:
  - *Align symbols* in the center or the bottom
  - configure the horizontal leader *Line symbol* from the symbol to the corresponding legend text.

A preview of the legend is displayed in the right panel of the dialog and updated as you set the parameters.



Слика12.16: Setting size scaled legend

Белешка

Currently, data-defined size legend for layer symbology can only be applied to point layer using single, categorized or graduated symbology.

Animation settings

To allow any symbol to become an *animated symbol*, you can utilize *Animation settings* panel. In this panel, you can enable animation for the symbol and set a specific frame rate for the symbol's redrawing.

1. Start by going to the top symbol level and select *Advanced* menu in the bottom right of the dialog
2. Find *Animation settings* option
3. Check *Is Animated* to enable animation for the symbol
4. Configure the *Frame rate*, i.e. how fast the animation would be played

5. You can now use `@symbol_frame` variable in any sub-symbol data defined property in order to animate that property.

For example, setting the symbol's rotation to data defined expression `@symbol_frame % 360` will cause the symbol to rotate over time, with rotation speed dictated by the symbol's frame rate:

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. Start by going to the top symbol level and select *Advanced* menu in the bottom right of the dialog
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)`:

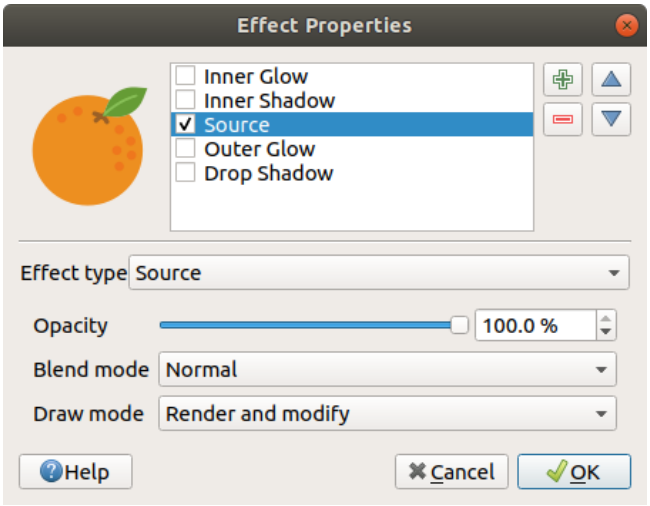
Draw effects

In order to improve layer rendering and avoid (or at least reduce) the resort to other software for final rendering of maps, QGIS provides another powerful functionality: the *Draw Effects* options, which adds paint effects for customizing the visualization of vector layers.

The option is available in the *Layer Properties* ► *Symbology* dialog, under the *Layer rendering* group (applying to the whole layer) or in *symbol layer properties* (applying to corresponding features). You can combine both usage.

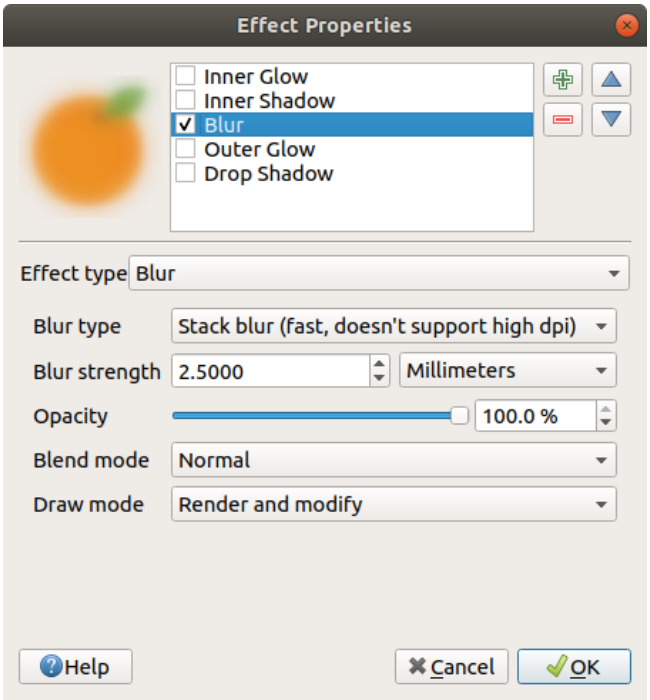
Paint effects can be activated by checking the *Draw effects* option and clicking the Customize effects button. That will open the *Effect Properties* Dialog (see Сл.12.17). The following effect types, with custom options are available:

- **Source**: Draws the feature's original style according to the configuration of the layer's properties. The *Opacity* of its style can be adjusted as well as the *Blend mode* and *Draw mode*. These are common properties for all types of effects.



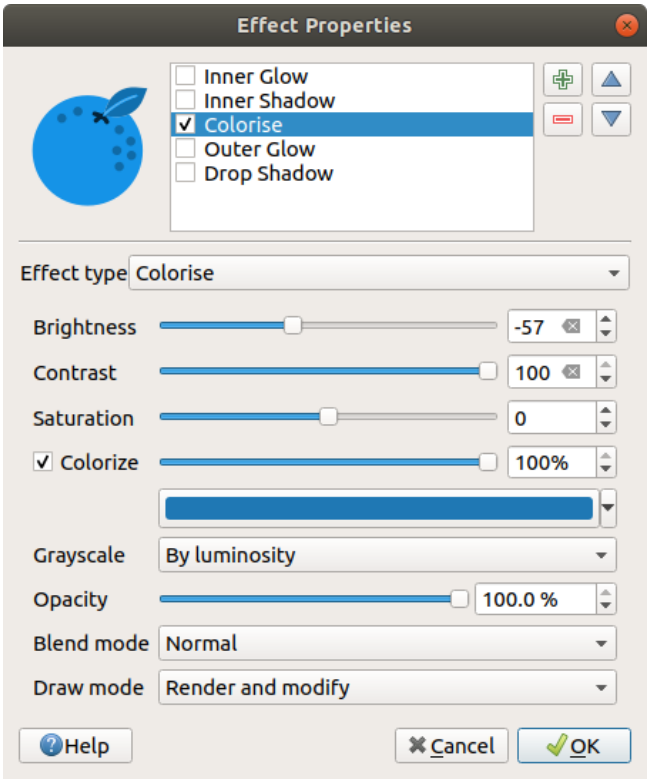
Слика12.17: Draw Effects: Source dialog

- **Blur**: Adds a blur effect on the vector layer. The custom options that you can change are the *Blur type* (*Stack blur (fast)* or *Gaussian blur (quality)*) and the *Blur strength*.



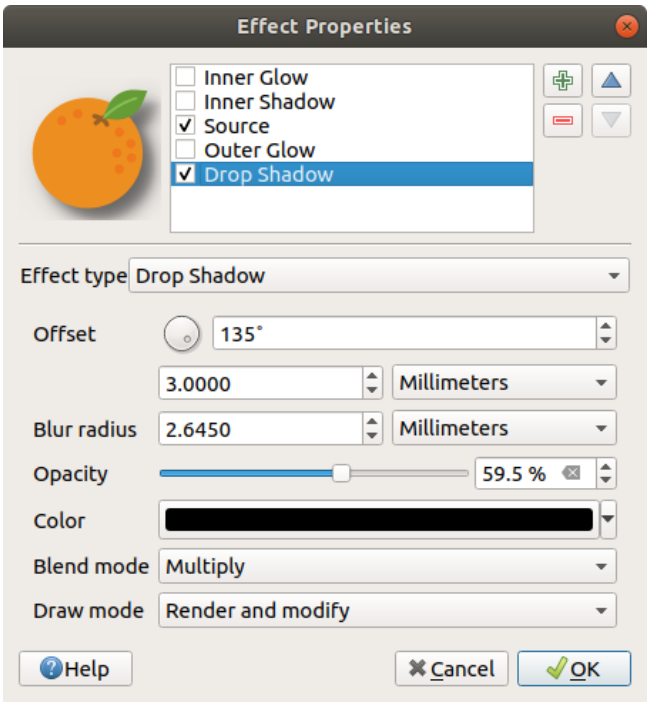
Слика12.18: Draw Effects: Blur dialog

- **Colorise**: This effect can be used to make a version of the style using one single hue. The base will always be a grayscale version of the symbol and you can:
  - Use the  *Grayscale* to select how to create it: options are 'By lightness', 'By luminosity', 'By average' and 'Off'.
  - If  *Colorise* is selected, it will be possible to mix another color and choose how strong it should be.
  - Control the *Brightness*, *Contrast* and *Saturation* levels of the resulting symbol.



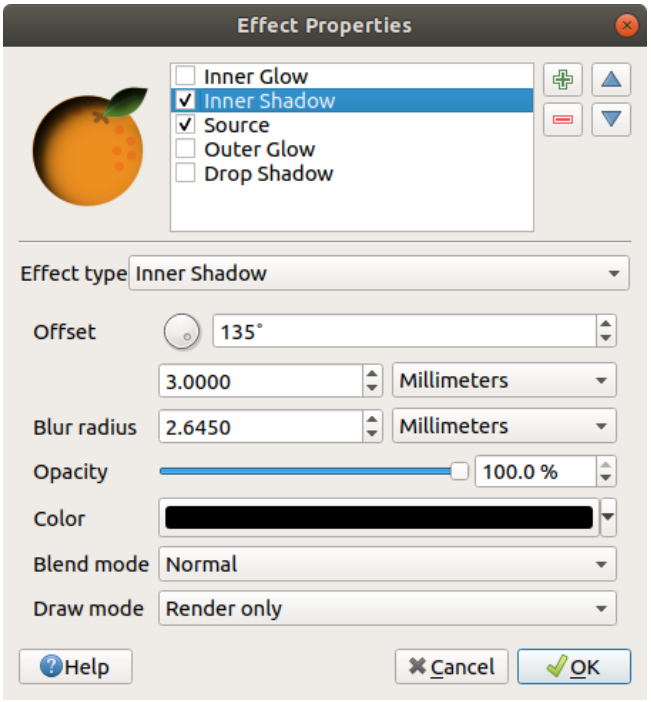
Слика12.19: Draw Effects: Colorize dialog

- **Drop Shadow**: Using this effect adds a shadow on the feature, which looks like adding an extra dimension. This effect can be customized by changing the *Offset* angle and distance, determining where the shadow shifts towards to and the proximity to the source object. *Drop Shadow* also has the option to change the *Blur radius* and the *Color* of the effect.



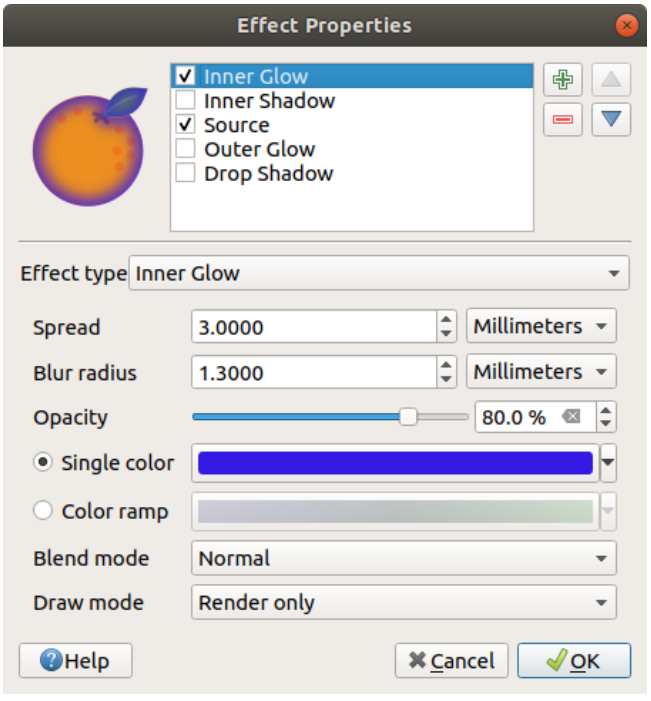
Слика12.20: Draw Effects: Drop Shadow dialog

- **Inner Shadow**: This effect is similar to the *Drop Shadow* effect, but it adds the shadow effect on the inside of the edges of the feature. The available options for customization are the same as the *Drop Shadow* effect.



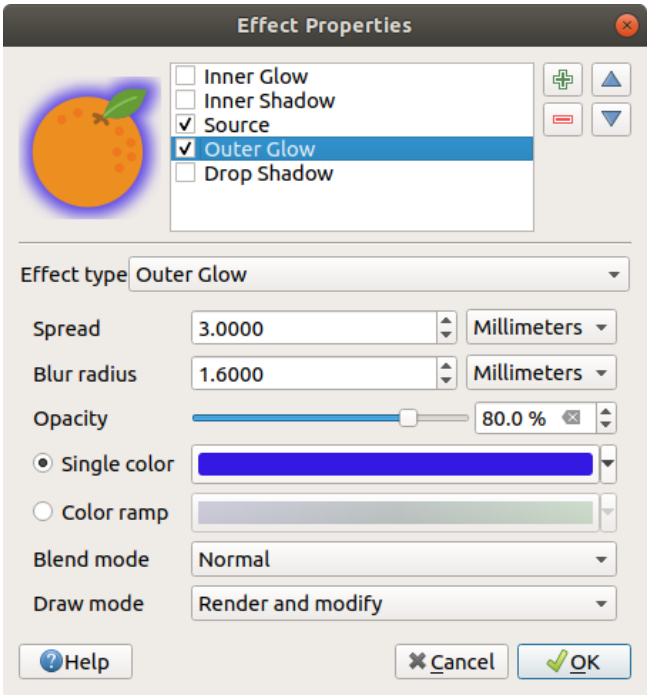
Слика12.21: Draw Effects: Inner Shadow dialog

- **Inner Glow**: Adds a glow effect inside the feature. This effect can be customized by adjusting the *Spread* (width) of the glow, or the *Blur radius*. The latter specifies the proximity from the edge of the feature where you want any blurring to happen. Additionally, there are options to customize the color of the glow using a *Single color* or a *Color ramp*.



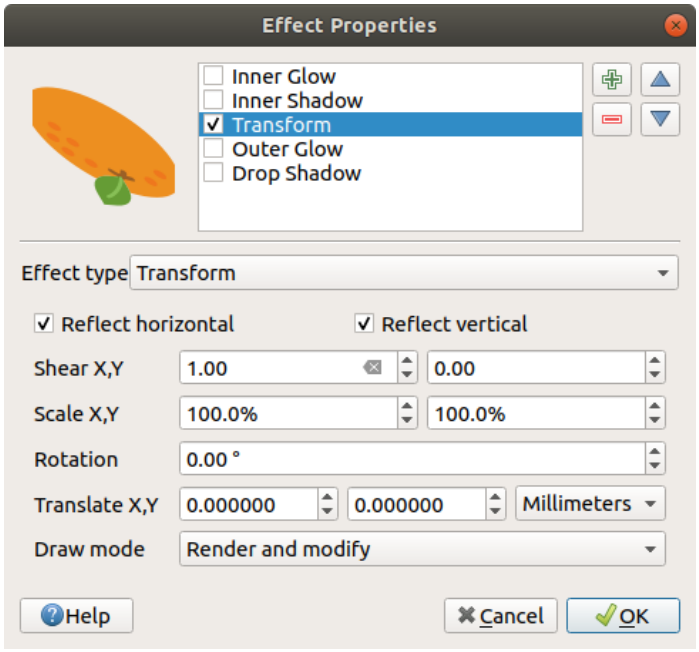
Слика12.22: Draw Effects: Inner Glow dialog

- **Outer Glow**: This effect is similar to the *Inner Glow* effect, but it adds the glow effect on the outside of the edges of the feature. The available options for customization are the same as the *Inner Glow* effect.



Слика12.23: Draw Effects: Outer Glow dialog

- **Transform**: Adds the possibility of transforming the shape of the symbol. The first options available for customization are the *Reflect horizontal* and *Reflect vertical*, which actually create a reflection on the horizontal and/or vertical axes. The other options are:
  - *Shear X,Y*: Slants the feature along the X and/or Y axis.
  - *Scale X,Y*: Enlarges or minimizes the feature along the X and/or Y axis by the given percentage.
  - *Rotation*: Turns the feature around its center point.
  - and *Translate X,Y* changes the position of the item based on a distance given on the X and/or Y axis.



Слика12.24: Draw Effects: Transform dialog

One or more effect types can be used at the same time. You (de)activate an effect using its checkbox in the effects list. You can change the selected effect type by using the *Effect type* option. You can reorder the effects using Move up and Move down buttons, and also add/remove effects using the Add new effect and Remove effect buttons.

There are some common options available for all draw effect types. *Opacity* and *Blend mode* options work similar to the ones described in *Layer rendering* and can be used in all draw effects except for the transform one.

There is also a *Draw mode* option available for every effect, and you can choose whether to render and/or modify the symbol, following some rules:

- Effects render from top to bottom.
- *Render only* mode means that the effect will be visible.
- *Modifier only* mode means that the effect will not be visible but the changes that it applies will be passed to the next effect (the one immediately below).
- The *Render and Modify* mode will make the effect visible and pass any changes to the next effect. If the effect is at the top of the effects list or if the immediately above effect is not in modify mode, then it will use the original source symbol from the layers properties (similar to source).

12.1.4 Labels Properties

The *Labels* properties provides you with all the needed and appropriate capabilities to configure smart labeling on vector layers. This dialog can also be accessed from the *Layer Styling* panel, or using the Layer Labeling Options button of the **Labels toolbar**.

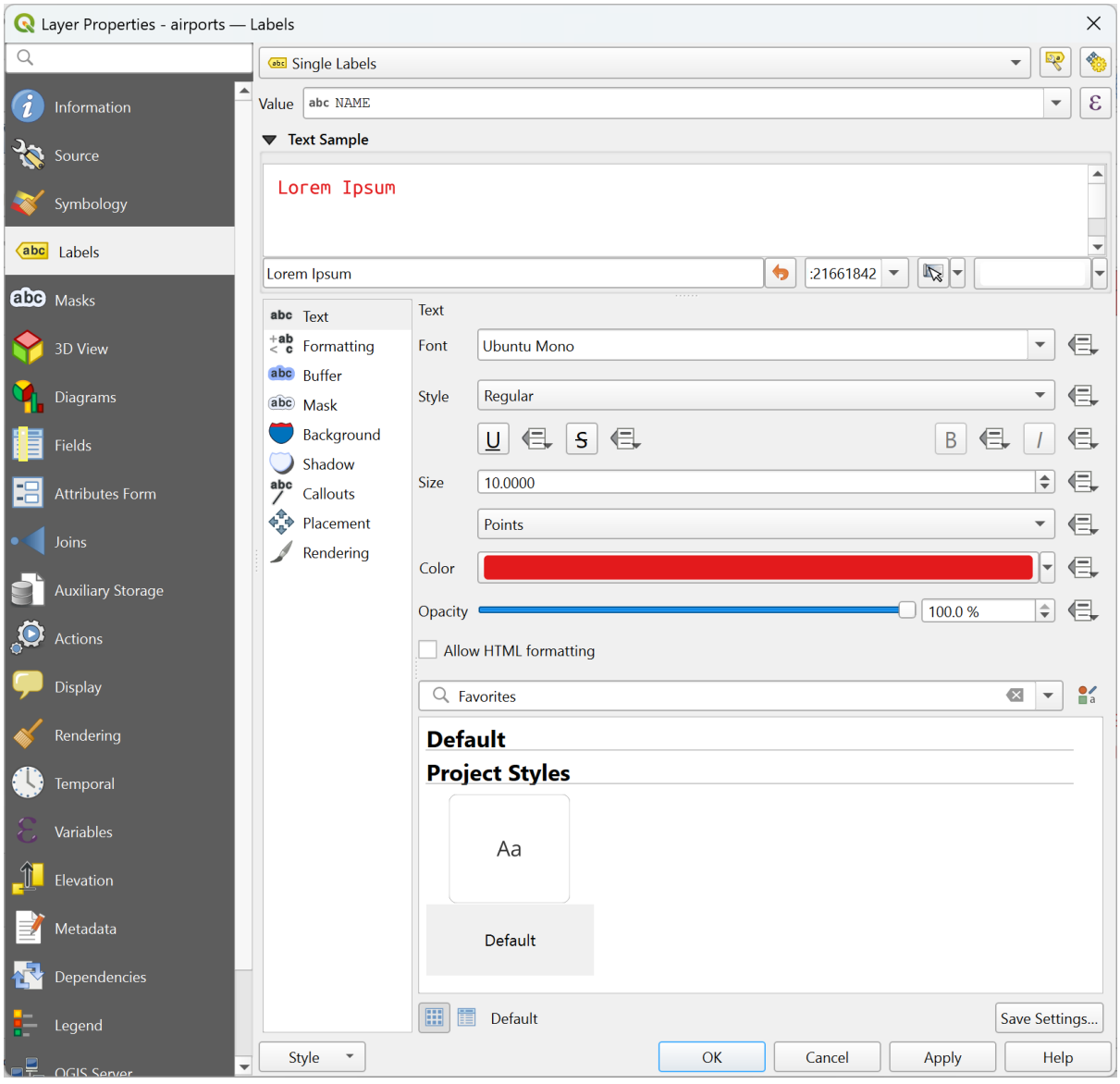
At the top of the dialog, you have:

- 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 *Configuring project labeling rules*.
- the Automated placement settings (applies to all layers) button: configure general properties on label placement and conflicts resolution. More details at *Setting the automated placement engine*.

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
- *Single labels*: Show labels on the map using a single attribute or an expression
- *Rule-based labeling*
- and *Blocking*: allows to set a layer as just an obstacle for other layer's labels without rendering any labels of its own.

The next steps assume you select the *Single labels* option, opening the following dialog.



Слика12.25: Layer labeling settings - Single labels

At the top of the dialog, a *Value* drop-down list is enabled. You can select an attribute column to use for labeling. By default, the *display field* is used. Click ε if you want to define labels based on expressions - See *Define labels based on expressions*.

Белешка

Labels with their formatting can be displayed as entries in the legends, if enabled in the *Legend* tab.

Below are displayed options to customize the labels, under various tabs:

- abc *Text*
- *Formatting*
- abc *Buffer*
- abc *Mask*
- *Background*
- *Shadow*
- abc *Callouts*
- *Placement*
- *Rendering*

Description of how to set each property is exposed at *Setting a label*.

Configuring project labeling rules

Pressing the Configure project labeling rules button next to the labeling method drop-down selector, you can create rules that controls how labels from a layer can interact with labels or features from another layer.

1. Press the Add rule button and in the drop-down menu, select one of the rule types:
   - *Prevent labels overlapping features*: prevents labels being placed overlapping features from a different layer.
   - *Pull labels towards features*: prevents labels being placed too far from features from a different layer. The maximum distance can be set in the unit of your choice.
   - *Push labels away from features*: prevents labels being placed too close to features from a different layer. The minimum distance can be set in the unit of your choice, as well as the *rule's priority* (The highest-priority rules are more important to respect in the event of a label placement conflict).
   - *Push labels away from other labels*: prevents labels being placed too close to labels from a different layer.

   Пажња

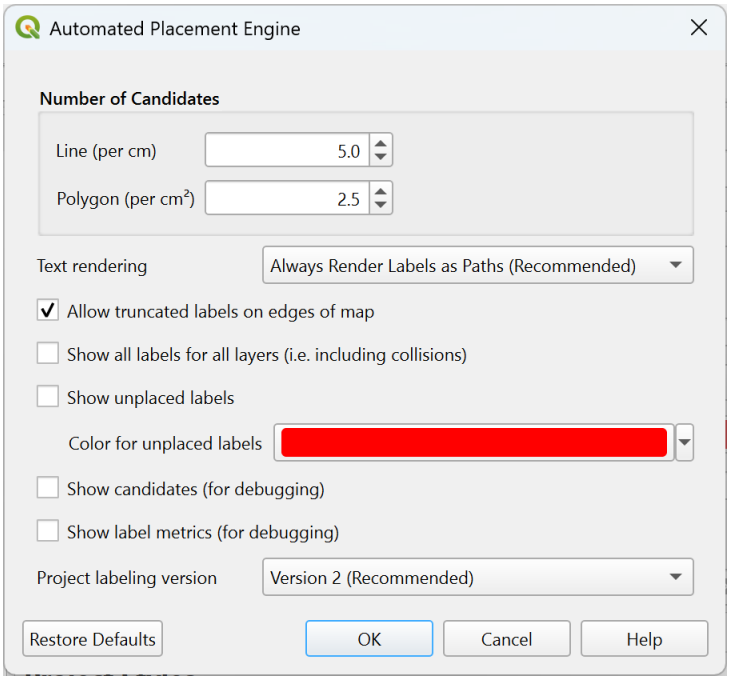
   The last three options are only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

2. Fill the properties at your will; you can provide a more meaningful name to the rule.
3. Press *OK*.
4. Add as many rules as necessary.
5. If necessary, press Edit rule to modify the selected rule or Remove rule to delete it from the project.

The set rules are available from any layer *Labels* properties tab, pressing the Configure project labeling rules button. You can temporary enable or disable any of them, using the checkbox next to the name. Hover over a rule to preview its details.

Setting the automated placement engine

You can use the automated placement settings to configure a project-level automated behavior of the labels. In the top right corner of the *Labels* tab, click the Automated placement settings (applies to all layers) button, opening a dialog with the following options:



Слика12.26: The labels automated placement engine

- *Number of candidates*: calculates and assigns to line and polygon features the number of possible labels placement based on their size. The longer or wider a feature is, the more candidates it has, and its labels can be better placed with less risk of collision.
- *Text rendering*: sets the default value for label rendering widgets when *exporting a map canvas* or *a layout* to PDF or SVG. If *Always render labels as text* is selected then labels can be edited in external applications (e.g. Inkscape) as normal text. BUT the side effect is that the rendering quality is decreased, and there are issues with rendering when certain text settings like buffers are in place. That's why *Always render labels as paths (recommended)* which exports labels as outlines but guarantees complete compatibility with the full range of formatting options available, is recommended. With *Prefer rendering labels as text* option, labels are rendered as text objects, unless doing so results in rendering artifacts or poor quality rendering (depending on text format settings).

  Белешка

  When rendering labels as text to a vector based device (e.g. PDF or SVG), care must be taken to ensure that the required fonts are available to users when opening the created files, or default fallback fonts will be used to display the output instead. (Although PDF exports MAY automatically embed some fonts when possible, depending on the user's platform).

- *Allow truncated labels on edges of map*: controls whether labels which fall partially outside of the map extent should be rendered. If checked, these labels will be shown (when there's no way to place them fully within the visible area). If unchecked then partially visible labels will be skipped. Note that this setting has no effects on labels' display in the *layout map item*.
- *Show all labels for all layers (i.e. including colliding objects)*. Note that this option can be also set per layer (see *Rendering tab*)

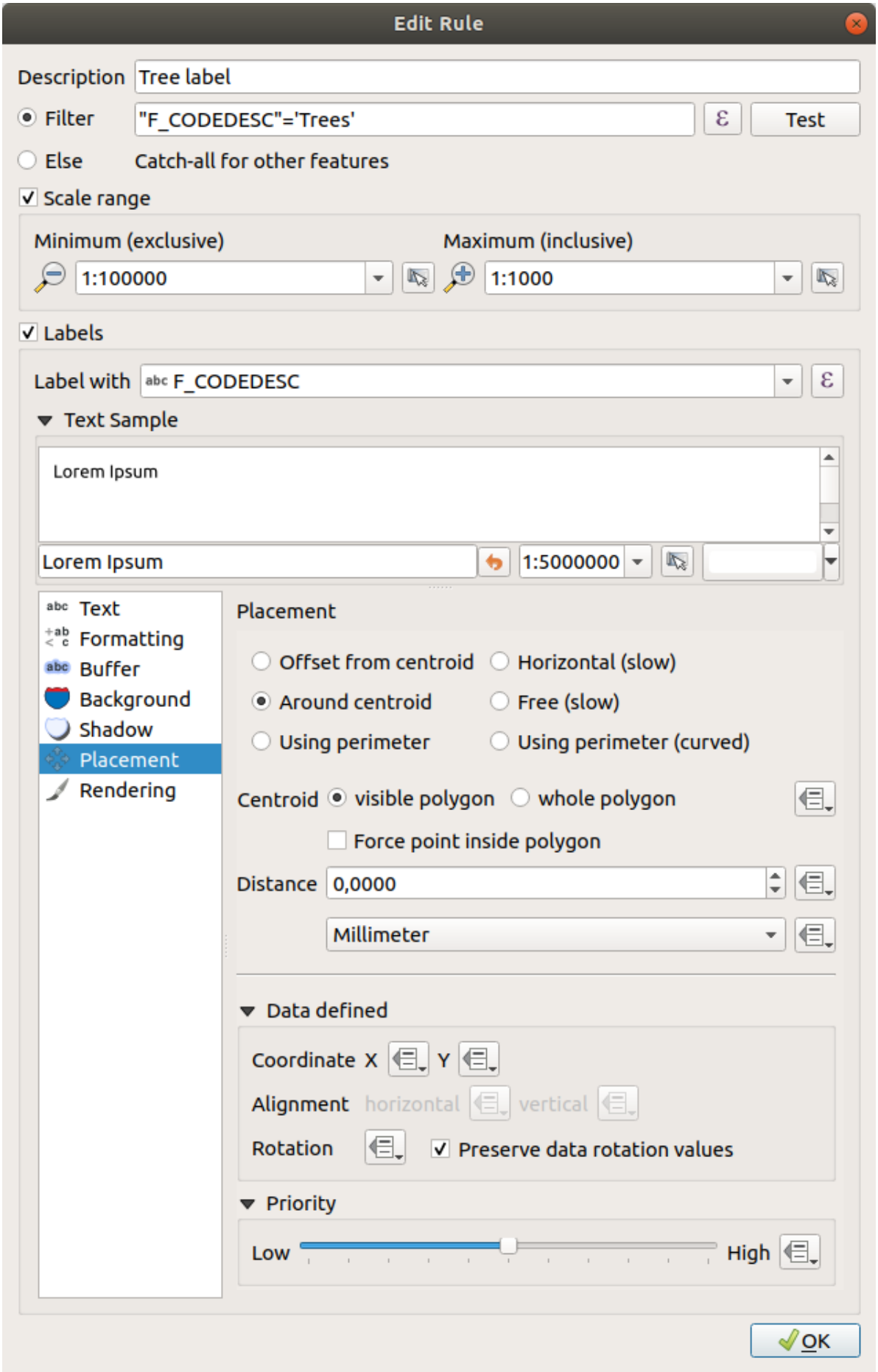
- *Show unplaced labels*: allows to determine whether any important labels are missing from the maps (e.g. due to overlaps or other constraints). They are displayed using a customizable color.
- *Show candidates (for debugging)*: controls whether boxes should be drawn on the map showing all the candidates generated for label placement. Like the label says, it's useful only for debugging and testing the effect different labeling settings have. This could be handy for a better manual placement with tools from the *label toolbar*.
- *Show label metrics (for debugging)*: displays the text bounds of the label in red and baselines in blue
- *Project labeling version*: QGIS supports two different versions of label automatic placement:
  - *Version 1*: the old system (used by QGIS versions 3.10 and earlier, and when opening projects created in these versions in QGIS 3.12 or later). Version 1 treats label and obstacle priorities as „rough guides" only, and it's possible that a low-priority label will be placed over a high-priority obstacle in this version. Accordingly, it can be difficult to obtain the desired labeling results when using this version and it is thus recommended only for compatibility with older projects.
  - *Version 2 (recommended)*: this is the default system in new projects created in QGIS 3.12 or later. In version 2, the logic dictating when labels are allowed to overlap *obstacles* has been reworked. The newer logic forbids any labels from overlapping any obstacles with a greater obstacle weight compared to the label's priority. As a result, this version results in much more predictable and easier to understand labeling results.

Rule-based labeling

With rule-based labeling multiple label configurations can be defined and applied selectively on the base of expression filters and scale range, as in *Rule-based rendering*.

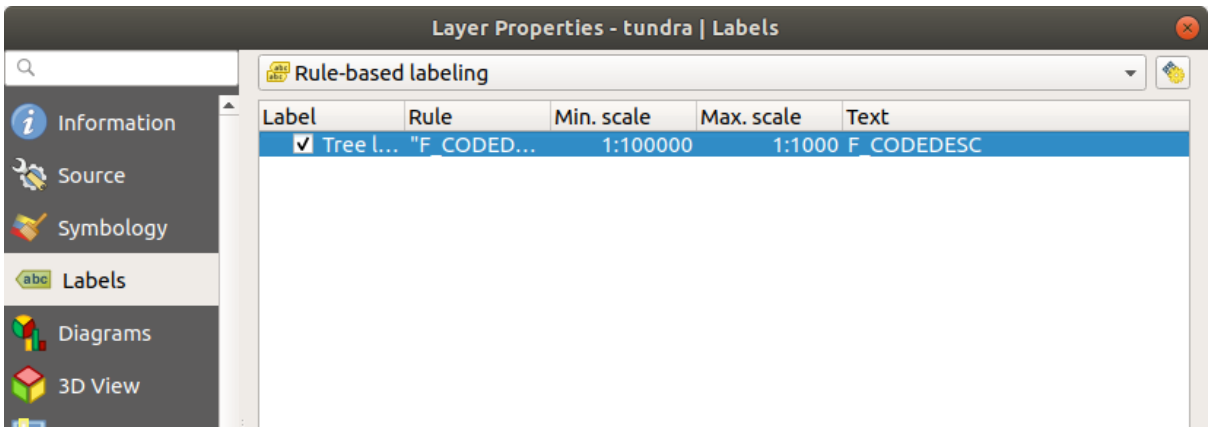
To create a rule:

1. Select the **Rule-based labeling** option in the main drop-down list from the *Labels* tab
2. Click the Add rule button at the bottom of the dialog.
3. Fill the new dialog with:
   - *Description*: a text used to identify the rule in the *Labels* tab and as a *label legend entry* in the print layout legend
   - *Filter*: an expression to select the features to apply the label settings to
   - If there are rules already set, the *Else* option can be used to select all the features not matching any filter of the rules in the same group.
4. You can set a *scale range* in which the label rule should be applied.
5. The options available under the *Labels* group box are the usual *label settings*. Configure them and press *OK*.



Слика12.27: Rule settings

A summary of existing rules is shown in the main dialog (see Сл.12.28). You can add multiple rules, reorder or imbricate them with a drag-and-drop. You can as well remove them with the button or edit them with button or a double-click.



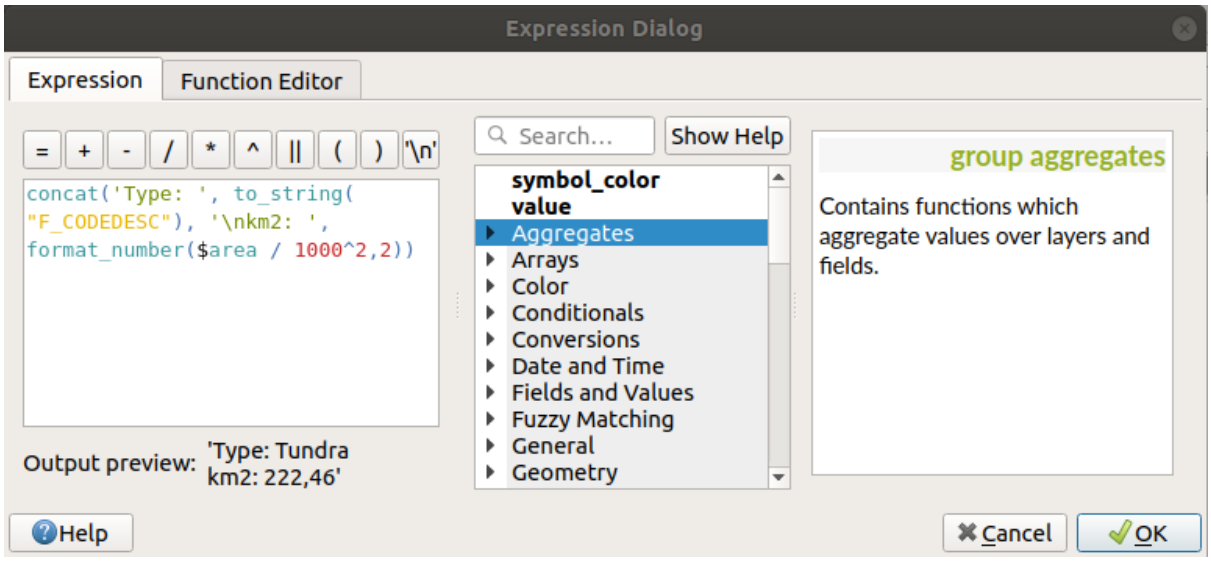
Слика12.28: Rule based labeling panel

Define labels based on expressions

Whether you choose single or rule-based labeling type, QGIS allows using expressions to label features.

Assuming you are using the *Single labels* method, click the button near the *Value* drop-down list in the *Labels* tab of the properties dialog.

In Сл.12.29, you see a sample expression to label the alaska trees layer with tree type and area, based on the field 'VEGDESC', some descriptive text, and the function `$area` in combination with `format_number()` to make it look nicer.



Слика12.29: Using expressions for labeling

Expression based labeling is easy to work with. All you have to take care of is that:

- You may need to combine all elements (strings, fields, and functions) with a string concatenation function such as `concat`, `+` or `||`. Be aware that in some situations (when null or numeric value are involved) not all of these tools will fit your need.

- Strings are written in 'single quotes'.
- Fields are written in „double quotes" or without any quote.

Let's have a look at some examples:

1. Label based on two fields 'name' and 'place' with a comma as separator:

```
"name" || ', ' || "place"
```

   Returns:

```
John Smith, Paris
```

2. Label based on two fields 'name' and 'place' with other texts:

```
'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")
```

   Returns:

```
My name is John Smith and I live in Paris
```

3. Label based on two fields 'name' and 'place' with other texts combining different concatenation functions:

```
concat('My name is ', name, ' and I live in ' || place)
```

   Returns:

```
My name is John Smith and I live in Paris
```

   Or, if the field 'place' is NULL, returns:

```
My name is John Smith
```

4. Multi-line label based on two fields 'name' and 'place' with a descriptive text:

```
concat('My name is ', "name", '\n' , 'I live in ' , "place")
```

   Returns:

```
My name is John Smith
I live in Paris
```

5. Label based on a field and the $area function to show the place's name and its rounded area size in a converted unit:

```
'The area of ' || "place" || ' has a size of '
|| round($area/10000) || ' ha'
```

   Returns:

```
The area of Paris has a size of 10500 ha
```

6. Create a CASE ELSE condition. If the population value in field *population* is <= 50000 it is a town, otherwise it is a city:

```
concat('This place is a ',
CASE WHEN "population" <= 50000 THEN 'town' ELSE 'city' END)
```

   Returns:

```
This place is a town
```

7. Display name for the cities and no label for the other features (for the „city" context, see example above):

```
CASE WHEN "population" > 50000 THEN "NAME" END
```

Returns:

```
Paris
```

As you can see in the expression builder, you have hundreds of functions available to create simple and very complex expressions to label your data in QGIS. See *Expressions* chapter for more information and examples on expressions.

Using data-defined override for labeling

With the Data defined override function, the settings for the labeling are overridden by entries in the attribute table or expressions based on them. This feature can be used to set values for most of the labeling options described above.

For example, using the Alaska QGIS sample dataset, let's label the `airports` layer with their name, based on their militarian `USE`, i.e. whether the airport is accessible to :

- military people, then display it in gray color, size 8;
- others, then show in blue color, size 10.

To do this, after you enabled the labeling on the `NAME` field of the layer (see *Setting a label*):

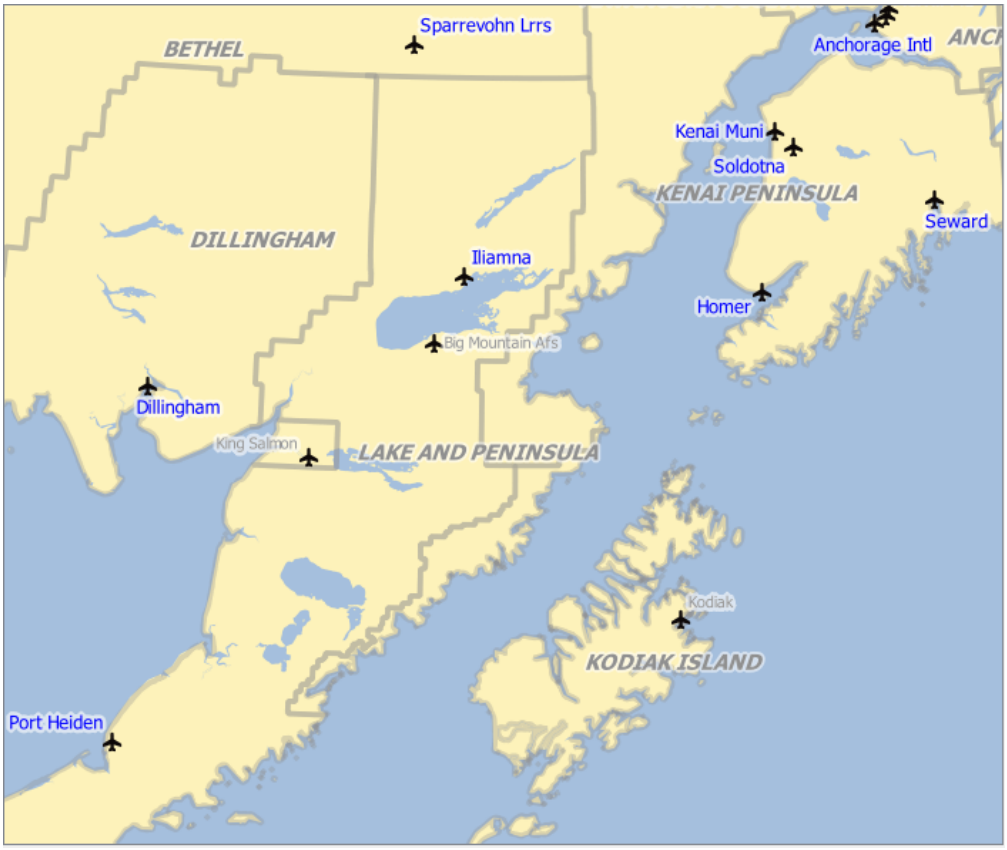
1. Activate the *Text* tab.
2. Click on the icon next to the *Size* property.
3. Select *Edit…* and type:

```
CASE
  WHEN "USE" like '%Military%' THEN 8 -- because compatible values are
↪'Military'
                                      -- and 'Joint Military/Civilian'
  ELSE 10
END
```

4. Press *OK* to validate. The dialog closes and the button becomes meaning that an rule is being run.
5. Then click the button next to the color property, type the expression below and validate:

```
CASE
  WHEN "USE" like '%Military%' THEN '150, 150, 150'
  ELSE '0, 0, 255'
END
```

Likewise, you can customize any other property of the label, the way you want. See more details on the Data-define override widget's description and manipulation in *Data defined override setup* section.



Слика12.30: Airports labels are formatted based on their attributes

Савет

Use the data-defined override to label every part of multi-part features

There is an option to set the labeling for multi-part features independently from your label properties. Choose the *Rendering*, `Feature options`, go to the Data-define override button next to the checkbox *Label every part of multipart-features* and define the labels as described in *Data defined override setup*.

The Label Toolbar

The *Label Toolbar* provides some tools to manipulate *label* (including their *callout*) or *diagram* properties:



Слика12.31: The Label toolbar

- Highlight Pinned Labels, Diagrams and Callouts. If the vector layer of the item is editable, then the highlighting is green, otherwise it's blue.
- Toggle Display of Unplaced Labels: Allows to determine whether any important labels are missing from the maps (e.g. due to overlaps or other constraints). They are displayed with a customizable color (see *Setting the automated placement engine*).
- Pin/Unpin Labels and Diagrams. By clicking or dragging an area, you pin overlaid items. If you click or drag an

area holding `Shift`, the items are unpinned. Finally, you can also click or drag an area holding `Ctrl` to toggle their pin status.

- Show/Hide Labels and Diagrams. If you click on the items, or click and drag an area holding `Shift`, they are hidden. When an item is hidden, you just have to click on the feature to restore its visibility. If you drag an area, all the items in the area will be restored.
- Move a Label, Diagram or Callout: click to select the item and click to move it to the desired place. The new coordinates are stored in *auxiliary fields*. Selecting the item with this tool and hitting the `Delete` key will delete the stored position value.
- Rotate a Label. Click to select the label and click again to apply the desired rotation. Likewise, the new angle is stored in an auxiliary field. Selecting a label with this tool and hitting the `Delete` key will delete the rotation value of this label.
- Change Label Properties. It opens a dialog to change the clicked label properties; it can be the label itself, its coordinates, angle, font, size, multiline alignment … as long as this property has been mapped to a field. Here you can set the option to *Label every part of a feature*.

Упозорење

Label tools overwrite current field values

Using the *Label toolbar* to customize the labeling actually writes the new value of the property in the mapped field. Hence, be careful to not inadvertently replace data you may need later!

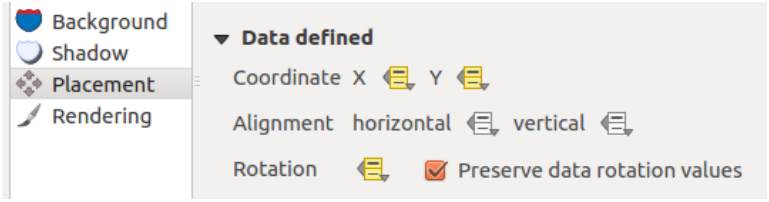
Белешка

The *Auxiliary Storage Properties* mechanism may be used to customize labeling (position, and so on) without modifying the underlying data source.

Customize the labels from the map canvas

Combined with the *Label Toolbar*, the data defined override setting helps you manipulate labels in the map canvas (move, edit, rotate). We now describe an example using the data-defined override function for the Move Label, Diagram or Callout function (see Сл.12.32).

1. Import `lakes.shp` from the QGIS sample dataset.
2. Double-click the layer to open the Layer Properties. Click on *Labels* and *Placement*. Select *Offset from centroid*.
3. Look for the *Data defined* entries. Click the icon to define the field type for the *Coordinate*. Choose `xlabel` for X and `ylabel` for Y. The icons are now highlighted in yellow.

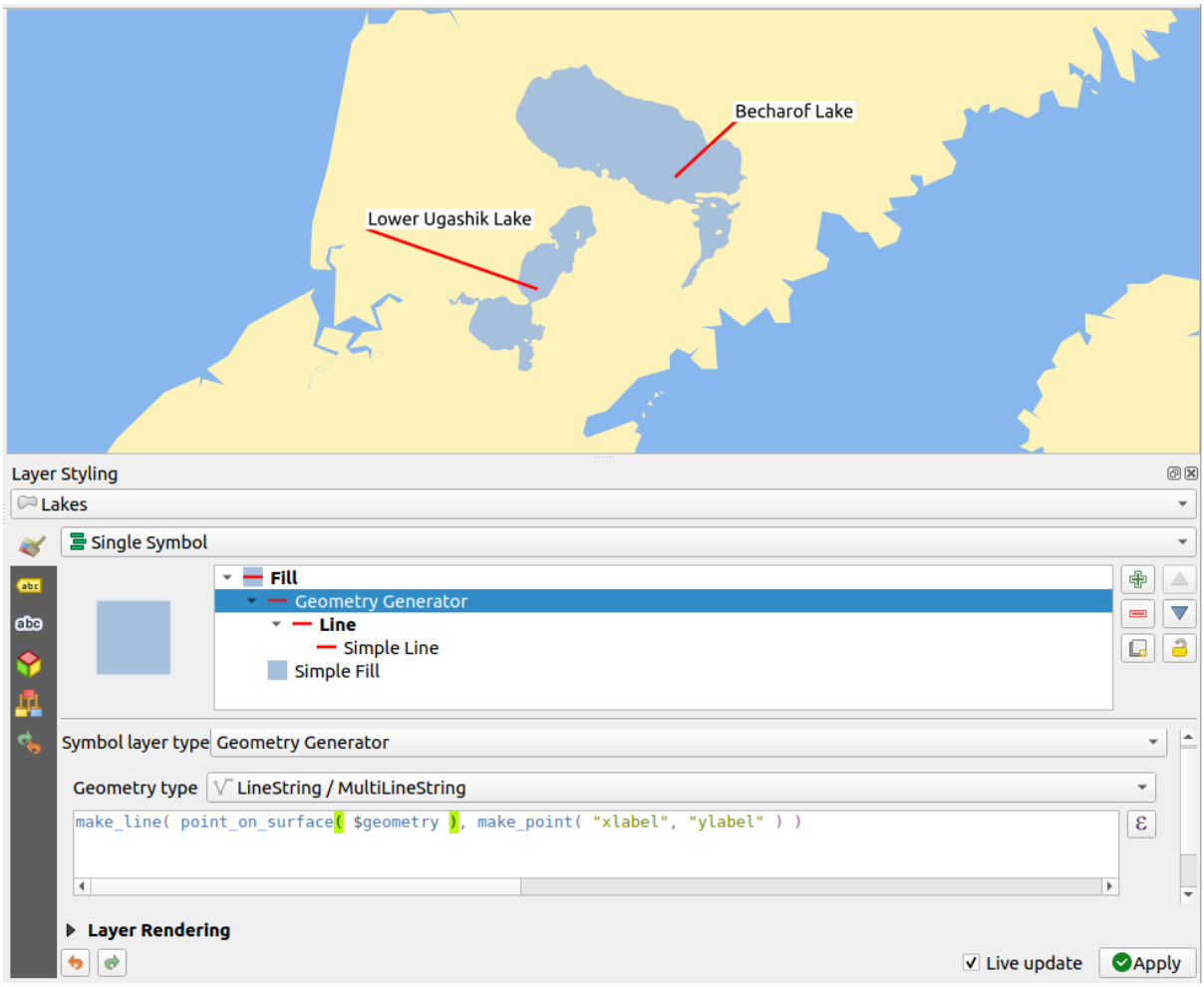


Слика12.32: Labeling of vector polygon layers with data-defined override

4. Zoom into a lake.

5. Set editable the layer using the Toggle Editing button.
6. Go to the Label toolbar and click the icon. Now you can shift the label manually to another position (see Сл.12.33). The new position of the label is saved in the `xlabel` and `ylabel` columns of the attribute table.
7. It's also possible to add a line connecting each lake to its moved label using:
   - the label's *callout property*
   - or the *geometry generator symbol layer* with the expression below:

```
make_line( centroid( $geometry ), make_point( "xlabel", "ylabel" ) )
```



Слика12.33: Moved labels

Белешка

The *Auxiliary Storage Properties* mechanism may be used with data-defined properties without having an editable data source.

12.1.5 Diagrams Properties

The *Diagrams* tab allows you to add a graphic overlay to a vector layer (see Сл.12.34). This dialog can also be accessed from the *Layer Styling* panel, or using the Layer Diagram Options button of the **Labels toolbar**.

The current core implementation of diagrams provides support for:

- *No diagrams*: the default value with no diagram displayed over the features;
- *Pie chart*, a circular statistical graphic divided into slices to illustrate numerical proportion. The arc length of each slice is proportional to the quantity it represents;
- *Text diagram*, a horizontally divided circle showing statistic values inside;
- *Histogram*, bars of varying colors for each attribute aligned next to each other;
- *Stacked bars*, stacks bars of varying colors for each attribute on top of each other vertically or horizontally;
- *Stacked diagram*, stacks diagrams of equal or varying types, next to each other, vertically or horizontally. More details at *Stacked Diagrams*.

In the top right corner of the *Diagrams* tab, the Automated placement settings (applies to all layers) button provides means to control diagram *labels placement* on the map canvas.

Савет

Switch quickly between types of diagrams

Given that the settings are almost common to the different types of diagram, when designing your diagram, you can easily change the diagram type and check which one is more appropriate to your data without any loss.

For each type of diagram, the properties are divided into several tabs:

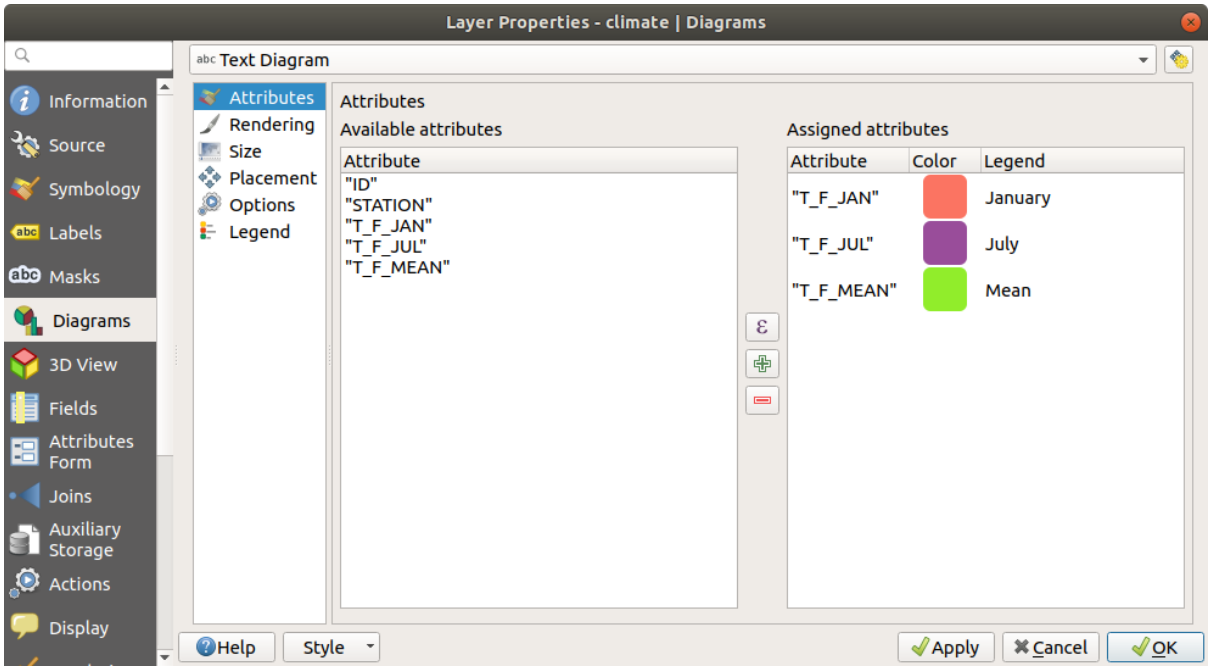
- *Attributes*
- *Rendering*
- *Size*
- *Placement*
- *Options*
- *Legend*

Attributes

Attributes defines which variables to display in the diagram. Use add item button to select the desired fields into the 'Assigned Attributes' panel. Generated attributes with *Expressions* can also be used.

You can move up and down any row with click and drag, sorting how attributes are displayed. You can also change the label in the 'Legend' column or the attribute color by double-clicking the item.

This label is the default text displayed in the legend of the print layout or of the layer tree.



Слика12.34: Diagram properties - Attributes tab

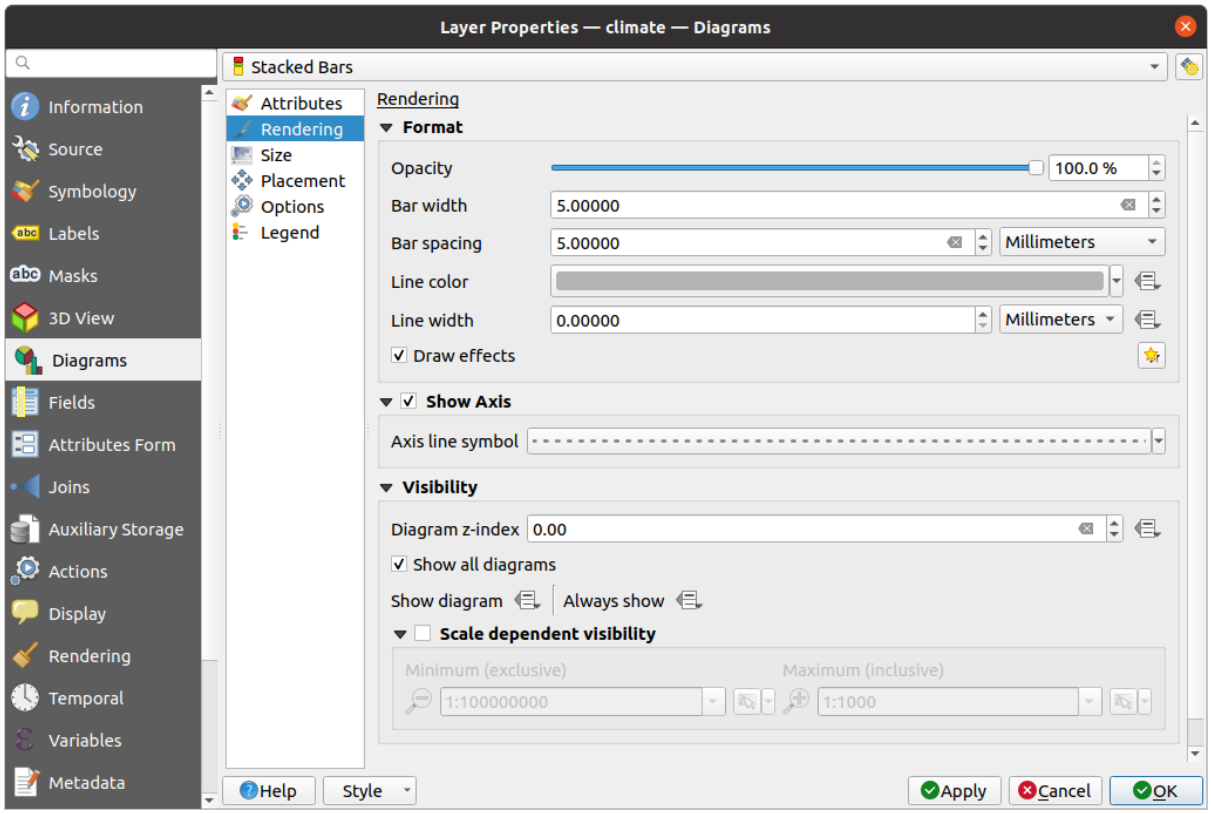
Rendering

Rendering defines how the diagram looks like. It provides general settings that do not interfere with the statistic values such as:

- the graphic's opacity, its outline width and color;
- depending on the type of diagram:
  - for histogram and stacked bars, the width of the bar and the spacing between the bars. You may want to set the spacing to `0` for stacked bars. Moreover, the *Axis line symbol* can be made visible on the map canvas and customized using *line symbol properties*.
  - for text diagram, the circle background color and the *font* used for texts;
  - for pie charts, the *Start angle* of the first slice and their *Direction* (clockwise or not).
- the use of *paint effects* on the graphics.

In this tab, you can also manage and fine tune the diagram visibility with different options:

- *Diagram z-index*: controls how diagrams are drawn on top of each other and on top of labels. A diagram with a high index is drawn over other diagrams and labels;
- *Show all diagrams*: shows all the diagrams even if they overlap each other;
- *Show diagram*: allows only specific diagrams to be rendered;
- *Always Show*: selects specific diagrams to always render, even when they overlap other diagrams or map labels;
- setting the *Scale dependent visibility*;

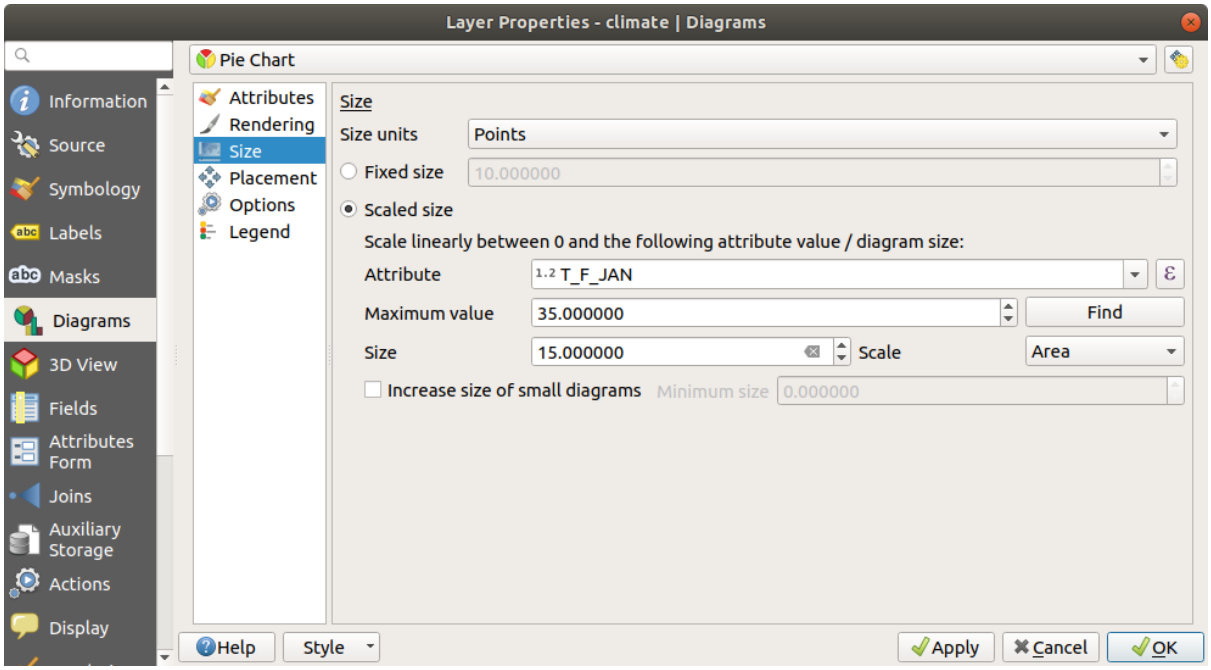


Слика12.35: Diagram properties - Rendering tab

Size

Size is the main tab to set how the selected statistics are represented. The diagram size *units* can be 'Millimeters', 'Points', 'Pixels', 'Map Units' or 'Inches'. You can use:

- *Fixed size*, a unique size to represent the graphic of all the features (not available for histograms)
- or *Scaled size*, based on an expression using layer attributes:
  1. In *Attribute*, select a field or build an expression
  2. Press *Find* to return the *Maximum value* of the attribute or enter a custom value in the widget.
  3. For histogram and stacked bars, enter a *Bar length* value, used to represent the *Maximum value* of the attributes. For each feature, the bar length will then be scaled linearly to keep this matching.
  4. For pie chart and text diagram, enter a *Size* value, used to represent the *Maximum value* of the attributes. For each feature, the circle area or diameter will then be scaled linearly to keep this matching (from `0`). A *Minimum size* can however be set for small diagrams.



Слика12.36: Diagram properties - Size tab

Placement

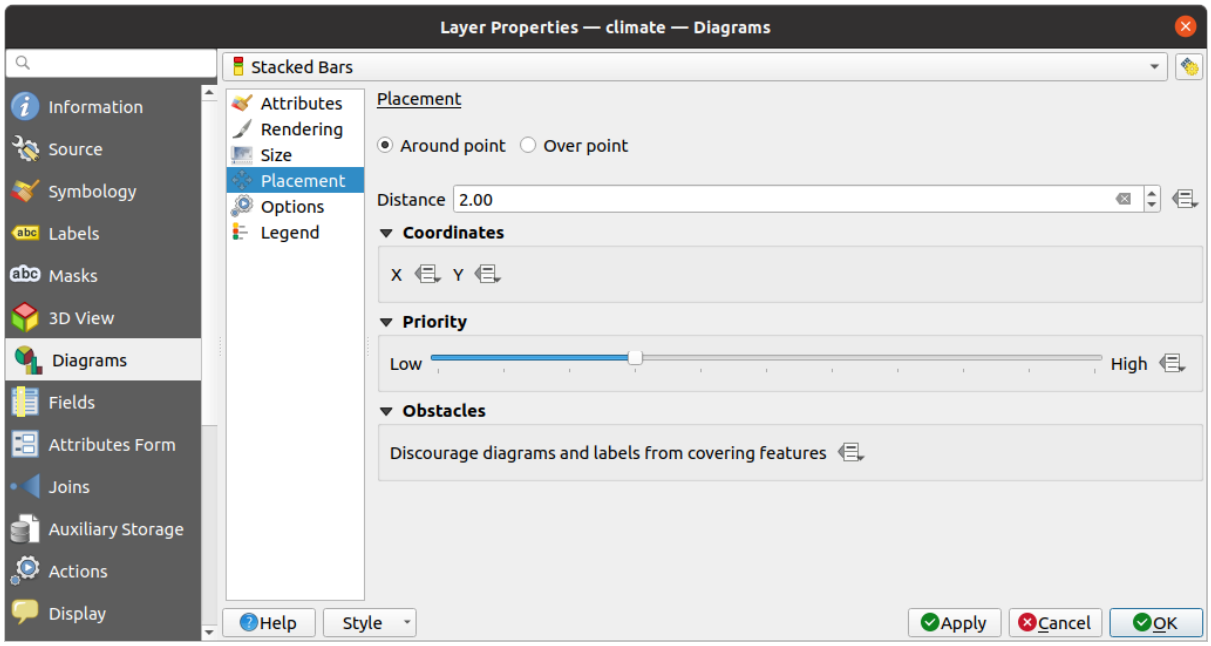
Placement defines the diagram position. Depending on the layer geometry type, it offers different options for the placement (more details at *Placement*):

- *Around point* or *Over point* for point geometry. The former variable requires a radius to follow.
- *Around line* or *Over line* for line geometry. Like point feature, the first variable requires a distance to respect and you can specify the diagram placement relative to the feature ('above', 'on' and/or 'below' the line) It's possible to select several options at once. In that case, QGIS will look for the optimal position of the diagram. Remember that you can also use the line orientation for the position of the diagram.
- *Around centroid* (at a set *Distance*), *Over centroid*, *Using perimeter* and *Inside polygon* are the options for polygon features.

The *Coordinate* group provides direct control on diagram placement, on a feature-by-feature basis, using their attributes or an expression to set the *X* and *Y* coordinate. The information can also be filled using the *Move labels and diagrams* tool.

In the *Priority* section, you can define the placement priority rank of each diagram, i.e. if there are different diagrams or labels candidates for the same location, the item with the higher priority will be displayed and the others could be left out.

Discourage diagrams and labels from covering features defines features to use as *obstacles*, i.e. QGIS will try to not place diagrams nor labels over these features. The priority rank is then used to evaluate whether a diagram could be omitted due to a greater weighted obstacle feature.



Слика12.37: Vector properties dialog with diagram properties, Placement tab

Options

The *Options* tab has settings for histograms and stacked bars. You can choose whether the *Bar orientation* should be *Up*, *Down*, *Right* or *Left*, for horizontal and vertical diagrams.

Legend

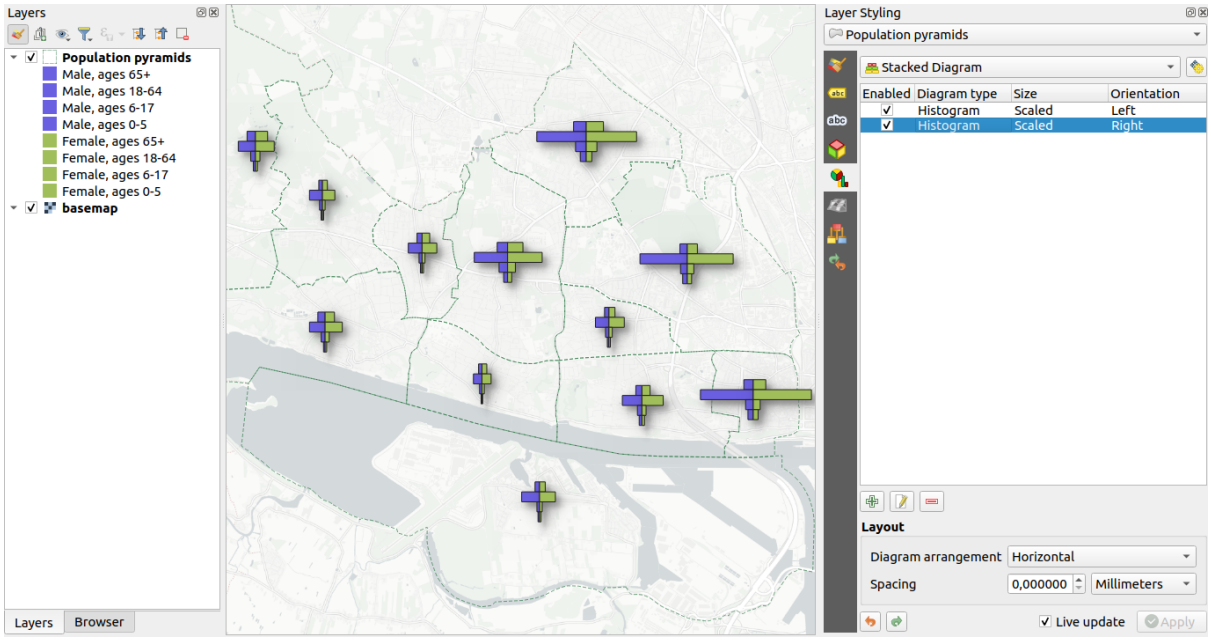
From the *Legend* tab, you can choose to display items of the diagram in the *Layers panel*, and in the *print layout legend*, next to the layer symbology:

- check *Show legend entries for diagram attributes* to display in the legends the `Color` and `Legend` properties, as previously assigned in the *Attributes* tab;
- and, when a *scaled size* is being used for the diagrams, push the *Legend Entries for Diagram Size…* button to configure the diagram symbol aspect in the legends. This opens the *Data-defined Size Legend* dialog whose options are described in *Data-defined size legend*.

When set, the diagram legend items (attributes with color and diagram size) are also displayed in the print layout legend, next to the layer symbology.

Stacked Diagrams

Stacked diagrams allow users to create complex diagrams like population pyramids, where two subdiagrams, namely histograms, are located side by side and displayed horizontally.



Слика12.38: Population pyramids built for each layer feature

Multi-temporal diagrams can also be constructed as stacked diagrams. The number of subdiagrams, as well as the spacing between them can be configured.

Moreover, subdiagrams can have different types (e.g., a pie chart alongside a histogram) and have their own independent settings like *Attributes*, *Rendering*, *Size*, *Options* and *Legend*.

Placement settings in a stacked diagram, as well as some visibility settings (located in the *Rendering* tab), are determined by the placement and visibility settings of the first subdiagram in the stack.

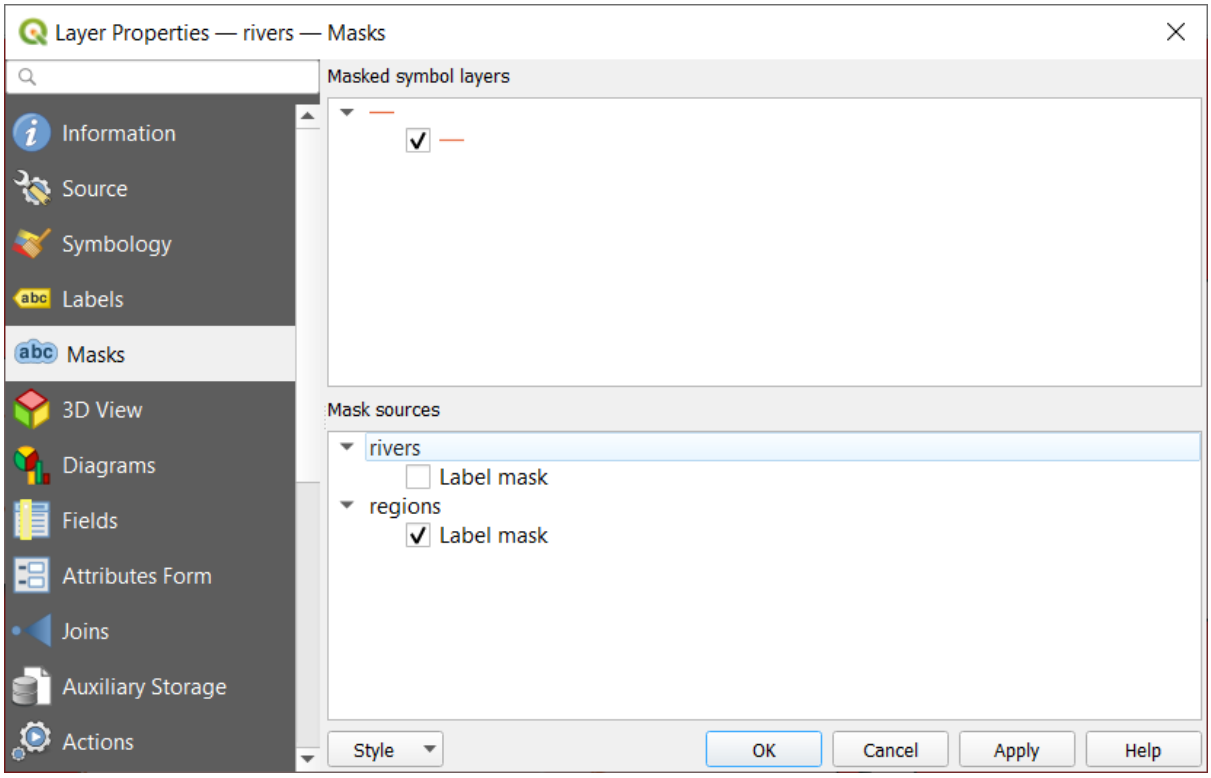
Finally, subdiagram ordering is given by the item ordering in the Stacked Diagram's list. The first subdiagram appears to the left in a horizontal stacked diagram, or in the upper part of a vertical one.

12.1.6 Masks Properties

The *Masks* tab helps you configure the current layer symbols overlay with other symbol layers or labels, from any layer. This is meant to improve the readability of symbols and labels whose colors are close and can be hard to decipher when overlapping; it adds a custom and transparent mask around the items to „hide" parts of the symbol layers of the current layer.

To apply masks on the active layer, you first need to enable in the project either *mask symbol layers* or *mask labels*. Then, from the *Masks* tab, check:

- the *Masked symbol layers*: lists in a tree structure all the symbol layers of the current layer. There you can select the symbol layer item you would like to transparently „cut out" when they overlap the selected mask sources
- the *Mask sources* tab: list all the mask labels and mask symbol layers defined in the project. Select the items that would generate the mask over the selected masked symbol layers



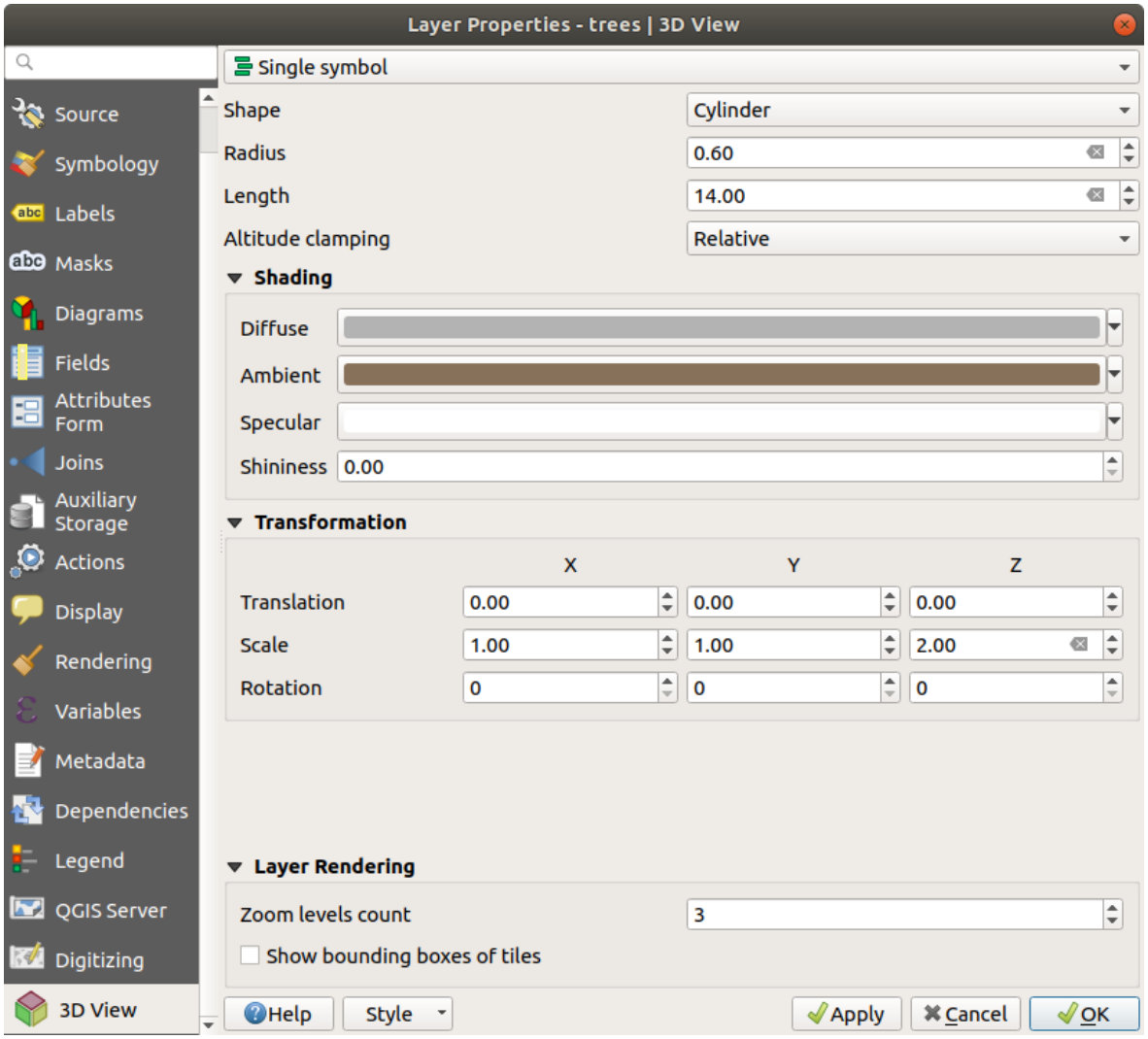
Слика12.39: Layer properties - Masks tab

12.1.7 3D View Properties

The *3D View* tab provides settings for vector layers that should be depicted in the *3D Map view* tool.

To display a layer in 3D, select from the combobox at the top of the tab, either:

- *Single symbol*: features are rendered using a common 3D symbol whose properties can be *data-defined* or not. Read details on *setting a 3D symbol* for each layer geometry type.
- *Rule-based*: multiple symbol configurations can be defined and applied selectively based on expression filters and scale range. More details on how-to at *Rule-based rendering*.



Слика12.40: 3D properties of a point layer

Пажња

Prefer the *Elevation* tab for symbol elevation and terrain settings

Features' elevation and altitude related properties (*Altitude clamping*, *Altitude binding*, *Extrusion* or *Height*) in the *3D View* tab inherit their default values from the layer's *Elevation properties* and should preferably be set from within the *Elevation* tab.

For better performance, data from vector layers are loaded in the background, using multithreading, and rendered in tiles whose size can be controlled from the *Layer rendering* section of the tab:

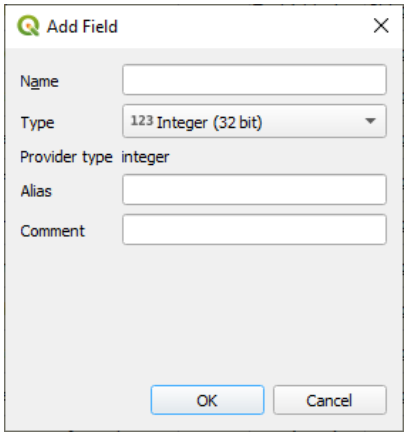
- *Zoom levels count*: determines how deep the quadtree will be. For example, one zoom level means there will be a single tile for the whole layer. Three zoom levels means there will be 16 tiles at the leaf level (every extra zoom level multiplies that by 4). The default is `3` and the maximum is `8`.
- *Show bounding boxes of tiles*: especially useful if there are issues with tiles not showing up when they should.

12.1.8 Fields Properties

The *Fields* tab provides information on fields related to the layer and helps you organize them.

The layer can be made *editable* using the Toggle editing mode. At this moment, you can modify its structure using the New field and Delete field buttons.

When creating New field, the *Comment* option is available only for data sources that allow editing comments (See *Database entries* for more details). You can also set aliases within *Add Field* dialog, for supported OGR formats (GeoPackage and ESRI File Geodatabase).



Слика12.41: Add Field Dialog

You can also rename fields by double-clicking its name. This is only supported for data providers like PostgreSQL, Oracle, Memory layer and some GDAL layers depending on the GDAL version.

If set in the underlying data source or in the *forms properties*, the field's alias is also displayed. An alias is a human readable field name you can use in the feature form or the attribute table. Aliases are saved in the project file.

Other than the fields contained in the dataset, *virtual fields* and *Auxiliary Storage* included, the *Fields* tab also lists fields from any *joined layers*. Depending on the origin of the field, a different background color is applied to it.

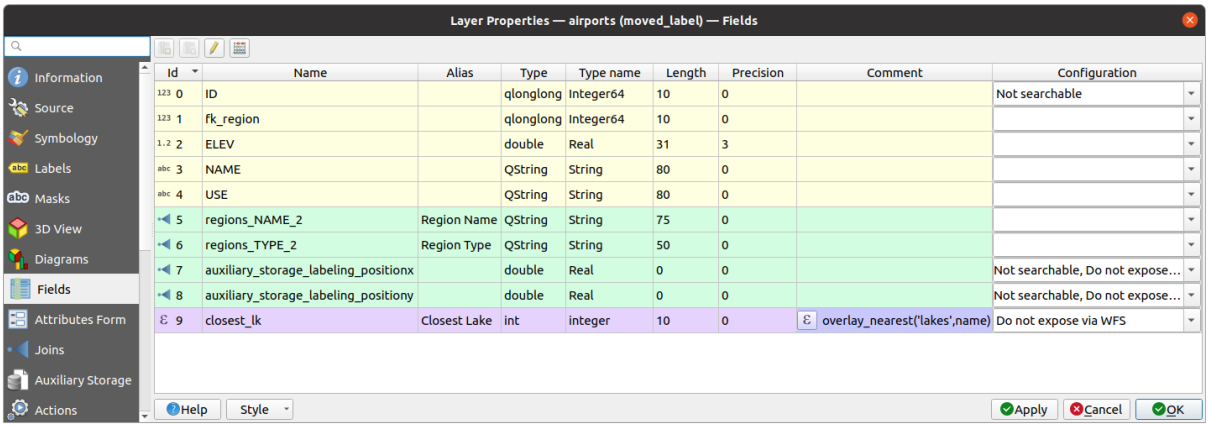
For each listed field, the dialog also lists read-only characteristics such as its *Type*, *Type name*, *Length* and *Precision*`.

Depending on the data provider, you can associate a comment with a field, for example at its creation. This information is retrieved and shown in the *Comment* column and is later displayed when hovering over the field label in

a feature form.

Under the *Configuration* column, you can set how the field should behave in certain circumstances:

- `Not searchable`: check this option if you do not want this field to be queried by the *search locator bar*
- `Do not expose via WMS`: check this option if you do not want to display this field if the layer is served as WMS from QGIS server
- `Do not expose via WFS`: check this option if you do not want to display this field if the layer is served as WFS from QGIS server



Слика12.42: Fields properties tab

12.1.9 Attributes Form Properties

The *Attributes Form* tab helps you set up the form to display when creating new features, editing or querying existing one. This affects them in both *table and form views*. You can define:

- the look and the behavior (label, visibility, widget, constraints…) of each field in the current layer, the *joined layers* and the *relational ones*;
- the placement of enabled *actions*;
- the form's structure (custom or autogenerated);
- extra HTML, QML, text widgets to enhance the form;
- extra logic in Python to handle interaction with the form or field widgets.

At the top right of the dialog, you can set whether the form is opened by default when creating new features. This can be configured per layer or globally with the *Suppress attribute form pop-up after feature creation* option in the *Settings* ► *Options* ► *Digitizing* menu.

Customizing a form for your data

By default, when you click on a feature with the Identify Features tool or switch the attribute table to the *form view* mode, QGIS displays a basic form with predefined widgets (generally spinboxes and textboxes — each field is represented on a dedicated row by its label next to the widget). If *relations* are set on the layer, fields from the referencing layers are shown in an embedded frame at the bottom of the form, following the same basic structure.

This rendering is the result of the default `Autogenerate` value of the *Attribute editor layout* setting in the *Layer properties* ► *Attributes Form* tab. This property holds three different values:

- `Autogenerate`: keeps the basic structure of „one row - one field" for the form but allows to customize each corresponding widget.
- `Drag-and-drop designer`: other than widget customization, the form structure can be made more complex eg, with widgets embedded in groups and tabs.

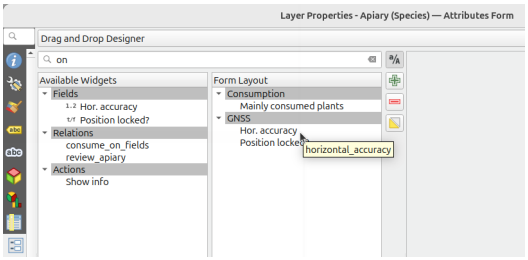
- `Provide ui file`: allows to use a Qt designer file, hence a potentially more complex and fully featured template, as feature form.

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.



Слика12.43: Search fields using aliases and names in Attributes Form page

The autogenerated form

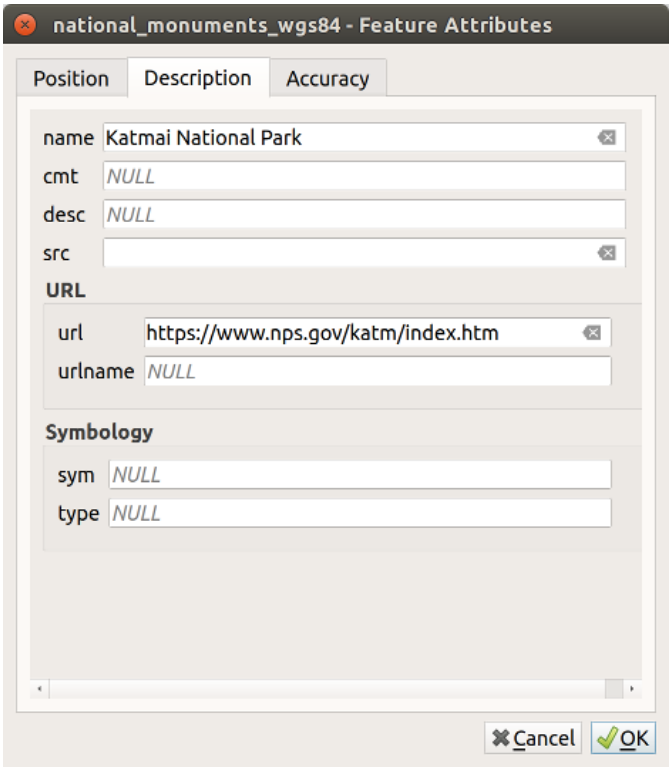
When the `Autogenerate` option is on, the *Available widgets* panel displays lists of fields (from the current layer, the join and relation layers) and the enabled actions that would be shown in the form. Select a field and you can configure its appearance and behavior in the right panel:

- adding *custom label and automated checks* to the field;
- setting a *particular widget* to use.

In this editor mode, only the properties can be viewed for the actions. Moreover, they will be displayed through an *Actions* drop-down menu in the feature form, and following the *Show in attribute table* property in other contexts.

The drag and drop designer

The drag and drop designer allows you to create a form with several containers (tabs or groups) to present the attribute fields or other widgets that are not directly linked to a particular field (like the HTML/QML widgets or the *actions* defined for the layer), as shown for example in Сл.12.44.



Слика12.44: Resulting built-in form with tabs and named groups

1. Choose `Drag and drop designer` from the *Select attribute layout editor* combobox. This enables the *Form Layout* panel next to the *Available widgets* panel, filled with existing fields. The selected field displays its *properties* in a third panel.

2. Select fields you do not want to use in your *Form Layout* panel and hit the button to remove them. You can also toggle the selection with the Invert selection button.

3. Drag and drop fields from the first panel to the *Form Layout* one to re-add them. The same field can be added multiple times.

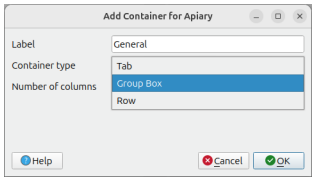
4. Drag and drop fields within the *Form Layout* panel to reorder their position.

5. Add containers to associate fields that belong to the same category and better structure the form.

    1. The first step is to use the Add a new tab or group to the form layout icon. Fields and other groups will be displayed in it.

    2. Then set the properties of the container, i.e.:

        - the *Label*: the title that will be used for the container
        - the *Container Type*: it can be a *Tab*, *Group box in container* (a collapsible group box inside a tab or another group) or a *Row* (a container type that allows you to arrange your widgets in a horizontal row, automatically determining the number of columns based on the number of widgets),
        - the *Within*: this optional feature allows you to select an existing container in which the new container (*Group box in container* or *Row*) will be embedded.
        - and the *Number of columns* the embedded fields should be distributed over



Слика12.45: Dialog to create containers with the **Attribute editor layout**

These, and other properties can later be updated by selecting the item and, from the third panel:

- hide or show the container's label
- rename the container
- set the number of columns
- enter an expression to control the container's visibility. The expression will be re-evaluated every time values in the form change, and the tab or group box shown/hidden accordingly
- *Show as Group Box*: converts a tab into a top-level group box and vice versa
- in case of a group box, set whether at the form opening, it should show as *Collapsed* for every features, or only for those matching an expression (*Control collapsed by expression*).
- configure the *Style* look of the container, with custom *Background color*, label color and font properties

3. You can create and embed as many containers as you want by pressing again the Add a new tab or group to the form layout icon.

6. The next step is to assign the relevant fields to each container, by simple drag and drop. Group boxes and tabs can also be moved the same way.
7. *Customize the widget* of the fields in use
8. In case the layer is involved in a *one or many to many relation*, drag-and-drop the relation name from the *Available Widgets* panel to the *Form Layout* panel. The associated layer attribute form will be embedded at the chosen place in the current layer's form. As for the other items, select the relation label to configure some properties:
   - hide or show the relation label
   - show the link button
   - show the unlink button
9. In case the layer has one or more *actions* enabled for `Layer` or `Feature` scope, the actions will be listed under *Actions* and you can drag and drop them as with the other fields. The associated action will be embedded at the chosen place in the current layer's form.
10. Further customize the form by adding one or more widgets from *Other Widgets* (see *Other Widgets*)
11. Apply the layer's properties dialog
12. Open a feature attribute form (eg, using the Identify features tool) and it should display the new form.

Other Widgets

The drag and drop designer offers a number of widgets that are not connected to a particular field. They can be used to enhance the appearance of the form or to display dynamically calculated values.

- *HTML Widget*: embeds an HTML page, the HTML source may contain the result of dynamically calculated expressions.

  HTML widgets can be used for example to display images stored as BLOB in a field (let's call it `photo`):

1. In the `Drag-and-drop designer` mode, add a *HTML Widget* to your *Form Layout*.
2. Double-click on the *HTML Widget* to configure it.
3. Change the default *Title* or hide it.
4. Press the  button and enter the following QGIS expression:

   ```
   '<img src= "' || to_base64("photo") || '">'
   ```

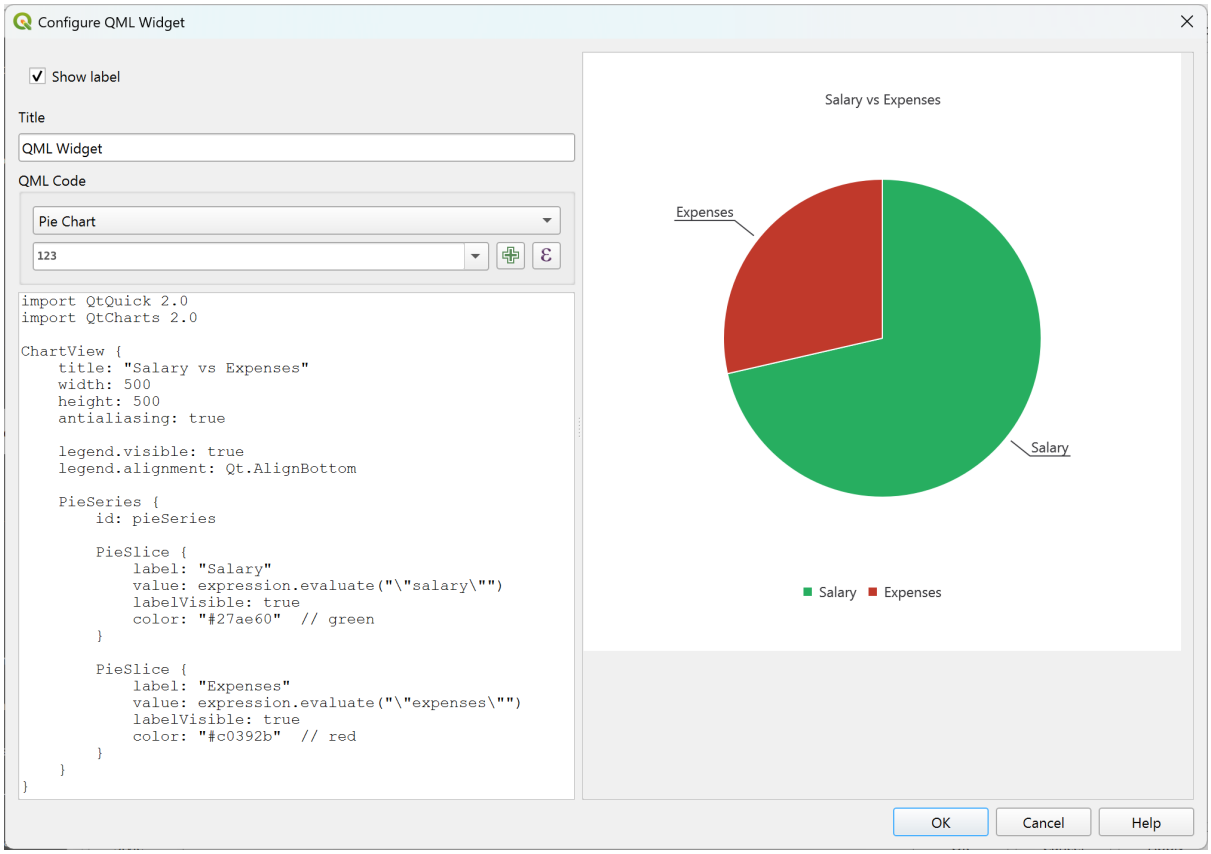
   Ensure that you replace *photo* with your own BLOB field name. The above expression creates a string with HTML image tag in which the BLOB file is encoded.
5. Apply the dialog and then press the  button.
6. QGIS automatically applies HTML formatting and functions to evaluate your expression, resulting in following code:

   ```
   <script>document.write(expression.evaluate("'<img src=\"data:image/png;
   ↪base64,' || to_base64(\"photo\") || '\">'"));</script>
   ```

   A preview of your image is displayed on the right.

- *QML Widget*: embeds a Qt QML document, displaying graphical elements in the attribute form. Beside the custom *Title* of the added widget and whether it should be shown or not, you can select from predefined *QML code* elements:
  - *Free text…*: allows you to write from scratch or paste an existing code
  - *Rectangle*: provides a minimal code for displaying a rectangle
  - *Bar chart*: provides a minimal code for displaying a bar chart
  - *Pie chart*: provides a minimal code for displaying a pie chart

  You can extend the code with QML syntax, use layer fields or QGIS expressions that are dynamically calculated.



Слика12.46: Setting a QML graph to display in attribute form

- *Text Widget*: displays a text widget which supports basic HTML markup and may contain the result of dynamically calculated expressions.
- *Spacer Widget*: inserts an empty transparent rectangle, increasing the vertical distance between two widgets.

Савет

Display Dynamic Content

The widgets mentioned above (except the *Spacer Widget*) support expressions that can be used to display content that dynamically changes whenever another field in the form changes. This can be achieved using the `current_value('field_name')` function in the expression to examine the value of another field.

Using custom ui-file

The `Provide ui-file` option allows you to use complex dialogs made with Qt-Designer. Using a UI-file allows a great deal of freedom in creating a dialog. Note that, in order to link the graphical objects (textbox, combobox…) to the layer's fields, you need to give them the same name.

Use the *Edit UI* to define the path to the file to use.

UI-files can also be hosted on a remote server. In this case, you provide the URL of the form instead of the file path in *Edit UI*.

You'll find some example in the Creating a new form lesson of the QGIS-training-manual--index-reference. For more advanced information, see https://woostuff.wordpress.com/2011/09/05/qgis-tips-custom-feature-forms-with-python-logic/.

Enhance your form with custom functions

QGIS forms can have a Python function that is called when the dialog is opened. Use this function to add extra logic to your dialogs. The form code can be specified in three different ways:

- `load from the environment`: use a function, for example in `startup.py` or from an installed plugin
- `load from an external file`: a file chooser will let you select a Python file from your filesystem or enter a URL for a remote file.
- `provide code in this dialog`: a Python editor will appear where you can directly type the function to use.

In all cases you must enter the name of the function that will be called (`open` in the example below).

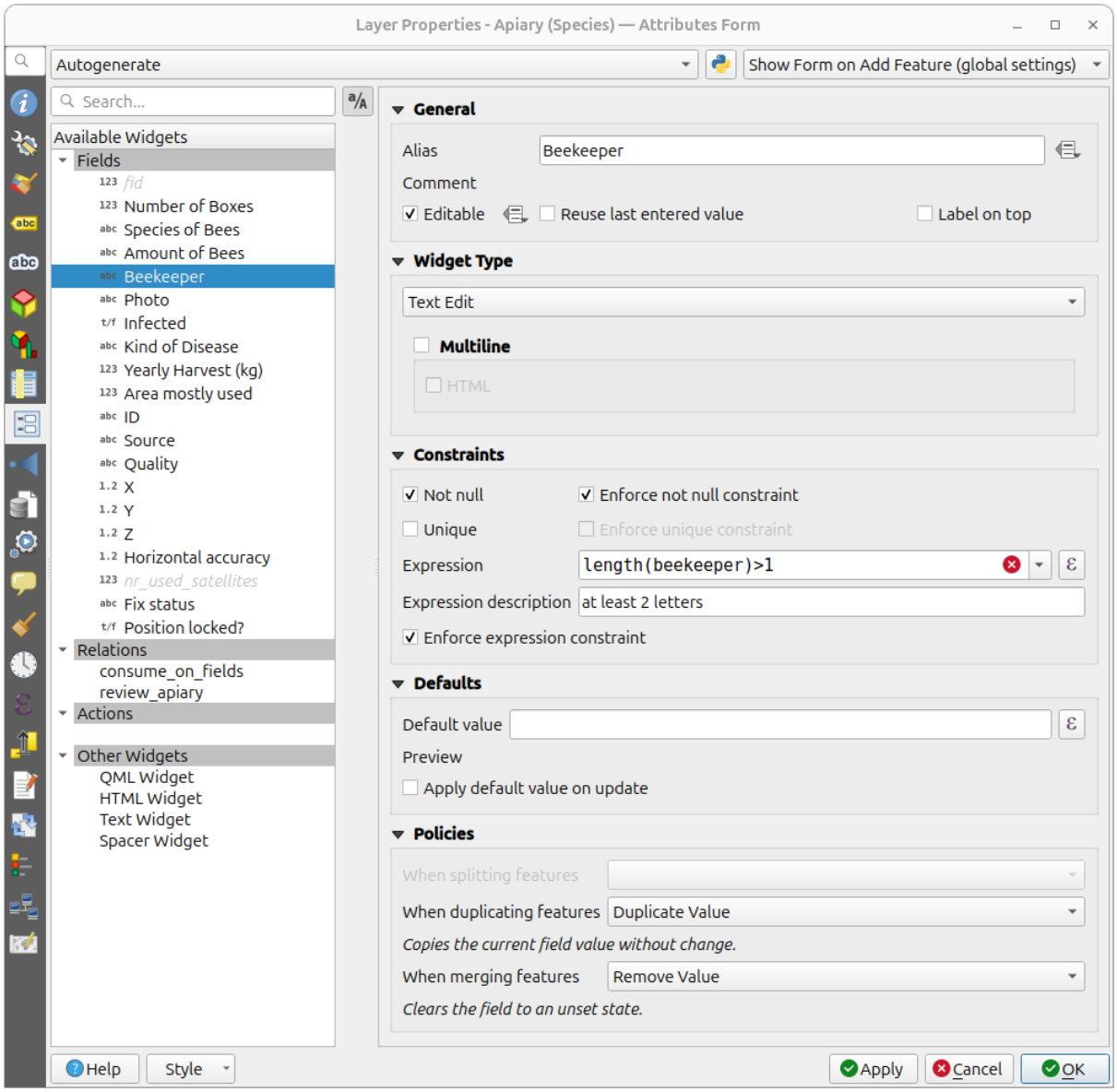
An example is (in module MyForms.py):

```
def open(dialog,layer,feature):
    geom = feature.geometry()
    control = dialog.findChild(QWidget,"My line edit")
```

Reference in Python Init Function like so: `open`

Configure the field behavior

The main part of the *Attributes Form* tab helps you set the type of widget used to fill or display values of the field, in the attribute table or the feature form: you can define how user interacts with each field and the values or range of values that are allowed to be added to each.



Слика12.47: Dialog to select an edit widget for an attribute column

Савет

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.

Common settings

Regardless the type of widget applied to the field, there are some common properties you can set to control whether and how a field can be edited.

Widget display

Only available for the *Drag and drop* designer mode, this group helps you configure the look of the widget assigned to the field:

- *Show label*: indicates whether the field name should be displayed in the form
- *Override label color*: applies specific color to the field's label
- *Override label font*: applies specific font properties (bold, italic, underline, strikethrough, or font family) to the field's label
- *Size*: allows to control how widgets will relatively resize when resizing an attribute form.
  - *Horizontal stretch*: sets a higher horizontal value for widgets that need more horizontal space.
  - *Vertical stretch*: determines how widgets resize vertically when the form is resized.

General options

- *Alias*: a human readable name to use for fields. The alias will be displayed in the feature form, the attribute table, or in the *Identify results* panel. It can also be used as field name replacement in the *expression builder*, easing expressions understanding and reviews. Aliases are saved in project file.
- *Comment*: displays the field's comment as shown in the *Fields* tab, in a read-only state. This information is shown as tooltip when hovering over the field label in a feature form.
- *Editable*: uncheck this option to set the field read-only (not manually modifiable) even when the layer is in edit mode. Note that checking this setting doesn't override any edit limitation from the provider. This option can be controlled by a data-defined property thanks to the Data-defined override button.
- *Reuse last entered value*: remembers the last value entered in this field and uses it as default for the next feature being edited in the layer.
- *Label on top*: places the field name above or beside the widget in the feature form.

Constraints

You can constrain the value to insert in the field. This constraint can be:

- *Not null*: requires the user to provide a value;

- *Unique*: guarantee the inserted value to be unique throughout the field;
- based on a custom *expression*: e.g. `not regexp_match(col0,'[^A-Za-z]')` will ensure that the value of the field *col0* has only alphabet letters. A short description can be added to help you remember the constraint.

Whenever a value is added or edited in a field, it's submitted to the existing constraints and:

- if it meets all the requirements, a green check is shown beside the field in the form;
- if it does not meet all the requirements, then the field is colored in yellow or orange and a corresponding cross is displayed next to the widget. You can hover over the cross to remind which constraints are applied to the field and fix the value:
  - A yellow cross appears when the unmet constraint is an unenforced one (`soft constraint`) and it does not prevent you to save the changes with the „wrong" values;
  - An orange cross can not be ignored and does not allow you to save your modifications until they meet the constraints. It appears when the *Enforce constraint* option is checked (`hard constraint`).

Default values

- *Default value*: for new features, automatically populates by default the field with a predefined value or an *expression-based one*. For example, you can:
  - use `$x`, `$length`, `$area` to automatically populate a field with the feature's X coordinate, length, area or any geometric information at its creation;
  - increment a field by 1 for each new feature using `maximum("field")+1`;
  - save the feature creation datetime using `now()`;
  - use *variables* in expressions, making it easier to e.g. insert the operator name (`@user_full_name`), the project file path (`@project_path`), …

  A preview of the resulting default value is displayed at the bottom of the widget.

  Белешка

  The `Default value` option is not aware of the values in any other field of the feature being created so it won't be possible to use an expression combining any of those values i.e using an expression like `concat(field1, field2)` may not work.

- *Apply default value on update*: whenever the feature attribute or geometry is changed, the default value is recalculated. This could be handy to save values like last user that modifies data, last time it was changed…

Policies

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*:

- *Duplicate Values*: Keeps the existing value of the field for the new features.
- *Use Default Value*: Resets the field by recalculating its *default value*. If no default value clause exists, the existing value is kept for the new features.
- *Remove Value*: Clears the field to an unset state.
- *Use Ratio Geometries*: Recalculates the field value for all split portions by multiplying the existing value by ratio of the split parts lengths or areas.

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.

Edit widgets

Based on the field type, QGIS automatically determines and assigns a default widget type to it. You can then replace the widget with any other compatible with the field type. The available widgets are:

- **Binary (BLOB)**: Available only for binary fields, it displays by default a label with the size of the embedded data, if not empty. A drop-down button next to the label allows to:
  - *Embed file*, replacing or filling the field
  - *Clear contents*, removing any data in the field
  - *Save contents to file*, exporting the data as a file on disk

  It is also possible to preview the embedded binary file in the field, if combined in a drag-and-drop form with e.g. a *QML or HTML widget*.
- **Checkbox**: Displays a checkbox whose state defines the value to insert. Internally, the widget stores a value (e.g. 1 for checked, 0 for unchecked) while representing it visually as a toggled box. You can configure the stored values for the checked and unchecked states in the representation. It is also possible to display the checkbox state as a plain text using „True"/"False" labels. These settings influence how the values are shown in classification renderers.
- **Classification**: Only available when a *categorized symbology* is applied to the layer, displays a combo box with the values of the classes.
- **Color**: Displays a *color widget* allowing to select a color; the color value is stored as a html notation in the attribute table.
- **Date/Time**: Displays a line field which can open a calendar widget to enter a date, a time or both. Column type must be text. You can select a custom format, pop-up a calendar, etc.
- **Enumeration**: Opens a combo box with predefined values fetched from the database. This is currently only supported by the PostgreSQL provider, for fields of `enum` type.
- **Attachment**: Uses a „Open file" dialog to store file path in a relative or absolute mode. It can be used to display the document path as a hyperlink or render the document within a dedicated widget in the form. Supported document types are image, web page, audio and video, and the supported file formats depend on the Operating System. You can also configure an *external storage system* to fetch/store resources.

> **Савет**
>
> **Relative Path in Attachment widget**
>
> If the path which is selected with the file browser is located in the same directory as the `.qgs` project file or below, paths are converted to relative paths. This increases portability of a `.qgs` project with multimedia information attached.

- **Geometry**: Compatible with the *Geometry* field type. For example, PostgreSQL and Memory layers can contain multiple geometry fields. QGIS uses the first geometry column as the layer geometry. Applying the Geometry widget to additional geometry fields allows lossless use of the data in the attribute form:
  - Copy in the formats WKT (Well-known text representation, e.g. `POINT (30 10)`) or GeoJSON
  - Paste
  - Clear
- **Hidden**: A hidden attribute column is invisible. The user is not able to see its contents.
- **Key/Value**: Displays a two-columns table to store sets of key/value pairs within a single field. This is currently supported by the PostgreSQL provider, for fields of `hstore` type.
- **JSON View**: Displays JSON data in a syntax highlighted text edit or in tree view. This widget is currently read only. Several options are available to change how the data is displayed. 'Default view' specify if the widget should appear in Text or Tree mode. 'Format JSON' has three options which are related to the tree view only:
  - Indented: Display data in a human readable form with newlines and four space characters for indentation.
  - Compact: Display data in a one-line size optimized string without newlines or spaces.
  - Disabled: Display data as it comes from the provider.
- **List**: Displays a single column table to add different values within a single field. This is currently supported by the PostgreSQL provider, for fields of `array` type.
- **Range**: Allows you to set numeric values from a specific range. The edit widget can be either a slider or a spin box.

  Белешка

  Some layers, such as GeoPackage or ESRI File Geodatabase, with predefined **range Field Domains** will be automatically recognized by QGIS and assigned a **Range** widget for the relevant fields. The widget will be prefilled with the minimum and maximum values specified in the domain.

- **Relation Reference**: This is the default widget assigned to the referencing field (i.e., the foreign key in the child layer) when a *relation* is set. It provides direct access to the parent feature's form which in turn embeds the list and form of its children. The number of entries in the widget can be limited for efficiency, and if limit is not set, all entries will be loaded. The field defined in the display expression is used as representable value in classification renderers.
- **Text Edit** (default): This opens a text edit field that allows simple text or multiple lines to be used. If you choose multiple lines you can also choose html content.
- **Unique Values**: You can select one of the values already used in the attribute table. If 'Editable' is activated, a line edit is shown with autocompletion support, otherwise a combo box is used.
- **Uuid Generator**: Generates a read-only UUID (Universally Unique Identifiers) field, if empty.
- **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.

  Белешка

  Some layers, such as GeoPackage or ESRI File Geodatabase, with predefined **coded Field Domains** will be automatically recognized by QGIS and assigned a **Value Map** widget for the relevant fields.

- **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.

If a layer that is stored in PostgreSQL, GeoPackage or SpatiaLite is configured to use a value relation widget, but the required layer is not already loaded into the project, QGIS will automatically search for the layer in the same database/connection.

12.1.10 Joins Properties

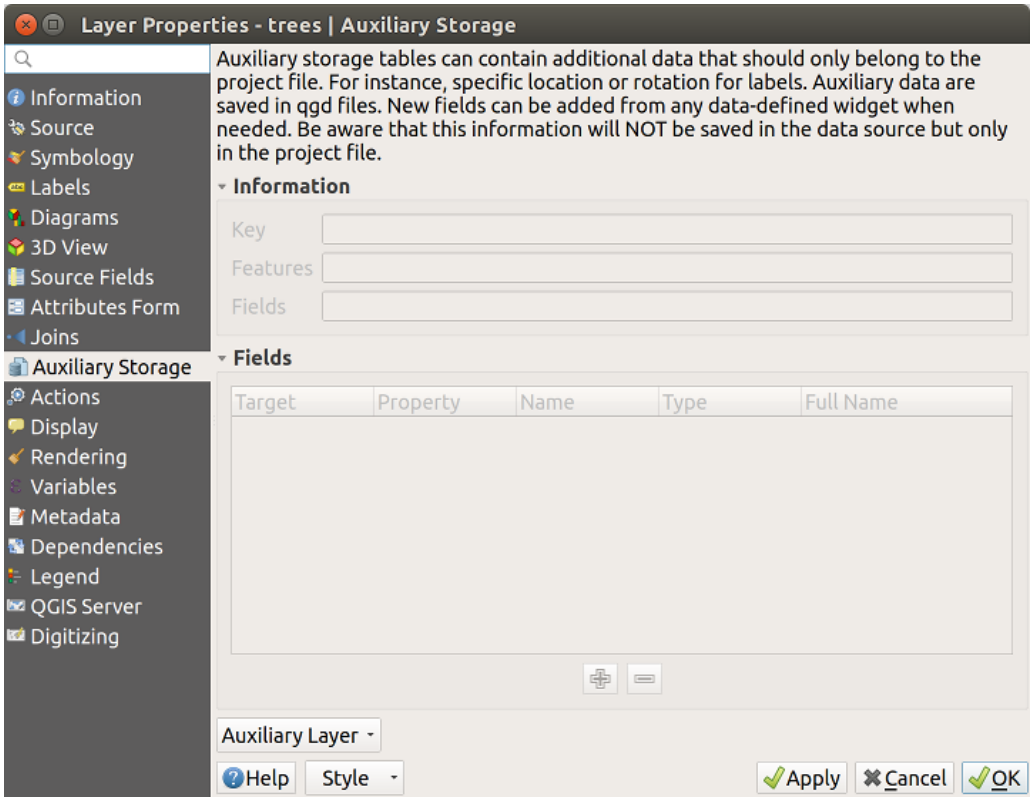
The *Joins* tab allows you to associate features of the current layer to features from another loaded vector layer (or table). The join is based on an attribute that is shared by the layers, in a one-to-one relationship. For more details on joins, please read *Joining features between two layers*.

12.1.11 Auxiliary Storage Properties

The regular way to customize styling and labeling is to use data-defined properties as described in *Data defined override setup*. However, it may not be possible if the underlying data is read only. Moreover, configuring these data-defined properties may be very time consuming or not desirable! For example, if you want to fully use map tools coming with *The Label Toolbar*, then you need to add and configure more than 20 fields in your original data source (X and Y positions, rotation angle, font style, color and so on).

The Auxiliary Storage mechanism provides the solution to these limitations and awkward configurations. Auxiliary fields are a roundabout way to automatically manage and store these data-defined properties (labels, diagram, symbology…) in a SQLite database thanks to editable joins. This allows you to store properties for layers that aren't editable.

A tab is available in vector layer properties dialog to manage auxiliary storage:

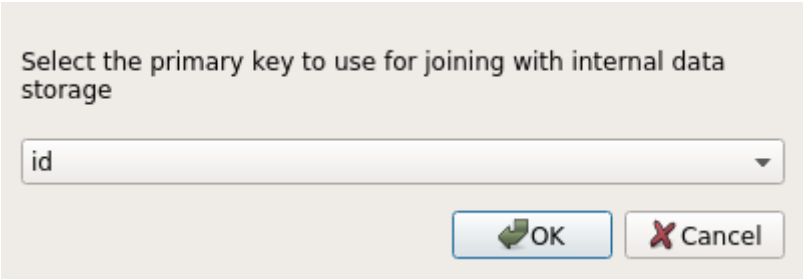


Слика12.48: Auxiliary Storage tab

Labeling

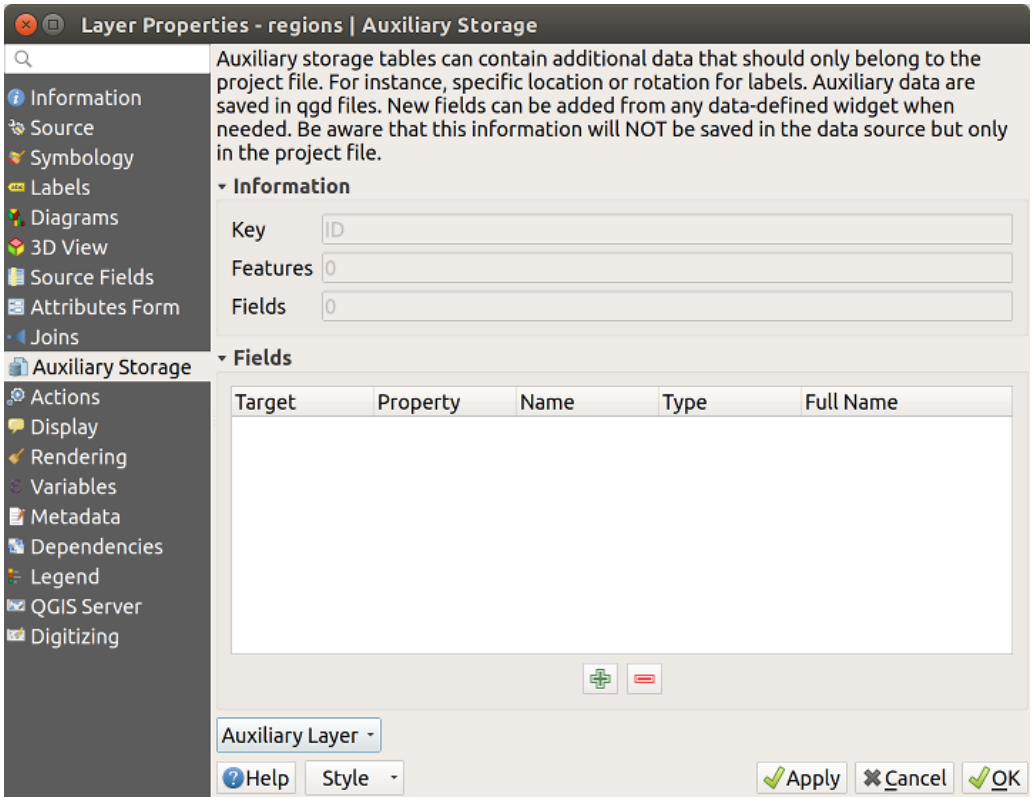
Considering that the data source may be customized thanks to data-defined properties without being editable, labeling map tools described in *The Label Toolbar* are always available as soon as labeling is activated.

Actually, the auxiliary storage system needs an auxiliary layer to store these properties in a SQLite database (see *Auxiliary storage database*). Its creation process is run the first time you click on the map while a labeling map tool is currently activated. Then, a window is displayed, allowing you to select the primary key to use for joining (to ensure that features are uniquely identified):



Слика12.49: Auxiliary Layer creation dialog

As soon as an auxiliary layer is configured for the current data source, you can retrieve its information in the tab:

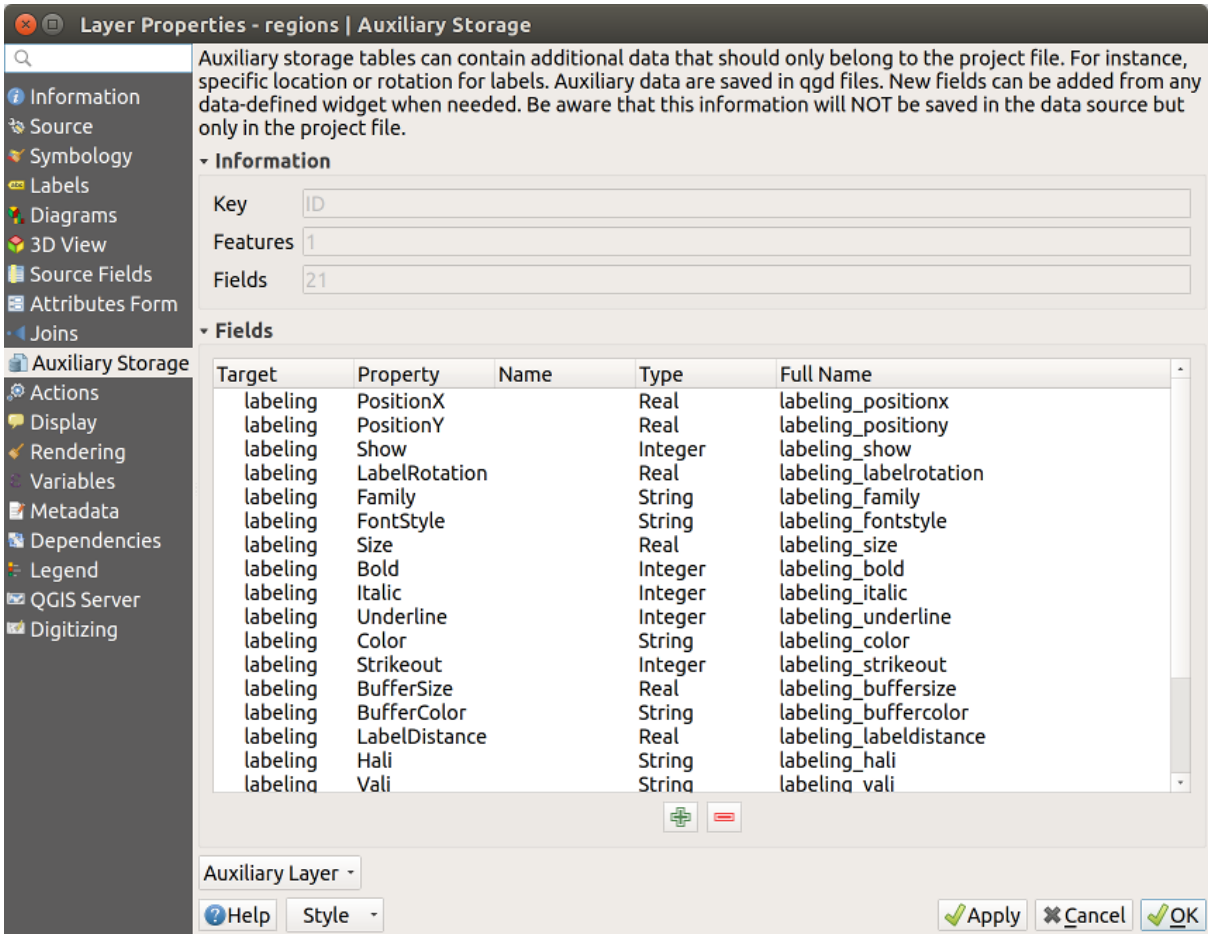


Слика12.50: Auxiliary Layer key

The auxiliary layer now has these characteristics:

- the primary key is `ID`,
- there are `0` features using an auxiliary field,
- there are `0` auxiliary fields.

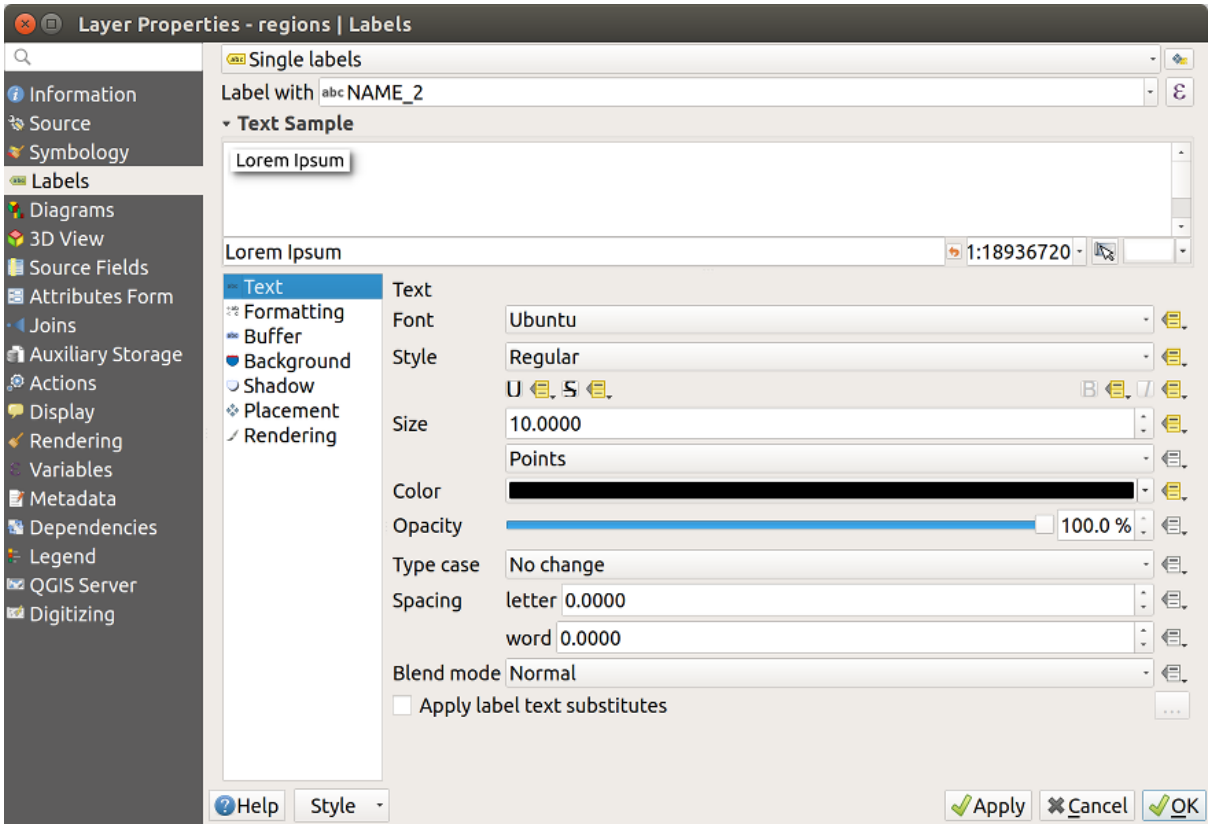
Now that the auxiliary layer is created, you can edit the layer labels. Click on a label while the Change Label map tool is activated, then you can update styling properties like sizes, colors, and so on. The corresponding data-defined properties are created and can be retrieved:



Слика12.51: Auxiliary Fields

As you can see in the figure above, `21` fields are automatically created and configured for labeling. For example, the `FontStyle` auxiliary field type is a `String` and is named `labeling_fontstyle` in the underlying SQLite database. There is also `1` feature which is currently using these auxiliary fields.

Notice that the icon is displayed in the *Labels* properties tab indicating that the data-defined override options are set correctly:

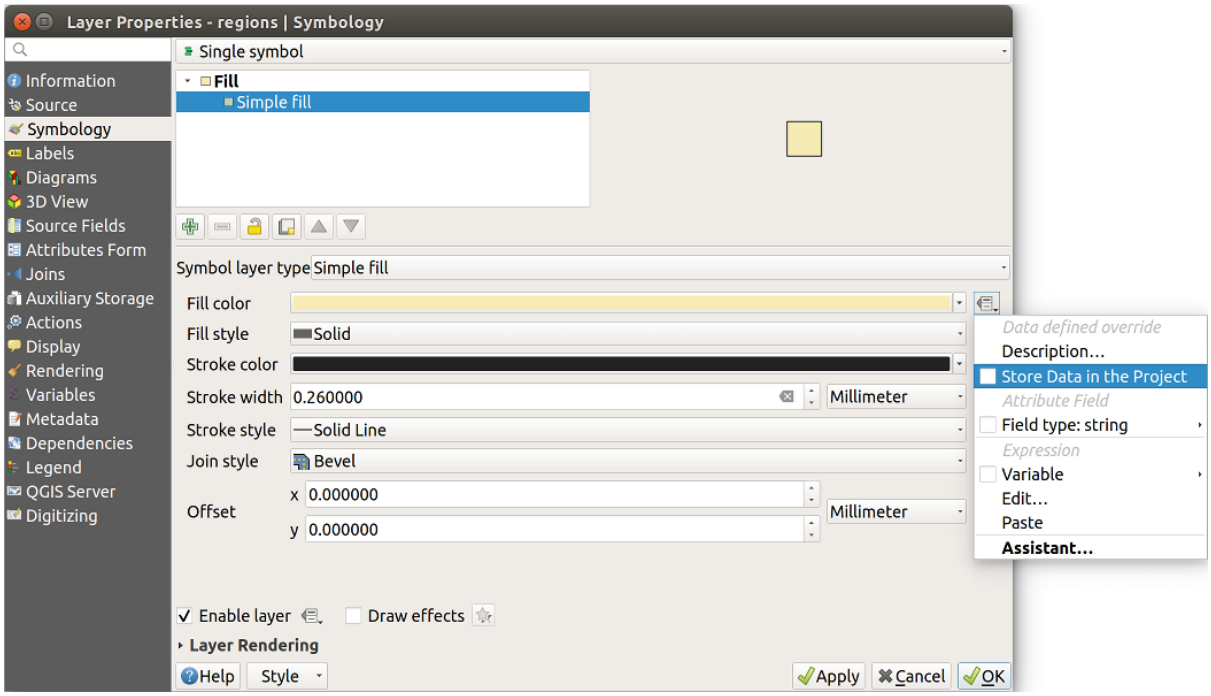


Слика12.52: Data-defined properties automatically created

Otherwise, there's another way to create an auxiliary field for a specific property thanks to the Data-defined override button. By clicking on *Store data in the project*, an auxiliary field is automatically created for the *Opacity* field. If you click on this button and the auxiliary layer is not created yet, a window (Сл.12.49) is first displayed to select the primary key to use for joining.

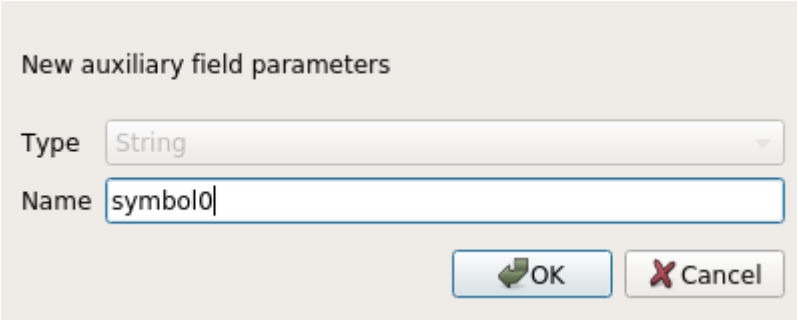
Symbology

Like the method described above for customizing labels, auxiliary fields can also be used to stylize symbols and diagrams. To do this, click on Data-defined override and select *Store data in the project* for a specific property. For example, the *Fill color* field:



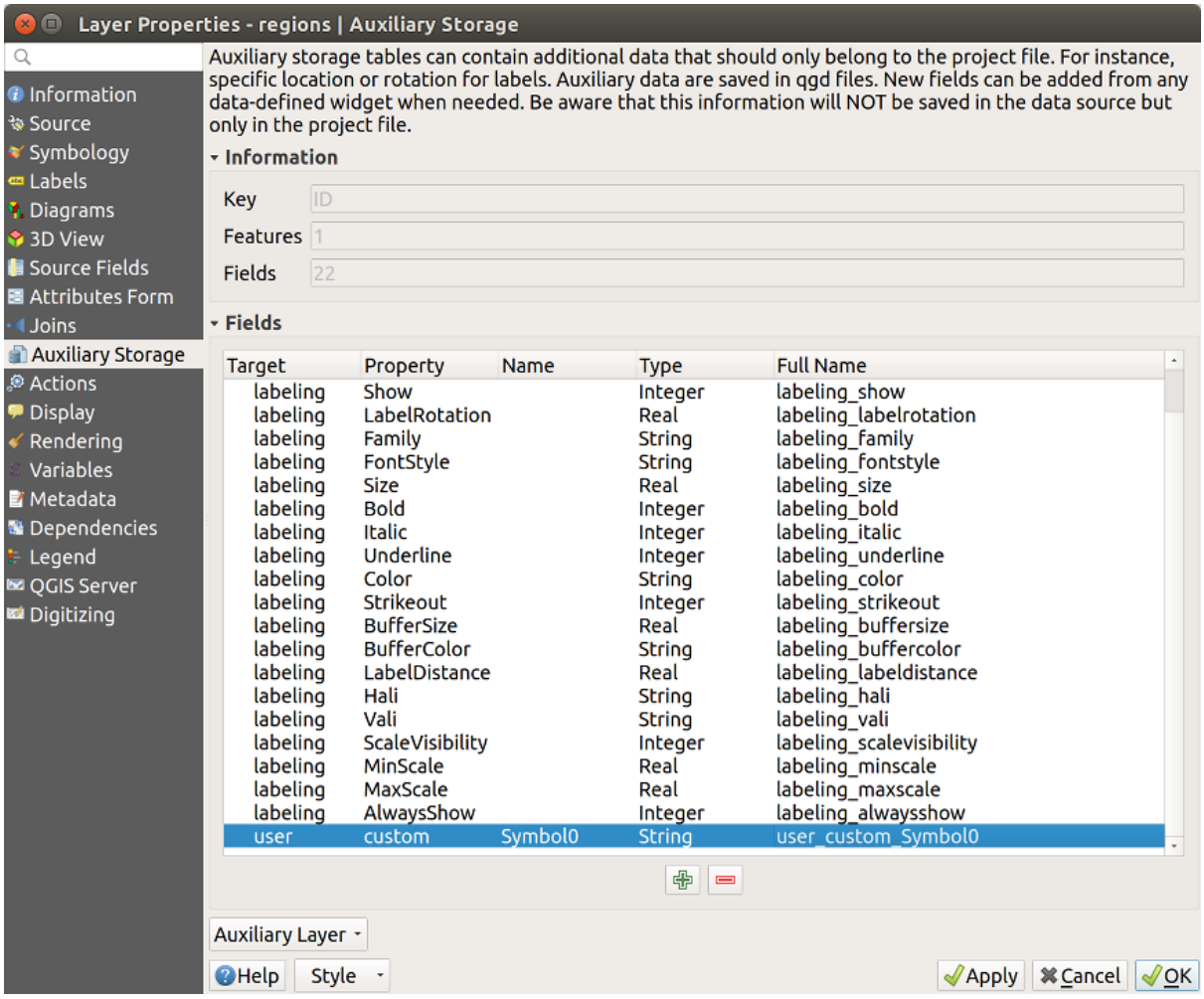
Слика12.53: Data-defined property menu for symbol

There are different attributes for each symbol (e.g. fill style, fill color, stroke color, etc…), so each auxiliary field representing an attribute requires a unique name to avoid conflicts. After selecting *Store data in the project*, a window opens and displays the *Type* of the field and prompts you to enter a unique name for the auxiliary field. For example, when creating a *Fill color* auxiliary field the following window opens:



Слика12.54: Name of the auxiliary field for a symbol

Once created, the auxiliary field can be retrieved in the auxiliary storage tab:



Слика12.55: Auxiliary field symbol

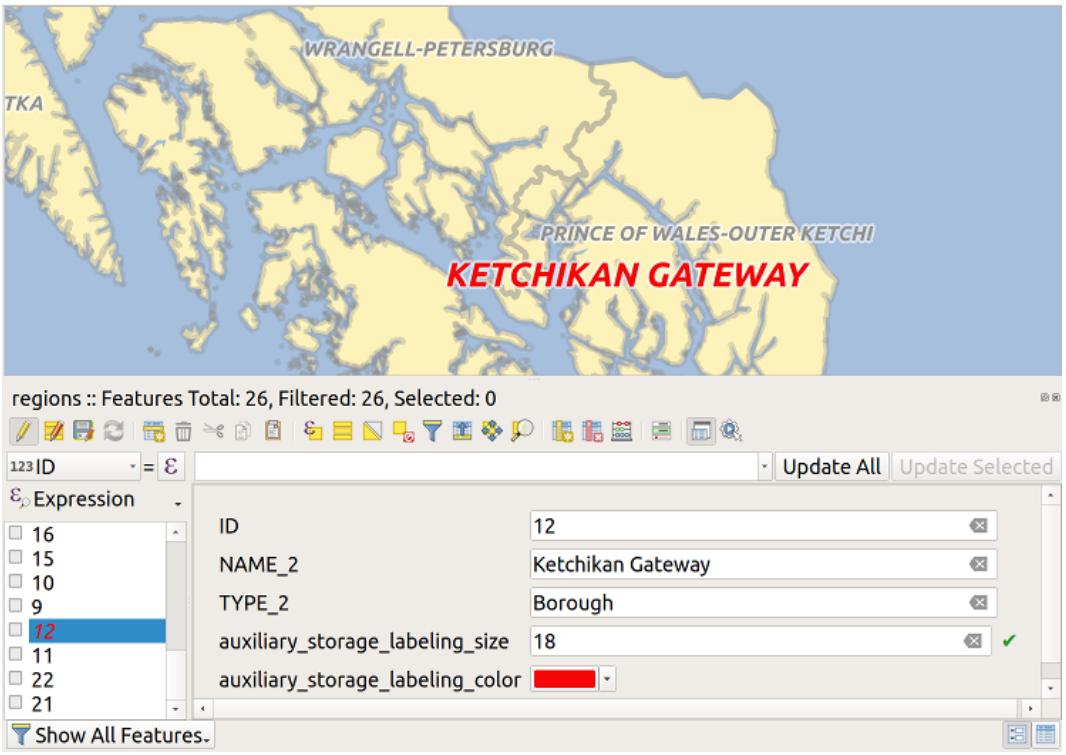
Attribute table and widgets

Auxiliary fields can be edited using the *attribute table*. However, not all auxiliary fields are initially visible in the attribute table.

Auxiliary fields representing attributes of a layer's symbology, labeling, appearance, or diagrams will appear automatically in the attribute table. The exception are attributes that can be modified using the *Label Toolbar* which are hidden by default. Auxiliary fields representing a `Color` have a widget **Color** set by default, otherwise auxiliary fields default to the **Text Edit** widget.

Auxiliary fields that represent attributes that can be modified using the *Label toolbar* are **Hidden** in the attribute table by default. To make a field visible, open the *Attribute Form properties tab* and change the value of an auxiliary field *Widget Type* from **Hidden** to another relevant value. For example, change the **auxiliary_storage_labeling_size** to **Text Edit** or change **auxiliary_storage_labeling_color** to the **Color** widget. Those fields will now be visible in the attribute table.

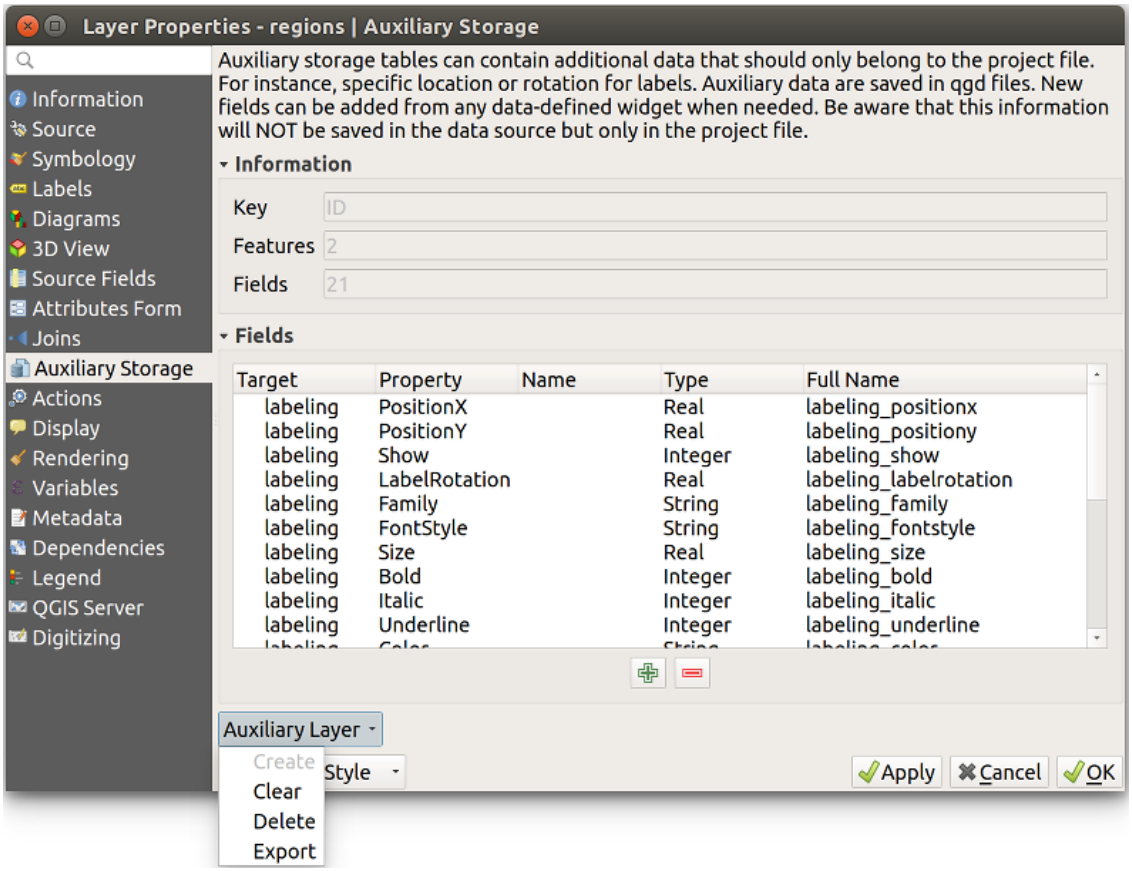
Auxiliary fields in the attribute table will appear like the following image:



Слика12.56: Form with auxiliary fields

Management

The *Auxiliary Layer* menu allows you to manage the auxiliary fields:



Слика12.57: Auxiliary layer management

The first item *Create* is disabled in this case because the auxiliary layer is already created. But in case of a fresh work, you can use this action to create an auxiliary layer. As explained in *Labeling*, a primary key will be needed then.

The *Clear* action allows to keep all auxiliary fields, but remove their contents. This way, the number of features using these fields will fall to `0`.

The *Delete* action completely removes the auxiliary layer. In other words, the corresponding table is deleted from the underlying SQLite database and properties customization are lost.

Finally, the *Export* action allows to save the auxiliary layer as a *new vector layer*. Note that geometries are not stored in auxiliary storage. However, in this case, geometries are exported from the original data source too.

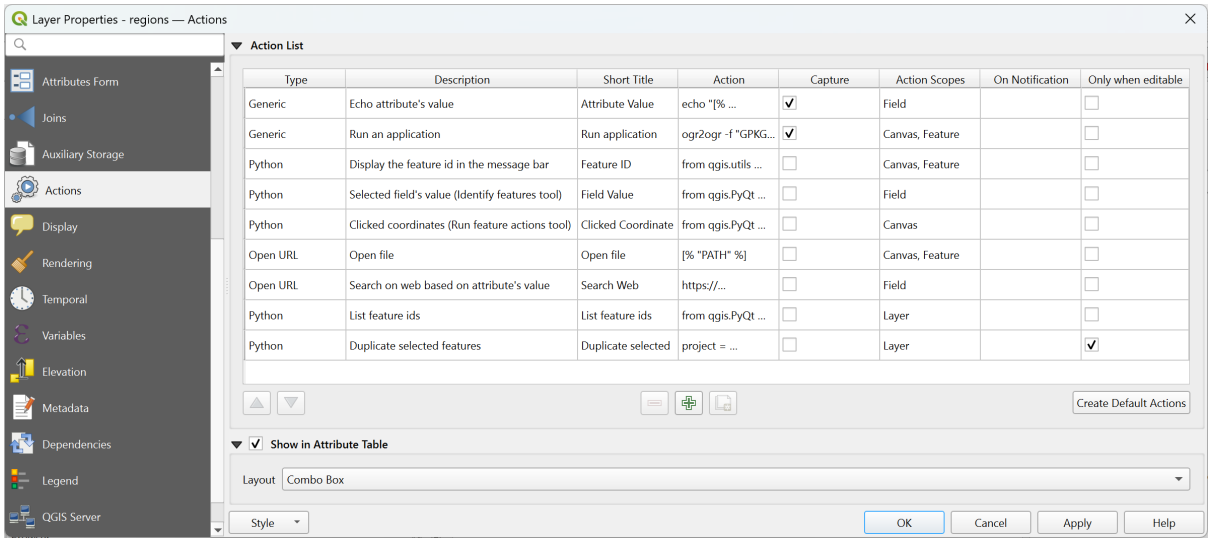
Auxiliary storage database

When you save your project with the `.qgs` format, the SQLite database used for auxiliary storage is saved at the same place but with the extension `.qgd`.

For convenience, an archive may be used instead thanks to the `.qgz` format. In this case, `.qgd` and `.qgs` files are both embedded in the archive.

12.1.12 Actions Properties

The  *Actions* tab provides the ability to perform an action based on the attributes of a feature. This can be used to perform any number of actions, for example, running a program with arguments built from the attributes of a feature or passing parameters to a web reporting tool.



Слика12.58: Overview action dialog with some sample actions

Actions are useful when you frequently want to run an external application or view a web page based on one or more values in your vector layer. They are of different types and can be used like this:

- *Generic*, *macOS*, *Windows* and *Unix* actions start an external process.
- *Python* actions execute a Python expression.
- *Generic* and *Python* actions are visible everywhere.
- *macOS*, *Windows* and *Unix* actions are visible only on the respective platform (i.e., you can define three „Edit" actions to open an editor and the users can only see and execute the one „Edit" action for their platform to run the editor).
- *Open URL*: Uses a HTTP GET request to open a provided URL.
- *Submit URL (urlencoded or JSON)*: Same as the *Open URL* action but using a HTTP POST request. Data are posted to a URL, using „application/x-www-form-urlencoded" or „application/json" if the body is a valid JSON.

  An example of action call could be:

  ```
  http://localhost:8000?/[% url_encode(map('file', 'index.php')) %]
  ```

- *Submit URL (multipart)*: Same as the *Open URL* action but using a HTTP POST request. Data are posted to a URL, using „multipart/form-data".

There are several examples included in the dialog. You can load them by clicking on *Create Default Actions*. To edit any of the examples, double-click its row. One example is performing a search based on an attribute value. This concept is used in the following discussion.

The *Show in Attribute Table* allows you to display in the attribute table dialog the checked feature-scoped actions, either as *Combo Box* or as *Separate Buttons* (see *Konfigurisanje kolona*).

Defining Actions

To define an attribute action, open the vector *Layer Properties* dialog and click on the *Actions* tab. In the *Actions* tab, click the Add a new action to open the *Edit Action* dialog.

Select the action *Type* and provide a descriptive name for the action. The action itself must contain the name of the application that will be executed when the action is invoked. You can add one or more attribute field values as arguments to the application. When the action is invoked, any set of characters that start with a `%` followed by the name of a field will be replaced by the value of that field. The special characters `%%` will be replaced by the value of

the field that was selected from the identify results or attribute table (see *Using Actions*). Double quote marks can be used to group text into a single argument to the program, script or command. Double quotes will be ignored if preceded by a backslash.

Actions can invoke a single process, with arguments, so Boolean operators (such as `&`, `&&`, `;`, `|`) will not work. In UNIX-like operating systems multiple commands can be executed via `bash -c`.

The *Action Scopes* allows you to define where the action should be available. You have following choices:

1. *Field*: action is available when right click in the cell within the attribute table, in the feature form and in the default action button of the main toolbar.
2. *Feature*: action is available when right click in the cell within the attribute table.
3. *Canvas*: action is available in the main action button in the toolbar.
4. *Form*: action is available only in a feature form designed using the *drag-and-drop* mode.
5. *Layer*: action is available in the action button in the attribute table toolbar. Be aware that this type of action involves the entire layer and not the single features.

If you have field names that are substrings of other field names (e.g., `col1` and `col10`), you should indicate that by surrounding the field name (and the % character) with square brackets (e.g., `[%col10]`). This will prevent the `%col10` field name from being mistaken for the `%col1` field name with a `0` on the end. The brackets will be removed by QGIS when it substitutes in the value of the field. If you want the substituted field to be surrounded by square brackets, use a second set like this: `[[%col10]]`.

Using the *Identify Features* tool, you can open the *Identify Results* dialog. It includes a *(Derived)* item that contains information relevant to the layer type. The values in this item can be accessed in a similar way to the other fields by proceeding the derived field name with `(Derived).`. For example, a point layer has an `X` and `Y` field, and the values of these fields can be used in the action with `%(Derived).X` and `%(Derived).Y`. The derived attributes are only available from the *Identify Results* dialog box, not the *Attribute Table* dialog box.

Two example actions are shown below:

- `konqueror https://www.google.com/search?q=%name`
- `konqueror https://www.google.com/search?q=%%`

In the first example, the web browser konqueror is invoked and passed a URL to open. The URL performs a Google search on the value of the `name` field from our vector layer. Note that the application or script called by the action must be in the path, or you must provide the full path. To be certain, we could rewrite the first example as: `/opt/kde3/bin/konqueror https://www.google.com/search?q=%name`. This will ensure that the konqueror application will be executed when the action is invoked.

The second example uses the %% notation, which does not rely on a particular field for its value. When the action is invoked, the %% will be replaced by the value of the selected field in the identify results or attribute table.

Duplicating Actions

QGIS allows you to duplicate existing actions. To duplicate an attribute action, open the vector *Layer Properties* dialog and click on the *Actions* tab. In the *Actions* tab, click the  Duplicate an action to open the *Duplicate Action* dialog. You must have selected at least one existing action in order to create a duplicate.

In the dialogue that appears, make any changes that are necessary. See *Defining Actions* for further information. Once finished, press *OK* to create a duplicate of the action with any changes that you made. If you did not edit the description, or if you changed it to be identical to the description of any other existing action, „_1" will be added to the end of it.

Using Actions

QGIS offers many ways to execute actions you enabled on a layer. Depending on their settings, they can be available:

- in the drop-down menu of  Run Feature Action button from the *Attributes toolbar* or *Attribute table* dialog;
- when right-clicking a feature with the  Identify Features tool (see *Identifying Features* for more information);

- from the *Identify Results* panel, under the *Actions* section;
- as items of an *Actions* column in the *Attribute Table* dialog.

If you are invoking an action that uses the `%%` notation, right-click on the field value in the *Identify Results* dialog or the *Attribute Table* dialog that you wish to pass to the application or script.

Here is another example that pulls data out of a vector layer and inserts it into a file using bash and the `echo` command (so it will only work on or perhaps). The layer in question has fields for a species name `taxon_name`, latitude `lat` and longitude `long`. We would like to be able to make a spatial selection of localities and export these field values to a text file for the selected record (shown in yellow in the QGIS map area). Here is the action to achieve this:

```
bash -c "echo \"%taxon_name %lat %long\" >> /tmp/species_localities.txt"
```

After selecting a few localities and running the action on each one, opening the output file will show something like this:

```
Acacia mearnsii -34.0800000000 150.0800000000
Acacia mearnsii -34.9000000000 150.1200000000
Acacia mearnsii -35.2200000000 149.9300000000
Acacia mearnsii -32.2700000000 150.4100000000
```

As an exercise, we can create an action that does a Google search on the `lakes` layer. First, we need to determine the URL required to perform a search on a keyword. This is easily done by just going to Google and doing a simple search, then grabbing the URL from the address bar in your browser. From this little effort, we see that the format is https://www.google.com/search?q=QGIS, where `QGIS` is the search term. Armed with this information, we can proceed:

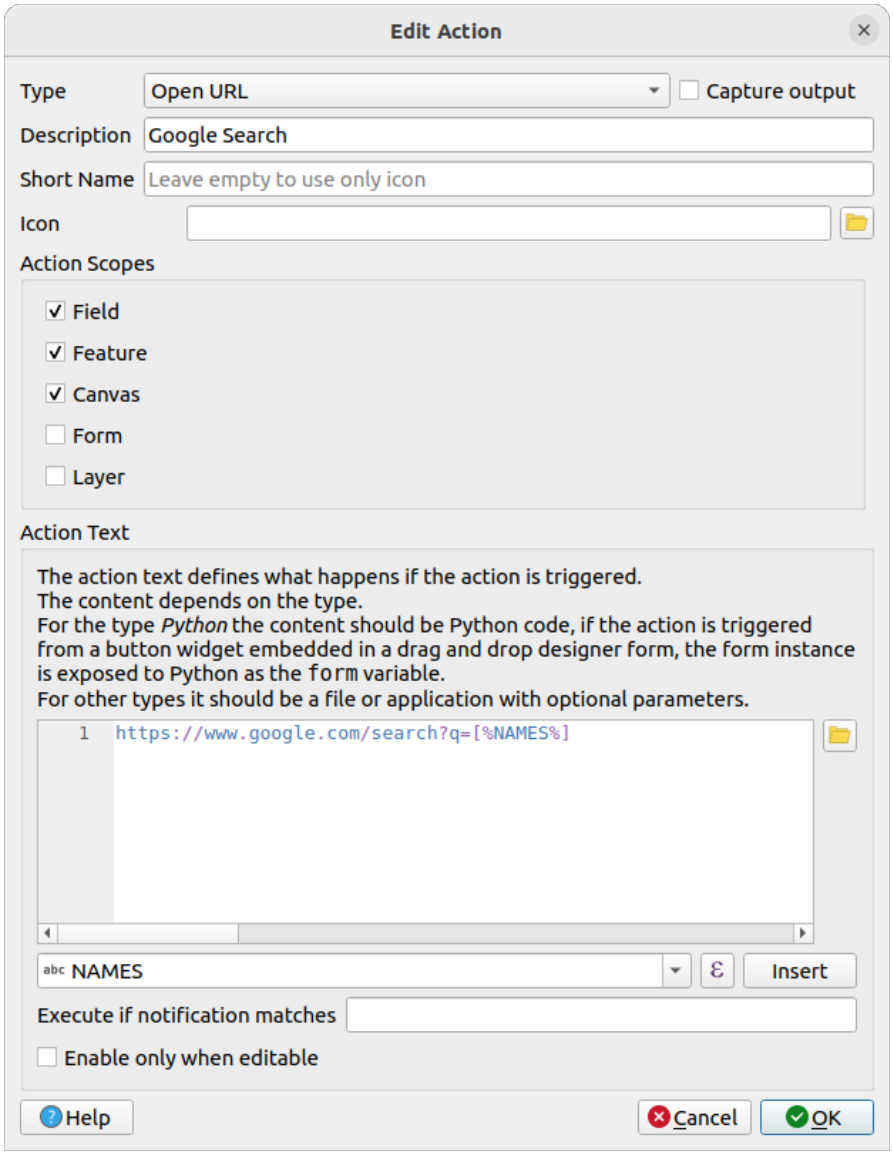
1. Make sure the `lakes` layer is loaded.
2. Open the *Layer Properties* dialog by double-clicking on the layer in the legend, or right-click and choose *Properties* from the pop-up menu.
3. Click on the *Actions* tab.
4. Click Add a new action.
5. Choose the *Open URL* action type,
6. Enter a name for the action, for example `Google Search`.
7. Additionally you can add a *Short Name* or even an *Icon*.
8. Choose the *Action Scopes*. See *Defining Actions* for further information. Leave the default settings for this example.
9. For the action, add the URL used for doing a Google search, up to but not including the search term: `https://www.google.com/search?q=`
10. The text in the *Action* field should now look like this:

    ```
    https://www.google.com/search?q=
    ```

11. Click on the drop-down box containing the field names for the `lakes` layer. It's located just to the left of the *Insert* button.
12. From the drop-down box, select *NAMES* and click *Insert*.
13. Your action text now looks like this:

    ```
    https://www.google.com/search?q=[%NAMES%]
    ```

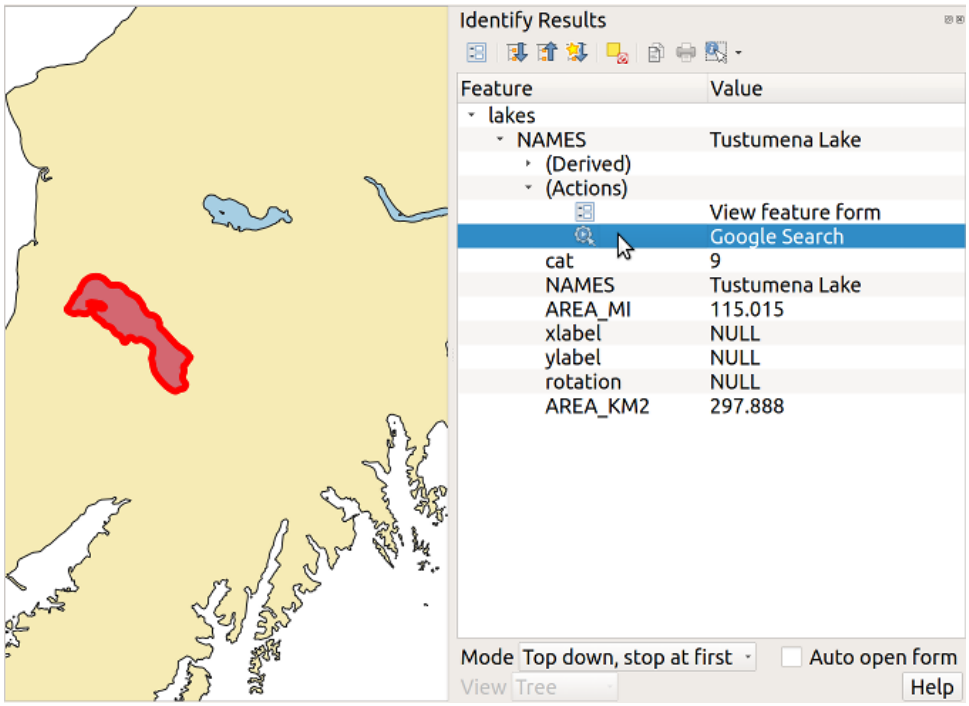
14. To finalize and add the action, click the *OK* button.



Слика12.59: Edit action dialog configured with the example

This completes the action, and it is ready to use.

Close the *Layer Properties* dialog and zoom in to an area of interest. Make sure the `lakes` layer is active and identify a lake. In the result box you'll now see that our action is visible:



Слика12.60: Select feature and choose action

When we click on the action, it brings up our default browser and navigates to the URL https://www.google.com/search?q=Tustumena. It is also possible to add further attribute fields to the action. Therefore, you can add a + to the end of the action text, select another field and click on *Insert Field*. In this example, there is just no other field available that would make sense to search for.

You can define multiple actions for a layer, and each will show up in the *Identify Results* dialog.

You can also invoke actions from the attribute table by selecting a row and right-clicking, then choosing the action from the pop-up menu.

There are all kinds of uses for actions. For example, if you have a point layer containing locations of images or photos along with a file name, you could create an action to launch a viewer to display the image. You could also use actions to launch web-based reports for an attribute field or combination of fields, specifying them in the same way we did in our Google search example.

We can also make more complex examples, for instance, using **Python** actions.

Usually, when we create an action to open a file with an external application, we can use absolute paths, or eventually relative paths. In the second case, the path is relative to the location of the external program executable file. But what about if we need to use relative paths, relative to the selected layer (a file-based one, like Shapefile or SpatiaLite)? The following code will do the trick:

```
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 ] )
```

We just have to remember that the action is one of type *Python* and the *command* and *imagerelpath* variables must be changed to fit our needs.

But what about if the relative path needs to be relative to the (saved) project file? The code of the Python action would be:

```
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 ] )
```

Another Python action example is the one that allows us to add new layers to the project. For instance, the following examples will add to the project respectively a vector and a raster. The names of the files to be added to the project and the names to be given to the layers are data driven (*filename* and *layername* are column names of the table of attributes of the vector where the action was created):

```
qgis.utils.iface.addVectorLayer('/yourpath/[% "filename" %].shp',
  '[% "layername" %]', 'ogr')
```

To add a raster (a TIF image in this example), it becomes:

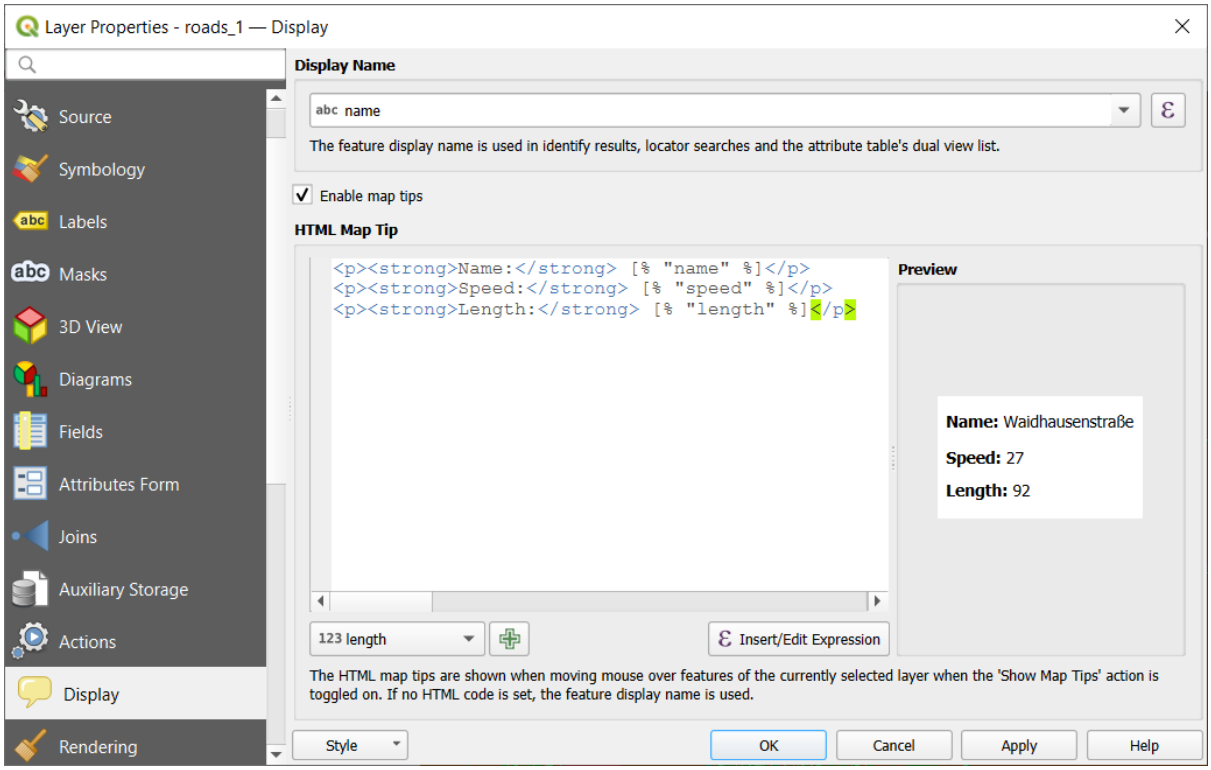
```
qgis.utils.iface.addRasterLayer('/yourpath/[% "filename" %].tif',
  '[% "layername" %]')
```

12.1.13 Display Properties

The *Display* tab helps you configure fields to use for feature identification:

- The *Display name*: based on a field or an *expression*. By default, this is set to the first field in the layer if no field with `<name>` component exists. This is used as:
  - the label shown on top of the feature information in the *Identify tool* results
  - the field used in the *locator bar* when looking for features in all layers
  - the feature identifier in the attribute table *form view*
  - the feature identifier when the map or layout is exported to a layered output format such as Geospatial PDF
  - the map tip information, i.e. the message displayed in the map canvas when hovering over a feature of the active layer with the Show Map Tips icon pressed. Applicable when *Enable Map Tips* is active and no *HTML Map Tip* is set.
- *Enable Map Tips* controls whether to display map tips for the layer
- 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 (also convenient for previewing the *Display name* output). Additionally, you can select and edit existing expressions using the *Insert/Edit Expression* button.

> **Белешка**
>
> Understanding the *Insert/Edit Expression* button behavior
>
> If you select some text within an expression (between „[%” and „%]”), or if no text is selected but the cursor is inside an expression, the whole expression will be automatically selected for editing. If the cursor or a selected text is outside an expression, the dialog opens with the selection.



Слика12.61: HTML code for map tip

To display map tips:

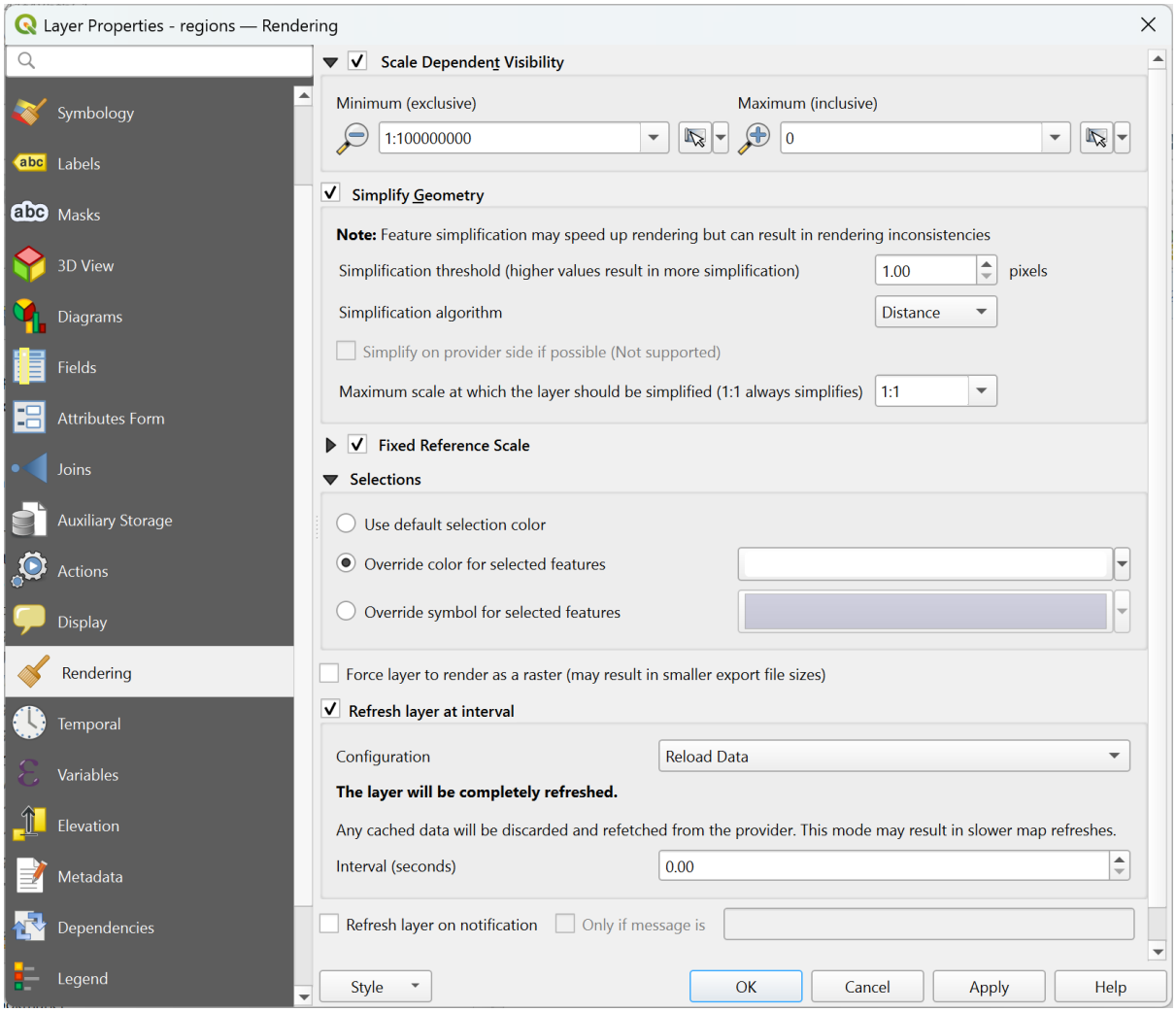
1. Select the menu option *View* ► *Show Map Tips* or click on the Show Map Tips icon of the *Attributes Toolbar*.
2. Make sure that the layer you target is active and has the *Enable Map Tips* property checked.
3. Move over a feature, and the corresponding information will be displayed over.

Map tip is a cross-layer feature meaning that once activated, it stays on and applies to any map tip enabled layer in the project until it's toggled off.



Слика12.62: Map tip made with HTML code

12.1.14 Rendering Properties



Слика12.63: Layer Rendering Properties dialog

The *Rendering* tab offers following properties:

- Under *Scale dependent visibility*, you can set the *Maximum (inclusive)* and *Minimum (exclusive)* scales, defining a range of scales in which features 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 *Visibility Scale Selector* for more information.

  Белешка

  You can also activate scale dependent visibility on a layer from within the *Layers* panel: right-click on the layer and in the contextual menu, select *Set Layer Scale Visibility*.

- QGIS offers support for on-the-fly feature generalisation. This can improve rendering times when drawing many complex features at small scales. This feature can be enabled or disabled in the layer settings using the *Simplify geometry* option. There is also a global setting that enables generalisation by default for newly added layers (see *global simplification* for more information).

Белешка

Feature generalisation may introduce artefacts into your rendered output in some cases. These may include slivers between polygons and inaccurate rendering when using offset-based symbol layers.

- The *Fixed reference scale* indicates the map scale at which symbology and labeling sizes which uses paper--based units (such as millimeters or points) relate to. The sizes will be scaled accordingly whenever the map is viewed at a different scale.

  For instance, a line layer using a 2mm wide line with a 1:2000 *Reference scale* set will be rendered using 4mm wide lines when the map is viewed at 1:1000.

- The *Selections* group allows you to control whether a specific color or symbol should be used in place of the defaults (*Project properties* ► *General* ► *Selection color*) for a particular layer. This is useful to improve the visibility of selected features with certain symbology:

  - *Use default selection color*
  - *Override color for selected features*: e.g. if the layer uses a yellow color by default and the standard yellow selection is not visible.
  - *Override symbol for selected features*: e.g. if a line layer uses a thin symbol, and coloring the lines might not make them visible enough, overriding the symbol with a thicker line can help. Also, if the layer uses raster symbology or gradient fills/lines/shapeburst with color ramp symbology, then the default selection color is not applied at all; being able to set a specific simpler symbol to use for selected features in the layer can help.

- Rendering extremely detailed layers (e.g. polygon layers with a huge number of nodes), can cause layout exports in PDF/SVG format to be huge as all nodes are included in the exported file. This can also make the resultant file very slow to work with/open in other programs.

  Checking *Force layer to render as raster* forces these layers to be rasterised so that the exported files won't have to include all the nodes contained in these layers and the rendering is therefore sped up.

  You can also do this by forcing the layout to export as a raster, but that is an all-or-nothing solution, given that the rasterisation is applied to all layers. Alternatively, you can rely on geometry simplification in *layout export settings*.

- *Refresh layer at interval*: controls whether and how regular a layer can be refreshed. Available *Configuration* options are:

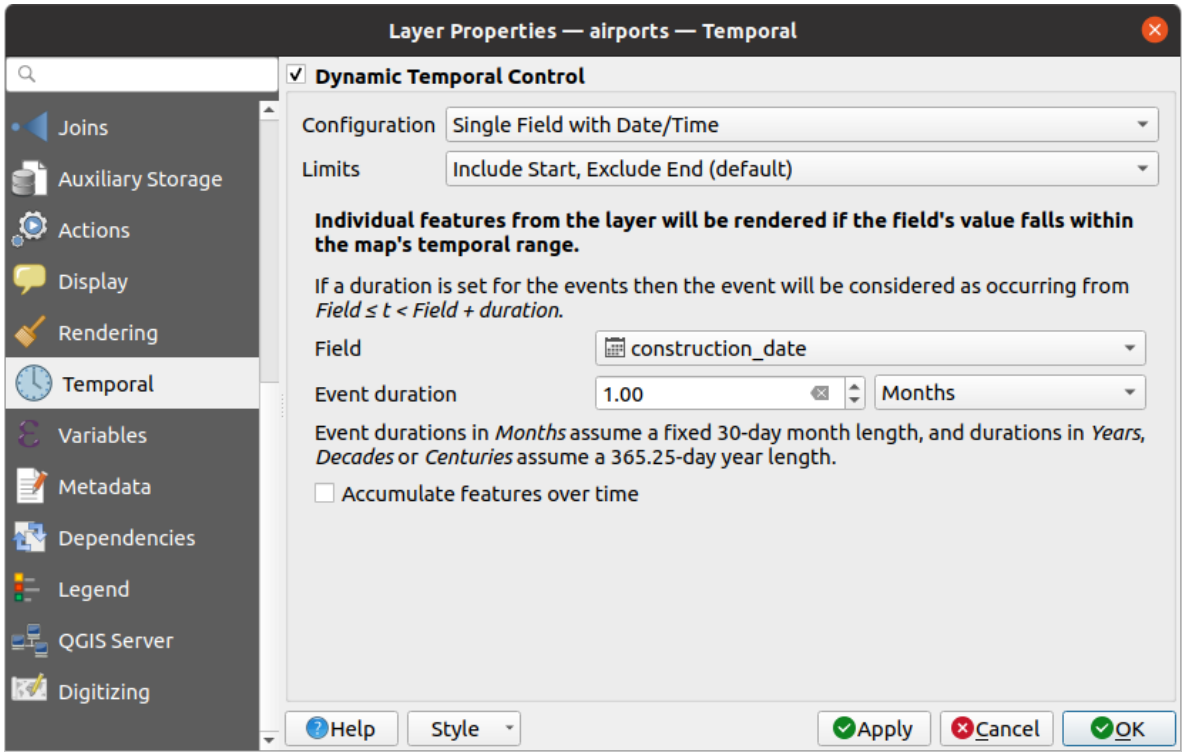
  - *Reload data*: the layer will be completely refreshed. Any cached data will be discarded and refetched from the provider. This mode may result in slower map refreshes.
  - *Redraw layer only*: this mode is useful for animation or when the layer's style will be updated at regular intervals. Canvas updates are deferred in order to avoid refreshing multiple times if more than one layer has an auto update interval set.

  It is also possible to set the *Interval (seconds)* between consecutive refreshments.

- Depending on the data provider (e.g. PostgreSQL), notifications can be sent to QGIS when changes are applied to the data source, out of QGIS. Use the *Refresh layer on notification* option to trigger an update. You can also limit the layer refresh to a specific message set in the *Only if message is* text box.

12.1.15 Temporal Properties

The *Temporal* tab provides options to control the rendering of the layer over time. Such dynamic rendering requires the *temporal navigation* to be enabled over the map canvas.



Слика12.64: Vector layer temporal properties dialog

Check the *Dynamic Temporal Control* option to configure the vector layer temporal rendering. Depending on the structure of your dataset, you may want to use one of the provided *Configuration* options:

- *Fixed time range*: all the features are rendered if the map canvas temporal frame overlaps the given *Start date* and *End date* range.
- *Single field with date/time*: features are rendered if their *Field*'s value falls within the map canvas temporal frame. An *Event duration* can be set. With checking the *Accumulate features over time* option, all features which occur before or within the map's temporal range will continue to be rendered. The event duration is thus ignored.
- *Separate fields for start and end date/time*: features are rendered if the range specified by their *Start field* and *End field* values overlaps the map canvas temporal.
- *Separate fields for start and event duration*: features are rendered if the range defined by their *Start field* and *Event duration field* values overlaps the map canvas temporal.
- *Start and end date/time from expressions*: features are rendered if the time range specified by the fields *Start expression* and *End expression* overlaps the map canvas temporal.
- *Redraw layer only*: the layer is redrawn at each new animation frame but no time-based filtering is applied to the features. It's useful when the layer uses time-based expression values for renderer settings (e.g. data--defined symbology).

It is also possible to set the *Limits* of the features time range as:

- *Include start, exclude end*
- *Include start, include end*

12.1.16 Variables Properties

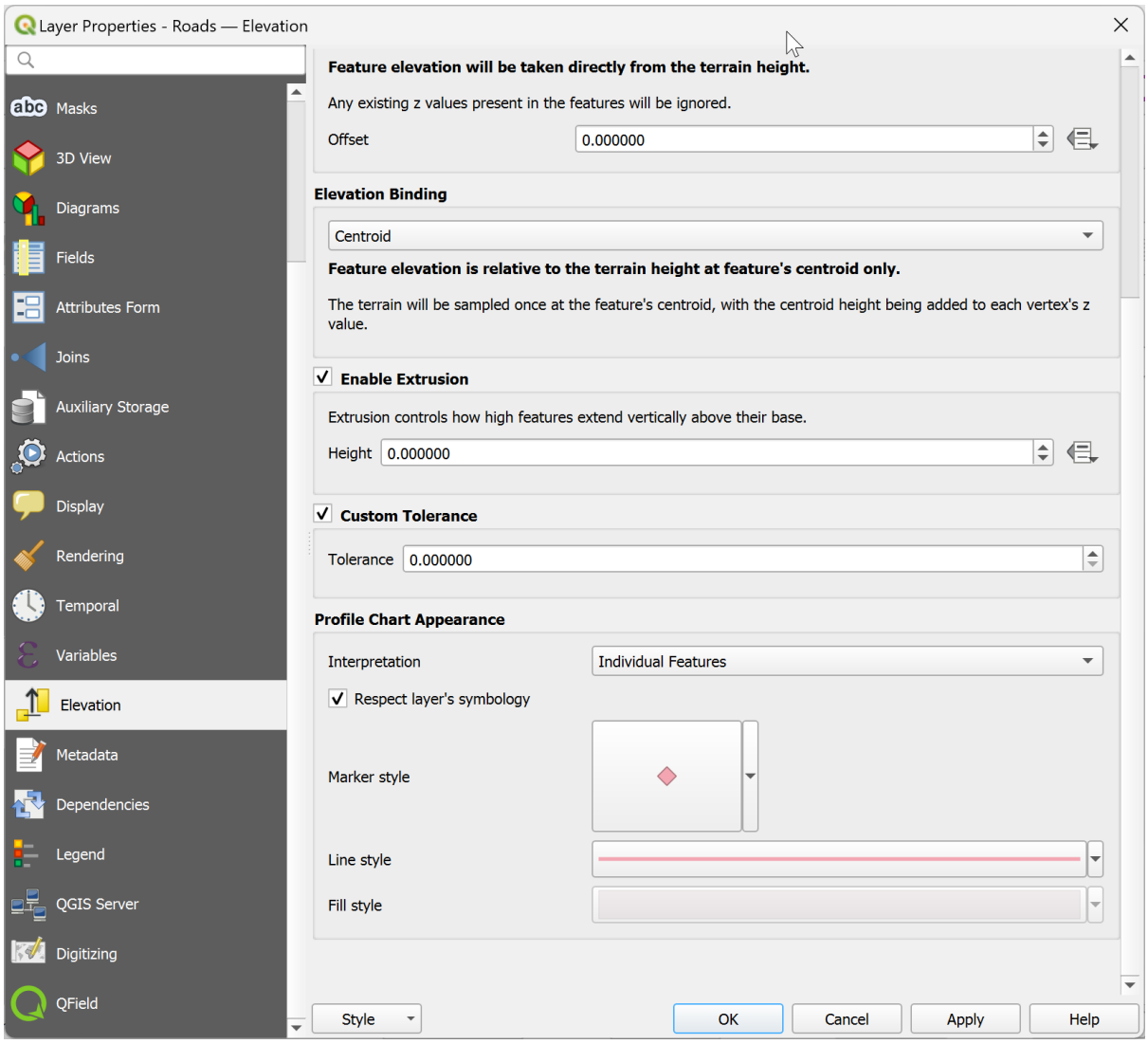
The *Variables* tab lists all the variables available at the layer's level (which includes all global and project's variables).

It also allows the user to manage layer-level variables. Click the button to add a new custom layer-level variable. Likewise, select a custom layer-level variable from the list and click the button to remove it.

More information on variables usage in the General Tools *Storing values in Variables* section.

12.1.17 Elevation Properties

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 set:



Слика12.65: Vector layer elevation properties dialog

- *Vertical Reference System*: If the CRS of your vector layer is a compound one (including a Z dimension), then the vertical CRS used for the layer will be the vertical component of the layer CRS. In this case, you cannot manually set a different vertical CRS. If your layer CRS is horizontal (2D), then you can select a specific vertical CRS by clicking on the Select CRS. Vertical reference systems are supported for vector layers by:
  - *Elevation profiles*
  - *Identify Tool Results*
  - *3D map views*

- *Elevation Clamping*: defines how and whether the features altitude should be:
  - *Clamped to terrain*: takes elevation directly from the terrain height and ignores any existing Z values in the features. A data-defined *Offset* value from the terrain can also be filled.
  - *Relative to terrain*: any existing Z values in the features are added to the terrain height. A *Scale* factor followed by a data-defined *Offset* can be used to adjust the elevation. This option is not available for 2D geometry layers.
  - *Absolute*: ignores the terrain height and directly takes Z values from the features for the elevation. A *Scale* factor followed by a data-defined *Offset* can be used to adjust the elevation. For 2D geometry layers (with no Z values), a data-defined *Base height* can instead be set.
- ☐ *Enable extrusion*: you can set a *Height* to control how high features vertically extend above their base. This is convenient to indicate that a 2D geometry layers, e.g. a polygon building footprints layer, actually represents 3D objects.
- ☐ *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.
- *Elevation Binding*: only relevant when combining an *Elevation clamping* relying on the terrain with a line or polygon layer, this option controls how feature elevation is set relative to the terrain height. The terrain can be sampled:
  - at the feature's *Centroid*, with the centroid height being added to each vertex's z value
  - at every individual *Vertex* before being added to the vertex's z value
- *Profile Chart Appearance*: controls how features are rendered when drawing a profile chart. Two main *Interpretation* modes are available:
  - as *Individual features*: samples discrete positions where the cross section profile line intersects the vector features. That intersection can be represented as point, line or surface depending on the layer type and whether an extrusion is applied.

    With checking ☑ *Respect layer symbology*, features will be rendered on the profile chart with their corresponding *layer styling* (allowing e.g. categorized classes to be visible on the profile chart). If the profile symbol type does not match the layer's renderer symbol types, only the symbol color from the renderer is applied to the profile symbol.

    Depending on the layer settings, profile symbols can be represented with a custom style, using:
    - *Marker style*: for non-extruded point and line features, and for non-extruded polygon features touched by the profile line
    - *Line style*: for extruded point and line features, and for non-extruded polygon features intersected by the profile line
    - *Fill style*: for extruded polygon features
  - as *Continuous Surface (e.g. contours)*: the elevation chart will be rendered as a surface instead of separate features by joining the sampled elevation results into a continuous line. This can enhance visualisation and is designed for vector layers which represent a continuous elevation surface, e.g. contour lines or surveyed elevation points. The profile *Style* can be set as:
    - a *Line* with a specific *Line style*
    - an elevation surface rendered using a fill symbol either above (*Fill above*) or below (*Fill below*) the elevation curve line. The surface symbology is represented using:
      - a *Fill style*
      - and a *Limit*: the maximum (respectively minimum) altitude determining how high the fill surface will be

Moreover, you can check *Show markers at sampled points* to make them visible over the interpretation line and assign them a *Marker style*.

12.1.18 Metadata Properties

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

12.1.19 Dependencies Properties

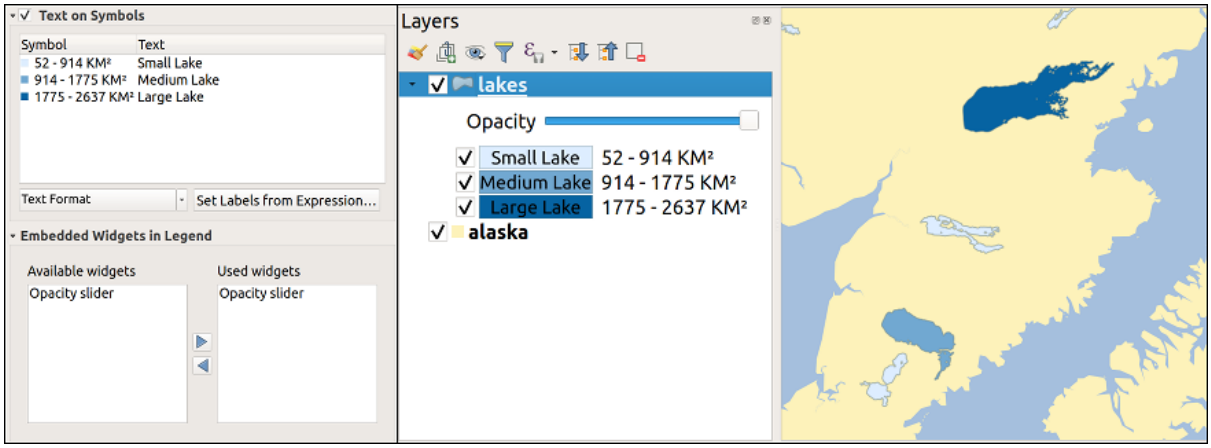
The *Dependencies* tab allows to declare data dependencies between layers. A data dependency occurs when a data modification in a layer, not by direct user manipulation, may modify data of other layers. This is the case for instance when geometry of a layer is updated by a database trigger or custom PyQGIS scripting after modification of another layer's geometry.

In the *Dependencies* tab, you can select any layers which may externally alter the data in the current layer. Correctly specifying dependent layers allows QGIS to invalidate caches for this layer when the dependent layers are altered.

12.1.20 Legend Properties

The *Legend* properties tab provides you with advanced settings for the *Layers panel* and/or the *print layout legend*. These options include:

- Depending on the symbology applied to the layer, you may end up with several entries in the legend, not necessarily readable/useful to display. The *Legend placeholder image* helps you *select an image* for replacement, displayed both in the *Layers* panel and the print layout legend.
- *Show label legend*: Displays overviews of the different label settings as entries in the legends. The *label style* is previewed along with the description.
- *Text on symbols*: In some cases it can be useful to add extra information to the symbols in the legend. With this frame, you can affect to any of the symbols used in the layer symbology a text that is displayed over the symbol, in both *Layers* panel and print layout legend. This mapping is done by typing each text next to the symbol in the table widget or filling the table using the *Set Labels from Expression* button. Text appearance is handled through the font and color selector widgets of the *Text Format* button.



Слика12.66: Setting text on symbols (left) and its rendering in the *Layers* panel (right)

- a list of widgets you can embed within the layer tree in the Layers panel. The idea is to have a way to quickly access some actions that are often used with the layer (setup transparency, filtering, selection, style or other stuff…).

  By default, QGIS provides transparency widget but this can be extended by plugins registering their own widgets and assign custom actions to layers they manage.

12.1.21 QGIS Server Properties

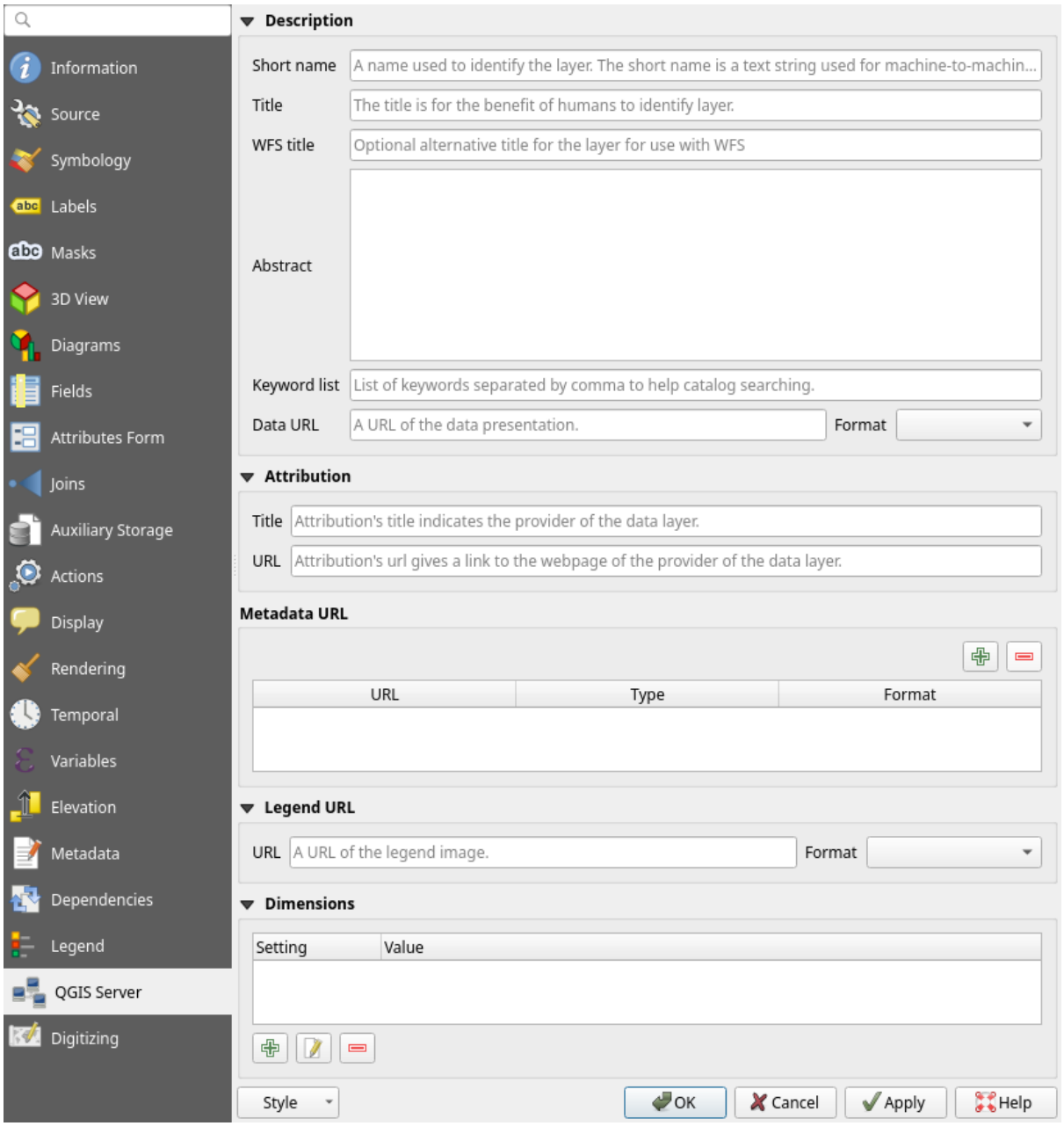
The *QGIS Server* tab consists of *Description*, *Attribution*, *Metadata URL*, and *Legend URL* sections.

From the *Description* section, you can change the *Short name* used to reference the layer in requests (to learn more about short names, read services_basics_short_name). You can also add or edit a *Title*, an alternative *WFS Title* and *Abstract* for the layer, or define a *Keyword list* here. These keyword lists can be used in a metadata catalog. If you want to use a title from an XML metadata file, you have to fill in a link in the *Data URL* field.

Use *Attribution* to get attribute data from an XML metadata catalog.

In *Metadata URL*, you can add the general paths to the XML metadata catalog. This information will be saved in the QGIS project file for subsequent sessions and will be used for QGIS Server.

In the *Legend URL* section, you can provide the url of a legend image in the url field. You can use the Format drop-down option to apply the appropriate format of the image. Currently png, jpg and jpeg image formats are supported.

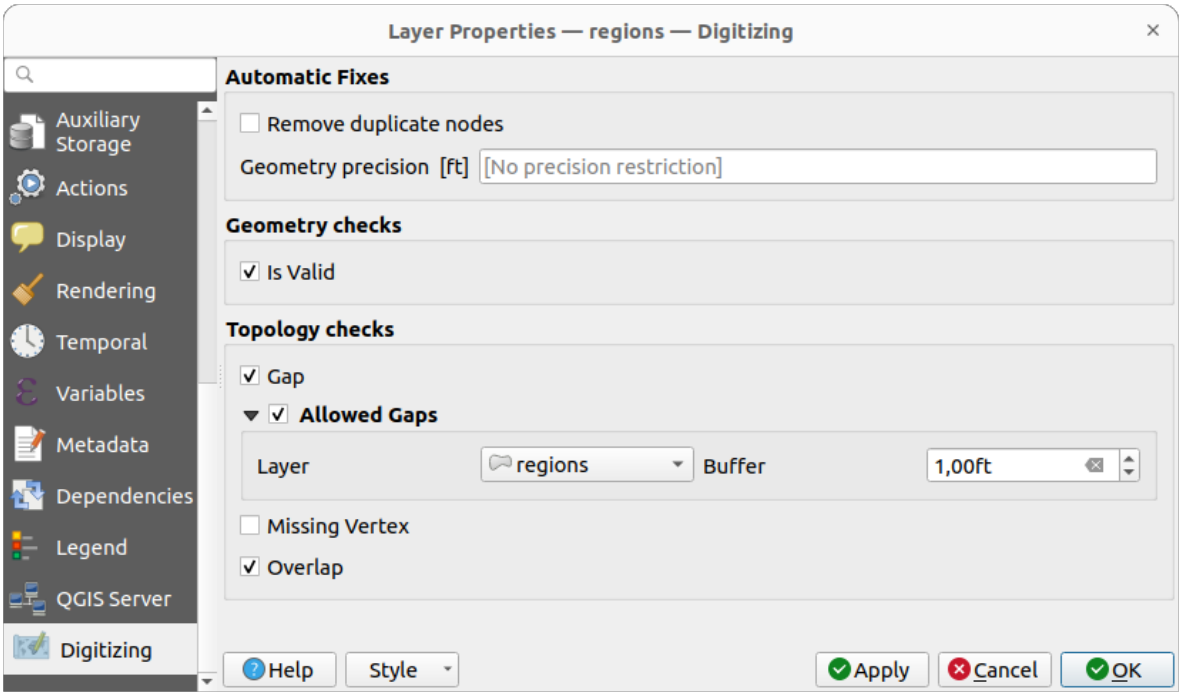


Слика12.67: QGIS Server tab in vector layers properties dialog

To learn more about QGIS Server, read the QGIS-Server-manual.

12.1.22 Digitizing Properties

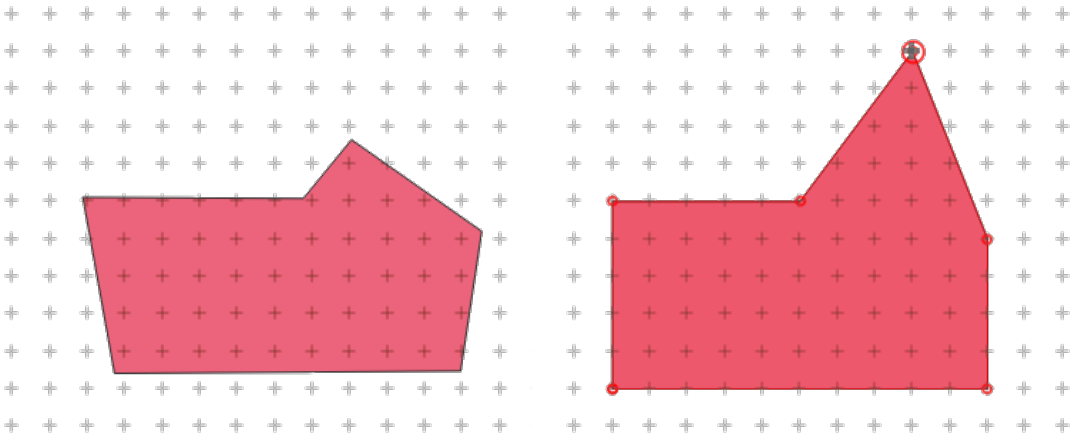
The *Digitizing* tab gives access to options that help to ensure the quality of digitized geometries.



Слика12.68: The QGIS Digitizing tab in the vector layers properties dialog

Automatic Fixes

Options in the *Automatic Fixes* section will directly affect the vertices of any geometry which is added or modified. If the *Remove duplicate nodes* option is checked, any two subsequent vertices with exactly the same coordinates will be removed. If the *Geometry precision* is set, all vertices will be rounded to the closest multiple of the configured geometry precision. The rounding will happen in the layer coordinate reference system. Z and M values are not rounded. With many map tools, a grid is shown on the canvas while digitizing.



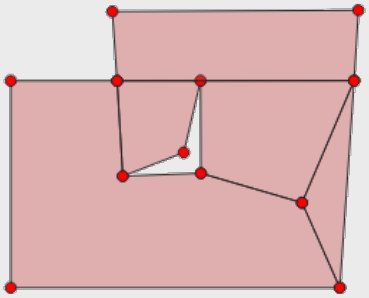
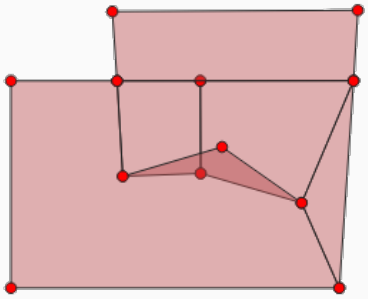
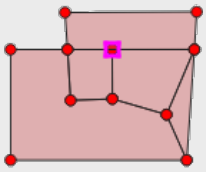
Слика12.69: Moving the top vertex snaps all the vertices to the grid

Geometry Checks

In the *Geometry checks* section, additional validations on a per geometry basis can be activated. Immediately after any geometry modification, failures in these checks are reported to the user in the *Geometry validation* panel. As long as a check is failing, it is not possible to save the layer. The *Is valid* check will run basic validity checks like self intersection on geometries.

Topology Checks

In the *Topology checks* section, additional topology validation checks can be activated. Topology checks will be executed when the user saves the layer. Check errors will be reported in the *Geometry validation* panel. As long as validation errors are present, the layer can not be saved. Topology checks are executed in the area of the bounding box of the modified features. Since other features may be present in the same area, topological errors concerning these features are reported as well as errors introduced in the current edit session.

Topology check option	Illustration
The *Gap* check will check for gaps between neighbouring polygons.	
The *Overlap* check will check for overlaps between neighbouring polygons.	
The *Missing vertex* check will check for shared boundaries of neighbouring polygons where one border misses a vertex which is present on the other one.	

Gap check exceptions

Sometimes it is desirable to keep gaps inside an area in a polygon layer that otherwise is fully covered by polygons. For example, a land use layer may have acceptable holes for lakes. It is possible to define areas that are ignored in the gap check. Since gaps inside these areas are allowed, we will refer to them as *Allowed Gaps* areas.

In the options for the gap checks under *Allowed Gaps*, an *Allowed Gaps layer* can be configured.

Whenever the gap check is executed, gaps which are covered by one or more polygons in the *Allowed Gaps Layer* are not reported as topology errors.

It is also possible to configure an additional *Buffer*. This buffer is applied to each polygon on the *Allowed Gaps Layer*. This makes it possible to make the tests less susceptible to small changes in the outlines at the borders of gaps.

When *Allowed Gaps* are enabled, an additional button (*Add Allowed Gap*) for detected gap errors is available in the geometry validation dock, where gaps are reported during digitizing. If the *Add Allowed Gap* button is pushed, a new polygon with the geometry of the detected gap is inserted into the *Allowed Gaps Layer*. This makes it possible to quickly flag gaps as allowed.

Geometry validation panel

The *Geometry Validation* panel is triggered when any of the abovementioned digitizing checks finds an error. The dialog provides you with the list of errors and their description, and you can to browse the list using the keyboard arrows or dedicated arrows.

You'll need to address all the issues before you can save edits to the layer. To do so:

1. Select an error, and it's possible to:
   -  *Zoom to Feature(s)*
   -  *Zoom to problem*
2. Pick the usual *digitizing tools* to fix the issue.

12.2 Rad sa atributskom tabelom

Atributska tabela prikazuje informacije o objektima izabranog sloja. Svaki red u tabeli predstavlja objekat (sa geometrijom ili bez nje), a svaka kolona sadrži određenu informaciju o objektu. Objekti u tabeli mogu se pretraživati, birati, pomerati ili čak uređivati.

12.2.1 Uvod: prostorne i neprostorne tabele

QGIS vam omogućava da učitate prostorne i neprostorne slojeve. Ovo trenutno uključuje tabele koje podržavaju GDAL i razdvojen tekst, kao i PostgreSQL, MS SQL Server, SpatiaLite i Oracle provajdere. Svi učitani slojevi navedeni su u panelu *Slojevi*. To da li je sloj prostorno omogućen ili ne određuje da li možete da komunicirate s njim na karti.

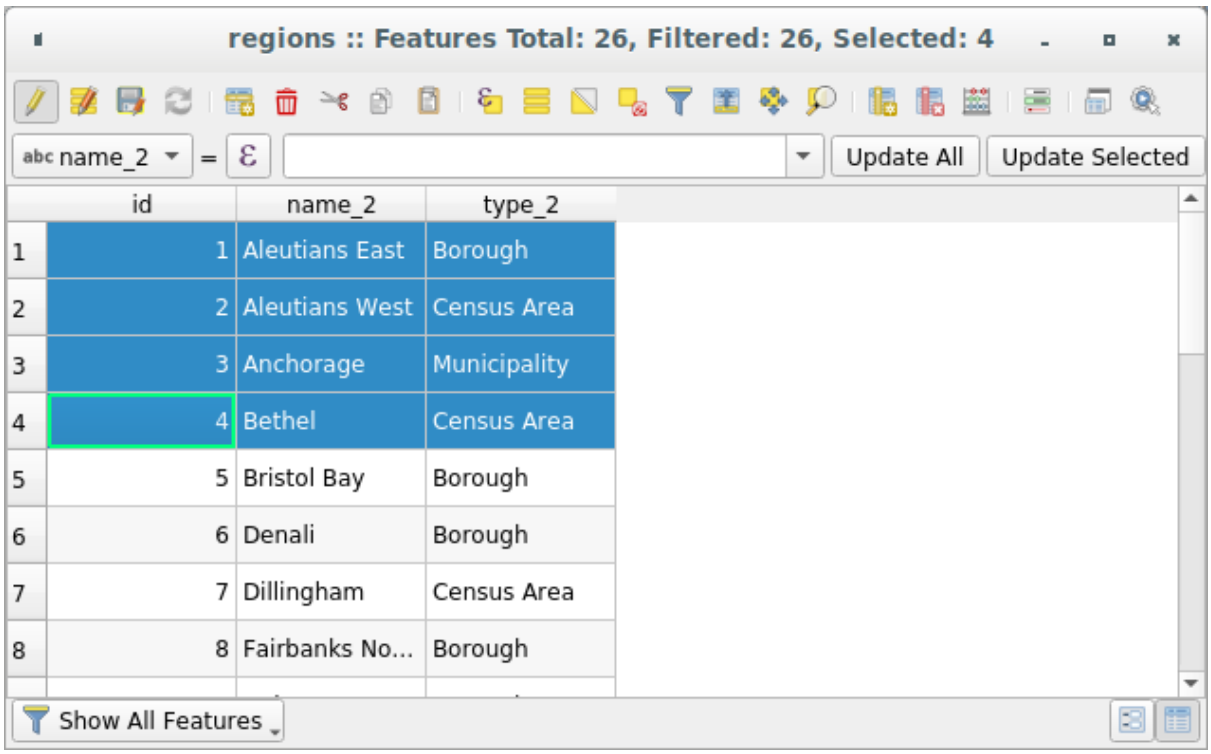
Neprostorne tabele mogu se pregledati i uređivati pomoću prikaza atributske tabele. Osim toga, mogu se koristiti za pretraživanje polja. Na primer, možete koristiti kolone neprostorne tabele da definišete vrednosti atributa, ili opseg dozvoljenih vrednosti, koje se dodaju određenom vektorskom sloju tokom digitalizovanja. Pogledajte pobliže vidžet za uređivanje u odeljku *Attributes Form Properties* da biste saznali više.

12.2.2 Predstavljanje interfejsa atributske tabele

Da biste otvorili atributsku tabelu za vektorski sloj, aktivirajte sloj klikom na njega u *Layers Panel*. Zatim iz glavnog menija *Sloj* izaberite  *Otvori atributsku tabelu*. Takođe je moguće kliknuti desnim tasterom na sloj i izabrati  *Otvori atributsku tabelu* iz padajućeg menija, ili kliknuti na dugme  *Otvori atributsku tabelu* u paleti alatki

Atributi. Ako preferirate prečice, `F6` će otvoriti atributsku tabelu. `Shift+F6` će otvoriti atributsku tabelu filtriranu na izabrane objekte, a `Ctrl+F6` će otvoriti atributsku tabelu filtriranu na vidljive objekte.

Time će se otvoriti novi prozor koji prikazuje atribute objekata sloja (*figure_attributes_table*). Prema podešavanju u meniju *Podešavanja* ► *Opcije* ► *Izvori podataka*, atributska tabela će se otvoriti u usidrenom prozoru ili običnom prozoru. Ukupan broj objekata u sloju i broj trenutno izabranih/filtriranih objekata prikazani su u naslovu atributske tabele, kao i da li je sloj prostorno ograničen.



	id	name_2	type_2
1	1	Aleutians East	Borough
2	2	Aleutians West	Census Area
3	3	Anchorage	Municipality
4	4	Bethel	Census Area
5	5	Bristol Bay	Borough
6	6	Denali	Borough
7	7	Dillingham	Census Area
8	8	Fairbanks No...	Borough

Слика12.70: Atributska tabela za sloj regiona

Dugmad na vrhu prozora atributske tabele pružaju sledeću funkcionalnost:

Табела12.1: Dostupne alatke

Ikona	Oznaka	Svrha	Podrazumevana prečica
	Uključi/isključi režim uređivanja	Omogući funkcionalnosti uređivanja	`Ctrl+E`
	Uključi/isključi režim višestrukog uređivanja	Ažuriraj više polja mnogih objekata	
	Sačuvaj izmene	Sačuvaj trenutne izmene	
	Ponovo učitaj tabelu		
► ([1])	Dodaj objekat	Dodaj novi objekat bez geometrije	
►	Dodaj objekat preko atributske tabele	Umeće novi prazan red u atributsku tabelu koji treba popuniti	
►	Dodaj objekat preko atributskog obrasca	Otvara obrazac atributa objekta za unos podataka	

наставак на следећој страни

Табела 12.1 – настављено са претходне стране

Ikona	Oznaka	Svrha	Podrazumevana prečica
	Obriši izabrane objekte	Ukloni izabrane objekte iz sloja	
	Iseci izabrane objekte u ostavu		`Ctrl+X`
	Kopiraj izabrane objekte u ostavu		`Ctrl+C`
	Nalepi objekte iz ostave	Umetni nove objekte iz kopiranih	`Ctrl+V`
	Izaberi objekte pomoću izraza		
	Izaberi sve	Izaberi sve objekte u sloju	`Ctrl+A`
	Obrni izbor	Obrni trenutni izbor u sloju	`Ctrl+R`
	Poništi ceo izbor	Poništi izbor svih objekata u trenutnom sloju	`Ctrl+Shift+A`
	Filtriraj/izaberi objekte pomoću obrasca		`Ctrl+F`
	Pomeri izabrano na vrh	Pomeri izabrane redove na vrh tabele	
	Pomeri kartu na izabrane redove		`Ctrl+P`
	Uveličaj kartu na izabrane redove		`Ctrl+J`
	Novo polje	Dodaj novo polje u izvor podataka	`Ctrl+W`
	Obriši polje	Ukloni polje iz izvora podataka	
	Organizuj kolone	Prikaži/sakrij polja iz atributske tabele	
	Otvori kalkulator polja	Ažuriraj polje za mnoge objekte odjednom	`Ctrl+I`
	Uslovno formatiranje	Omogući formatiranje tabele	
	Usidri atributsku tabelu	Omogućava usidravanje/odsidravanje atributske tabele	
	Actions	Navodi radnje povezane sa slojem	

[1] Pritisnite i držite da biste izabrali da li da dodate objekat preko atributske tabele ili obrasca.

Пажња

U zavisnosti od formata podataka i GDAL biblioteke sa kojom je izgrađena vaša verzija QGIS-a, neke alatke možda neće biti dostupne.

Ispod ove dugmadi je traka za brzo računanje polja (omogućena samo u *režimu uređivanja*), koja omogućava brzu primenu izračunavanja na sve ili deo objekata u sloju. Ova traka koristi iste *izraze* kao  Kalkulator polja (pogledajte *Uređivanje vrednosti atributa*).

Prikaz tabele naspram prikaza obrasca

QGIS pruža dva režima prikaza za laku manipulaciju podacima u atributskoj tabeli:

-  Prikaz tabele prikazuje vrednosti više objekata u tabelarnom režimu, gde svaki red predstavlja objekat, a svaka kolona polje. Desni klik na zaglavlje kolone omogućava vam da *konfigurišete prikaz tabele*, dok desni klik na ćeliju pruža *interakciju sa objektom*.

  Atributska tabela podržava skrolovanje pomoću `Shift`+`Mouse Wheel` u režimu prikaza tabele radi prebacivanja između vertikalnog i horizontalnog kretanja skrolovanja. Ovo se takođe može postići zamenom miša trekpadom na macOS-u.

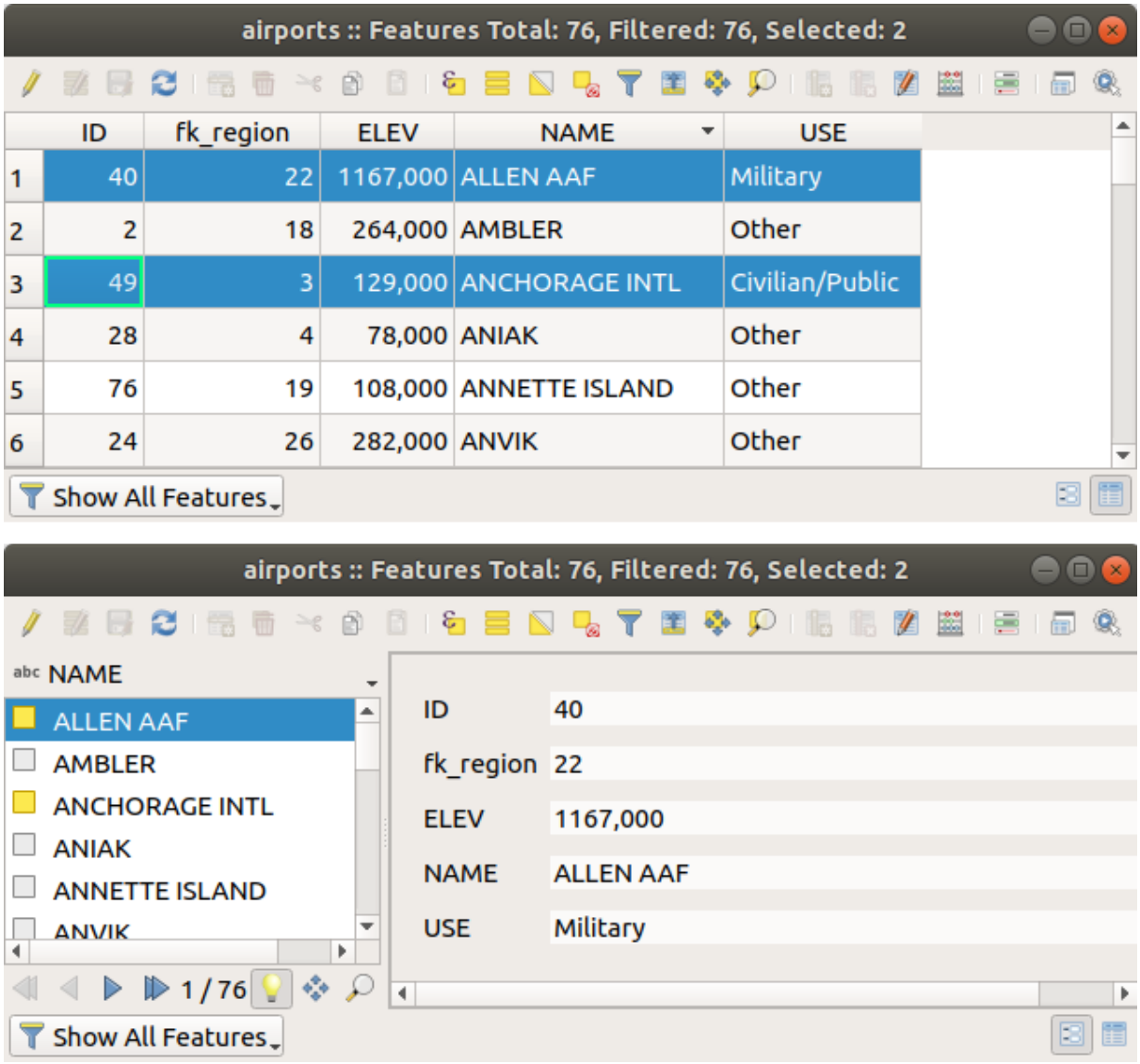
-  Prikaz obrasca prikazuje *identifikatore objekata* u prvom panelu i prikazuje samo atribute objekta na koji je kliknuto u drugom. Na vrhu prvog panela nalazi se padajući meni gde se „identifikator“ može ažurirati pomoću atributa (*Kolone* ►) ili *izraza*. Meni *Sortiraj…* ► omogućava vam da sortirate vraćene identifikatore *Po prikazanom nazivu (rastuće)*, *Po prikazanom nazivu (opadajuće)* ili *Po prilagođenom izrazu*. Padajući meni takođe uključuje poslednjih 10 izraza za ponovnu upotrebu. Prikaz obrasca koristi konfiguraciju polja sloja (pogledajte *Attributes Form Properties*).

  Kroz identifikatore objekata možete se kretati pomoću strelica na dnu prvog panela. Atributi objekata se ažuriraju u drugom panelu kako idete. Takođe je moguće identifikovati ili preći na aktivni objekat na platnu karte pritiskanjem bilo kog dugmeta na dnu:

  -  Istakni trenutni objekat ako je vidljiv na platnu karte
  -  Automatski pomeri na trenutni objekat
  -  Uveličaj na trenutni objekat

Iz jednog režima u drugi možete se prebaciti klikom na odgovarajuću ikonu u donjem desnom uglu dijaloga.

Takođe možete zadati režim *Podrazumevani prikaz* pri otvaranju atributske tabele u meniju *Podešavanja* ► *Opcije* ► *Izvori podataka*. To može biti „Zapamti poslednji prikaz“, „Prikaz tabele“ ili „Prikaz obrasca“.



Слика12.71: Atributska tabela u prikazu tabele (gore) naspram prikaza obrasca (dole)

Konfigurisanje kolona

Kliknite desnim tasterom u zaglavlje kolone dok ste u prikazu tabele da biste imali pristup alatkama koje vam pomažu da kontrolišete:

- *veličinu kolone(a)*
- *vidljivost i redosled kolone(a)*
- *redosled sortiranja podataka*

Promena širine kolona

Širina kolona može se postaviti desnim klikom na zaglavlje kolone i izborom bilo:

- *Postavi širinu…* da biste uneli željenu vrednost. Podrazumevano se u vidžetu prikazuje trenutna vrednost
- *Postavi širine svih kolona…* na istu vrednost
- *Automatska veličina* za promenu veličine kolone na najbolje uklapanje.
- *Automatska veličina svih kolona*

Veličina kolone se takođe može promeniti prevlačenjem granice sa desne strane njenog zaglavlja. Nova veličina kolone se održava za sloj i vraća pri sledećem otvaranju atributske tabele.

U *Podešavanjima izvora podataka* možete izabrati *Podrazumevano automatski postavi veličinu svih kolona pri otvaranju atributske tabele*, čime će „Automatska veličina svih kolona“ postati podrazumevani prikaz svaki put kada se atributske tabele otvore u QGIS-u.

Skrivanje i organizovanje kolona i omogućavanje radnji

Desnim klikom u zaglavlje kolone možete izabrati *Sakrij kolonu* iz atributske tabele (u režimu „prikaz tabele“). Za naprednije kontrole pritisnite dugme Organizuj kolone… iz palete alatki dijaloga ili izaberite *Organizuj kolone…* u kontekstnom meniju zaglavlja kolone.

U novom dijalogu možete:

- označiti/odznačiti kolone koje želite da prikažete ili sakrijete: skrivena kolona će nestati iz svake instance dijaloga atributske tabele dok se aktivno ne vrati. Takođe je moguće:
  - izabrati *Prikaži sve* da biste prikazali sva polja (kolone) i radnje u tabeli
  - izabrati *Sakrij sve* da biste sakrili sva polja (kolone) i radnje u tabeli
  - koristiti *Preokreni izbor* da biste obrnuli vidljivost trenutnog izbora kolona. Za izbor više kolona možete koristiti *kombinaciju tastera*.
- prevući i ispustiti stavke radi preuređivanja kolona u atributskoj tabeli. Imajte u vidu da je ova izmena za iscrtavanje tabele i ne menja redosled polja u izvoru podataka sloja
- dodati novu virtuelnu kolonu *Radnje* koja u svakom redu prikazuje padajući okvir ili listu dugmadi omogućenih radnji. Za više informacija o radnjama pogledajte *Actions Properties*.

Sortiranje redova

Redovi se mogu sortirati po bilo kojoj koloni, klikom na zaglavlje kolone. Mala strelica označava redosled sortiranja (strelica nadole znači opadajuće vrednosti od gornjeg reda naniže, strelica nagore znači rastuće vrednosti od gornjeg reda naniže). Takođe možete izabrati sortiranje redova pomoću opcije *Sortiraj…* u kontekstnom meniju zaglavlja kolone i napisati izraz. Npr. za sortiranje redova pomoću više kolona možete napisati `concat(col0, col1)`.

U prikazu obrasca, identifikatori objekata mogu se sortirati pomoću opcije *Sortiraj po izrazu pregleda*.

Imajte u vidu da sortiranje redova utiče samo na iscrtavanje tabele i ne menja redosled objekata u izvoru podataka sloja.

Савет

Sortiranje na osnovu kolona različitih tipova

Pokušaj sortiranja atributske tabele na osnovu kolona niskovnog i numeričkog tipa može dovesti do neočekivanog rezultata jer izraz `concat("USE", "ID")` vraća niskovne vrednosti (tj. `'Borough105' < 'Borough6'`). Ovo možete zaobići korišćenjem npr. `concat("USE", lpad("ID", 3, 0))` koji vraća `'Borough105' > 'Borough006'`.

Formatiranje ćelija tabele pomoću uslova

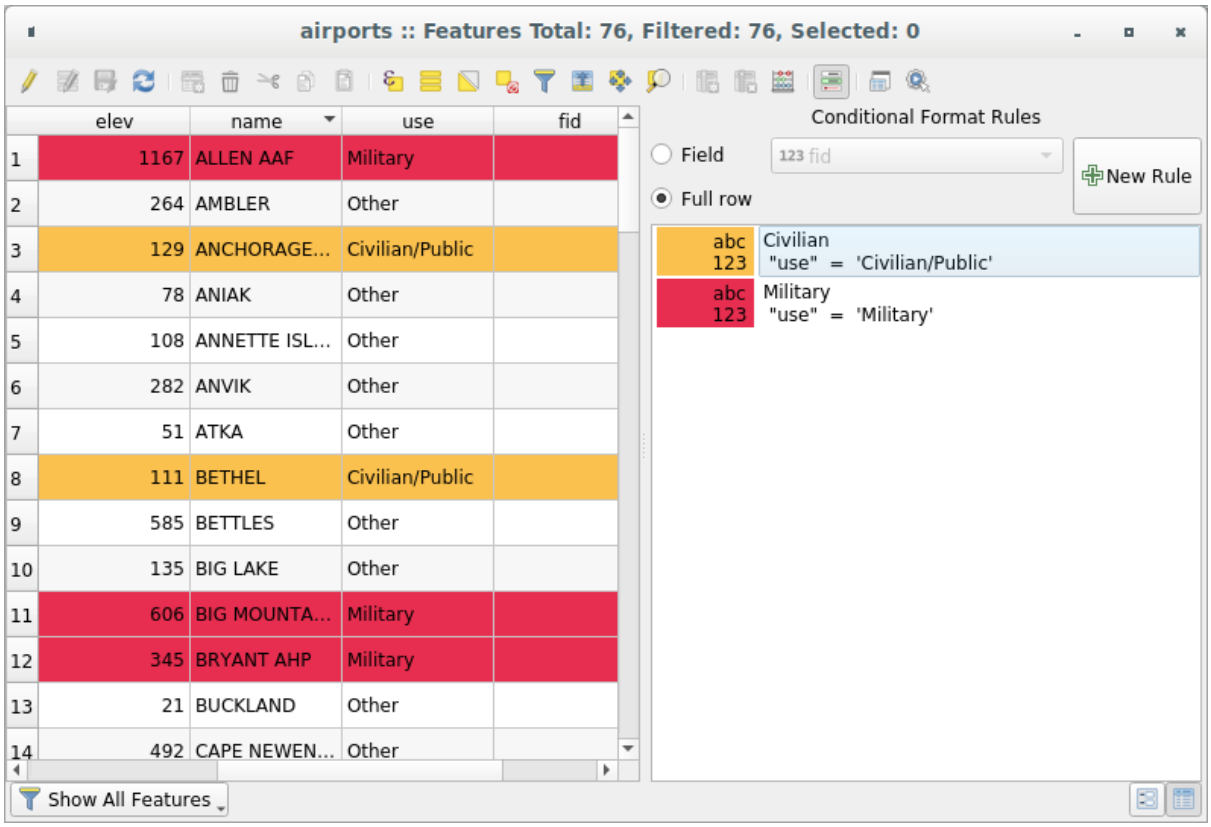
Podešavanja uslovnog formatiranja mogu se koristiti za isticanje objekata u atributskoj tabeli na koje biste možda želeli da stavite poseban fokus, pomoću prilagođenih uslova na:

- geometriju objekta (npr. prepoznavanje višedelnih objekata, onih male površine ili u definisanom obuhvatu karte…);
- ili vrednost polja (npr. poređenje vrednosti sa pragom, prepoznavanje praznih ćelija, duplikata, …).

Panel uslovnog formatiranja možete omogućiti klikom na dugme Uslovno formatiranje u gornjem desnom uglu prozora atributa u prikazu tabele (ne pokreće se u prikazu obrasca).

Novi panel omogućava korisniku da doda nova pravila za formatiranje iscrtavanja *Polja* ili *Celog reda*. Dodavanje novog pravila otvara obrazac za definisanje:

- naziva pravila;
- uslova pomoću bilo koje funkcije *graditelja izraza*;
- formatiranja: može se izabrati iz liste unapred definisanih formata ili napraviti na osnovu svojstava kao što su:
  - boje pozadine i teksta;
  - korišćenje ikone;
  - podebljano, kurziv, podvučeno ili precrtano;
  - font.



Слика12.72: Uslovno formatiranje atributske tabele

12.2.3 Interakcija sa objektima u atributskoj tabeli

Izbor objekata

U prikazu tabele, svaki red u atributskoj tabeli prikazuje atribute jedinstvenog objekta u sloju. Izbor reda bira objekat, a isto tako, izbor objekta na platnu karte (u slučaju sloja sa geometrijom) bira red u atributskoj tabeli. Ako se skup objekata izabranih na platnu karte (ili atributskoj tabeli) promeni, tada se izbor u skladu s tim ažurira i u atributskoj tabeli (ili na platnu karte).

Redovi se mogu izabrati klikom na broj reda sa leve strane reda. **Više redova** može se označiti držanjem tastera `Ctrl`. **Neprekidan izbor** može se napraviti držanjem tastera `Shift` i klikom na nekoliko zaglavlja redova sa leve

strane redova. Svi redovi između trenutnog položaja pokazivača i reda na koji je kliknuto biće izabrani. Pomeranje položaja pokazivača u atributskoj tabeli, klikom na ćeliju u tabeli, ne menja izbor reda. Promena izbora na glavnom platnu ne pomera položaj pokazivača u atributskoj tabeli.

U prikazu obrasca atributske tabele, objekti se podrazumevano identifikuju u levom panelu vrednošću svog prikazanog polja (pogledajte *Display Properties*). Ovaj identifikator se može zameniti pomoću padajuće liste na vrhu panela, bilo izborom postojećeg polja ili korišćenjem prilagođenog izraza. Takođe možete izabrati sortiranje liste objekata iz padajućeg menija.

Kliknite na vrednost u levom panelu da biste prikazali atribute objekta u desnom. Da biste izabrali objekat, potrebno je da kliknete unutar kvadratnog simbola sa leve strane identifikatora. Podrazumevano simbol postaje žut. Kao u prikazu tabele, možete izabrati više objekata pomoću prethodno izloženih kombinacija tastera.

Osim izbora objekata mišem, možete izvršiti automatski izbor na osnovu atributa objekta pomoću alatki dostupnih u paleti alatki atributske tabele, kao što su (za više informacija i primere upotrebe pogledajte odeljak *Automatic selection* i naredne):

- *Izaberi pomoću izraza…*
- *Izaberi objekte po vrednosti…*
- *Poništi izbor svih objekata iz sloja*
- *Izaberi sve objekte*
- *Obrni izbor objekata.*

Takođe je moguće *izabrati objekte pomoću obrazaca*.

Filtriranje objekata

Kada izaberete objekte u atributskoj tabeli, možda ćete želeti da prikažete samo te zapise u tabeli. To se lako radi pomoću stavke *Prikaži izabrane objekte* iz padajuće liste u donjem levom uglu dijaloga atributske tabele. Ova lista nudi sledeće filtere:

- *Prikaži sve objekte*
- *Prikaži izabrane objekte* - isto kao korišćenje *Otvori atributsku tabelu (izabrani objekti)* iz menija *Sloj* ili *palete alatki Atributi* ili pritiskanje `Shift+F6`
- *Prikaži objekte vidljive na karti* - isto kao korišćenje *Otvori atributsku tabelu (vidljivi objekti)* iz menija *Sloj* ili *palete alatki Atributi* ili pritiskanje `Ctrl+F6`
- *Prikaži objekte sa neispunjenim ograničenjima* - objekti će biti filtrirani da prikažu samo one koji imaju neispunjena *ograničenja*. U zavisnosti od toga da li je neispunjeno ograničenje tvrdo ili meko, neispravne vrednosti polja prikazuju se redom u tamno ili svetlo narandžastim ćelijama.
- *Prikaži uređene i nove objekte* - isto kao korišćenje *Otvori atributsku tabelu (uređeni i novi objekti)* iz menija *Sloj* ili *palete alatki Atributi*
- *Filter polja* - omogućava korisniku da filtrira na osnovu vrednosti polja: izaberite kolonu iz liste, unesite ili izaberite vrednost i pritisnite `Enter` za filtriranje. Zatim se u atributskoj tabeli prikazuju samo objekti koji odgovaraju izrazu `num_field = value` ili `string_field ilike '%value%'`. Možete označiti *Osetljivo na velika/mala slova* da biste bili manje popustljivi sa niskama.
- *Napredni filter (izraz)* - Otvara dijalog graditelja izraza. Unutar njega možete praviti *složene izraze* za podudaranje redova tabele. Na primer, možete filtrirati tabelu pomoću više od jednog polja. Kada se primeni, izraz filtera će se pojaviti na dnu obrasca.
- *Sačuvani izrazi filtera* ►: prečica do *sačuvanih izraza* koji se često koriste za filtriranje vaše atributske tabele.

Takođe je moguće *filtrirati objekte pomoću obrazaca*.

Белешка

Filtriranje zapisa iz atributske tabele ne filtrira objekte iz sloja; oni su samo trenutno sakriveni iz tabele i mogu se pristupiti sa platna karte ili uklanjanjem filtera. Za filtere koji sakrivaju objekte iz sloja koristite *Graditelj upita*.

Савет

Ažuriraj filtriranje izvora podataka pomoću `Prikaži objekte vidljive na karti`

Kada su, iz razloga performansi, objekti prikazani u atributskoj tabeli prostorno ograničeni na obuhvat platna pri njenom otvaranju (za uputstvo pogledajte *Opcije izvora podataka*), izbor *Prikaži objekte vidljive na karti* na novom obuhvatu platna ažurira prostorno ograničenje.

Čuvanje izraza filtera

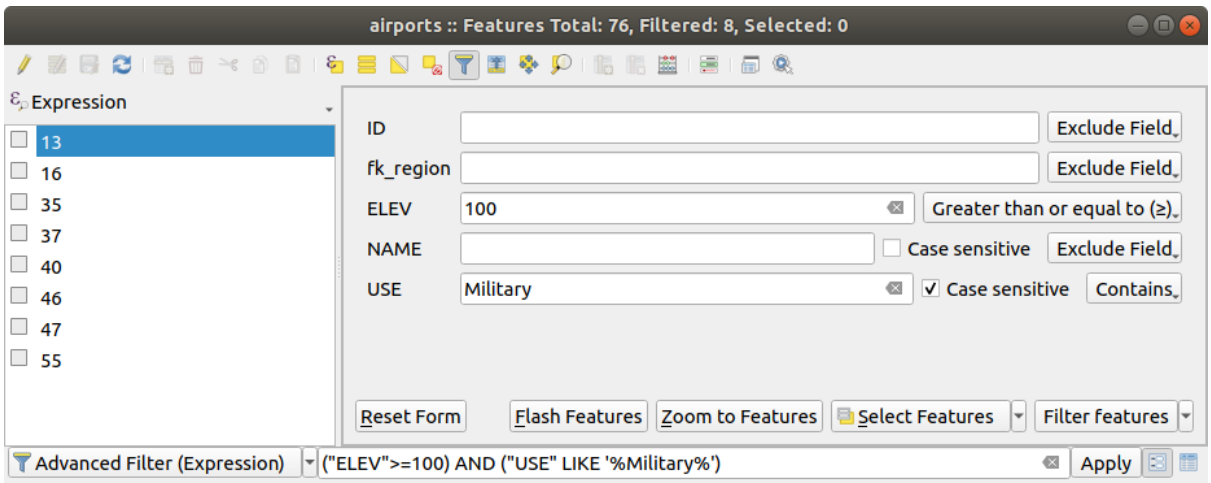
Izrazi koje koristite za filtriranje atributske tabele mogu se sačuvati za naredne pozive. Pri korišćenju stavki *Filter polja* ili *Napredni filter (izraz)*, korišćeni izraz se prikazuje u tekstualnom vidžetu na dnu dijaloga atributske tabele. Pritisnite ☆ Sačuvaj izraz sa tekstom kao nazivom pored okvira da biste sačuvali izraz u projektu. Pritiskanje padajućeg menija pored dugmeta omogućava čuvanje izraza sa prilagođenim nazivom (*Sačuvaj izraz kao…*). Kada se sačuvani izraz prikaže, aktivira se dugme ★, a njegov padajući meni omogućava vam da *Uredite izraz* i naziv ako postoji, ili *Obrišete sačuvani izraz*.

Sačuvani izrazi filtera čuvaju se u projektu i dostupni su preko menija *Sačuvani izrazi filtera* atributske tabele. Razlikuju se od *korisničkih izraza*, koje dele svi projekti aktivnog korisničkog profila.

Filtriranje i izbor objekata pomoću obrazaca

Klik na Filtriraj/izaberi objekte pomoću obrasca ili pritiskanje `Ctrl+F` će prebaciti dijalog atributske tabele u prikaz obrasca i zameniti svaki vidžet njegovom varijantom za pretragu.

Od ove tačke nadalje, funkcionalnost ovog alata slična je onoj opisanoj u *Select Features By Value*, gde možete pronaći opise svih operatora i režima izbora.



Слика12.73: Atributska tabela filtrirana obrascem za filtriranje

Pri izboru / filtriranju objekata iz atributske tabele, postoji dugme *Filtriraj objekte* koje omogućava definisanje i

preciziranje filtera. Njegovo korišćenje pokreće opciju *Napredni filter (izraz)* i prikazuje odgovarajući izraz filtera u tekstualnom vidžetu koji se može uređivati na dnu obrasca.

Ako već postoje filtrirani objekti, filter možete precizirati pomoću padajuće liste pored dugmeta *Filtriraj objekte*. Opcije su:

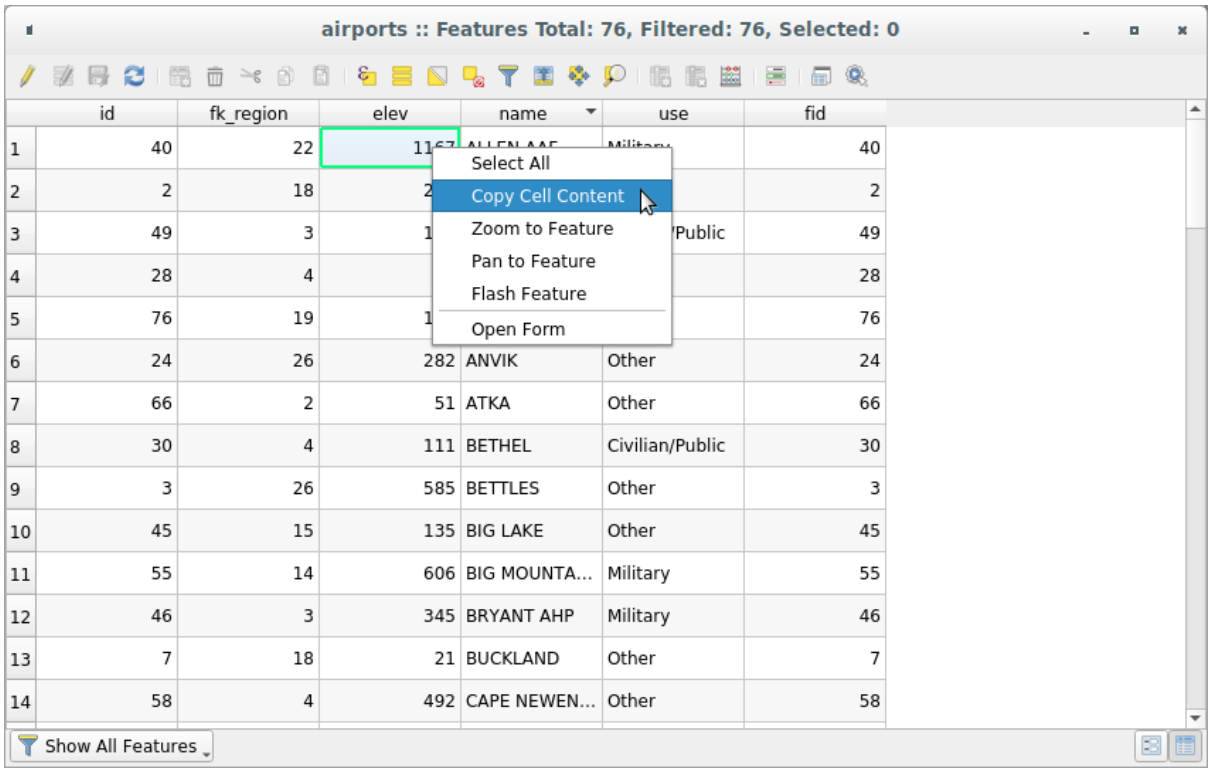
- *Filtriraj unutar („AND“)*
- *Proširi filter („OR“)*

Da biste očistili filter, ili izaberite opciju *Prikaži sve objekte* iz padajućeg menija u donjem levom uglu, ili očistite izraz i kliknite na *Primeni* ili pritisnite `Enter`.

Više radnji nad objektima

Korisnici imaju nekoliko mogućnosti za manipulaciju objektom u atributskoj tabeli. Kliknite desnim tasterom u ćeliju i možete:

- *Izabrati sve* (`Ctrl+A`) objekte;
- Kopirati sadržaj ćelije u ostavu pomoću *Kopiraj sadržaj ćelije*;
- *Uveličati na objekat* bez potrebe da ga prethodno izaberete;
- *Pomeriti na objekat* bez potrebe da ga prethodno izaberete;
- *Bljesni objekat*, da biste ga istakli na platnu karte;
- *Otvori obrazac*: prebacuje atributsku tabelu u prikaz obrasca sa fokusom na objekat na koji je kliknuto.
- Prikazati *listu radnji*, prethodno omogućenih u kartici *Svojstva sloja* ► *Radnje*.



	id	fk_region	elev	name	use	fid
1	40	22	11			40
2	2	18	2			2
3	49	3	1		Public	49
4	28	4				28
5	76	19	1			76
6	24	26	282	ANVIK	Other	24
7	66	2	51	ATKA	Other	66
8	30	4	111	BETHEL	Civilian/Public	30
9	3	26	585	BETTLES	Other	3
10	45	15	135	BIG LAKE	Other	45
11	55	14	606	BIG MOUNTA...	Military	55
12	46	3	345	BRYANT AHP	Military	46
13	7	18	21	BUCKLAND	Other	7
14	58	4	492	CAPE NEWEN...	Other	58

Слика12.74: Dugme Kopiraj sadržaj ćelije

Ako želite da koristite atributske podatke u spoljnim programima (kao što su Excel, LibreOffice ili prilagođena veb aplikacija), izaberite jedan ili više redova i koristite dugme Kopiraj izabrane redove u ostavu ili pritisnite `Ctrl+C`. Osim toga, u meniju *Podešavanja* ► *Opcije* ► *Izvori podataka* možete definisati format u koji se lepi pomoću opcije *Kopiraj objekte kao*. Više detalja na *Data Sources settings*.

12.2.4 Uređivanje vrednosti atributa

Da biste izmenili podatke u atributskoj tabeli, prvo treba da prebacite sloj u režim uređivanja. Pritisnite dugme  Uključi/isključi uređivanje. U zavisnosti od tipa geometrije sloja i stanja ostave, još nekoliko alatki se omogućava u gornjoj paleti alatki atributske tabele.

Uređivanje vrednosti atributa se zatim može obaviti:

- kucanjem nove vrednosti direktno u ćeliju, bilo da je atributska tabela u prikazu tabele ili obrasca. Izmene se dakle vrše ćeliju po ćeliju, objekat po objekat;
- korišćenjem *kalkulatora polja*: ažuriranje odjednom polja koje možda već postoji ili tek treba da bude napravljeno, ali za više objekata. Može se koristiti za pravljenje virtuelnih polja;
- korišćenjem trake za brzo *računanje polja*: isto kao gore, ali samo za postojeće polje;
- ili korišćenjem režima *višestrukog uređivanja*: ažuriranje odjednom više polja za više objekata.

Prebacivanje sloja u režim uređivanja takođe će vam omogućiti da  Nalepite objekte iz ostave (`Ctrl+V`),  Isečete izabrane redove u ostavu (`Ctrl+X`) ili  Obrišete izabrane objekte. Više detalja na *Cutting, Copying and Pasting Features*.

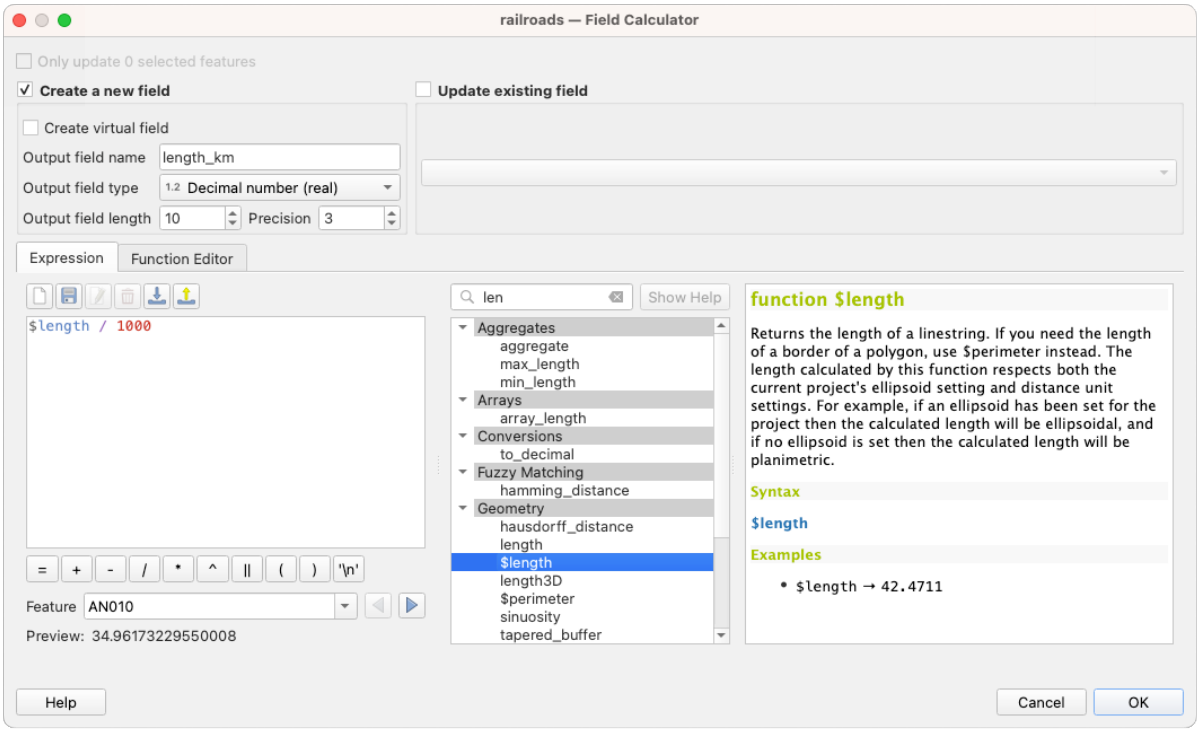
Korišćenje kalkulatora polja

Dugme  Kalkulator polja u atributskoj tabeli omogućava vam da vršite izračunavanja na osnovu postojećih vrednosti atributa ili definisanih funkcija, na primer, za računanje dužine ili površine geometrijskih objekata. Rezultati se mogu koristiti za ažuriranje postojećeg polja ili upisati u novo polje (koje može biti *virtuelno*).

Kalkulator polja dostupan je na bilo kom sloju koji podržava uređivanje. Kada kliknete na ikonu kalkulatora polja, otvara se dijalog (pogledajte Сл.12.75). Ako sloj nije u režimu uređivanja, prikazuje se upozorenje, a korišćenje kalkulatora polja uzrokovaće da sloj bude prebačen u režim uređivanja pre nego što se izvrši izračunavanje.

Zasnovan na dijalogu *Graditelj izraza*, dijalog kalkulatora polja nudi kompletan interfejs za definisanje izraza i njegovu primenu na postojeće ili novonapravljeno polje. Da biste koristili dijalog kalkulatora polja, morate izabrati da li želite da:

1. primenite izračunavanje na ceo sloj ili samo na izabrane objekte
2. napravite novo polje za izračunavanje ili ažurirate postojeće.



Слика12.75: Kalkulator polja

Ako izaberete da dodate novo polje, potrebno je da unesete naziv polja, tip polja (ceo broj, realni, datum ili niska) i, ako je potrebno, ukupnu dužinu polja i preciznost polja. Na primer, ako izaberete dužinu polja 10 i preciznost polja 3, to znači da imate 7 cifara pre tačke i 3 cifre za decimalni deo.

Kratak primer ilustruje kako kalkulator polja radi pri korišćenju kartice *Izraz*. Želimo da izračunamo dužinu u km sloja `railroads` iz QGIS uzornog skupa podataka:

1. Učitajte shapefile `railroads.shp` u QGIS i pritisnite Otvori atributsku tabelu.
2. Kliknite na Uključi/isključi režim uređivanja i otvorite dijalog Kalkulator polja.
3. Izaberite polje za potvrdu *Napravi novo polje* da biste sačuvali izračunavanja u novo polje.
4. Postavite *Naziv izlaznog polja* na `length_km`
5. Izaberite `Decimalni broj (realni)` kao *Tip izlaznog polja*
6. Postavite *Dužinu izlaznog polja* na `10` i *Preciznost* na `3`
7. Dvokliknite na `$length` u grupi *Geometrija* da biste dodali dužinu geometrije u okvir izraza kalkulatora polja (počećete da vidite pregled izlaza, do 60 znakova, ispod okvira izraza koji se ažurira u realnom vremenu kako se izraz sastavlja).
8. Dovršite izraz kucanjem `/ 1000` u okvir izraza kalkulatora polja i kliknite na *OK*.
9. Sada u atributskoj tabeli možete pronaći novo polje *length_km*.

Pravljenje virtuelnog polja

Virtuelno polje je polje zasnovano na izrazu koji se izračunava u hodu, što znači da se njegova vrednost automatski ažurira čim se promeni neki od parametara koji stoje u osnovi. Izraz se primenjuje na sve objekte u sloju i postavlja se jednom; više ne morate ponovo računati polje svaki put kada se osnovne vrednosti promene. Na primer, možda ćete želeti da koristite virtuelno polje ako je potrebno da se površina izračunava dok digitalizujete objekte ili da se automatski računa trajanje između datuma koji se mogu promeniti (npr. pomoću funkcije `now()`).

Pravljenje virtuelnog polja obavlja se kroz dijalog *Kalkulator polja* i prati *isti postupak* kao za obična polja. Samo zapamtite da označite opciju *Napravi virtuelno polje* i koristite tip polja kompatibilan sa podacima koje bi vaš izraz generisao.

Uređivanje virtuelnog polja obavlja se kroz karticu *Polja* dijaloga svojstava sloja (pogledajte *Fields Properties*). Izraz koji definiše polje prikazan je u koloni *Komentar*, a pritiskanje dugmeta ε pored njega otvara prozor uređivača izraza za ažuriranje.

Белешка

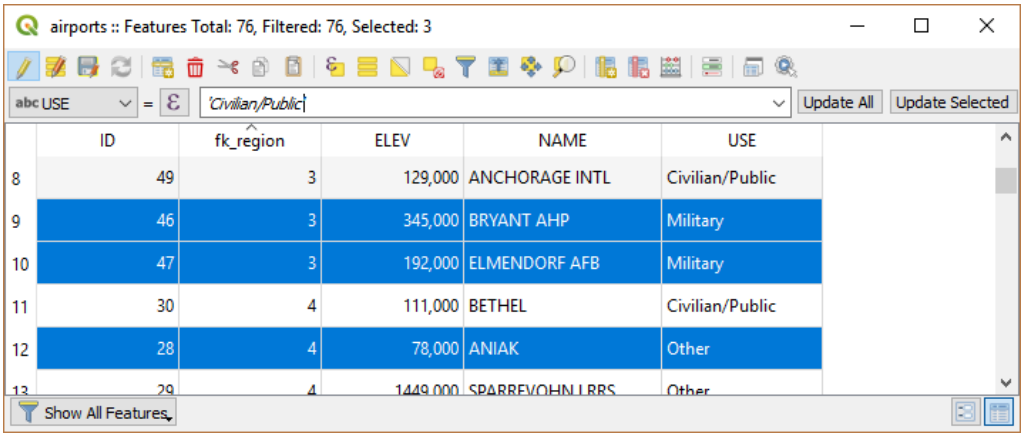
Korišćenje virtuelnih polja

- Polje se može postaviti kao virtuelno samo pri njegovom pravljenju.
- Virtuelna polja nisu trajna u atributima sloja, što znači da se čuvaju i dostupna su samo u datoteci projekta u kojoj su napravljena.

Korišćenje trake za brzo računanje polja

Dok je kalkulator polja uvek dostupan, traka za brzo računanje polja na vrhu atributske tabele vidljiva je samo ako je sloj u režimu uređivanja. Zahvaljujući mehanizmu za izraze, ona nudi brži pristup uređivanju već postojećeg polja:

1. Izaberite polje za ažuriranje u padajućoj listi.
2. Popunite tekstualni okvir vrednošću, izrazom koji direktno napišete ili napravite pomoću dugmeta izraza ε.
3. Kliknite na dugme *Ažuriraj sve*, *Ažuriraj izabrane* ili *Ažuriraj filtrirane* prema svojoj potrebi.



Слика12.76: Traka za brzo računanje polja

Uređivanje više polja

Za razliku od prethodnih alatki, režim višestrukog uređivanja omogućava istovremeno uređivanje više atributa različitih objekata. Kada se sloj prebaci u uređivanje, mogućnosti višestrukog uređivanja su dostupne:

- pomoću dugmeta Uključi/isključi režim višestrukog uređivanja iz palete alatki unutar dijaloga atributske tabele;
- ili izborom menija *Uređivanje* ► *Izmeni atribute izabranih objekata*.

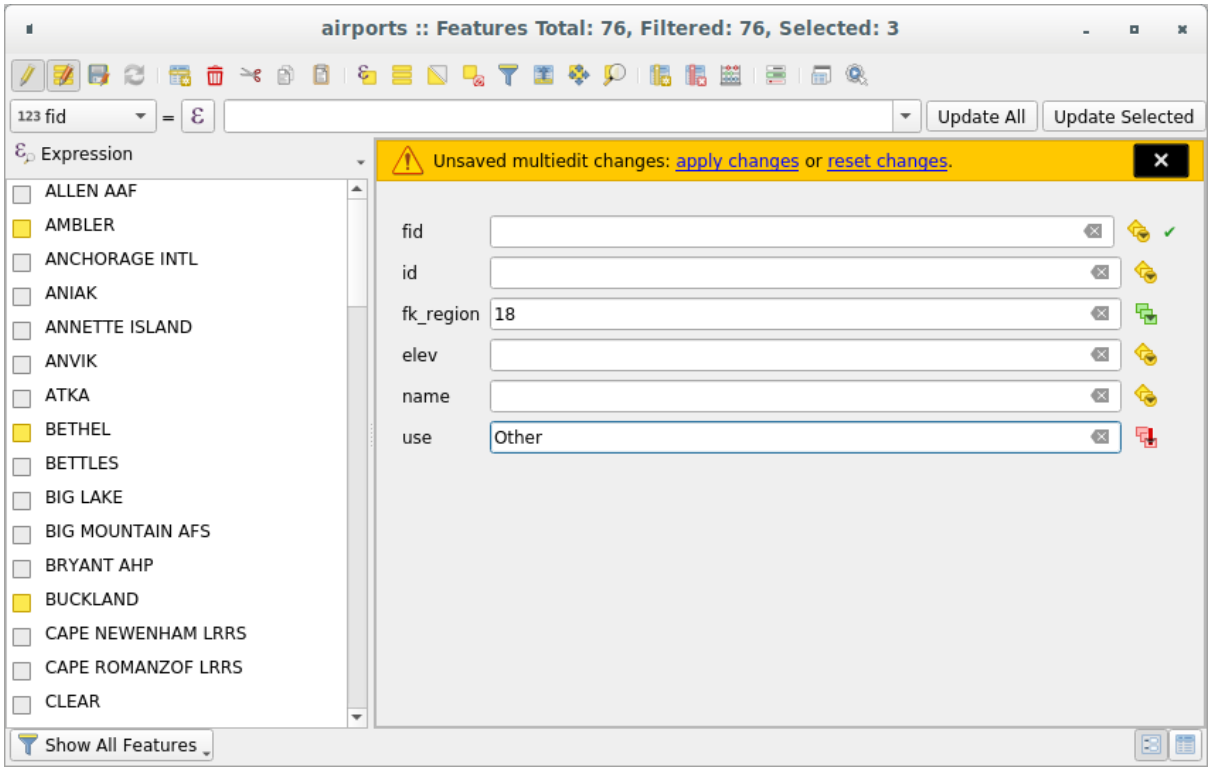
Белешка

Za razliku od alata iz atributske tabele, izbor opcije *Uređivanje* ► *Izmeni atribute izabranih objekata* pruža vam modalni dijalog za popunjavanje izmena atributa. Zato je izbor objekata potreban pre izvršavanja.

Da biste uredili više polja odjednom:

1. Izaberite objekte koje želite da uredite.
2. Iz palete alatki atributske tabele kliknite na dugme . Time će se dijalog prebaciti u prikaz obrasca. Izbor objekata takođe se može napraviti u ovom koraku.
3. Sa desne strane atributske tabele prikazana su polja (i vrednosti) izabranih objekata. Novi vidžeti se pojavljuju pored svakog polja omogućavajući prikaz trenutnog stanja višestrukog uređivanja:
   - Polje sadrži različite vrednosti za izabrane objekte. Prikazano je prazno i svaki objekat će zadržati svoju originalnu vrednost. Vrednost polja možete resetovati iz padajuće liste vidžeta.
   - Svi izabrani objekti imaju istu vrednost za ovo polje i vrednost prikazana u obrascu će biti zadržana.
   - Polje je uređeno i unesena vrednost će biti primenjena na sve izabrane objekte. Poruka se pojavljuje na vrhu dijaloga, pozivajući vas da primenite ili resetujete svoju izmenu.

   Klik na bilo koji od ovih vidžeta omogućava vam da ili postavite trenutnu vrednost za polje ili resetujete na originalnu vrednost, što znači da možete poništiti izmene polje po polje.



Слика12.77: Uređivanje polja više objekata

4. Napravite izmene na poljima koja želite.
5. Kliknite na **Primeni izmene** u gornjem tekstu poruke ili na bilo koji drugi objekat u levom panelu.

Izmene će se primeniti na **sve izabrane objekte**. Ako nijedan objekat nije izabran, cela tabela se ažurira vašim

izmenama. Izmene se vrše kao jedna komanda uređivanja. Zato će pritiskanje  Opozovi poništiti izmene atributa za sve izabrane objekte odjednom.

Белешка

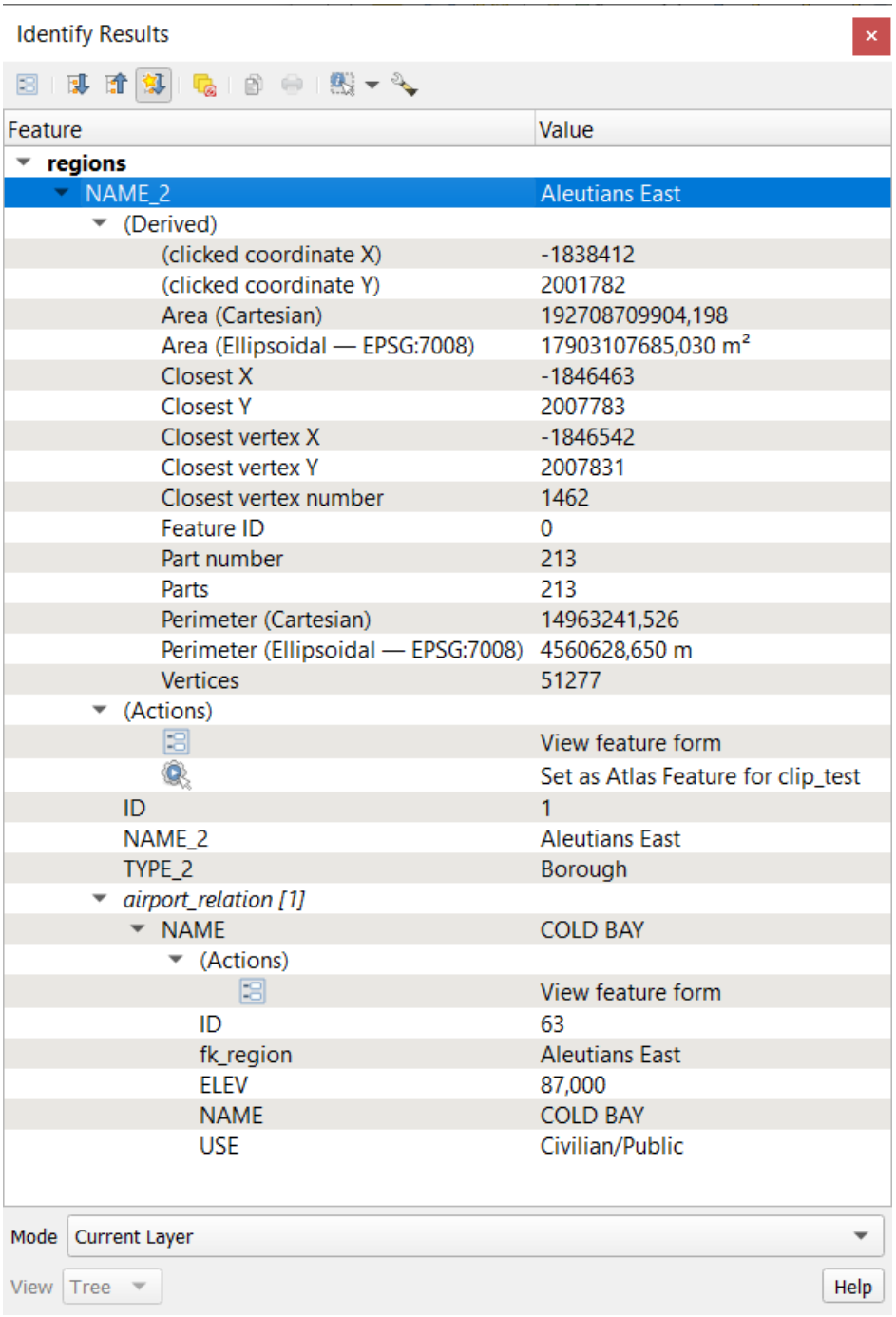
Režim višestrukog uređivanja dostupan je samo za automatski generisane obrasce i obrasce prevlačenja (pogledajte *Customizing a form for your data*); ne podržavaju ga prilagođeni ui obrasci.

12.2.5 Istraživanje atributa objekata pomoću alatke Identifikuj

Alatka  *Identifikuj objekte* može se koristiti za prikaz svih atributa objekta na platnu karte. To je brz način da se svi podaci pregledaju i provere bez potrebe za njihovim traženjem u atributskoj tabeli.

Da biste koristili alatku *Identifikuj objekte* za vektorske slojeve, pratite ove korake:

1. Izaberite vektorski sloj u panelu Slojevi.
2. Kliknite na alatku *Identifikuj objekte* u paleti alatki ili pritisnite `Ctrl+Shift+I`.
3. Kliknite na objekat u prikazu karte.



Слика12.78: Dijalog Rezultati identifikacije

Panel *Rezultati identifikacije* prikazaće različite informacije o objektima u prikazu stabla. U panelu postoje dve kolone: *Objekat* i *Vrednost*. Prva stavka je naziv sloja, a njena deca su njegov(i) identifikovan(i) objekat(i). Svaki objekat se identifikuje poljem *prikazanog naziva* zajedno sa njegovom vrednošću. Slede druge informacije o objektu:

- Odeljak **Izvedeno** - to su informacije izračunate ili izvedene iz drugih informacija u sloju. Na primer, površina poligona ili dužina linije. Opšte informacije koje se mogu naći u ovom odeljku:
  - U zavisnosti od tipa geometrije, kartezijanska merenja dužine, obima ili površine u jedinicama CRS-a sloja. Za 3D linijske vektore, dostupna je kartezijanska dužina linije.
  - U zavisnosti od tipa geometrije i ako je elipsoid postavljen u dijalogu *Svojstva projekta* (*Opšte* ► *Merenja*), elipsoidne vrednosti dužine, obima ili površine pomoću zadatih jedinica.

  - Broj delova geometrije u objektu i broj dela na koji je kliknuto.
  - Broj temena u objektu.
  - Informacije o koordinatama, pomoću podešavanja *Prikaz koordinata* u svojstvima projekta:
    * `X` i `Y` vrednosti koordinata tačke na koju je kliknuto
    * broj najbližeg temena tački na koju je kliknuto
    * `X` i `Y` vrednosti koordinata najbližeg temena (i `Z`/`M` ako je primenljivo)
    * ako kliknete na zakrivljeni segment, prikazuje se i poluprečnik tog dela.
    * ako i vektorski sloj i projekat imaju postavljene vertikalne datume i oni se razlikuju, `Z` vrednost će biti prikazana za oba datuma.

- **Radnje**: Radnje se mogu dodati u prozore identifikacije objekta. Radnja se pokreće klikom na natpis radnje. Podrazumevano se dodaje samo jedna radnja, tj. `Prikaži obrazac objekta` za uređivanje. Više radnji možete definisati u dijalogu svojstava sloja (pogledajte *Actions Properties*).
- **Atributi podataka**: Ovo je lista atributskih polja i vrednosti za objekat na koji je kliknuto.

  Белешка

  Linkovi u atributima objekta mogu se kliknuti iz panela *Rezultati identifikacije* i otvoriće se u vašem podrazumevanom veb pregledaču.

- Kada vektorski sloj ima definisane *relacije*, panel *Rezultati identifikacije* može prikazati i **referencirane** i **referencirajuće** povezane objekte. Da biste videli ove relacije, uverite se da je opcija *Prikaži relacije* omogućena u *Podešavanjima identifikacije*. Dostupne informacije za svaki povezani objekat uključuju:
  - naziv relacije i povezani sloj
  - unos u referenciranom ili referencirajućem polju, koji identifikuje objekat
  - aktivirane radnje
  - listu atributskih polja i vrednosti

  Povezane objekte možete proširiti da biste istražili povezane zapise kroz više nivoa, uključujući relacije više-prema-više (n:m) i polimorfne relacije. Učitavaju se samo čvorovi koje eksplicitno proširite, sprečavajući preterano ugnežđivanje.

  Da biste se fokusirali na određeni povezani zapis, kliknite desnim tasterom na njega i izaberite *Identifikuj objekat*. Time se rezultati identifikacije ponovo centriraju na izabrani objekat, počinjući novo stablo identifikacije od njega i efektivno ograničavajući vidljivu dubinu ugnežđivanja. Objekti već prikazani u nadređenim čvorovima automatski se izostavljaju radi izbegavanja duplikata ili kružnih relacija.

12.3 Uređivanje

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.

Белешка

The procedure for editing GRASS layers is different - see section *Digitizing and editing a GRASS vector layer* for details.

Пажња

Concurrent Edits

QGIS does not track if somebody else is editing the same feature at the same time as you are. The last person to save the edits wins.

Савет

Validating Edits

Continuous validation can be activated on a layer basis in the *Layer Properties* ► *Digitizing* tab. More at *Digitizing Properties*.

12.3.1 Setting the snapping tolerance and search radius

Under the *Settings* ► *Options...* ► *Digitizing* menu, QGIS provides a number of parameters to configure default behaviour of editing tools. More information at *Digitizing settings*.

For optimal and accurate editing of vector layer geometries, we need to set an appropriate value of snapping tolerance and search radius for features vertices. The *Snapping* group provides related options, namely handling of the snapping tolerance and the search radius.

- Snapping tolerance: When you add a new vertex or move an existing one, the snapping tolerance is the distance QGIS uses to search for the closest vertex or segment you are trying to connect to. If you are not within the snapping tolerance, QGIS will leave the vertex where you release the mouse button, instead of snapping it to an existing vertex or segment.

  The tolerance setting affects all tools that work with snapping and applies by default to new layers and projects. It can however be overridden at layer level (see *Snapping and Digitizing Options*).

- Search radius: *Search radius for vertex edits* is the distance QGIS uses to `search` for the vertex to select when you click on the map. If you are not within the search radius, QGIS will not find and select any vertex for editing.

Snap tolerance and search radius are set in `map units` or `pixels`. You may need to experiment to get them right. If you specify a too big tolerance, QGIS may snap to the wrong vertex, especially if you are dealing with a large number of vertices in close proximity. The smaller the search radius, the more difficult it will be to hit what you want to move.

12.3.2 Snapping and Digitizing Options

Global *snapping and digitizing settings* (snapping mode, tolerance value, and units…) can be overridden in the project from the *Project* ► *Snapping Options...* menu. In the *Snapping and Digitizing Options*, you can also configure some other properties (snapping layers, scale limit, topology…) The *Snapping Toolbar* gives access to most of these features.

By default, snapping is disabled in a project until you press the  Enable snapping button or press `S`. The snapping mode, tolerance value, and units can also be configured in this toolbar.

Snapping properties

There are three options to select the layer(s) to snap to:

- *All layers*: quick setting for all visible layers in the project so that the pointer snaps to all vertices and/or segments. In most cases, it is sufficient to use this snapping mode, but beware when using it for projects with many vector layers, as it may affect performance.
- *Current layer*: only the active layer is used, a convenient way to ensure topological consistency within the layer being edited.

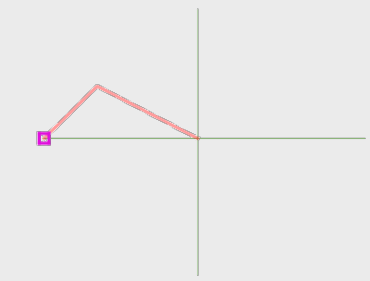
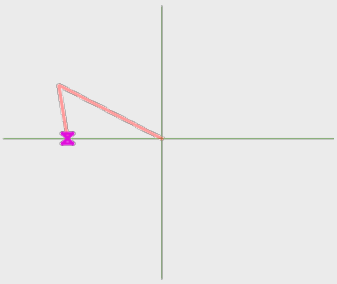
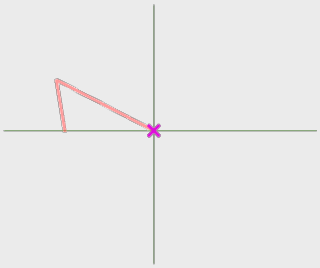
- *Advanced Configuration*: allows you to enable and adjust snapping mode, tolerance and units, overlaps and scales of snapping on a layer basis (see Сл.12.79). If you need to edit a layer and snap its vertices to another, make sure that the target layer is checked and increase the snapping tolerance to a higher value. Snapping will not occur to a layer that is not checked in the snapping options dialog.

When moving or creating vertex, you can opt for the following snapping modes:

- *Vertex*
- *Segment*: snaps along a line or a polygon perimeter. If topological editing is enabled, then a new vertex is added at the snapping location.
- *Area*: guarantees that the snap point lies anywhere on a polygon's area, not necessarily on its boundary
- *Centroid*: snaps to the centroid of the geometry of a feature. In case of a multipart geometry, the target point may be distinct from the existing parts.
- *Middle of Segments* on line or polygon feature
- *Line Endpoints*: snaps to the first or last vertex of every part of a line or polygon feature.

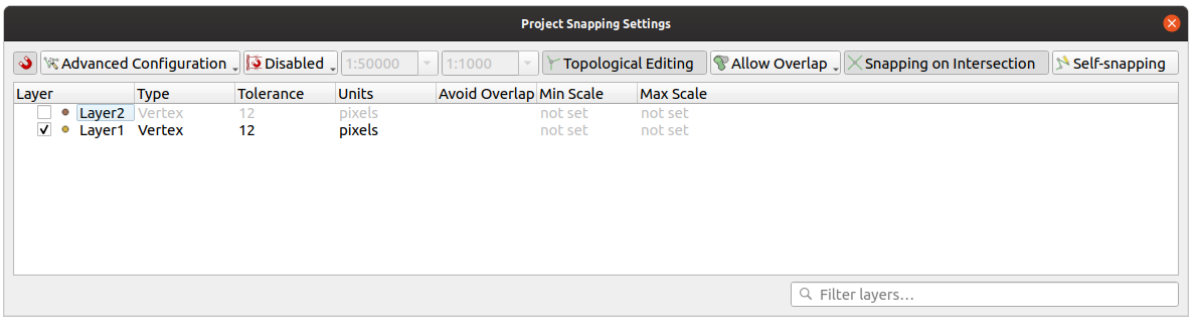
QGIS will show different *snap* icons depending on the kind of *snap*:

Табела12.2: Snapping icons

		
Snapping to a vertex: box icon	Snapping to a segment: hourglass icon	Snapping to an intersection: cross icon

Note that it is possible to change the color of these icons in the *Digitizing* part of the global settings.

The tolerance values can be set either in the project's `map units` or in `pixels`. The advantage of choosing `pixels` is that it keeps the snapping constant at different map scales. 10 to 12 pixels is normally a good value, but it depends on the DPI of your screen. Using map units allows the tolerance to be related to real ground distances. For example, if you have a minimum distance between elements, this option can be useful to ensure that you don't add vertices too close to each other.



Слика12.79: Snapping options (Advanced Configuration mode)

Белешка

By default, only visible features (the features whose style is displayed, except for layers where the symbology is „No symbols") can be snapped. You can enable the snapping on invisible features by checking *Enable snapping on invisible features* under the *Settings* ► *Options* ► *Digitizing* tab.

Савет

Enable snapping by default

You can set snapping to be enabled by default on all new projects in the *Settings* ► *Options* ► *Digitizing* tab. You can also set the default snapping mode, tolerance value, and units, which will populate the *Snapping Options* dialog.

Enable snapping on intersections

Another available option is to use *snapping on intersection*, which allows you to snap to geometry intersections of snapping enabled layers, even if there are no vertices at the intersections.

Limit snapping to a scale range

In some cases snapping can become very slow. This is often caused by the amount of features in some layers that require a heavy index to compute and maintain. Some parameters exist to enable snapping only when the map view is inside a relevant scale range. This allows to only do the costly index computation related to snapping at a scale where drawing is relevant.

Scale limit to snapping is configured in *Project* ► *Snapping Options…*. Limiting snapping to scale is only available in *Advanced Configuration* mode.

To limit snapping to a scale range you have three modes available:

- *Disabled*: Snapping is enabled whatever the current map scale is. This is the default mode.
- *Global*: Snapping is limited and only enabled when the current scale of the map is between a global minimum and a global maximum value. When selecting this mode two widgets become available to configure the range of scales in which snapping is enabled.
- *Per layer*: The snapping scale range limit is defined for each layer. When selecting this mode two columns become available to configure the minimum and maximum scales for each layer.

Please note that the minimum and maximum scales follow the QGIS convention: minimum scale is the most „zoomed out" scale while maximum scale is the most „zoomed in". A minimum or maximum scale that is set to „0" or „not set" is considered not limiting.

Self-snapping

The Self-snapping option allows you to snap to the geometry that is being edited. Combined with the *advanced digitizing panel*, this provides a handy way to digitize new edges relative to the previous edges or vertices. Self--snapping can cause invalid geometries, use with caution.

Snapping on custom grid

A snapping distance can also be customized on a layer basis in the *Digitizing* tab of the layer properties dialog. With setting the *Geometry precision* distance, you enable a dotted grid visible when the map canvas is at a coherent scale for display. Snapping can then be performed on the dots of the grid: an added or modified geometry will have all of its vertices snapped automatically to the closest node of the grid. More information at *Digitizing Properties*.

12.3.3 Topological editing

In addition to these snapping options, the *Snapping options…* dialog (*Project* ► *Snapping options*) and the *Snapping* toolbar allow you to enable / disable some other topological functionalities.

Enable topological editing

The  Topological editing button helps when editing and maintaining features with common boundaries. With this option enabled, QGIS 'detects' shared boundaries. When you move common vertices/segments, QGIS will also move them in the geometries of the neighboring features.

Topological editing works with features from different layers, as long as the layers are visible and in editing mode.

In layer with Z or M values, topological editing will interpolate the Z or M value of the vertex based on the value of the edge used for the connection.

Overlapping control

Overlapping prevents you from drawing new features that overlap existing ones in the selected layer, speeding up digitizing of adjacent polygons. It can be controlled by the overlap tool. Three modes are available:

1.  *Allow Overlap* (default)
2.  *Avoid Overlap on Active Layer*: prevents any overlap with other features from the layer being edited. Digitize the new geometries so that they overlap their neighbours and QGIS will cut the overlapping part(s) of the new geometries and snap them to the boundary of the existing features. The advantage is that you don't have to digitize the common vertices on boundary.
3.  *Follow Advanced Configuration*: allows the overlapping setting to be set on a layer basis in the *Advanced configuration* view mode.

Белешка

If the new geometry is totally covered by existing ones, it gets cleared, and QGIS will show an error message.

Упозорење

Use cautiously the *Avoid overlap* **option**

Since this option will cut new overlapping geometries of any polygon layer, you can get unexpected geometries if you forget to uncheck it when no longer needed.

Automatic Tracing

Usually, when using capturing map tools (add feature, add part, add ring, reshape and split), you need to click each vertex of the feature. With the automatic tracing mode, you can speed up the digitization process as you no longer need to manually place all the vertices during digitization:

1. Enable the  Tracing tool (in the *Snapping* toolbar) by pushing the icon or pressing `T` key.
2. *Snap to* a vertex or segment of a feature you want to trace along.
3. Move the mouse over another vertex or segment you'd like to snap and, instead of the usual straight line, the digitizing rubber band represents a path from the last point you snapped to the current position. The tool also works with curved geometries.

   QGIS actually uses the underlying features topology to build the shortest path between the two points. Tracing requires snapping to be activated in traceable layers to build the path. You should also snap to an existing vertex

or segment while digitizing and ensure that the two nodes are topologically connectable through existing features edges, otherwise QGIS is unable to connect them and thus traces a single straight line.

4. Click and QGIS places the intermediate vertices following the displayed path.

Unfold the  Enable Tracing icon and set the *Offset* option to digitize a path parallel to the features instead of tracing along them. A positive value shifts the new drawing to the left side of the tracing direction and a negative value does the opposite.

Белешка

Adjust map scale or snapping settings for an optimal tracing

If there are too many features in map display, tracing is disabled to avoid potentially long tracing structure preparation and large memory overhead. After zooming in or disabling some layers the tracing is enabled again.

Белешка

Does not add topological points

This tool does not add points to existing polygon geometries even if *Topological editing* is enabled. If geometry precision is activated on the edited layer, the resulting geometry might not exactly follow an existing geometry.

Савет

Quickly enable or disable automatic tracing by pressing the `T` key

By pressing the `T` key, tracing can be enabled/disabled anytime (even while digitizing a feature), so it is possible to digitize parts of the feature with tracing enabled and other parts with tracing disabled. Tools behave as usual when tracing is disabled.

Савет

Convert tracing to curved geometries

By using *Settings* ► *Options* ► *Digitizing* ► *Tracing* you can create curved geometries while digitizing. See *digitizing options*.

12.3.4 Digitizing an existing layer

By default, QGIS loads layers read-only. This is a safeguard to avoid accidentally editing a layer if there is a slip of the mouse. However, you can choose to edit any layer as long as the data provider supports it (see *Exploring Data Formats and Fields*), and the underlying data source is writable (i.e., its files are not read-only).

Савет

Restrict edit permission on layers within a project

From the *Project* ► *Properties...* ► *Data Sources* ► *Layers Capabilities* table, you can choose to set any layer read-only regardless the provider permission. This can be a handy way, in a multi-users environment to avoid unauthorized users to mistakenly edit layers (e.g., Shapefile), hence potentially corrupt data. Note that this setting only applies inside the current project.

In general, tools for editing vector layers are divided into a digitizing and an advanced digitizing toolbar, described

in section *Advanced digitizing*. You can select and unselect both under *View* ► *Toolbars* ►.

Using the basic digitizing tools, you can perform the following functions:

Табела12.3: Vector layer basic editing toolbar

Alatka	Svrha	Alatka	Svrha
	Access to save, rollback or cancel changes in all or selected layers simultaneously		Turn on or off edit status of selected layer(s) based on the active layer status
	Save edits to the active layer		
	Digitize using straight segments		Digitize using curve lines
	Enable freehand digitizing		Digitize polygon of regular shape
	Add new record		Add Feature: Capture Point
	Add Feature: Capture Line		Add Feature: Capture Polygon
	Vertex Tool (All Layers)		Vertex Tool (Current Layer)
 *Show Vertex Editor*	Set whether the vertex editor panel should auto-open		Modify the attributes of all selected features simultaneously
	Delete Selected features from the active layer		Cut Features from the active layer
	Copy selected Features from the active layer		Paste Features into the active layer
	Undo changes in the active layer		Redo changes in active layer

Note that while using any of the digitizing tools, you can still *zoom or pan* in the map canvas without losing the focus on the tool.

All editing sessions start by choosing the  Toggle editing option found in the context menu of a given layer, from the attribute table dialog, the digitizing toolbar or the *Layer* menu.

Once the layer is in edit mode, additional tool buttons on the editing toolbar will become available and markers will appear at the vertices of all features unless *Show markers only for selected features* option under *Settings* ► *Options…* ► *Digitizing* menu is checked.

Савет

Save Regularly

Remember to  Save Layer Edits regularly. This will also check that your data source can accept all the changes.

Geometry editing techniques

When a geometry drawing tool (mainly the ones that add, split, reshape features) is enabled for a line or polygon based layer, you can select the technique for adding new vertices:

- The  Digitize with Segment: draws straight segment whose start and end points are defined by left clicks.
- The  Digitize with Curve: draws curve line based on three consecutive nodes defined by left clicks (start, point along the arc, end). If the geometry type does not support curves, then consecutive smaller segments are used to approximate the curvature.

- The  Stream Digitizing: draws lines in freehand mode, i.e. nodes are added following cursor movement in the map canvas and a *Streaming Tolerance*. The streaming tolerance defines the spacing between consecutive vertices. Currently, the only supported unit is pixels (`px`). Only the starting left click and the ending right click are necessary in this mode.
- The  Digitize Shape: triggers tools on the *Shape Digitizing Toolbar* to draw a polygon of a regular shape.

The selected technique remains while switching among the digitizing tools. You can combine any of the first three methods while drawing the same geometry.

Adding Features

Depending on the layer type, you can use the  Add Record,  Add Point Feature,  Add Line Feature or  Add Polygon Feature icons on the toolbar to add new features into the current layer.

To add a geometryless feature, click on the  Add Record button and you can enter attributes in the feature form that opens.

To create features with the spatially enabled tools, you first digitize the geometry then enter its attributes. To digitize the geometry:

1. (Optional as it is the default) Select the  Digitize With Segment geometry drawing method
2. Left-click on the map area to create the first point of your new feature. For point features, this should be enough and trigger, if required, the feature form to fill in their attributes.
3. For line or polygon geometries, keep on left-clicking for each additional point you wish to capture. You can rely on the *snapping to features* options, the *snap-to-grid* or the *advanced digitizing* panel to accurately position each vertex.

   Along with drawing straight segments between nodes you click one by one, lines and polygons can be:

   - *traced automatically*, accelerating the digitization. This will create consecutive straight lines between the vertices you place, following existing features.
   - free-hand digitized, pressing `R` or activating  Stream Digitizing.
   - drawn as curve, pressing `Ctrl+Shift+G` or activating  Digitize with Curve.

   Белешка

   While digitizing line or polygon geometries, you can switch back and forth between the geometry drawing methods, allowing you to create features mixing straight segments, free-hand ones and curved parts.

4. Press `Delete` or `Backspace` key to revert the last node(s) you may wrongly add.
5. When you have finished adding points, right-click anywhere on the map area to confirm you have finished entering the geometry of that feature.

   Савет

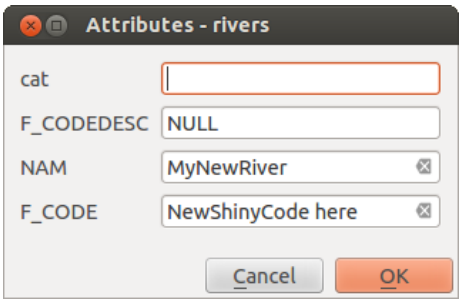
   Customize the digitizing rubber band

   While capturing polygon, the by-default red rubber band can hide underlying features or places you'd like to capture a point. This can be fixed by setting a lower opacity (or alpha channel) to the rubber band's *Fill Color* in *Settings* ► *Options* ► *Digitizing* menu. You can also avoid the use of the rubber band by checking *Don't update rubber band during node editing*.

6. For line feature pressing `Shift` + right-click will close the line automatically.

7. The attribute window will appear, allowing you to enter the information for the new feature. Сл.12.80 shows setting attributes for a fictitious new river. However, in the *Digitizing* menu under the *Settings* ► *Options* menu, you can also:

   - *Suppress attributes pop-up windows after each created feature* to avoid the form opening;
   - or *Reuse last entered attribute values* to have fields automatically filled at the opening of the form and just have to type changing values.



Слика12.80: Enter Attribute Values Dialog after digitizing a new vector feature

Vertex tool

QGIS provides two tools to interact with vector features vertices:

- Vertex Tool (Current Layer): only overlaid features in the active layer (in the *Layers* panel) are affected
- Vertex Tool (All Layers): any overlaid features in all editable layers are affected. This allows you to edit features without switching the active layer or edit multiple layers at once (e.g., country and their regions boundaries)

For any editable vector layer, the vertex tools provide manipulation capabilities of feature vertices similar to CAD programs. It is possible to select multiple vertices at once and to move, add or delete them altogether. The vertex tools also support the topological editing feature. They are selection persistent, so when some operation is done, selection stays active for this feature and tool.

Савет

Drawing a series of new vertices

The vertex tool does not support *automatic tracing* and is optimized for editing individual vertices and moving or deleting an arbitrary selection of multiple vertices. Try using the Reshape Features tool when you need to replace or insert a series of new vertices, especially if you want to trace other features.

It is important to set the property *Settings* ► *Options* ► *Digitizing* ► *Search Radius:* 1,00 to a number greater than zero. Otherwise, QGIS will not be able to tell which vertex is being edited and will display a warning.

Савет

Vertex Markers

QGIS supports different kinds of vertex markers: 'Semi-transparent circle', 'Cross' and 'None'. To change the marker style, choose *Options* from the *Settings* menu, click on the *Digitizing* tab and select the appropriate entry.

Basic operations

Given a layer in edit mode, start by activating the vertex tool. Red circles will appear when hovering vertices.

- **Selecting vertices**: You can select vertices by:
  - Clicking on them one at a time holding `Shift` key pressed
  - Click-and-dragging a rectangle surrounding the target vertices
  - Drawing a polygon surrounding the target vertices: Hold `Alt` and click using the vertex tool to start digitizing a polygon. Each subsequent click adds a new vertex to the rubberband polygon. `Backspace` or `Delete` removes last added rubberband vertex. `Esc` cancels the polygon selection mode, as also does backspacing/deleting all of the rubberband's vertices. Right click finalizes the polygon digitizing and selects all vertices within the rubberband polygon.

  When a vertex is selected, its color changes to blue. To add more vertices to the current selection, hold down the `Shift` key while proceeding as above. To remove vertices from the selection, hold down `Ctrl`.

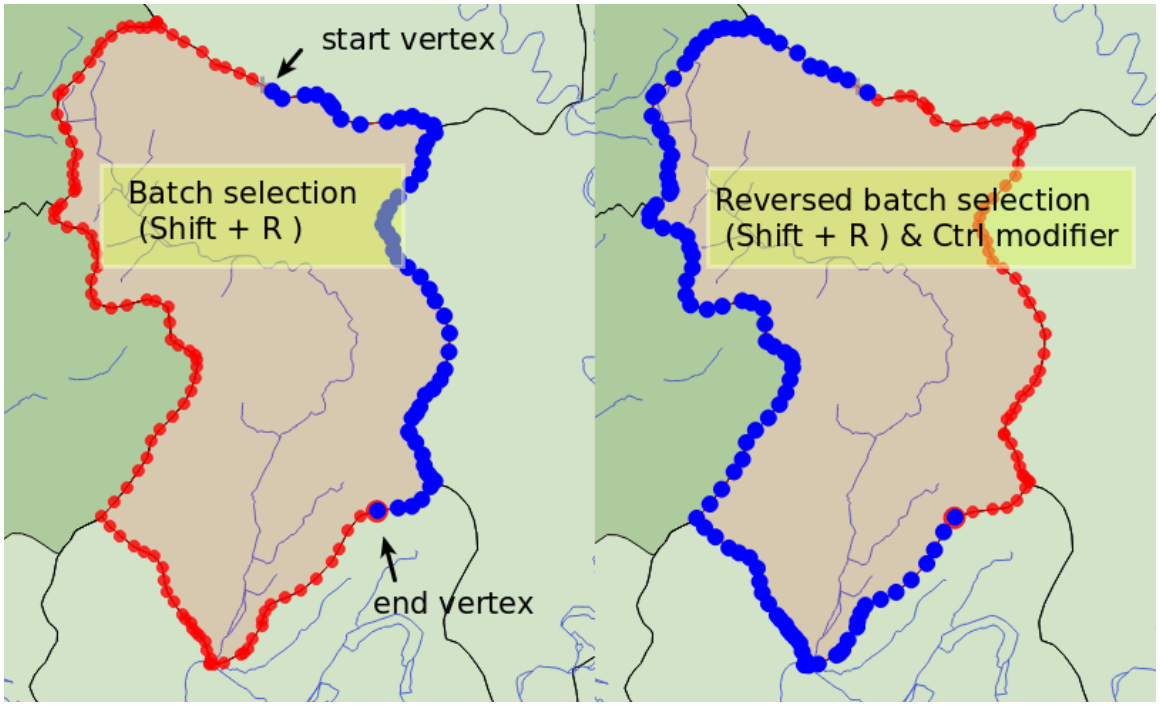
  Савет

  Feature selection bounds vertex tool

  Vertices can be selected across different features (or layers). If you are looking for vertices of a specific feature in a crowded place, first select that feature. Then draw the rectangle or polygon selector with the vertex tool around the vertices: only the selected feature's vertices are selected.

  This is also the case if you display the feature in the *vertex editor* panel.

- **Batch vertex selection mode**: The batch selection mode can be activated by pressing `Shift+R`. Select a first node with one single click, and then hover **without clicking** another vertex. This will dynamically select all the nodes in between using the shortest path (for polygons).

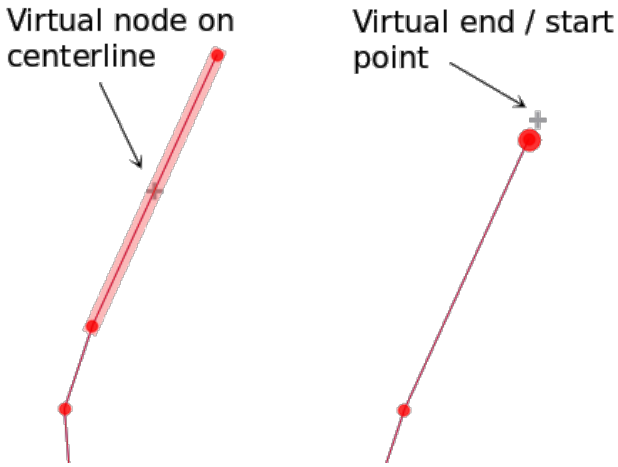


Слика12.81: Batch vertex selection using `Shift+R`

Press `Ctrl` will invert the selection, selecting the longest path along the feature boundary. Ending your node selection with a second click, or pressing `Esc` will escape the batch mode.

- **Adding vertices**: To add a vertex to a line or polygon geometry, hold `Shift` and double-click the place on the segment.

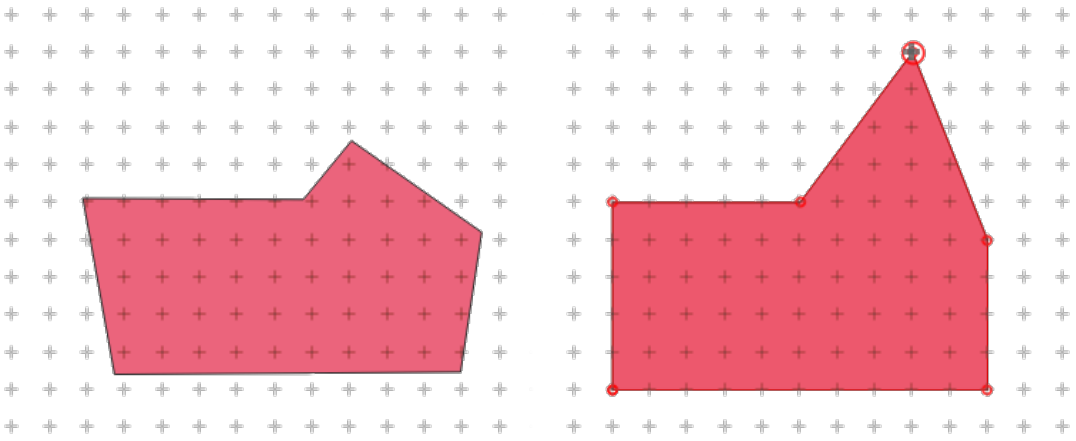
  When hovering a segment, a virtual new node appears on the center. Click on it, move the cursor to a new location and click again to add a new vertex. For lines, a virtual node is also proposed at both extremities: click on it, do subsequent clicks and finish with a right-click; this allows to easily extend an existing line.



Слика12.82: Virtual nodes for adding vertices

- **Deleting vertices**: Select the vertices and click the `Delete` key. Deleting all the vertices of a feature generates, if compatible with the datasource, a geometryless feature. Note that this doesn't delete the complete feature, just the geometry part. To delete a complete feature use the Delete Selected tool.
- **Moving vertices**: Select all the vertices you want to move, click on a selected vertex or edge, and click on the desired new location. You can use the *snapping to feature capabilities* and the *Advanced Digitizing Panel* constraints for distance, angles, exact X and Y location before the second click. All the selected vertices will be translated.

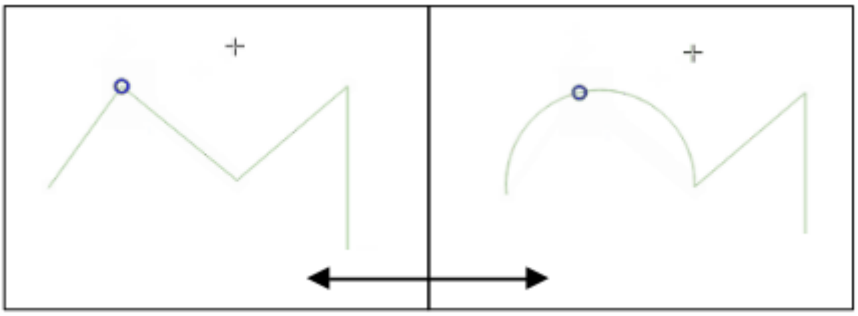
  However, if the *snap-to-grid* option is enabled, selected vertices are snapped to the closest grid intersection to their translated position. Unselected vertices are also moved to their closest grid intersection. There is no simple translation.



Слика12.83: Moving the top vertex snaps all the vertices to the grid

- **Converting adjacent segments to/from curve**: Select the center vertex of the segment you want to convert, hit the `O` letter key. If the vertex was in a curve, the curve is converted into straight lines. If the vertex was

between two straight lines, they are converted into a curve. A first or a last vertex of a line can't be converted to a center vertex curve. The layer must be compatible with curve geometry type.



Слика12.84: Switch from curve to straight lines with ○ letter

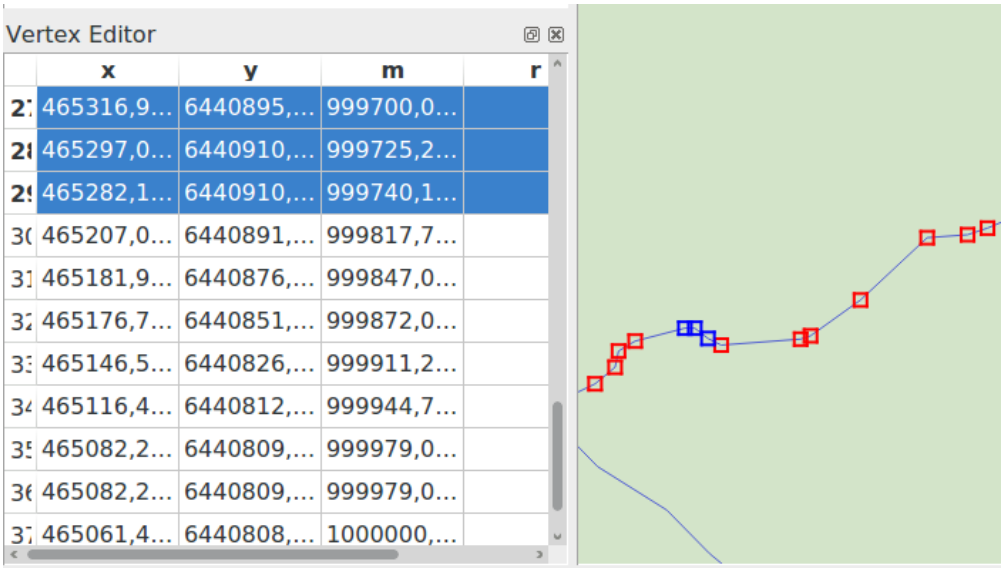
Each change made with the vertex tool is stored as a separate entry in the *Undo* dialog. Remember that all operations support topological editing when this is turned on. On-the-fly projection is also supported.

The Vertex Editor Panel

With enabling a vertex tool, you also open the *Vertex Editor* panel. Right-clicking over a feature fills the panel with the list of all the vertices of the feature with their *x*, *y* (*z*, *m* if applicable) coordinates and *r* (for the radius, in case of circular geometry). The feature is also made exclusive for editing, meaning that the edit of any other features is disabled:

- Selecting a row in the table does select the corresponding vertex in the map canvas, and vice versa.
- Clicking or dragging over the map canvas will only select or move vertices and segments of that feature
- Change a coordinate in the table and the vertex position is updated. This is a convenient way to edit Z coordinate or M value on vertices.
- You can also select multiple rows and delete them altogether.
- New vertices can only be added to the bound feature

If you do not want the *Vertex Editor* panel to immediately show each time you interact with vertex tools (and potentially hide other panels or disturb panels placement), uncheck the *Auto-open table* entry in the ☰ Options menu at the top of the panel. You can then also close the panel. To reopen the panel, you would need to right-click over a panel or toolbar and select it in the list or tick the *Show vertex editor* entry in the *Digitizing toolbar*.



Слика12.85: Vertex editor panel showing selected nodes

Rules of Z coordinate or M value assignment

Digitizing 3D vector features or features with M value is not that different from (X,Y) 2D layers'. Tools and options described in this chapter are still available and help you place the vertex or point in a planar environment. Then you may need to handle the Z coordinate (or M value) assignment:

- By default, QGIS will assign to new vertices the *Default Z value* (respectively *Default M value*) set in the *Settings* ► *Options* ► *Digitizing* tab. If the *Advanced Digitizing Panel* is in use, then the value is taken from its *z* (respectively *m*) widget.
- When snapping to a vertex, the new or moved vertex takes the snapped one's Z or M value.
- When snapping to a segment while the topological editing is on, then the new vertex Z or M value is interpolated along the segment.
- If the *z* (respectively *m*) widget of the *Advanced Digitizing Panel* is locked, then its value is applied to the vertex, taking precedence over any snapped vertex or segment Z or M value.

To edit Z or M values of an existing feature, you can use the *Vertex editor panel*. To create features with custom Z or M values you may want to rely on the *Advanced Digitizing Panel*.

Cutting, Copying and Pasting Features

Selected features can be cut, copied and pasted between layers in the same QGIS project, as long as destination layers are set to Toggle editing beforehand.

Савет

Transform polygon into line and vice-versa using copy/paste

Copy a line feature and paste it in a polygon layer: QGIS pastes in the target layer a polygon whose boundary corresponds to the closed geometry of the line feature. This is a quick way to generate different geometries of the same data.

Features can also be pasted to external applications as text. That is, the features are represented in CSV format, with the geometry data appearing in the OGC Well-Known Text (WKT) format. WKT and GeoJSON features from outside QGIS can also be pasted to a layer within QGIS.

When would the copy and paste function come in handy? Well, it turns out that you can edit more than one layer at a time and copy/paste features between layers. Why would we want to do this? Say we need to do some work on a new layer but only need one or two lakes, not the 5,000 on our `big_lakes` layer. We can create a new layer and use copy/paste to plop the needed lakes into it.

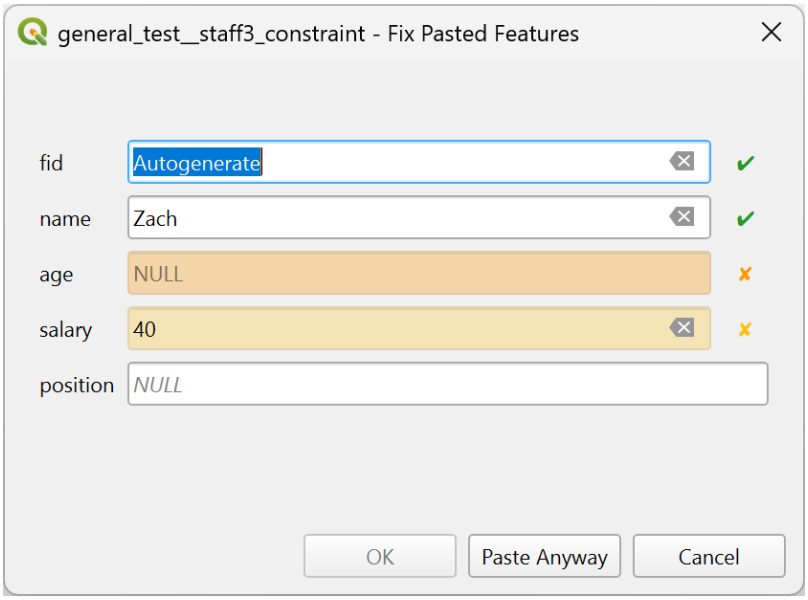
As an example, we will copy some lakes to a new layer:

1. Load the layer you want to copy from (source layer)
2. Load or create the layer you want to copy to (target layer)
3. Start editing for target layer
4. Make the source layer active by clicking on it in the legend
5. Use the Select Features by area or single click tool to select the feature(s) on the source layer
6. Click on the Copy Features tool
7. Make the destination layer active by clicking on it in the legend
8. Click on the Paste Features tool
9. Stop editing and save the changes

What happens if the source and target layers have different schemas (field names and types are not the same)? QGIS populates what matches and ignores the rest. If you don't care about the attributes being copied to the target layer, it doesn't matter how you design the fields and data types. If you want to make sure everything - the feature and its attributes - gets copied, make sure the schemas match.

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:

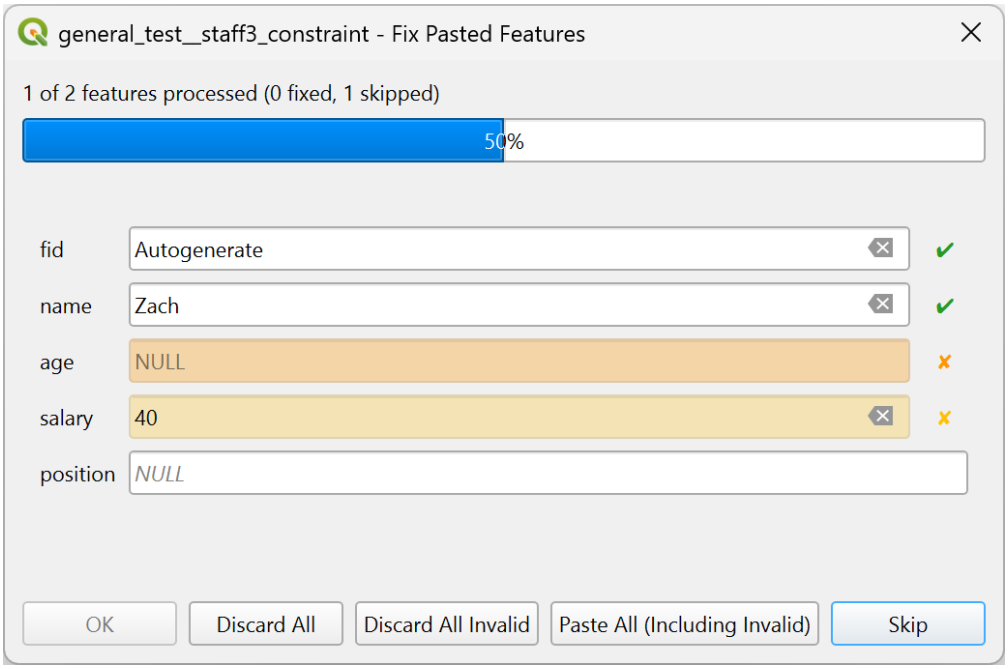
- Edit the invalid values directly in the dialog. The corrected values will be used when pasting.
- *Paste Anyway*: the feature is pasted as-is, even with invalid values.



Слика12.86: 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.



Слика12.87: Dialog for handling invalid attribute values when pasting multiple features

Белешка

Congruency of Pasted Features

If your source and destination layers use the same projection, then the pasted features will have geometry identical to the source layer. However, if the destination layer is a different projection, then QGIS cannot guarantee the geometry is identical. This is simply because there are small rounding-off errors involved when converting between projections.

Савет

Copy string attribute into another

If you have created a new column in your attribute table with type 'string' and want to paste values from another attribute column that has a greater length the length of the column size will be extended to the same amount. This is because the GDAL Shapefile driver knows to auto-extend string and integer fields to dynamically accommodate for the length of the data to be inserted.

Deleting Selected Features

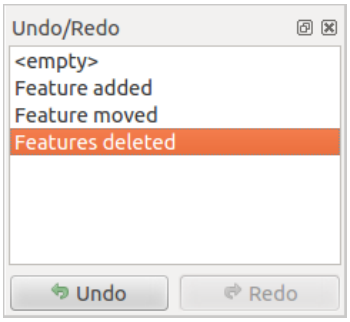
If we want to delete an entire feature (attribute and geometry), we can do that by first selecting the geometry using the regular Select Features by area or single click tool. Selection can also be done from the attribute table. Once you have

the selection set, press `Delete` or `Backspace` key or use the Delete Selected tool to delete the features. Multiple selected features can be deleted at once.

The Cut Features tool on the digitizing toolbar can also be used to delete features. This effectively deletes the feature but also places it on a „spatial clipboard". So, we cut the feature to delete. We could then use the Paste Features tool to put it back, giving us a one-level undo capability. Cut, copy, and paste work on the currently selected features, meaning we can operate on more than one at a time.

Undo and Redo

The Undo and Redo tools allows you to undo or redo vector editing operations. There is also a dockable widget, which shows all operations in the undo/redo history (see Сл.12.88). This widget is not displayed by default; it can be displayed by right-clicking on the toolbar and activating the *Undo/Redo Panel* checkbox. The Undo/Redo capability is however active, even if the widget is not displayed.



Слика12.88: Redo and Undo digitizing steps

When Undo is hit or `Ctrl+Z` (or `Cmd+Z`) pressed, the state of all features and attributes are reverted to the state before the reverted operation happened. Changes other than normal vector editing operations (for example, changes done by a plugin) may or may not be reverted, depending on how the changes were performed.

To use the undo/redo history widget, simply click to select an operation in the history list. All features will be reverted to the state they were in after the selected operation.

Saving Edited Layers

When a layer is in editing mode, any changes remain in the memory of QGIS. Therefore, they are not committed/saved immediately to the data source or disk. If you want to save edits to the current layer but want to continue editing without leaving the editing mode, you can click the Save Layer Edits button. When you turn editing mode off with Toggle editing (or quit QGIS for that matter), you are also asked if you want to save your changes or discard them.

If the changes cannot be saved (e.g., disk full, or the attributes have values that are out of range), the QGIS in--memory state is preserved. This allows you to adjust your edits and try again.

Савет

Data Integrity

It is always a good idea to back up your data source before you start editing. While the authors of QGIS have made every effort to preserve the integrity of your data, we offer no warranty in this regard.

Saving multiple layers at once

This feature allows the digitization of multiple layers. Choose  *Save for Selected Layers* to save all changes you made in multiple layers. You also have the opportunity to  *Rollback for Selected Layers*, so that the digitization may be withdrawn for all selected layers. If you want to stop editing the selected layers,  *Cancel for Selected Layer(s)* is an easy way.

The same functions are available for editing all layers of the project.

Савет

Use transaction group to edit, save or rollback multiple layers changes at once

When working with layers from the same PostGreSQL database, activate the *Automatically create transaction groups where possible* option in *Project* ► *Properties…* ► *Data Sources* to sync their behavior (enter or exit the edit mode, save or rollback changes at the same time).

12.3.5 Advanced digitizing

Табела12.4: Vector layer advanced editing toolbar

Ikona	Svrha	Ikona	Svrha
	Enable Advanced Digitizing Tools		
	Move Feature(s)		Copy and Move Feature(s)
	Rotate Feature(s)		Simplify Feature
	Scale Feature		
	Add Ring		Add Part
	Fill Ring		Swap direction
	Delete Ring		Delete Part
	Offset Curve		Reshape Features
	Split Parts		Split Features
	Merge Attributes of Selected Features		Merge Selected Features
	Rotate Point Symbols		Offset Point Symbols
	Trim or Extend Feature		

Move Feature(s)

The  Move Feature(s) tool allows you to move existing features:

1. Select the feature(s) to move.
2. Click on the map canvas to indicate the origin point of the displacement; you can rely on snapping capabilities to select an accurate point.

   You can also take advantages of the *advanced digitizing constraints* to accurately set the origin point coordinates. In that case:

   1. First click on the  button to enable the panel.
   2. Type `x` and enter the corresponding value for the origin point you'd like to use. Then press the  button next to the option to lock the value.
   3. Do the same for the `y` coordinate.
   4. Click on the map canvas and your origin point is placed at the indicated coordinates.
3. Move over the map canvas to indicate the destination point of the displacement, still using snapping mode or, as above, use the advanced digitizing panel which would provide complementary `distance` and `angle` placement constraints to place the end point of the translation.
4. Click on the map canvas: the whole features are moved to new location.

Likewise, you can create a translated copy of the feature(s) using the  Copy and Move Feature(s) tool.

Белешка

If no feature is selected when you first click on the map canvas with any of the *Move Feature(s)* or *Copy and Move Feature(s)* tools, then only the feature under the mouse is affected by the action. So, if you want to move several features, they should be selected first.

Rotate Feature(s)

Use the  Rotate Feature(s) tool to rotate one or multiple features in the map canvas:

1. Press the  Rotate Feature(s) icon
2. Then click on the feature to rotate. The feature's centroid is referenced as rotation center, a preview of the rotated feature is displayed and a widget opens showing the current *Rotation* angle.
3. Click on the map canvas when you are satisfied with the new placement or manually enter the rotation angle in the text box. You can also use the *Snap to* ° box to constrain the rotation values.
4. If you want to rotate several features at once, they shall be selected first, and the rotation is by default around the centroid of their combined geometries.

You can also use an anchor point different from the default feature centroid: press the `Ctrl` button, click on the map canvas and that point will be used as the new rotation center.

If you hold `Shift` before clicking on the map, the rotation will be done in 45 degree steps, which can be modified afterwards in the user input widget.

To abort feature rotation, press the `ESC` button or click on the  Rotate Feature(s) icon.

Scale Feature

The  Scale Feature tool is similar to the Rotate feature. Though instead of performing a rotation of selected features, it rescales their geometry. The change is performed in relation to the anchor point and the scale ratio can be manually specified in the widget that appears in the upper corner of the canvas.

Simplify Feature

The  Simplify Feature tool allows you to interactively reshape a line or polygon geometry by reducing or densifying the number of vertices, as long as the geometry remains valid:

1. Select the  Simplify Feature tool.
2. Click on the feature or drag a rectangle over the features.
3. A dialog pops up allowing you to define the *Method* to apply, ie whether you would like to:
   - *simplify the geometry*, meaning less vertices than the original. Available methods are `Simplify by distance`, `Simplify by snapping to grid` or `simplify by area (Visvalingam)`. You'd then need to indicate the value of *Tolerance* in `Layer units`, `Pixels` or `map units` to use for simplification. The higher the tolerance is the more vertices can be deleted.
   - or *densify the geometries* with new vertices thanks to the `Smooth` option: for each existing vertex, two vertices are placed on each of the segments originated from it, at an *Offset* distance representing the percentage of the segment length. You can also set the number of *Iterations* the placement would be processed: the more iterations, the more vertices and smoother is the feature.

   Settings that you used will be saved when leaving a project or an edit session. So you can go back to the same parameters the next time you simplify a feature.
4. A summary of the modifications that would apply is shown at the bottom of the dialog, listing number of features and number of vertices (before and after the operation and the ratio the change represents). Also, in the map canvas, the expected geometry is displayed over the existing one, using the rubberband color.
5. When the expected geometry fits your needs, click *OK* to apply the modification. Otherwise, to abort the operation, you can either press *Cancel* or right-click in the map canvas.

Белешка

Unlike the feature simplification option in *Settings* ► *Options* ► *Rendering* menu which simplifies the geometry just for rendering, the  Simplify Feature tool permanently modifies feature's geometry in data source.

Add Part

You can  Add Part to a selected feature generating a multipoint, multiline or multipolygon feature. The new part must be digitized outside the existing one which should be selected beforehand.

The  Add Part can also be used to add a geometry to a geometryless feature. First, select the feature in the attribute table and digitize the new geometry with the  Add Part tool.

Белешка

Order of vertices in polygon parts

Unlike the OGC standards, QGIS doesn't constrain vertices of the exterior boundary of a polygon feature to be ordered counterclockwise. Thus, you can find both directions in a layer. However, every parts of the same multipolygon feature will have their outer vertices ordered following the same direction.

You can however use the *Nametni pravilo desne ruke* algorithm to constrain features of a layer to have vertices of their outer boundaries ordered in the clockwise direction.

Delete Part

The  Delete Part tool allows you to delete parts from multifeatures (e.g., to delete polygons from a multi-polygon feature). This tool works with all multi-part geometries: point, line and polygon. Furthermore, it can be used to totally remove the geometric component of a feature. To delete a part, simply click within the target part.

Add Ring

You can create ring polygons using the  Add Ring icon in the toolbar. This means that inside an existing area, it is possible to digitize further polygons that will occur as a 'hole', so only the area between the boundaries of the outer and inner polygons remains as a ring polygon.

To add a ring:

1. Select the feature(s) to modify
2. Activate the  Add Ring tool
3. Draw a polygon within the selected geometries, using the aforementioned *techniques*. A hole appears in the selected geometries.
4. If no geometry is selected when the ring is drawn, then a hole is added to each of the polygons the ring is drawn over.

Белешка

Order of vertices in polygon rings

Unlike the OGC standards, QGIS doesn't constrain vertices of the exterior boundary of a polygon feature to be ordered counterclockwise. Thus, you can find both directions in a layer. However, every rings of the same (multi)polygon feature will have their vertices ordered in the opposite direction to the outer boundary's.

You can however use the *Nametni pravilo desne ruke* algorithm to constrain features of a layer to have vertices of their outer boundaries ordered in the clockwise direction, and vertices of their interior rings ordered in the counter-clockwise direction.

Fill Ring

The  Fill Ring tool helps you create polygon feature that totally falls within another one without any overlapping area; that is the new feature covers a hole within the existing one. To create such a feature:

1. Select the  Fill Ring tool.
2. Draw a new polygon over the existing feature: QGIS adds a ring to its geometry (like if you used the  Add Ring tool) and creates a new feature whose geometry matches the ring (like if you *traced* over the interior boundaries with the  Add polygon feature tool).
3. Or alternatively, if the ring already exists on the feature, place the mouse over the ring and left-click while pressing `Shift`: a new feature filling the hole is drawn at that place.

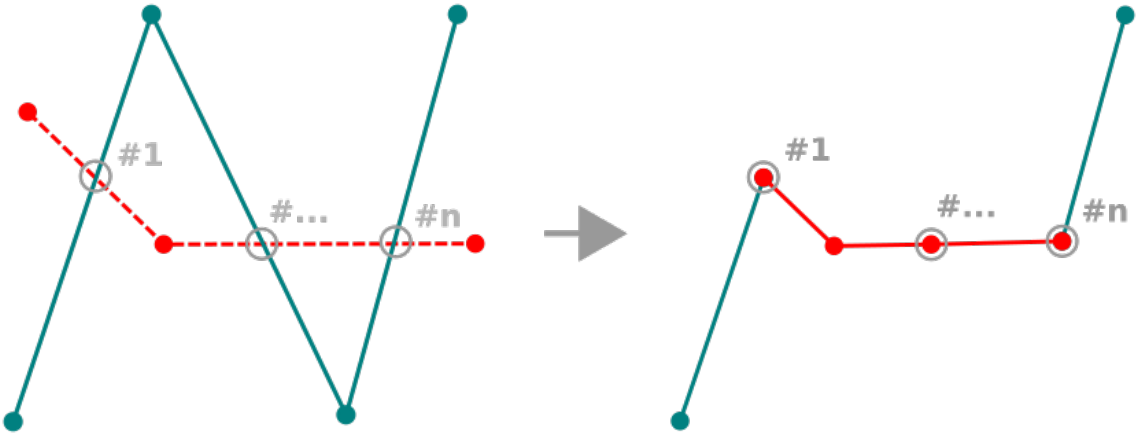
   The *Feature Attributes* form of the new feature opens, pre-filled with values of the „parent" feature and/or *fields constraints*.

Delete Ring

The  Delete Ring tool allows you to delete rings within an existing polygon, by clicking inside the hole. This tool only works with polygon and multi-polygon features. It doesn't change anything when it is used on the outer ring of the polygon.

Reshape Features

You can reshape line and polygon features using the  Reshape Features tool on the toolbar. For lines, it replaces the line part from the first to the last intersection with the original line.



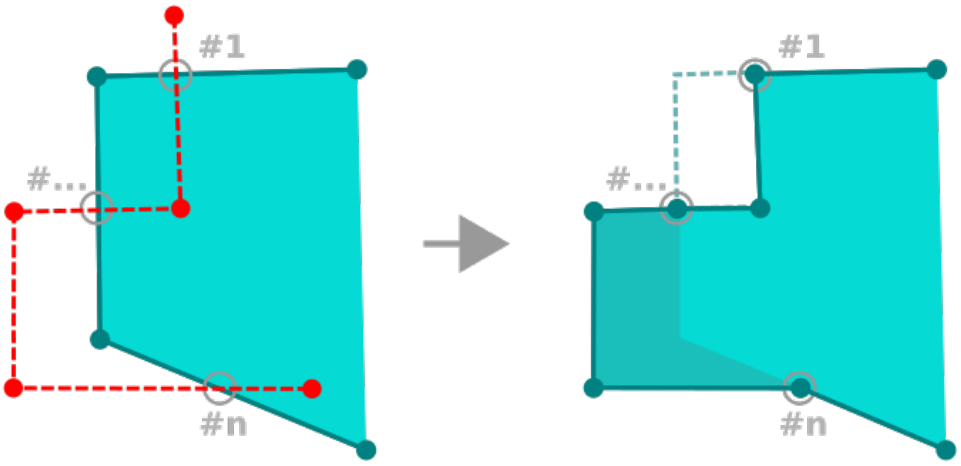
Слика12.89: Reshape line

Савет

Extend linestring geometries with reshape tool

Use the  Reshape Features tool to extend existing linestring geometries: snap to the first or last vertex of the line and draw a new one. Validate and the feature's geometry becomes the combination of the two lines.

For polygons, it will reshape the polygon's boundary. For it to work, the reshape tool's line must cross the polygon's boundary at least twice. To draw the line, click on the map canvas to add vertices. To finish it, just right-click. Like with the lines, only the segment between the first and the last intersections is considered. The reshape line's segments that are inside the polygon will result in cropping it, where the ones outside the polygon will extend it.



Слика12.90: Reshape polygon

With polygons, reshaping can sometimes lead to unintended results. It is mainly useful to replace smaller parts of a polygon, not for major overhauls, and the reshape line is not allowed to cross several polygon rings, as this would generate an invalid polygon.

Белешка

The reshape tool may alter the starting position of a polygon ring or a closed line. So, the point that is represented 'twice' will not be the same any more. This may not be a problem for most applications, but it is something to consider.

Offset Curves

The  Offset Curve tool creates parallel shifts of line layers. The tool can be applied to the edited layer (the geometries are modified) or also to background layers (in which case it creates copies of the lines / rings and adds them to the edited layer). It is thus ideally suited for the creation of distance line layers. The *User Input* dialog pops-up, showing the displacement distance and other settings.

To create a shift of a line or polygon layer, you must first go into editing mode and activate the  Offset Curve tool. Then click on a feature to shift it. Move the mouse and click where wanted or enter the desired distance in the user input widget. Holding `Ctrl` during the 2nd click will make an offset copy. Your changes may then be saved with the  Save Layer Edits tool.

For geometries on background layers make sure that snapping is on and hold `Ctrl` to select the geometry from the background. Also hold `Ctrl` when doing the second click. Geometries will be converted to the target layer geometry type.

QGIS options dialog (Digitizing tab then **Curve offset tools** section) or the  icon in the *User Input* dialog allows you to configure *some parameters* like **Join style**, **Quadrant segments**, **Miter limit** and **End cap style**.

Reverse Line

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.

To change a line direction:

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.

Савет

For bulk line direction changes, use the *Obrni smer linije* processing algorithm which can reverse all lines in a layer or a selection at once.

Split Features

Use the  Split Features tool to split a feature into two or more new and independent features, ie. each geometry corresponding to a new row in the attribute table.

To split line or polygon features:

1. Select the  Split Features tool.
2. Draw a line across the feature(s) you want to split. If a selection is active, only selected features are split. The original feature is then assigned the biggest geometry resulting from the splitting, and new features are created for the remaining parts. Fields of the features are filled/updated according to the datasource provider rules or their *splitting policy*.
3. You can then as usual modify any of the attributes of any resulting feature.

Савет

Split a polyline into new features in one-click

Using the  Split Features tool, snap and click on an existing vertex of a polyline feature to split that feature into two new features.

Split parts

In QGIS it is possible to split the parts of a multi part feature so that the number of parts is increased. Just draw a line across the part you want to split using the  Split Parts icon.

Савет

Split a polyline into new parts in one-click

Using the  Split Parts tool, snap and click on an existing vertex of a polyline feature to split the feature into two new polylines belonging to the same feature.

Merge selected features

The  Merge Selected Features tool allows you to create a new feature by merging existing ones: their geometries are merged to generate a new one. If features don't have common boundaries, a multipolygon/multipolyline/multipoint feature is created.

1. First, select the features you'd like to combine.
2. Then press the  Merge Selected Features button.
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:
   - manually replacing the value in the corresponding cell;
   - selecting a row in the table and pressing *Take attributes from selected feature* to use the values of this initial feature;
   - pressing the *Take attributes from the largest geometry* to use the attributes from the longest line feature, the largest polygon, or the multipoints with the most parts;
   - pressing *Skip all fields* to use empty attributes;
   - expanding the drop down menu at the top of the table, select any of the above options to apply to the corresponding field only. There, you can also choose to aggregate the initial features attributes (Minimum, Maximum, Median, Sum, Count, Concatenation… depending on the type of the field. see *Statistical Summary Panel* for the full list of functions).

   Белешка

   If the layer has default values or clauses present on fields, these are used as the initial value for the merged feature.

4. Press *OK* to apply the modifications. A single (multi)feature is created in the layer, replacing the previously selected ones.

Merge attributes of selected features

The  Merge Attributes of Selected Features tool allows you to apply same attributes to features without merging their boundaries. The dialog is the same as the `Merge Selected Features` tool's except that unlike that tool, selected objects are kept with their geometry while some of their attributes are made identical.

Rotate Point Symbols

The  Rotate Point Symbols allows you to individually change the rotation of point symbols in the map canvas.

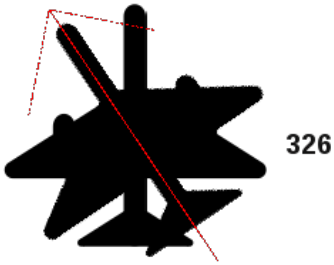
1. First, you need to indicate the field to store the rotation value in. This is made by assigning a field to the symbol *data-defined* rotation property:
   1. In the *Layer Properties* ► *Symbology* dialog, browse to the symbol editor dialog.
   2. Click the  *Data-defined override* widget near the *Rotation* option of the top *Marker* level (preferably) of the symbol layers.
   3. Choose a field in the *Field Type* combobox. Values of this field are hence used to rotate each feature's symbol accordingly.

      You can also check the Store data in project entry to generate an *auxiliary data storage* field to control the rotation value.

Белешка

Make sure that the same field is assigned to all the symbol layers

Setting the data-defined rotation field at the topmost level of the symbol tree automatically propagates it to all the symbol layers, a prerequisite to perform graphical symbol rotation with the *Rotate Point Symbols* tool. Indeed, if a symbol layer does not have the same field attached to its rotation property, the tool will not work.



Слика12.91: Rotating a point symbol

2. Then click on a point symbol in the map canvas with the Rotate Point Symbols tool
3. Move the mouse around. A red arrow with the rotation value will be visualized (see Сл.12.91). If you hold the `Ctrl` key while moving, the rotation will be done in 15 degree steps.
4. When you get the expected angle value, click again. The symbol is rendered with this new rotation and the associated field is updated accordingly.

   You can right-click to abort symbol rotation.

Offset Point Symbols

The Offset Point Symbols allows you to interactively change the rendered position of point symbols in the map canvas. This tool behaves like the Rotate Point Symbols tool except that it requires you to connect a field to the data-defined *Offset (X,Y)* property of each layer of the symbol. The field will then be populated with the offset coordinates for the features whose symbol is moved in the map canvas.

1. Associate a field to the data-defined widget of the *Offset (X,Y)* property of the symbol. If the symbol is made with many layers, you may want to assign the field to each of them
2. Select the Offset Point Symbols tool
3. Click a point symbol
4. Move to a new location
5. Click again. The symbol is moved to the new place. Offset values from the original position are stored in the linked field.

   You can right-click to abort symbol offset.

Белешка

The  Offset Point Symbols tool doesn't move the point feature itself; you should use the  Vertex Tool (Current Layer) or  Move Feature tool for this purpose.

Trim/Extend Feature

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:

- **Trim**: removes parts of the line segment or polygon boundary, beyond the cutting line
- **Extend**: extends polygon boundaries or line segments so that they can snap to the cutting line.

In order to trim or extend existing geometries:

1. Select the  Trim/Extend tool
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.

Белешка

Snapping is automatically enabled when this tool is activated. Your original snapping settings will be restored once the tool is deactivated.

Пажња

Pay attention to the modified geometry while using the  Trim/Extend tool. Depending on the inputs, it can create invalid geometries, potentially resulting in failure at layer saving.

12.3.6 Shape digitizing

The *Shape Digitizing* toolbar offers a set of tools to draw lines or polygons features of regular shape. It is synchronized with the  Digitize Shape *geometry drawing method* you can select on the *Digitizing Toolbar*. To use it:

1. Display the toolbar: *View* ► *Toolbars* ► *Shape Digitizing*

2. Select a tool that creates or modifies the shape of a geometry, e.g.  Add line feature,  Add polygon feature,  Add part,  Add ring,  Reshape Features, …
3. The  Digitize with segment button on the *Digitizing Toolbar* is enabled. The first time, you may need to switch it to the  Digitize Shape in order to enable tools on the *Shape Digitizing* toolbar.
4. Pick a shape digitizing tool and draw.

Circular string by radius

The  Circular string by radius button allows to add line or polygon features with a circular geometry, given two nodes on the curve and a radius:

1. Left click twice to place the two points on the geometry.
2. A *Radius* widget in the top right corner of the map canvas displays current radius (corresponding to distance between the points). Edit that field to the value you want.
3. An overview of the arcs matching these constraints is displayed while moving around the cursor. Right-click to validate when the expected arc is shown.
4. Add a new point to start shaping another arc.

Белешка

Curved geometries are stored as such only in compatible data provider

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.

Draw Circles

There is a set of tools for drawing circles. The tools are described below.

Circles are converted into circular strings. Therefore, as explained in *Circular string by radius*, if allowed by the data provider, it will be saved as a curved geometry, if not, QGIS will segmentize the circular arcs.

-  Circle from 2 points: The two points define the diameter and the orientation of the circle. (Left-click, right--click)
-  Circle from 3 points: Draws a circle from three known points on the circle. (Left-click, left-click, right-click)
-  Circle by a center point and another point: Draws a circle with a given center and a point on the circle (Left-click, right-click). When used with the *The Advanced Digitizing panel* this tool can become a „Add circle from center and radius" tool by setting and locking the distance value after first click.
-  Circle from 3 tangents: Draws a circle that is tangential to three segments. **Note that you must activate snapping to segments** (See *Setting the snapping tolerance and search radius*). Click on a segment to add a tangent. If two tangents are parallel, the coordinates of the click on the first parallel tangent are used to determine the positioning of the circle. If three tangents are parallel, an error message appears and the input is cleared. (Left-click, left-click, right-click)
-  Circle from 2 tangents and a point: Similar to circle from 3 tangents, except that you have to select two tangents, enter a radius and select the desired center.

Draw Ellipses

There is a set of tools for drawing ellipses. The tools are described below.

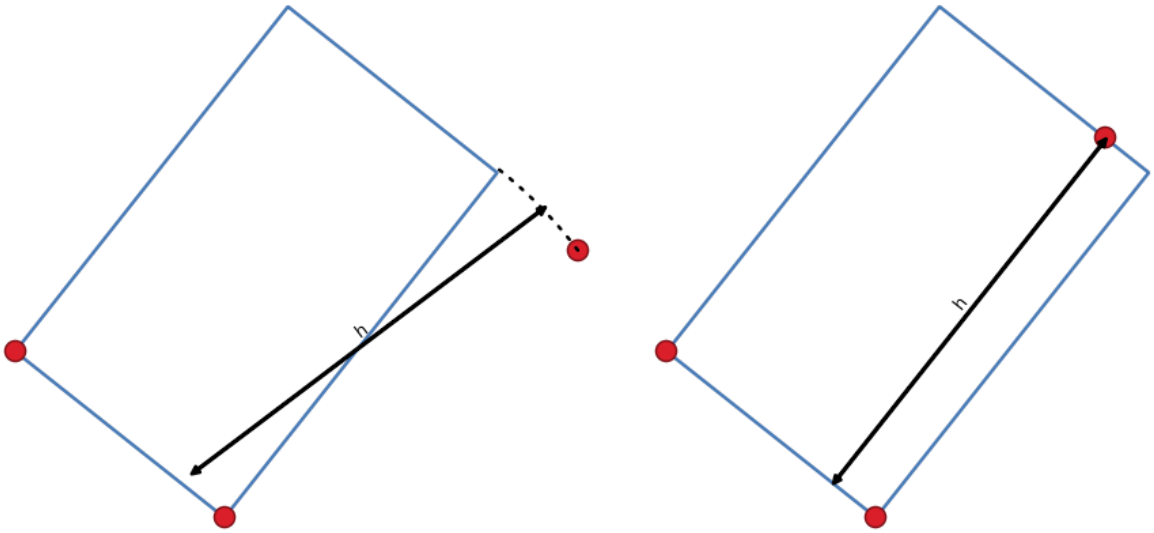
Ellipses cannot be converted as circular strings, so they will always be segmented.

- Ellipse from center and two points: Draws an ellipse with a given center, major axis and minor axis. (Left-click, left-click, right-click)
- Ellipse from center and a point: Draws an ellipse into a bounding box with the center and a corner. (Left-click, right-click)
- Ellipse from extent: Draws an ellipse into a bounding box with two opposite corners. (Left-click, right-click)
- Ellipse from foci: Draws an ellipse by 2 points for foci and a point on the ellipse. (Left-click, left-click, right-click)

Draw Rectangles

There is a set of tools for drawing rectangles. The tools are described below.

- Rectangle from center and a point: Draws a rectangle from the center and a corner. (Left-click, right-click)
- Rectangle from extent: Draws a rectangle from two opposite corners. (Left-click, right-click)
- Rectangle from 3 points (distance): Draws an oriented rectangle from three points. The first and second points determine the length and angle of the first edge. The third point determines the length of the other edge. One can use *The Advanced Digitizing panel* to set the length of the edges. (Left-click, left-click, right-click)
- Rectangle from 3 points (projected): Same as the preceding tool, but the length of the second edge is computed from the projection of the third point on the first edge. (Left-click, left-click, right-click)



Слика12.92: Draw rectangle from 3 points using distance (right) and projected (left)

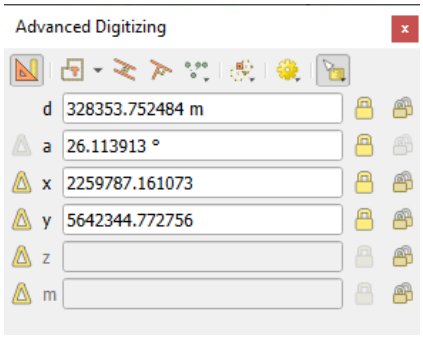
Draw Regular Polygons

There is a set of tools for drawing regular polygons. The tools are described below. Left-click to place the first point. A dialog appears, where you can set the number of polygon edges. Right-click to finish the regular polygon.

- Regular polygon from two points: Draws a regular polygon where the two points determine the length and angle of the first edge.
- Regular polygon from center and a point: Draws a regular polygon from the provided center point. The second point determines the angle and distance to the middle of an edge.
- Regular polygon from center and a corner: Same as the preceding tool, but the second point determines the angle and distante to a vertex.

12.3.7 The Advanced Digitizing panel

When capturing, reshaping, splitting new or existing geometries you also have the possibility to use the Advanced Digitizing panel. You can digitize lines exactly parallel or perpendicular to a particular angle or lock lines to specific angles. Furthermore, you can make a precise definition of your new geometry by entering X and Y coordinates as well as Z for 3D features, or M values.



Слика12.93: The Advanced Digitizing panel

The *Advanced Digitizing* panel can be opened either with a right-click on the toolbar, from *View* ► *Panels* ► menu or pressing `Ctrl+4`. Once the panel is visible, click the Enable advanced digitizing tools button to activate the set of tools.

Белешка

The tools are not enabled if the map view is in geographic coordinates.

The aim of the Advanced Digitizing tool is to lock coordinates, lengths, and angles when moving the mouse during the digitizing in the map canvas.

You can also create constraints with relative or absolute reference. Relative reference means that the next vertex constraints' values will be relative to the previous vertex or segment.

The toolbar

At the top of the *Digitizing panel*, you find the following buttons:

- Enable advanced digitizing tools
- Construction mode: allows to capture the clicks' positions to reuse as reference points to lock distance, angle, X, Y, Z or M relative values. More details at *Construction mode*.

-  Parallel to draw a line parallel to an existing one (more at *Parallel and perpendicular lines*)
-  Perpendicular to draw a line perpendicular to an existing one (more at *Parallel and perpendicular lines*)
-  Construction Tools provides a couple of options that constrain the vertices placement based on extrapolated coordinates of existing elements:
  -  *Line Extension*: hover over a segment and you get a purple dotted line extending the segment across the map canvas. You can snap the vertex anywhere on this virtual line.
  -  *X/Y Point*: hover over a vertex and you get a purple dotted line along its X or Y coordinate, across the map canvas. You can snap the vertex anywhere on this virtual line. It is even possible to hover over two different vertices, generating virtual coordinate lines for both, and snap to their intersection.
-  2-circle Point Intersection: allows you to digitize a point or vertex at the intersection of two circles. (more at *2-circle point intersection*).
-  Snap to common angles: when moving the cursor, displays a virtual line that you can snap to to add the next vertex. The snapping line is defined by the last added vertex and an (absolute or relative to previous segment) angle from a preset list (following steps of 0.1°, 0.5°, 1°, 5°, 10°, 15°, 18°, 22.5°, 30°, 45° or 90°). Choose *Do not snap to common angles* to disable this feature.

  Snapping to features can be used along with snapping to common angles for accurate digitizing. For a fine--grained control on how the target element to snap to is retained, you can indicate whether to prioritize snapping to features over common angles, and vice-versa under the *Snapping priority* entry. You can switch from one method to the other during the digitizing operation, and this avoids disabling any of the snapping options in the meantime. Press `N` (or `Shift+N`) during a digitizing operation to cycle through the angles list.
-  Floater settings: if the *Show floater* item is checked, a contextual menu with digitizing information follows the cursor during digitizing. The values can be accessed using the *panel's shortcuts*, edited and  Locked after validation (pressing `Enter`). The type of information to display can be selected in the bottom part of the menu:
  - *Show distance*
  - *Show angle*
  - *Show XY coordinates*
  - *Show Z value*
  - *Show M value*
  - *Show bearing/azimuth*
  - *Show common snapping angle*

Below the toolbar, you will find a number of text boxes whose value reflects by default the position or movement of the cursor in the map canvas. Editing these values helps you constrain the position of the items you edit:

- *d* for the distance from a reference position, usually the last edited vertex
- *a* for the angle (absolute or relative) from a reference position, usually the last edited segment
- *x* for the X coordinate of the pointer
- *y* for the Y coordinate of the pointer
- *z* for the default Z value or the Z coordinate of the vertex or segment under the pointer
- *m* for the default M value or the M value of the vertex or segment under the pointer

Keyboard shortcuts

To speed up the use of Advanced Digitizing Panel, there are a couple of keyboard shortcuts available:

Табела12.5: Keyboard shortcuts of the Advanced Digitizing Panel tools

Key	Simple	`Ctrl+` or `Alt+`	`Shift+`
`D`	Set distance	Lock distance	
`A`	Set angle	Lock angle	Toggle relative angle to last segment
`X`	Set X coordinate	Lock X coordinate	Toggle relative X to last vertex
`Y`	Set Y coordinate	Lock Y coordinate	Toggle relative Y to last vertex
`Z`	Set Z coordinate	Lock Z coordinate	Toggle relative Z to last vertex
`M`	Postavi M vrednost	Lock M value	Toggle relative M to last vertex
`C`	Toggle construction mode		
`P`	Toggle perpendicular and parallel modes		

Белешка

Z coordinate and M value options are available only if compatible with the layer geometry dimension.

Absolute reference digitizing

When drawing a new geometry from scratch, it is very useful to have the possibility to start digitizing vertices at given coordinates.

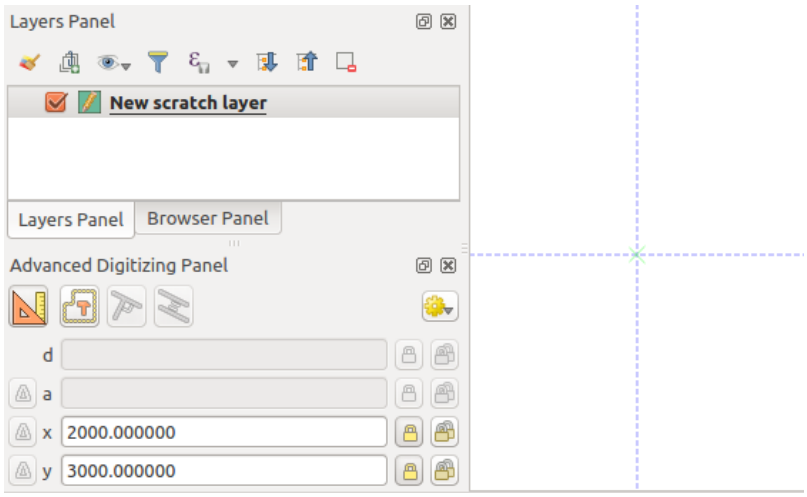
For example, to add a new feature to a polygonal layer, click the button. You can enter the exact coordinates where you want to start editing the feature, i.e.:

1. Click the *x* text box (or use the `X` keyboard shortcut).
2. Type the X coordinate value you want and press `Enter` or click the button to their right to lock the mouse to the X axis on the map canvas.
3. Click the *y* text box (or use the `Y` keyboard shortcut).
4. Type the Y coordinate value you want and press `Enter` or click the button to their right to lock the mouse to the Y axis on the map canvas.
5. If the layer has Z coordinate or M values, the corresponding *z* or *m* widget is enabled and displays its default value, as set in *Settings* ► *Options* ► *Digitizing* tab.
    1. Click the *z* or *m* text box (or use respectively the `Z` or `M` keyboard shortcut).
    2. Type the coordinate value you want and press `Enter` or click the button to their right to lock the value in the widget.

Белешка

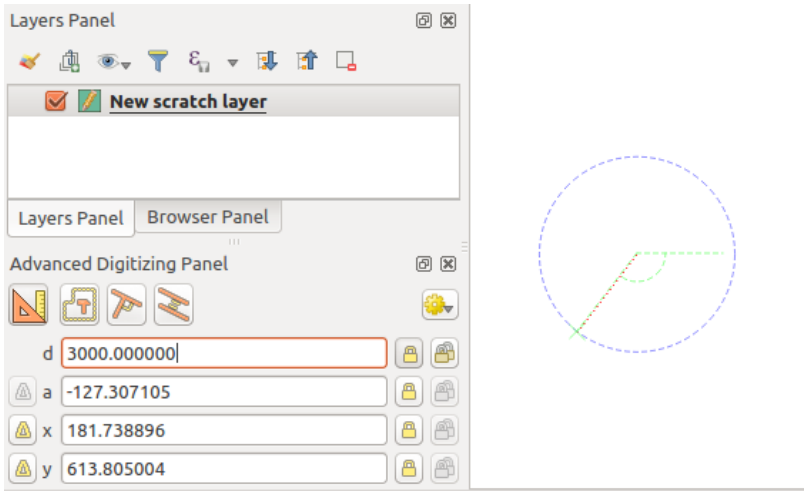
Read *Rules of Z coordinate or M value assignment* for details on how Z coordinate and M values are automatically determined from existing features.

6. Two blue dotted lines and a green cross identify the exact coordinates you entered. Click on the map canvas to add a vertex at the green cross position.



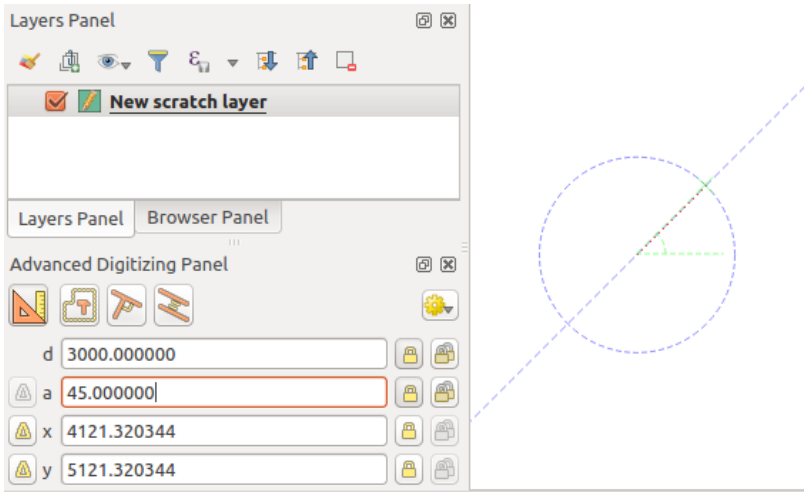
Слика12.94: Start drawing at given coordinates

7. You can proceed as above, adding a new set of coordinates for the next vertex, or switch to another *mode of digitizing* (e.g. segment, curve or stream).
8. If you want to draw a segment of a given length:
    1. Click the *d (distance)* text box (keyboard shortcut `D`)
    2. Type the distance value (in map units)
    3. Press `Enter` or click the button on the right to lock the mouse in the map canvas to the length of the segment. In the map canvas, the latest vertex is surrounded by a circle whose radius is the value entered in the distance text box. A cross on the circle shows the position of the next vertex if you click.



Слика12.95: Fixed length segment

9. You can also constrain the vertex position, setting the angle of the segment. As described before:
    1. Click the *a (angle)* text box (keyboard shortcut `A`)
    2. Type the angle value (in degrees)
    3. Press `Enter` or click the button on the right to lock it. A line going through the latest vertex and rotated based on the set angle appears in the map canvas and a cross on it shows the next vertex position if you click.



Слика12.96: Fixed angle segment

Путоказ

Pressing `Ctrl+<key>` or `Alt+<key>` automatically locks the target property and puts its value into edit. Modify, press `Enter` and you are done. Combined with the Toggle floater, this can be a real time saver, with keyboard digitizing.

Relative reference digitizing

Instead of using absolute values of angles or coordinates, you can also use values relative to the last digitized vertex or segment.

For angles, you can click the button on the left of the *a* text box (or press `Shift+A`) to toggle relative angles to the previous segment. With that option on, angles are measured between the last segment and the mouse pointer.

For coordinates, click the buttons to the left of the *x*, *y*, *z* or *m* text boxes (or press `Shift+<key>`) to toggle relative coordinates to the previous vertex. With these options on, coordinates measurement will consider the last vertex to be the origin of the set coordinates.

Continuous lock

Both in absolute or relative reference digitizing, angle, distance, X, Y, Z and M constraints can be locked continuously by clicking the *Continuous lock* buttons. Using continuous lock allows you to digitize several points or vertices using the same constraints.

Construction mode

You can enable and disable *construction mode* by clicking on the Construction mode icon or with the `C` keyboard shortcut. While in construction mode, clicking the map canvas won't add new vertices, but will capture the clicks' positions so that you can use them as reference points to then lock distance, angle or X, Y, Z, M relative values.

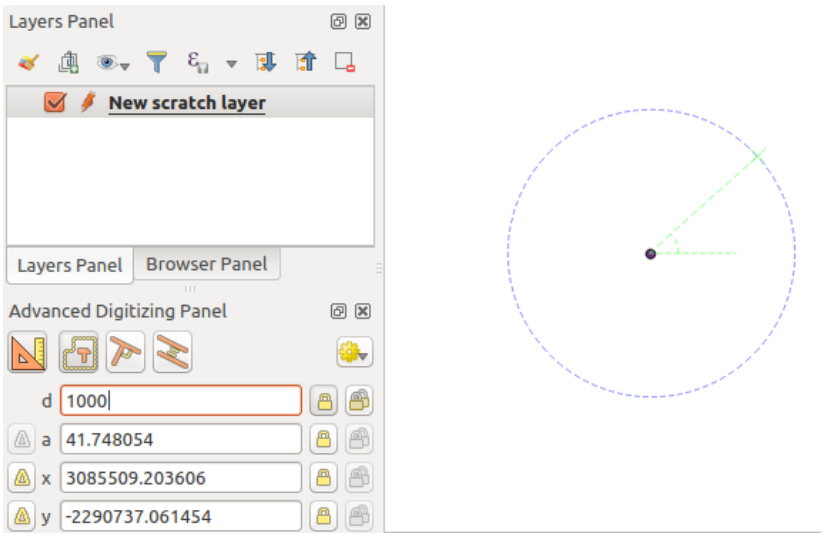
In the Construction mode you will find a drop-down menu where you can choose to:

- *Record Construction Guides*: All construction steps are rendered as dashed lines. Each step that you perform is visually represented, allowing you to trace the construction process. The guides are displayed as long as *Advanced Digitizing* remains active.
- *Show Construction Guides*: Allows you to make construction guides visible or hidden on the canvas. When enabled, all active guides are displayed, offering better spatial orientation and precision during the digitizing process.
- *Snap to Visible Construction Guides*: The guides are snap-able, allowing you to start new construction steps from any point along the existing guides.
- Choose *Clear Construction Guides* to remove all the guides from the canvas.

As an example, the construction mode can be used to draw some point at an exact distance from an existing point.

With an existing point in the map canvas and the snapping mode correctly activated, you can easily draw other points at given distances and angles from it. In addition to the button, you have to activate also the *construction mode* by clicking the Construction mode icon or with the `C` keyboard shortcut.

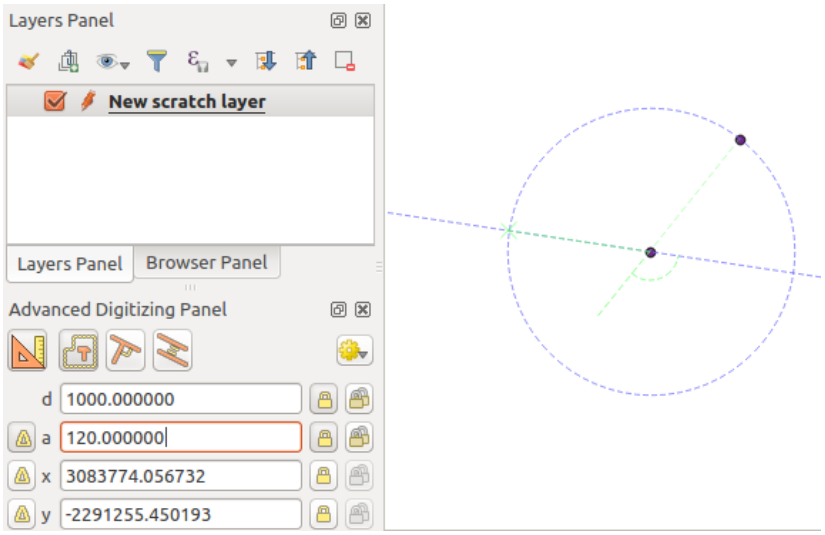
Click next to the point from which you want to calculate the distance and click on the *d* box (`D` shortcut) type the desired distance and press `Enter` to lock the mouse position in the map canvas:



Слика12.97: Distance from point

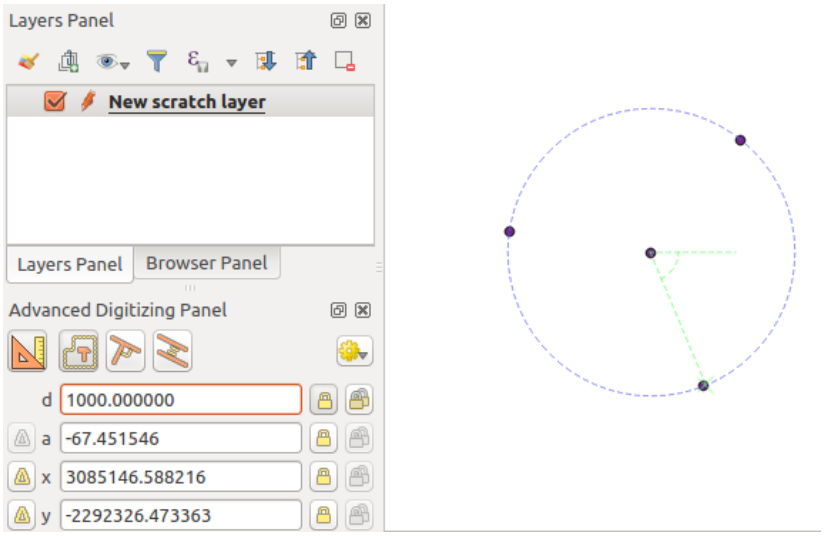
Before adding the new point, press `C` to exit the construction mode. Now, you can click on the map canvas, and the point will be placed at the distance entered.

You can also use the angle constraint to, for example, create another point at the same distance of the original one, but at a particular angle from the newly added point. Click the Construction mode icon or with the `C` keyboard shortcut to enter construction mode. Click the recently added point, and then the other one to set a direction segment. Then, click on the *d* text box (`D` shortcut) type the desired distance and press `Enter`. Click the *a* text box (`A` shortcut) type the angle you want and press `Enter`. The mouse position will be locked both in distance and angle.



Слика12.98: Distance and angle from points

Before adding the new point, press `C` to exit the construction mode. Now, you can click on the map canvas, and the point will be placed at the distance and angle entered. Repeating the process, several points can be added.



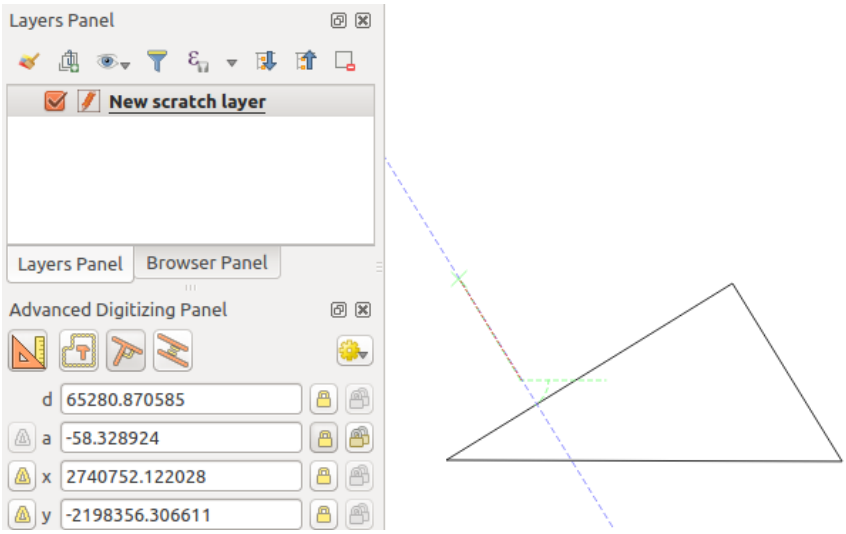
Слика12.99: Points at given distance and angle

Parallel and perpendicular lines

All the tools described above can be combined with the Perpendicular and Parallel tools. These two tools allow drawing segments perfectly perpendicular or parallel to another segment. The target segment can be on another layer, another feature within the layer or the feature being digitized (requires *self-snapping option*).

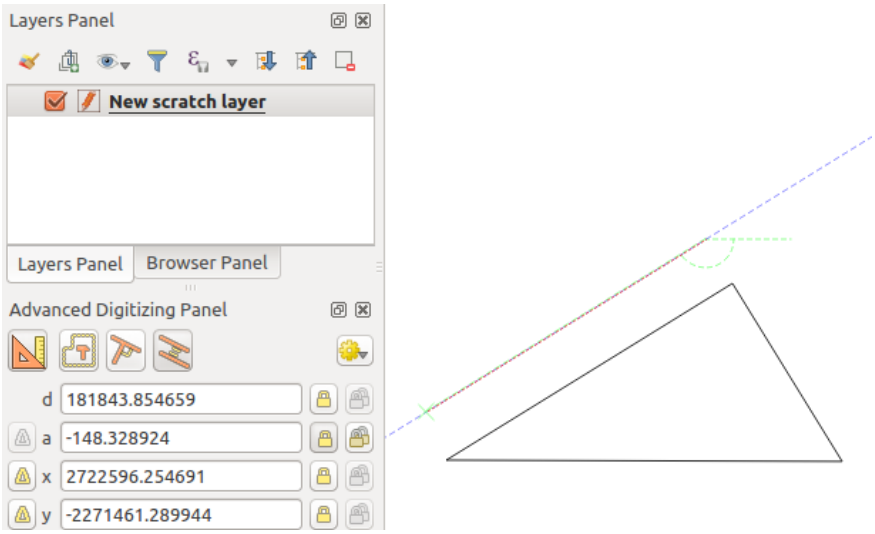
To draw a *perpendicular* segment:

1. First add one of the segment vertices.
2. Click the Perpendicular icon (keyboard shortcut `P`) to activate it.
3. Click on the segment that you want to be perpendicular to.
4. A virtual dotted line perpendicular to the segment through the previous vertex appears. The angle property is locked, constraining the next vertex on that line and, a cross indicates the projected position of the cursor on the line. Click to place the new vertex.



Слика12.100: Perpendicular digitizing

To draw a *parallel* segment, the steps are the same except that you need to click on the Parallel icon (keyboard shortcut `P` twice).



Слика12.101: Parallel digitizing

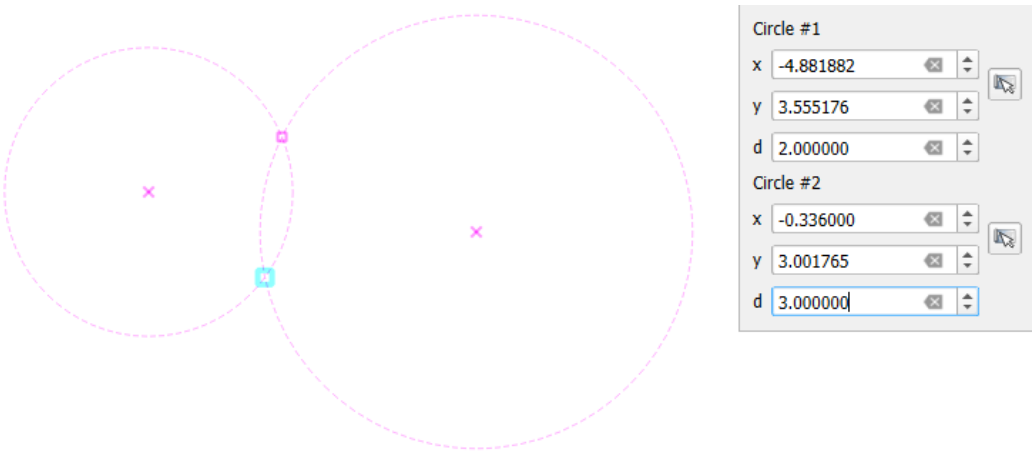
These two tools just find the right angle of the perpendicular and parallel angle and lock this parameter during your editing. Unlock the angle parameter to cancel their use in the middle of the process.

2-circle point intersection

To add a vertex at the intersection of two circles, follow these steps:

1. Click the 2-circle Point Intersection icon.
2. A dialog will open where you can define the parameters for *Circle #1* and *Circle #2*.
3. Click on the map canvas, and the tool will automatically calculate the *X* and *Y* coordinates for the centers of both circles.
4. Enter the distance *d* for each circle.
5. The tool will calculate and display the two intersection points of the circles.

6. Click one of the intersection points to add the new vertex.



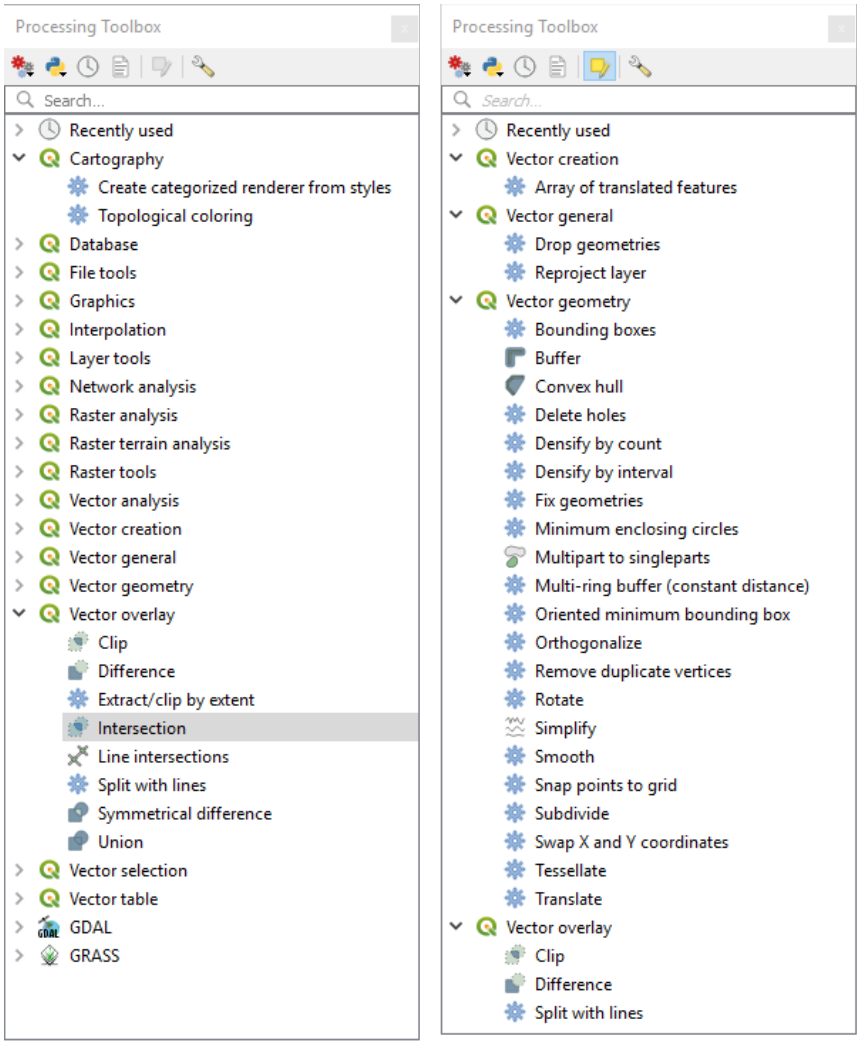
Слика12.102: 2-circle Point Intersection

12.3.8 The Processing in-place layer modifier

The *Processing menu* provides access to a large set of tools to analyze and create new features based on the properties of the input features or their relations with other features (within the same layer or not). While the common behavior is to create new layers as outputs, some algorithms also allow modifications to the input layer. This is a handy way to automate multiple features modification using advanced and complex operations.

To edit features in-place:

1. Select the layer to edit in the *Layers* panel.
2. Select the concerned features. You can skip this step, in which case the modification will apply to the whole layer.
3. Press the Edit Features In-Place button at the top of the *Processing toolbox*. The list of algorithms is filtered, showing only those compatible with in-place modifications, i.e.:
   - They work at the feature source and not at the layer level.
   - They do not change the layer structure, e.g. adding or removing fields.
   - They do not change the geometry type, e.g. from line to point layer.



Слика12.103: Processing algorithms: all (left) vs polygon in-place editors (right)

4. Find the algorithm you'd like to run and double-click it.

   Белешка

   If the algorithm does not need any additional user-set parameters (excluding the usual input and output layer parameters), then the algorithm is run immediately without any dialog popup.

   1. If parameters other than the usual input or output layers are needed, the algorithm dialog pops up. Fill in the required information.
   2. Click *Modify Selected Features* or *Modify All Features* depending on whether there's an active selection.

   Changes are applied to the layer and placed in the edit buffer: the layer is indeed toggled to editing mode with unsaved modification as indicated by the icon next to the layer name.

5. As usual, press Save layer edits to commit the changes in the layer. You can also press Undo to rollback the whole modification.

12.4 Connecting and Editing Data Across Layers

Ability to connect data from different layers is one of the duties of a GIS software. Such a connection can be based on the spatial relationship between the features, or on their shared attributes. QGIS provides tools to handle any of these associations, such as:

- Processing algorithms that can create a new layer as a result of the connection, namely *Spajanje atributa po lokaciji*, *Spajanje atributa po najbližem*, *Spajanje atributa po vrednosti polja*, …
- SQL queries to create a new layer from the *DB Manager* or as a *virtual layer*
- *Joins properties* or *relations settings* that temporarily extend attributes of features in a given layer, with those of features in another layer based on some matching attribute(s).

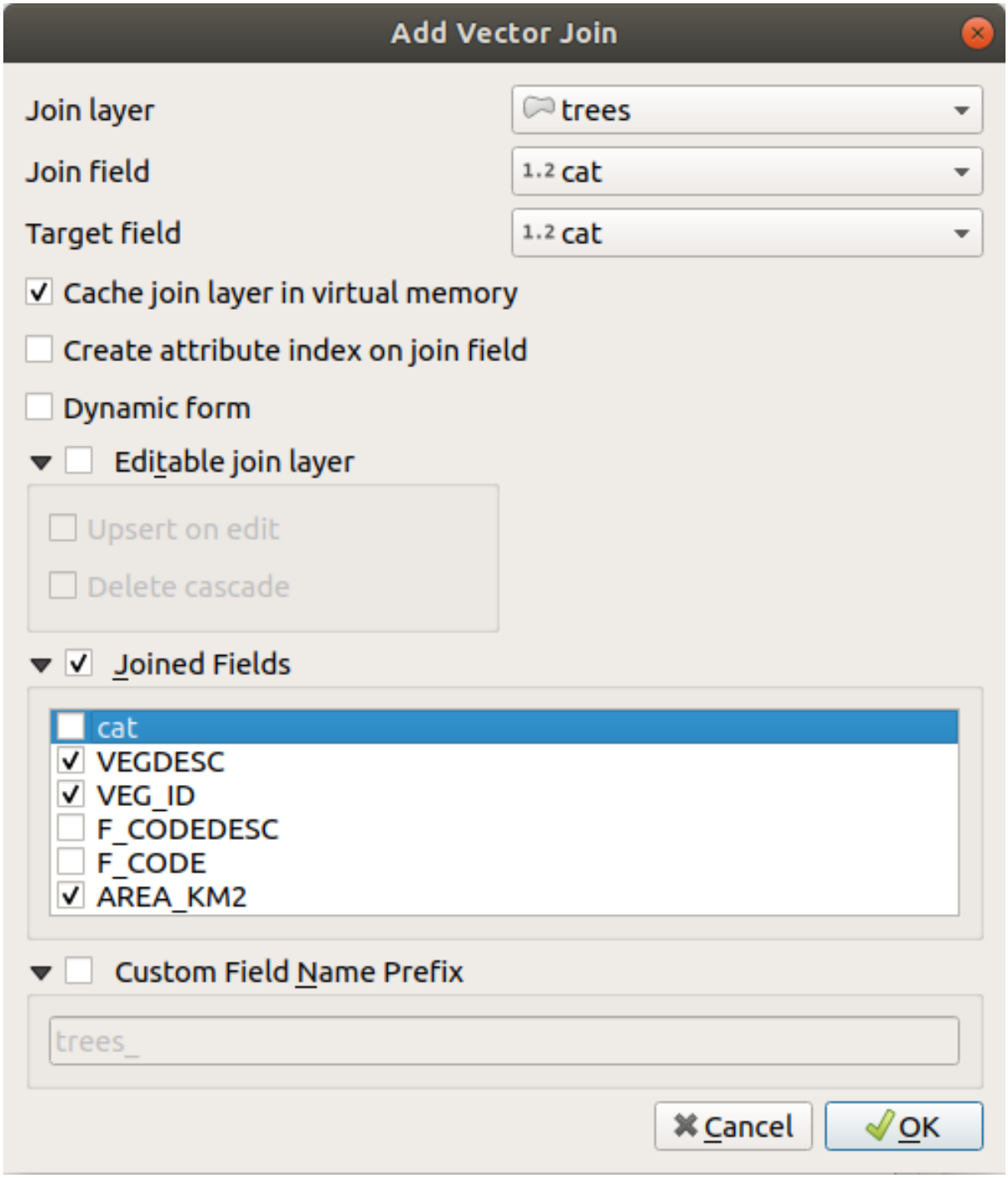
  Joins and relations are technical concepts borrowed from databases to get the most out of data stored in tables by combining their contents. The idea is that features (rows) of different layers (tables) can be associated to each other. The number of rows which are matching each other can be of any value (zero, one, many).

12.4.1 Joining features between two layers

Joins in QGIS allow you to associate features of the current layer to features from another loaded vector layer. Whether they are spatially enabled and the type of geometry do not matter. The join is based on an attribute that is shared by the layers, in a one-to-one relationship.

To create a join on a layer (identified below as `target layer`):

1. Go to the layer *Properties* ►  *Joins* tab
2. Click the  Add new join button. The *Add vector join* dialog appears.
3. Select the *Join layer* you want to connect with the target vector layer
4. Specify the *Join field* (from the `join layer`) and the *Target field* (from the `target layer`). These are the fields that are used to find matching feature in both layers hence they should have values in common.
5. Press *OK* and a summary of selected parameters is added to the *Join* panel.



Слика12.104: Join an attribute table to an existing vector layer

The steps above will create a join, where **ALL** the attributes of the **first matching feature** in the join layer is added to the target layer's feature. The following logic is used to pair features during a join process:

- All the features in the target layer are returned, regardless they have a match
- If the target field contains duplicate values, these features are assigned the same feature from the join layer.
- If the join field contains duplicate matching values, only the first fetched feature is picked.

Белешка

Joins in QGIS are based on a single field matching so most of the times, you would want to make sure that values in the matchable fields are unique.

QGIS provides some more options to tweak the join:

- *Cache join layer in virtual memory*: allows you to cache values in memory (without geometries) from the joined layer in order to speed up lookups.
- *Create attribute index on the join field* to speed up lookups
- *Dynamic form*: helps to synchronize join fields on the fly, according to the *Target field*. This way, constraints for join fields are also correctly updated. Note that it's deactivated by default because it may be very time consuming if you have a lot of features or a myriad of joins.
- If the target layer is editable, then some icons will be displayed in the attribute table next to fields, in order to inform about their status:
  - : the join layer is not configured to be editable. If you want to be able to edit join features from the target attribute table, then you have to check the option *Editable join layer*.
  - : the join layer is well configured to be editable, but its current status is read only.
  - : the join layer is editable, but synchronization mechanisms are not activated. If you want to automatically add a feature in the join layer when a feature is created in the target layer, then you have to check the option *Upsert on edit*. Symmetrically, the option *Delete cascade* may be activated if you want to automatically delete join features.
- *Joined fields*: instead of adding all the fields from the joined layer, you can specify a subset.
- *Custom field name prefix* for joined fields, in order to avoid name collision

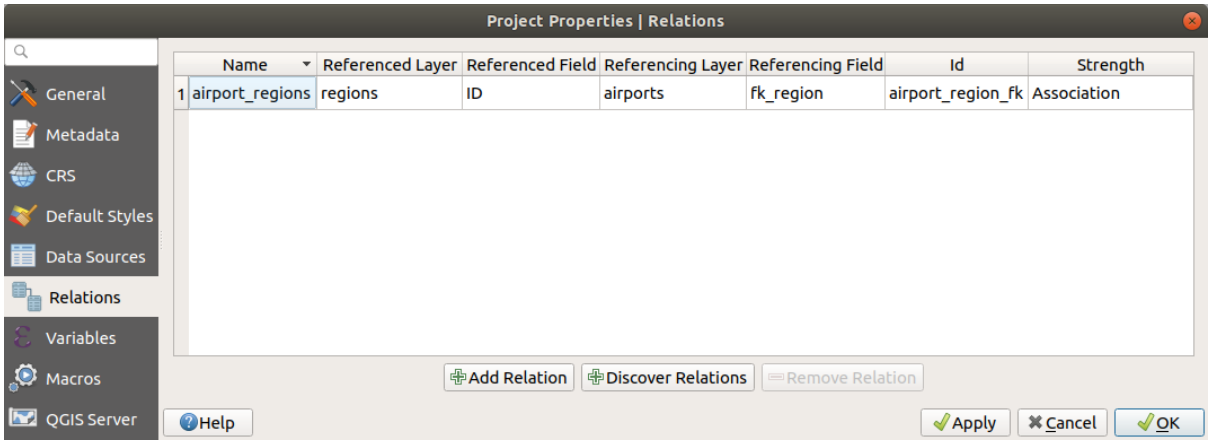
12.4.2 Setting relations between multiple layers

Unlike joins that define a one-to-one link between features across two layers, relations help you build interconnections between multiple features across two or more layers. As such, relations are project level settings and are set in *Project* ► *Properties* ► *Relations* tab. From there, you can:

- *Add relation* whose type can be:
  - *one to many relation*
  - *many to many relation*
  - *polymorphic relation* that you can add or edit with the dedicated tools in the action drop-down menu.

  Белешка

  There is no simple way yet to edit a non-polymorphic relation once it has been created. Only the name can be edited with a double-click. For any other parameters of such a relation you will have to remove and recreate it.
- *Discover relations*: QGIS is able to discover existing relations from supported database formats (PostgreSQL, GeoPackage, ESRI File Geodatabase, …). This can be a good way to ease the relations definition.
- *Remove relation*

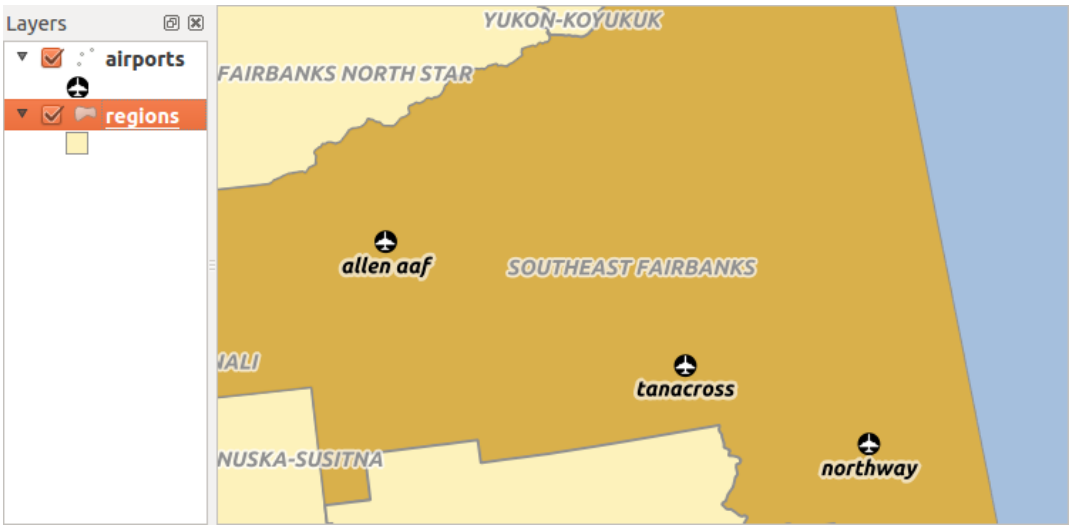


Слика12.105: Relations tab

One to many (1-N) relations

As an example you have a layer with all regions of Alaska (polygon) which provides some attributes about its name and region type and a unique id (which acts as primary key).

Then you get another point layer or table with information about airports that are located in the regions and you also want to keep track of these. If you want to add them to the regions layer, you need to create a one to many relation using foreign keys, because there are several airports in most regions.



Слика12.106: Alaska region with airports

Layers and keys

QGIS makes no difference between a table and a vector layer. Basically, a vector layer is a table with a geometry. So you can add your table as a vector layer. To demonstrate the 1-n relation, you can load the `regions` and `airports` layers in the sample dataset. In practice, each airport belongs to exactly one region while each region can have any number of airports (a typical one to many relation).

In addition to the attributes describing the airports, the aiports layer has another field `fk_region` which acts as a foreign key (if you have a database, you will probably want to define a constraint on it). This `fk_region` field will always contain an id of a region. It can be seen like a pointer to the region it belongs to.

All you have to do is to tell QGIS the relation between the layers so that you can design a custom edit form for editing and QGIS takes care of the setup. It works with different providers (so you can also use it with shape and csv files).

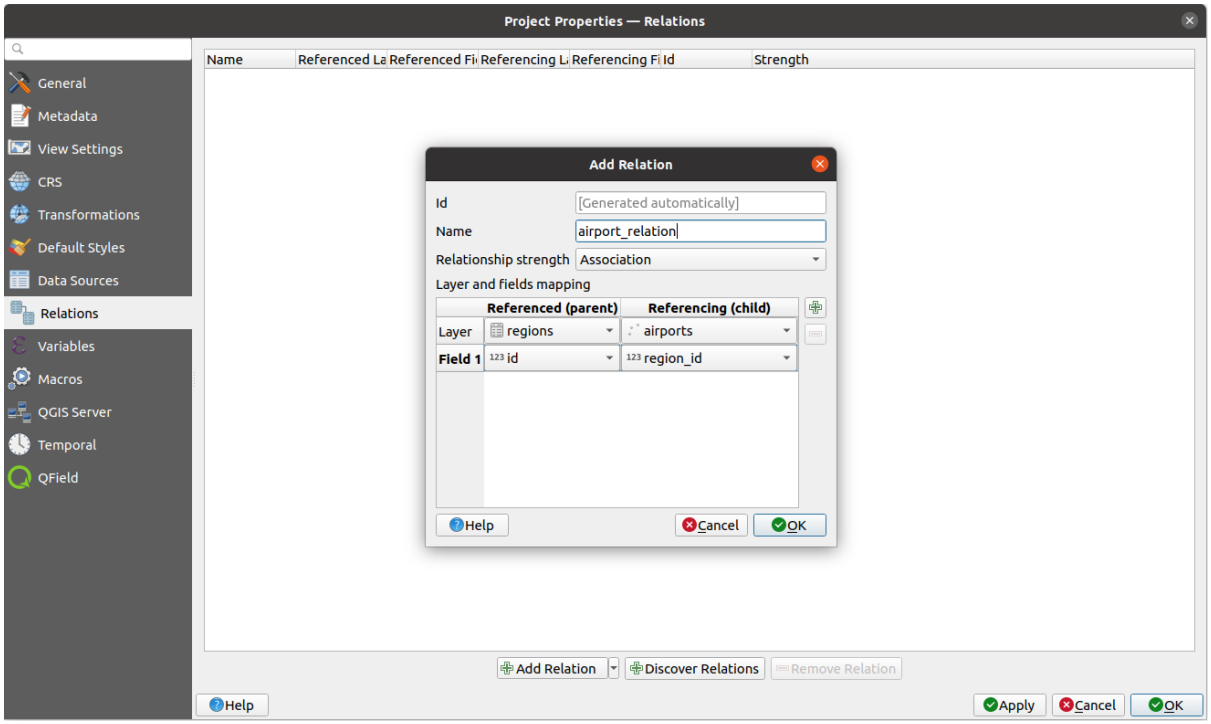
Defining 1-N relations

The first thing we are going to do is to let QGIS know about the relations between the layers. This is done in *Project ► Properties…*. Open the *Relations* tab and click on *Add Relation*.

- **Name** is going to be used as a title. It should be a human readable string describing what the relation is used for. We will just call say **airport_relation** in this case.
- **Referenced Layer (Parent)** also considered as parent layer, is the one with the primary key, pointed to, so here it is the `regions` layer. You need to define the primary key of the referenced layer, so it is `ID`.
- **Referencing Layer (Child)** also considered as child layer, is the one with the foreign key field on it. In our case, this is the `airports` layer. For this layer you need to add a referencing field which points to the other layer, so this is `fk_region`.

> **Белешка**
>
> Sometimes, you need more than a single field to uniquely identify features in a layer. Creating a relation with such a layer requires a **composite key**, i.e. more than a single pair of matching fields. Use the Add new field pair as part of a composite foreign key button to add as many pairs as necessary.

- **Id** will be used for internal purposes and has to be unique. You may need it to build *custom forms*. If you leave it empty, one will be generated for you but you can assign one yourself to get one that is easier to handle
- **Relationship strength** sets the strength of the relation between the parent and the child layer. The default *Association* type means that the parent layer is *simply* linked to the child one while the *Composition* type allows you to duplicate also the child features when duplicating the parent ones and on deleting a feature the children are deleted as well, resulting in cascade over all levels (means children of children of… are deleted as well).

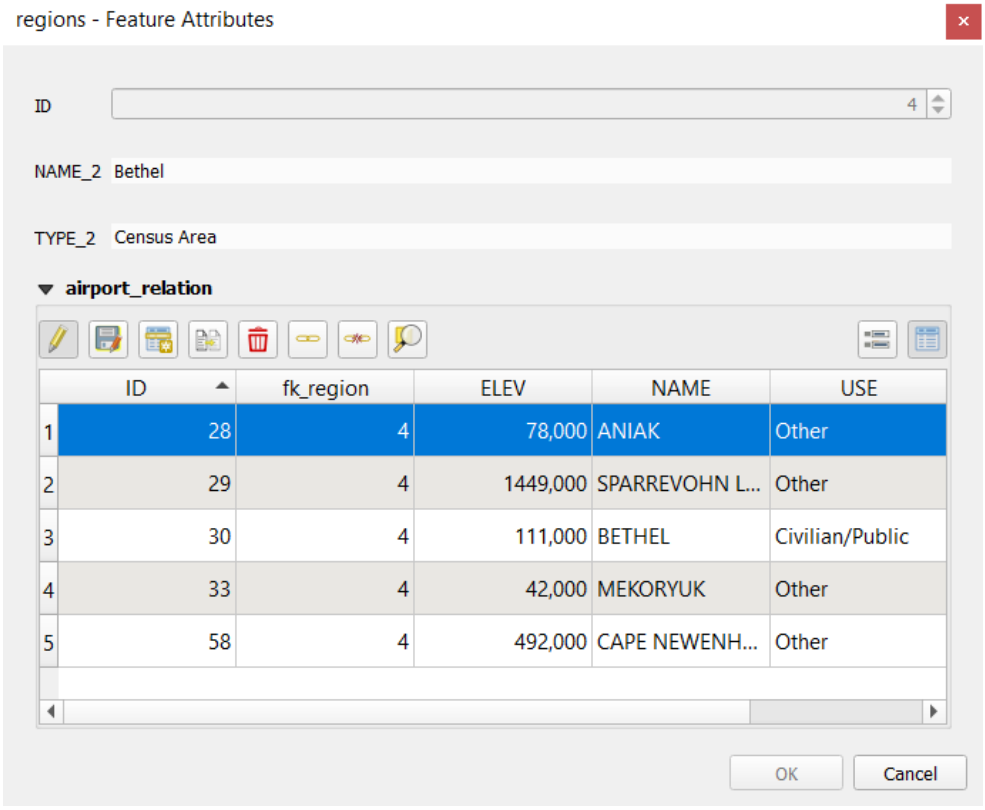


Слика12.107: Adding a relation between regions and airports layers

From the *Relations* tab, you can also press the *Discover Relation* button to fetch the relations available from the providers of the loaded layers. This is possible for layers stored in data providers like PostgreSQL or SpatiaLite.

Forms for 1-N relations

Now that QGIS knows about the relation, it will be used to improve the forms it generates. As we did not change the default form method (autogenerated), it will just add a new widget in our form. So let's select the layer region in the legend and use the identify tool. Depending on your settings, the form might open directly or you will have to choose to open it in the identification dialog under actions.



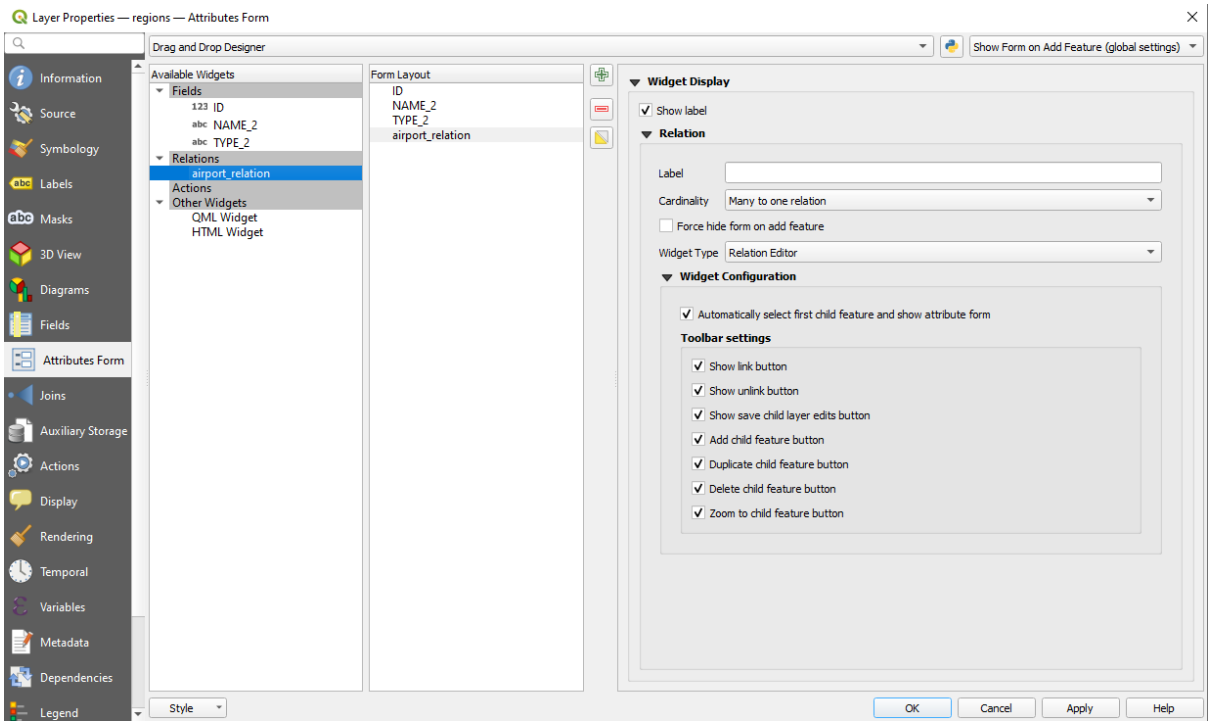
Слика12.108: Identification dialog regions with relation to airports

As you can see, the airports assigned to this particular region are all shown in a table. And there are also some buttons available. Let's review them shortly:

- The button is for toggling the edit mode. Be aware that it toggles the edit mode of the airport layer, although we are in the feature form of a feature from the region layer. But the table is representing features of the airport layer.
- The button is for saving all the edits in the child layer (airport).
- The button lets you digitize the airport geometry in the map canvas and assigns the new feature to the current region by default. Note that the icon will change according to the geometry type.
- The button adds a new record to the airport layer attribute table and assigns the new feature to the current region by default. The geometry can be drawn later with the *Add part* digitizing tool.
- The button allows you to copy and paste one or more child features within the child layer. They can later be assigned to a different parent feature or have their attributes modified.
- The button deletes the selected airport(s) permanently.
- The symbol opens a new dialog where you can select any existing airport which will then be assigned to the current region. This may be handy if you created the airport on the wrong region by accident.

- The symbol unlinks the selected airport(s) from the current region, leaving them unassigned (the foreign key is set to NULL) effectively.
- With the button you can zoom the map to the selected child features.
- The two buttons and to the right switch between the *table view and form view* of the related child features.

If you use the *Drag and Drop Designer* for the regions feature, you can select which tools are available. You can even decide whether to open a new form when a new feature is added using *Force hide form on add feature* option. Be aware that this option implies that not null attributes must take a valid default value to work correctly.



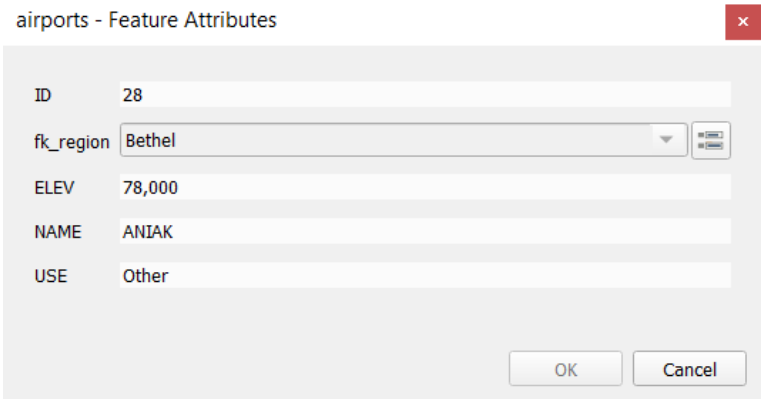
Слика12.109: Drag and Drop Designer for configure regions-airports relation tools

In the above example the referencing layer has geometries (so it isn't just an alphanumeric table) so the above steps will create an entry in the layer attribute table that has no corresponding geometric feature. To add the geometry:

1. Choose *Open Attribute Table* for the referencing layer.
2. Select the record that has been added previously within the feature form of the referenced layer.
3. Use the Add Part digitizing tool to attach a geometry to the selected attributes table record.

If you work on the airport table, the widget Relation Reference is automatically set up for the `fk_region` field (the one used to create the relation), see *Relation Reference widget*.

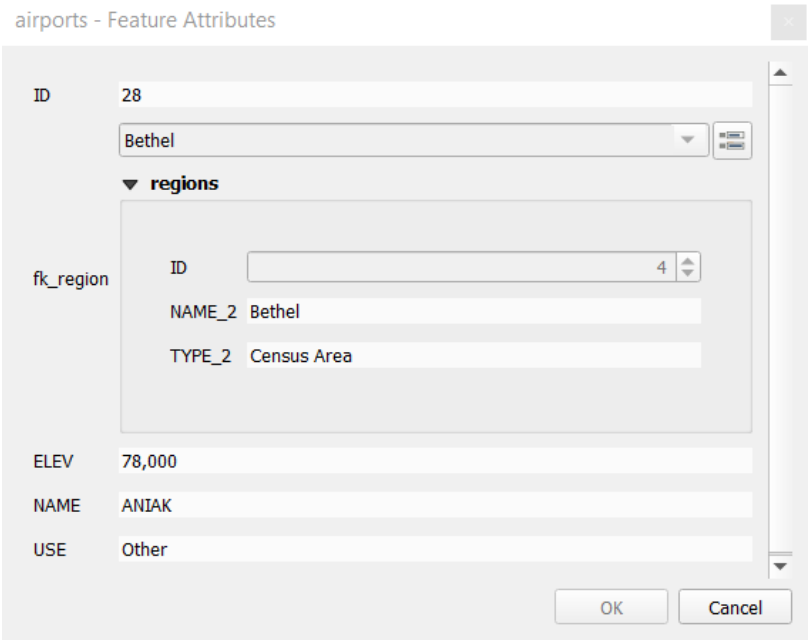
In the airport form you will see the button at the right side of the `fk_region` field: if you click on the button the form of the region layer will be opened. This widget allows you to easily and quickly open the forms of the linked parent features.



Слика12.110: Identification dialog airport with relation to regions

The Relation Reference widget has also an option to embed the form of the parent layer within the child one. It is available in the *Properties* ► *Attributes Form* menu of the airport layer: select the `fk_region` field and check the `Show embedded form` option.

If you look at the feature dialog now, you will see that the form of the region is embedded inside the airports form and will even have a combobox, which allows you to assign the current airport to another region.

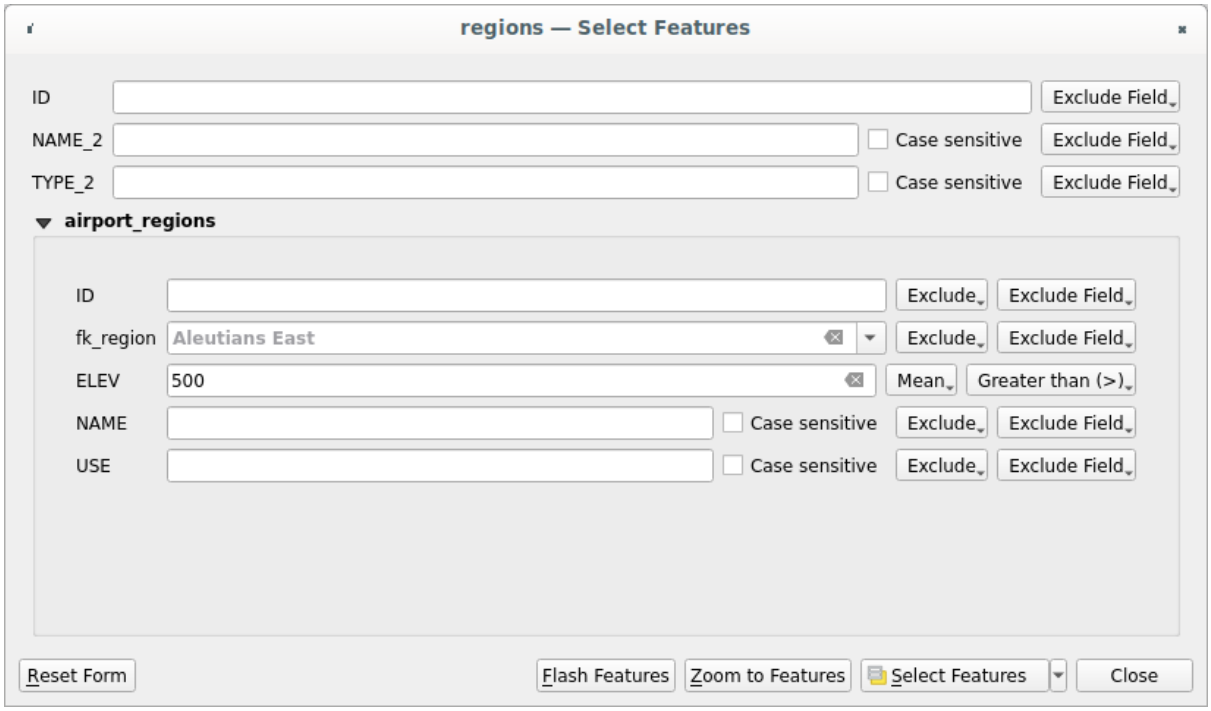


Moreover if you toggle the editing mode of the airport layer, the `fk_region` field has also an autocompleter function: while typing you will see all the values of the `id` field of the region layer. Here it is possible to digitize a polygon for the region layer using the button if you chose the option `Allow adding new features` in the *Properties* ► *Attributes Form* menu of the airport layer.

The child layer can also be used in the *Select Features By Value* tool in order to select features of the parent layer based on attributes of their children.

In Сл.12.111, all the regions where the mean altitude of the airports is greater than 500 meters above sea level are selected.

You will find that many different aggregation functions are available in the form.



Слика12.111: Select parent features with child values

Many-to-many (N-M) relations

N-M relations are many-to-many relations between two tables. For instance, the `airports` and `airlines` layers: an airport receives several airline companies and an airline company flies to several airports.

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)
);
```

(наставак на следећој страни)

(настављено са претходне стране)

```
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
 );
```

Instead of PostgreSQL you can also use GeoPackage. In this case, the three tables can be created manually using the *Database* ► *DB Manager*.... In GeoPackage there are no schemas so the *locations* prefix is not needed.

Foreign key constraints in `airports_airlines` table can´t be created using *Table* ► *Create Table*... or *Table* ► *Edit Table*... so they should be created using *Database* ► *SQL Window*.... GeoPackage doesn't support *ADD CONSTRAINT* statements so the `airports_airlines` table should be created in two steps:

1. Set up the table only with the `id` field using *Table* ► *Create Table*...
2. Using *Database* ► *SQL Window*..., type and execute this SQL code:

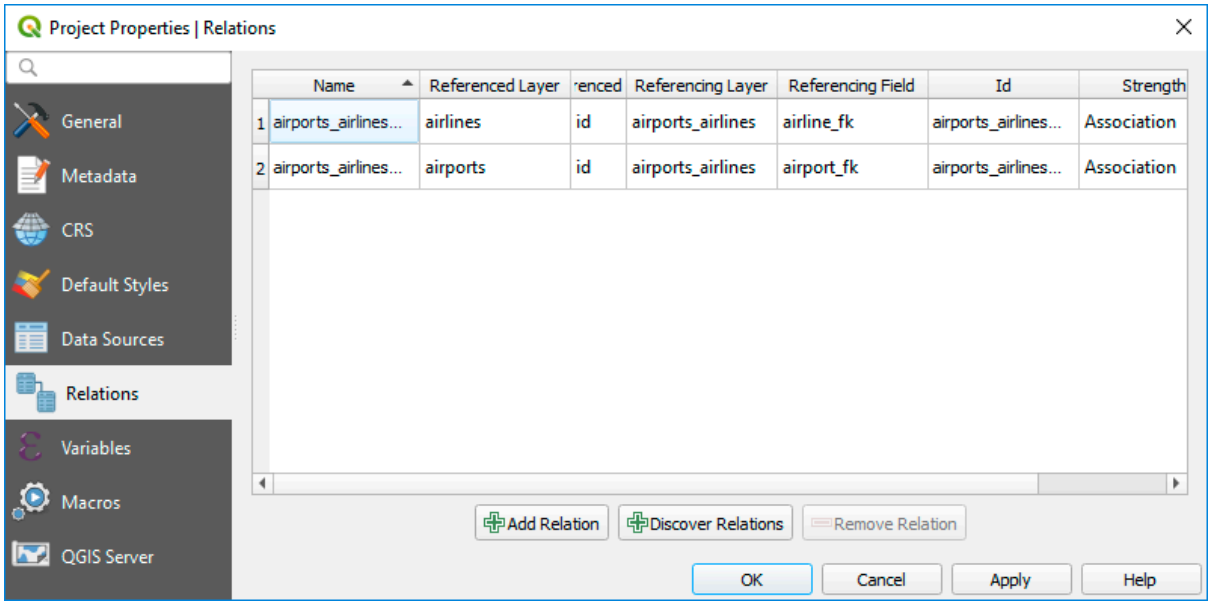
```
ALTER TABLE airports_airlines
   ADD COLUMN airport_fk INTEGER
   REFERENCES airports (id)
   ON DELETE CASCADE
   ON UPDATE CASCADE
   DEFERRABLE INITIALLY DEFERRED;

ALTER TABLE airports_airlines
   ADD COLUMN airline_fk INTEGER
   REFERENCES airlines (id)
   ON DELETE CASCADE
   ON UPDATE CASCADE
   DEFERRABLE INITIALLY DEFERRED;
```

Then in QGIS, you should set up two *one-to-many relations* as explained above:

- a relation between `airlines` table and the pivot table;
- and a second one between `airports` table and the pivot table.

An easier way to do it (only for PostgreSQL) is using the *Discover Relations* in *Project* ► *Properties* ► *Relations*. QGIS will automatically read all relations in your database and you only have to select the two you need. Remember to load the three tables in the QGIS project first.



Слика12.112: Relations and autodiscover

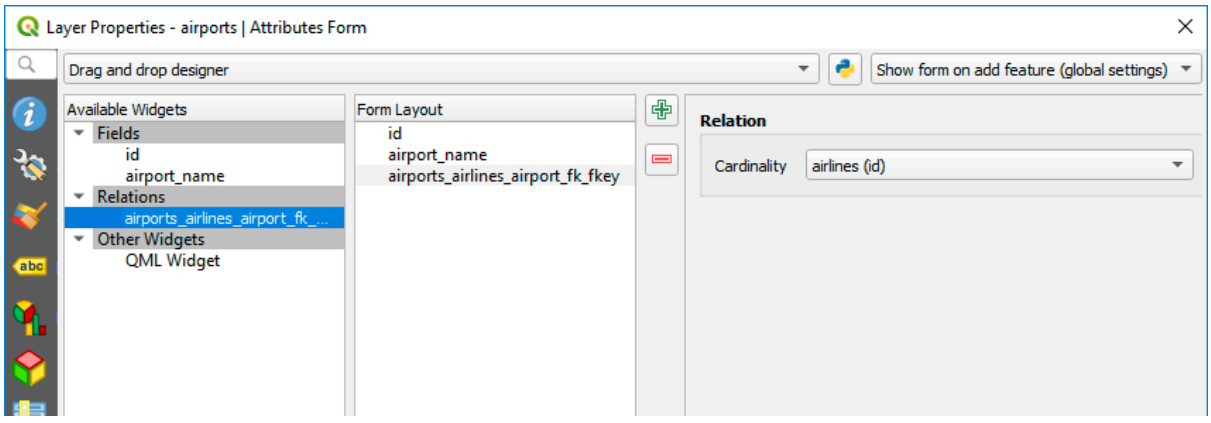
In case you want to remove an `airport` or an `airline`, QGIS won't remove the associated record(s) in `airports_airlines` table. This task will be made by the database if we specify the right *constraints* in the pivot table creation as in the current example.

Белешка

Combining N-M relation with automatic transaction group

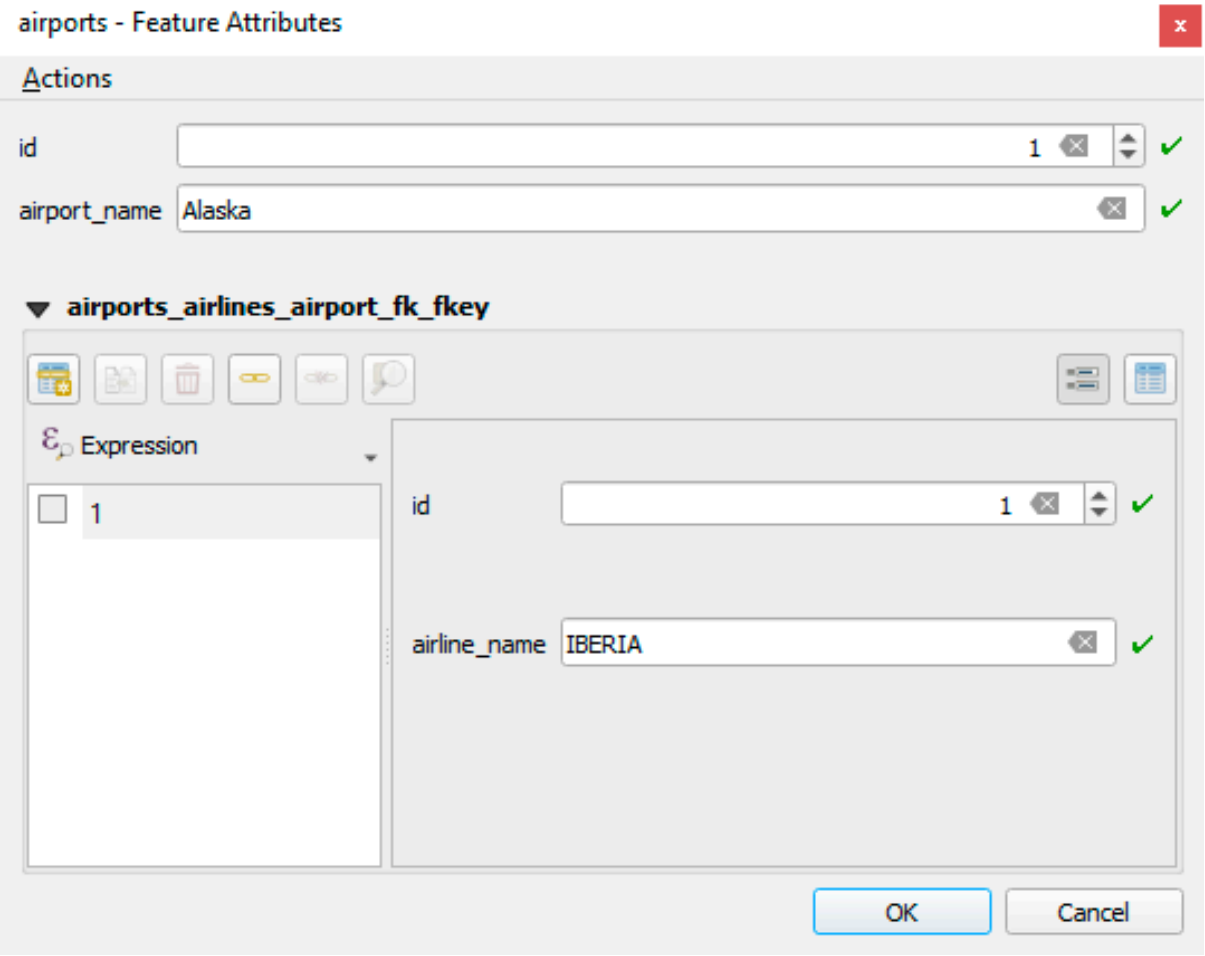
You should enable the transaction mode in *Project Properties* ► *Data Sources* ► when working on such context. QGIS should be able to add or update row(s) in all tables (airlines, airports and the pivot tables).

Finally we have to select the right cardinality in the *Layer Properties* ► *Attributes Form* for the `airports` and `airlines` layers. For the first one we should choose the **airlines (id)** option and for the second one the **airports (id)** option.



Слика12.113: Set relationship cardinality

Now you can associate an airport with an airline (or an airline with an airport) using *Add child feature* or *Link existing child feature* in the subforms. A record will automatically be inserted in the `airports_airlines` table.



Слика12.114: N-M relationship between airports and airlines

Белешка

Using **Many to one relation** cardinality

Sometimes hiding the pivot table in an N-M relationship is not desirable. Mainly because there are attributes in the relationship that can only have values when a relationship is established. If your tables have a geometry field, it could be interesting to activate the *On map identification* option (*Layer Properties* ► *Attributes Form* ► *Available widgets* ► *Fields*) for the foreign key fields in the pivot table.

Белешка

Pivot table primary key

Avoid using multiple fields in the primary key in a pivot table. QGIS assumes a single primary key so a constraint like `constraint airports_airlines_pkey primary key (airport_fk, airline_fk)` will not work.

Polymorphic relations

The purpose

Polymorphic relations are special case of 1-N relations, where a single referencing (document) layer contains the features for multiple referenced layers. This differs from normal relations which require different referencing layer for each referenced layer. A single referencing (document) layer is achieved by adding an adiditonal `layer_field` column in the referencing (document) layer that stores information to identify the referenced layer. In its most simple form, the referencing (document) layer will just insert the layer name of the referenced layer into this field.

To be more precise, a polymorphic relation is a set of normal relations having the same referencing layer but having the referenced layer dynamically defined. The polymorphic setting of the layer is solved by using an expression which has to match some properties of the referenced layer like the table name, layer id, layer name.

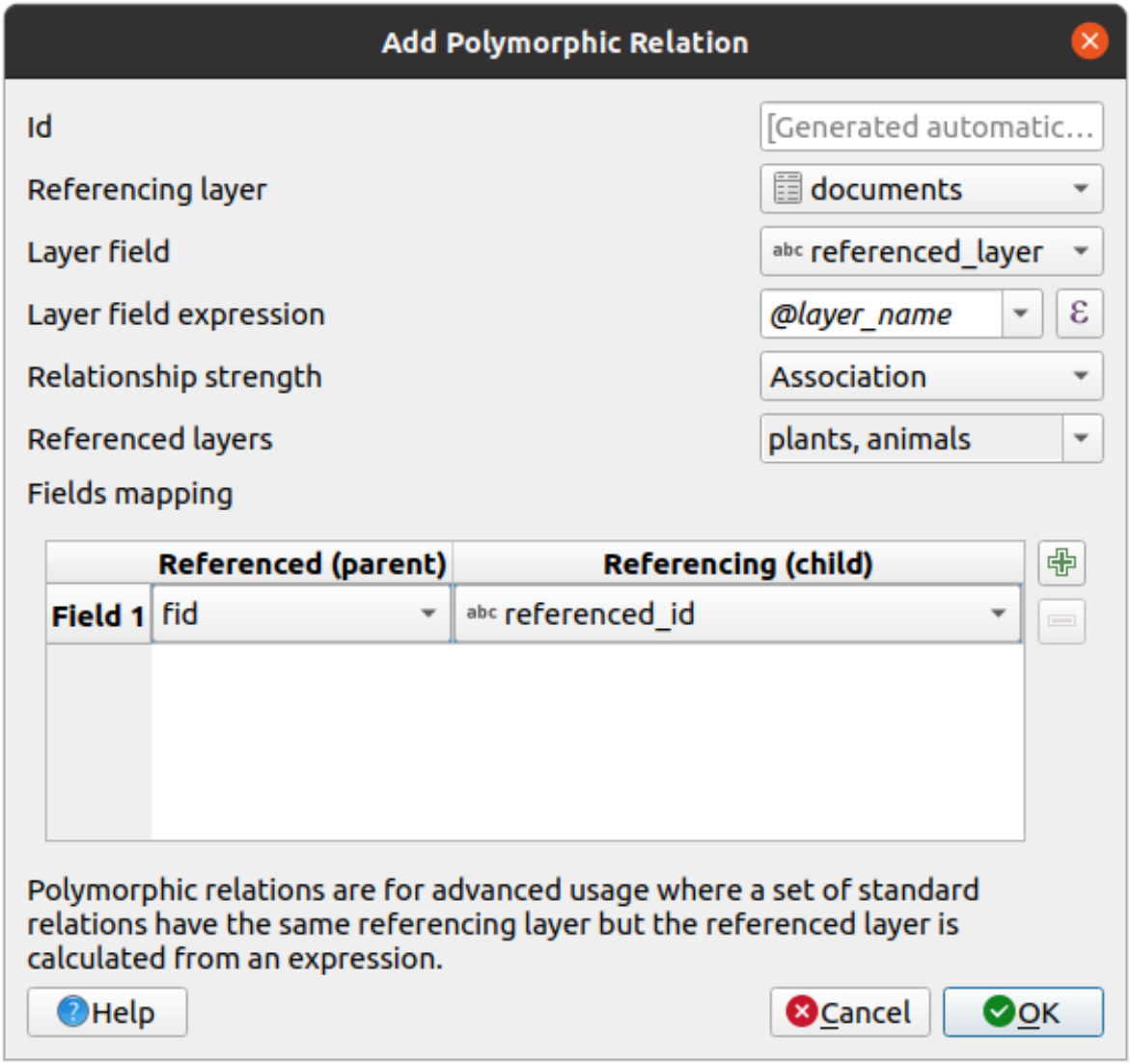
Imagine we are going to the park and want to take pictures of different species of `plants` and `animals` we see there. Each plant or animal has multiple pictures associated with it, so if we use the normal 1:N relations to store pictures, we would need two separate tables, `animal_images` and `plant_images`. This might not be a problem for 2 tables, but imagine if we want to take separate pictures for mushrooms, birds etc.

Polymorphic relations solve this problem as all the referencing features are stored in the same table `documents`. For each feature the referenced layer is stored in the `referenced_layer` field and the referenced feature id in the `referenced_id` field.

Defining polymorphic relations

First, let QGIS know about the polymorphic relations between the layers. This is done in *Project* ► *Properties…*

Open the *Relations* tab and click on the little down arrow next to the  *Add Relation* button, so you can select the *Add Polymorphic Relation* option from the newly appeared dropdown.



Слика12.115: Adding a polymorphic relation using `documents` layer as referencing and `animals` and `plants` as referenced layers.

- **Id** will be used for internal purposes and has to be unique. You may need it to build *custom forms*. If you leave it empty, one will be generated for you but you can assign one yourself to get one that is easier to handle
- **Referencing Layer (Child)** also considered as child layer, is the one with the foreign key field on it. In our case, this is the `documents` layer. For this layer you need to add a referencing field which points to the other layer, so this is `referenced_id`.

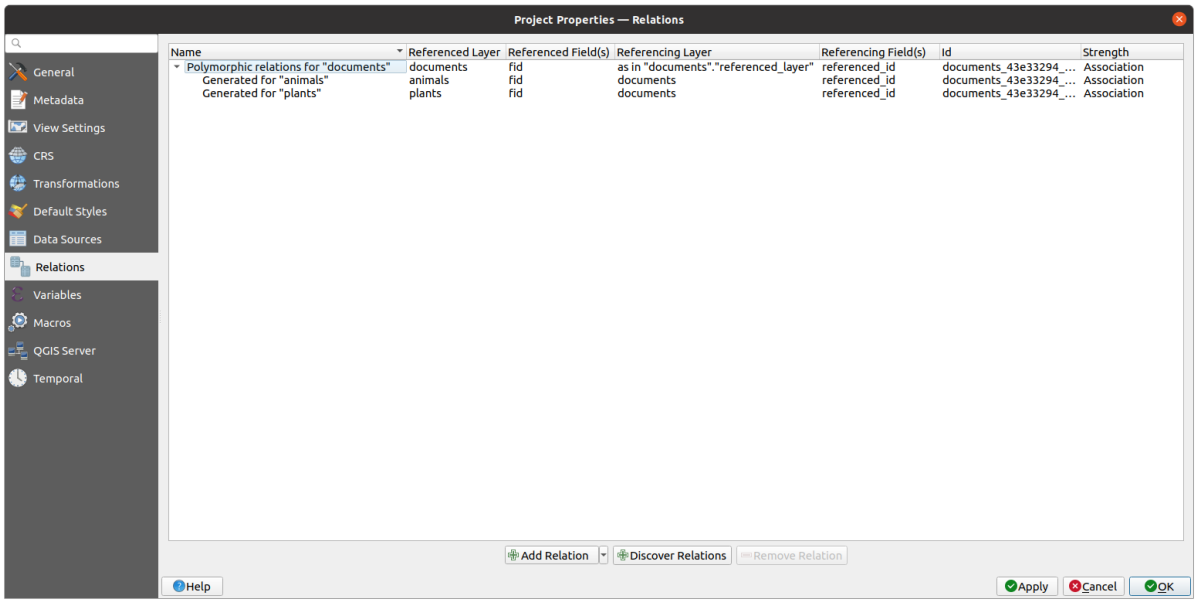
> **Белешка**
>
> Sometimes, you need more than a single field to uniquely identify features in a layer. Creating a relation with such a layer requires a **composite key**, i.e. more than a single pair of matching fields. Use the Add new field pair as part of a composite foreign key button to add as many pairs as necessary.

- **Layer Field** is the field in the referencing table that stores the result of the evaluated layer expression which is the referencing table that this feature belongs to. In our example, this would be the `referenced_layer` field.
- **Layer expression** evaluates to a unique identifier of the layer. This can be the layer name `@layer_name`,

the layer id `@layer_id`, the layer's table name `decode_uri(@layer, 'table')` or anything that can uniquely identifies a layer.

- **Relationship strength** sets the strength of the generated relations between the parent and the child layer. The default *Association* type means that the parent layer is *simply* linked to the child one while the *Composition* type allows you to duplicate also the child features when duplicating the parent ones and on deleting a feature the children are deleted as well, resulting in cascade over all levels (means children of children of… are deleted as well).
- **Referenced Layers** also considered as parent layers, are those with the primary key, pointed to, so here they would be `plants` and `animals` layers. You need to define the primary key of the referenced layers from the dropdown, so it is `fid`. Note that the definition of a valid primary key requires all the referenced layers to have a field with that name. If there is no such field you cannot save a polymorphic relation.

Once added, the polymorphic relation can be edited via the *Edit Polymorphic Relation* menu entry.



Слика12.116: Preview of the newly created polymorphic relation and its child relations for animals and plants.

The example above uses the following database schema:

```
CREATE SCHEMA park;

CREATE TABLE park.animals
(
   fid serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   animal_species text NOT NULL,
   CONSTRAINT animals_pkey PRIMARY KEY (fid)
);

CREATE INDEX animals_geom_idx ON park.animals USING gist (geom);

CREATE TABLE park.plants
(
   fid serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   plant_species text NOT NULL,
   CONSTRAINT plants_pkey PRIMARY KEY (fid)
```

(наставак на следећој страни)

(настављено са претходне стране)

```
);

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

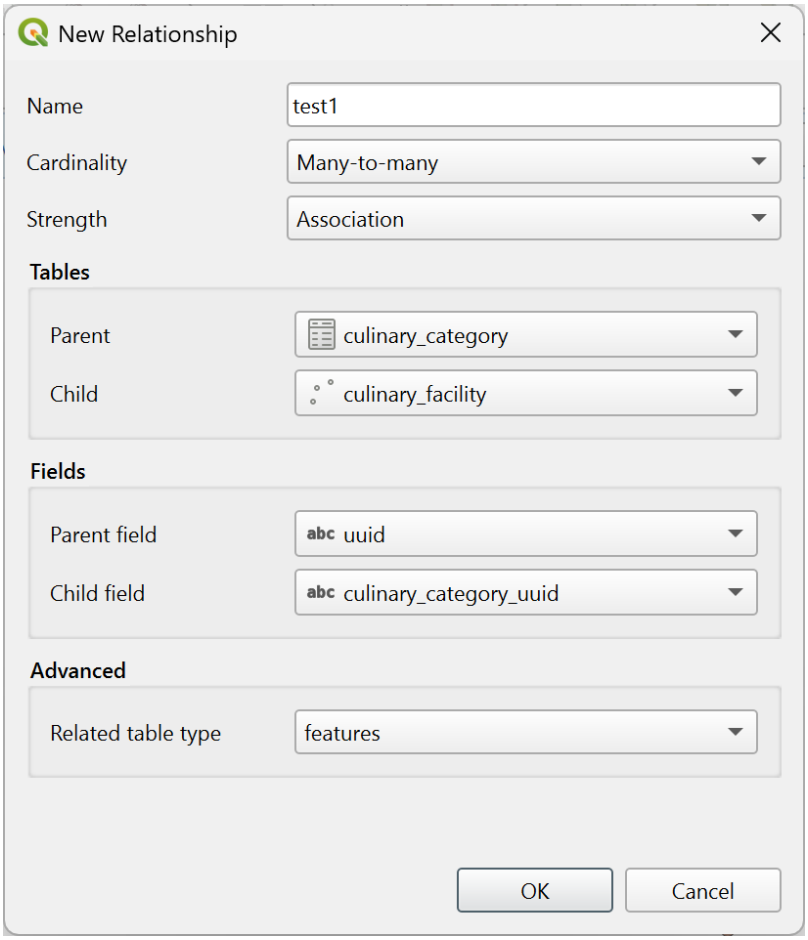
Other than the above mentioned relationships that are created in the project properties and apply only in the active project, QGIS can also read database-level relations, i.e., connections defined between layers in the dataset. This is supported for database providers such as PostgreSQL, SpatiaLite, 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.



Слика12.117: 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.



Слика12.118: 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).

Белешка

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.

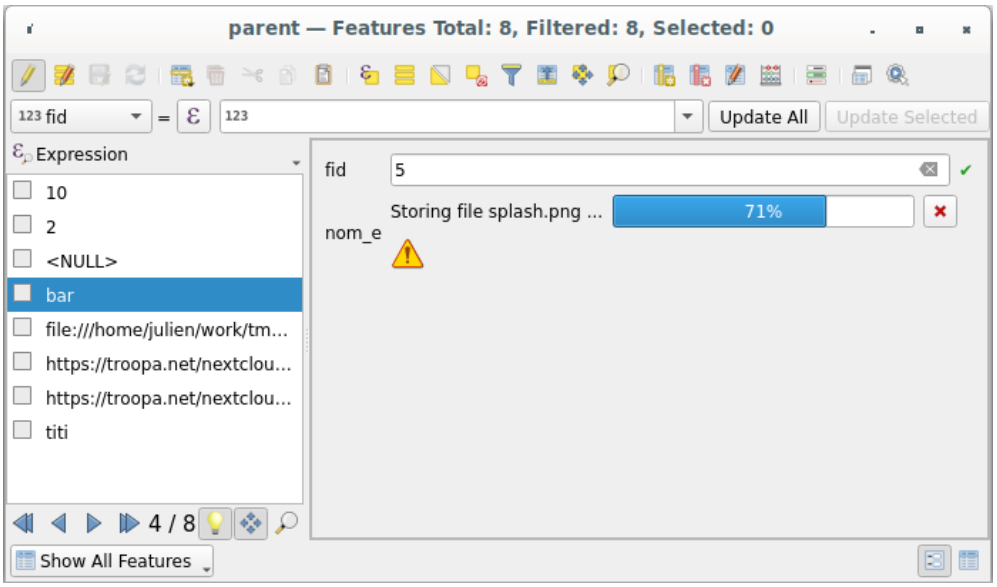
If the external storage system needs to, it's possible to configure an *authentication*.

Белешка

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.



Слика12.119: Storing a file to a WebDAV external storage

Белешка

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 *Log Messages Panel*.

ГЛАВА 13

Rad sa rasterskim podacima

13.1 Raster Properties Dialog

Raster data is made up of pixels (or cells), and each pixel has a value. It is commonly used to store various types of data, including:

- Imagery, such as satellite images, digital aerial photographs, scanned maps
- Elevation data, such as digital elevation models (DEMs), digital terrain models (DTMs)
- Other types of data, such as land cover, soil types, rainfall and many others.

Raster data can be stored in several supported formats, including GeoTIFF, ERDAS Imagine, ArcInfo ASCII GRID, PostgreSQL Raster and others. See more at *Opening Data*.

To view and set the properties for a raster layer, double click on the layer name in the map legend, or right click on the layer name and choose *Properties* from the context menu. This will open the *Raster Layer Properties* dialog.

There are several tabs in the dialog:

Information	*Source*	*Symbology* ([1])
Transparency ([1])	*Labels* ([1])	*Histogram* ([1])
Rendering	*Temporal*	*Pyramids*
Elevation	*Metadata*	*Legend*
Display	*Attribute Tables*	*QGIS Server*
External plugins ([2]) tabs		

Савет

Live update rendering

[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.

The *Layer Styling Panel* provides you with some of the common features of the Layer properties dialog and is a good modeless widget that you can use to speed up the configuration of the layer styles and view your changes on the map canvas.

Белешка

Because properties (symbology, label, actions, default values, forms…) of embedded layers (see *Embedding layers from external projects*) are pulled from the original project file, and to avoid changes that may break this behavior, the layer properties dialog is made unavailable for these layers.

13.1.1 Information Properties

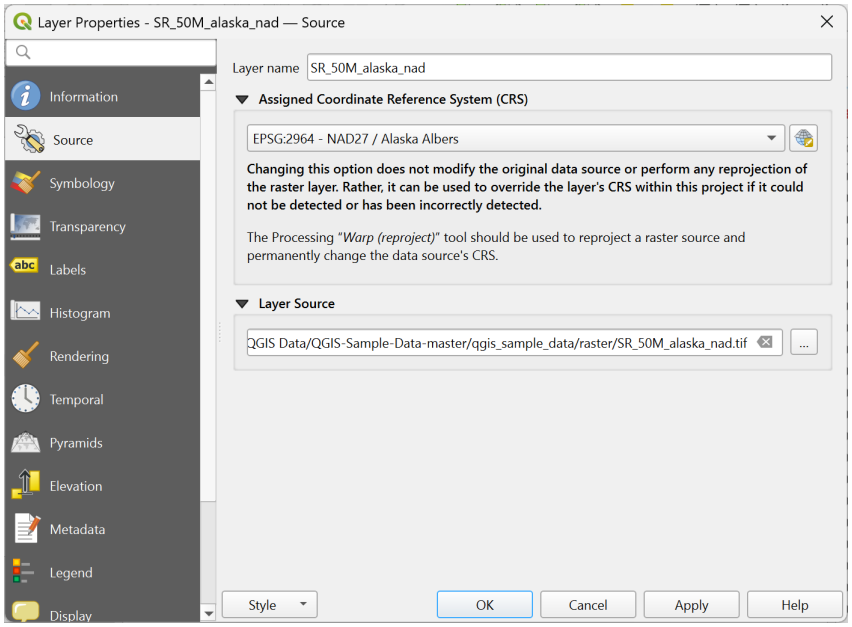
The  *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata for the current layer. Provided information are:

- general such as name in the project, source path, list of auxiliary files, last save time and size, the used provider
- custom properties, used to store in the active project additional information about the layer. Default custom properties include *Identify/format*, which influences how the results from using the  *Identify features* tool over a raster layer are formatted. More properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- based on the provider of the layer: extent, width and height, data type, GDAL driver, bands statistics
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e. whether it's static or dynamic)
- read from layer properties: data type, extent, width/height, compression, pixel size, statistics on bands, number of columns, rows and no-data values of the raster…
- picked from the *filled metadata*: access, extents, links, contacts, history…

13.1.2 Source Properties

The  *Source* tab displays basic information about the selected raster, including:

- the *Layer name* to display in the *Layers Panel*;
- the *Coordinate Reference System*: Displays the layer's *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 the  Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the layer CRS is a wrong or not specified. If you wish to reproject your data, use a reprojection algorithm from Processing or *Save it as new dataset*.
- 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.



Слика13.1: Raster Layer Properties - Source Dialog

13.1.3 Symbology Properties

The raster layer symbology tab is made of three different sections:

- The *Band rendering* where you can control the renderer type to use
- The *Layer rendering* to apply effects on rendered data
- The *Resampling* methods to optimize rendering on map

Band rendering

QGIS offers many different *Render types*. The choice of renderer depends on the data type and the information you'd like to highlight.

1. *Multiband color* - if the file comes with several bands (e.g. a satellite image with several bands).
2. *Paletted/Unique values* - for single band files that come with an indexed palette (e.g. a digital topographic map) or for general use of palettes for rendering raster layers.
3. *Singleband gray* - (one band of) the image will be rendered as gray. QGIS will choose this renderer if the file is neither multiband nor paletted (e.g. a shaded relief map).
4. *Singleband pseudocolor* - this renderer can be used for files with a continuous palette or color map (e.g. an elevation map).
5. *Single color* - the raster layer will be rendered with a single color.
6. *Hillshade* - Creates hillshade from a band.
7. *Contours* - Generates contours on the fly for a source raster band.

Multiband color

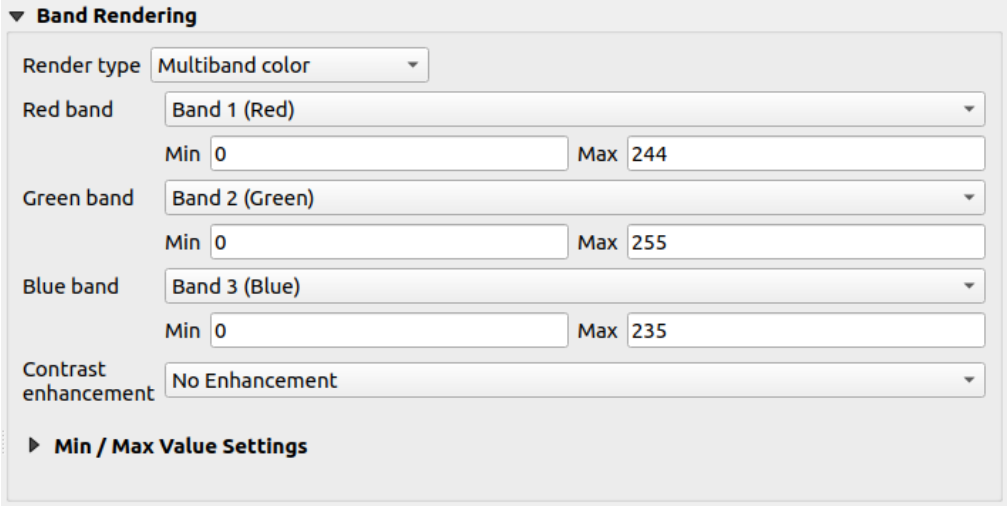
With the multiband color renderer, three selected bands from the image will be used as the red, green or blue component of the color image. QGIS automatically fetches *Min* and *Max* values for each band of the raster and scales the coloring accordingly. You can control the value ranges in the *Min/Max Value Settings* section.

A *Contrast enhancement* method can be applied to the values: 'No enhancement', 'Stretch to MinMax', 'Stretch and clip to MinMax' and 'Clip to min max'.

Белешка

Contrast enhancement

When adding GRASS rasters, the option *Contrast enhancement* will always be set automatically to *stretch to min max*, even if this is set to another value in the QGIS general options.



Слика13.2: Raster Symbology - Multiband color rendering

Савет

Viewing a Single Band of a Multiband Raster

If you want to view a single band of a multiband image (for example, Red), you might think you would set the Green and Blue bands to *Not Set*. But the preferred way of doing this is to set the image type to *Singleband gray*, and then select Red as the *Gray band* to use.

Paletted/Unique values

This is the standard render option for singleband files that include a color table, where a certain color is assigned to each pixel value. In that case, the palette is rendered automatically.

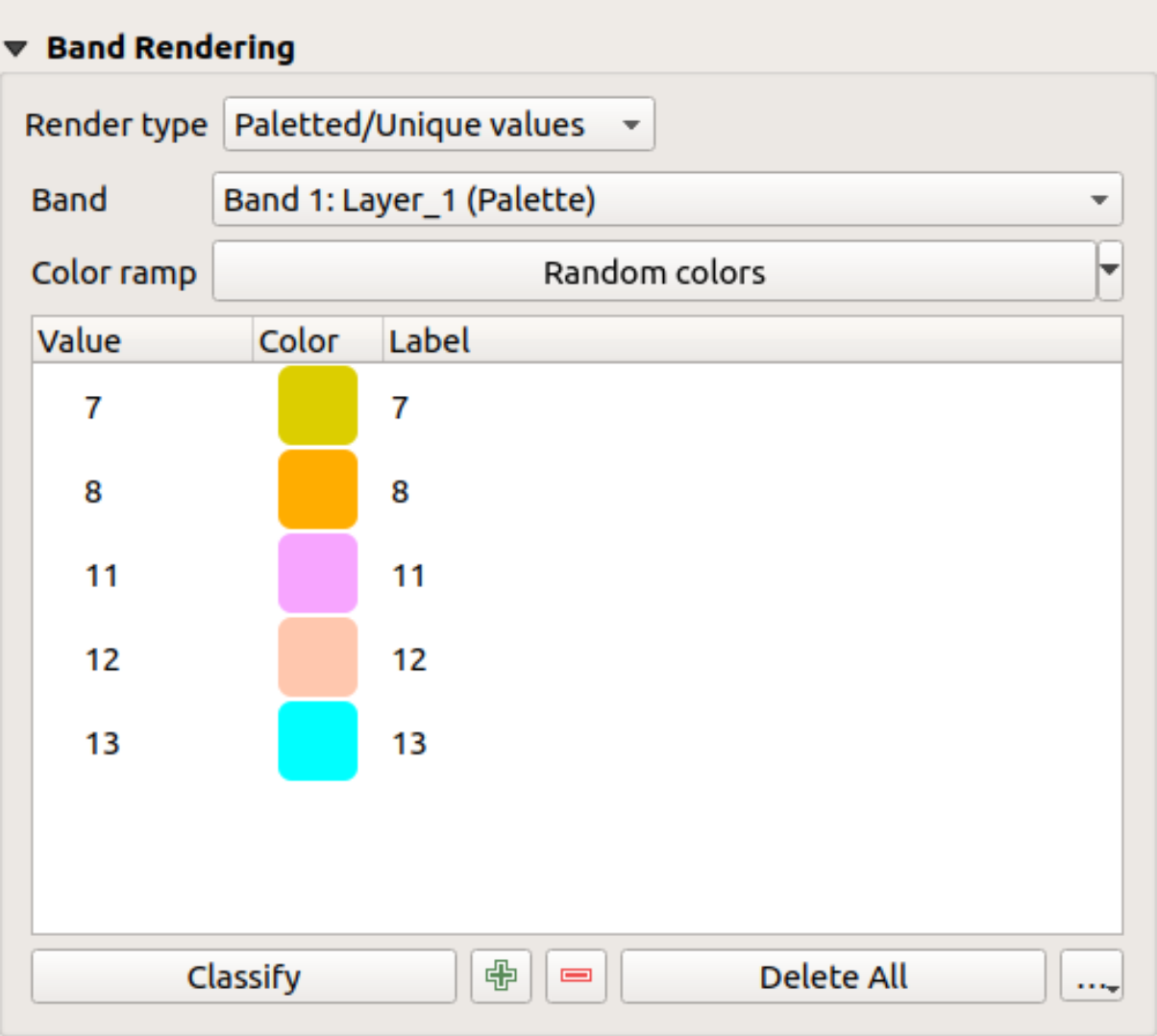
It can be used for all kinds of raster bands, assigning a color to each unique raster value.

If you want to change a color, just double-click on the color and the *Select color* dialog appears.

It is also possible to assign labels to the colors. The label will then appear in the legend of the raster layer.

Right-clicking over selected rows in the color table shows a contextual menu to:

- *Change Color…* for the selection
- *Change Opacity…* for the selection
- *Change Label…* for the selection



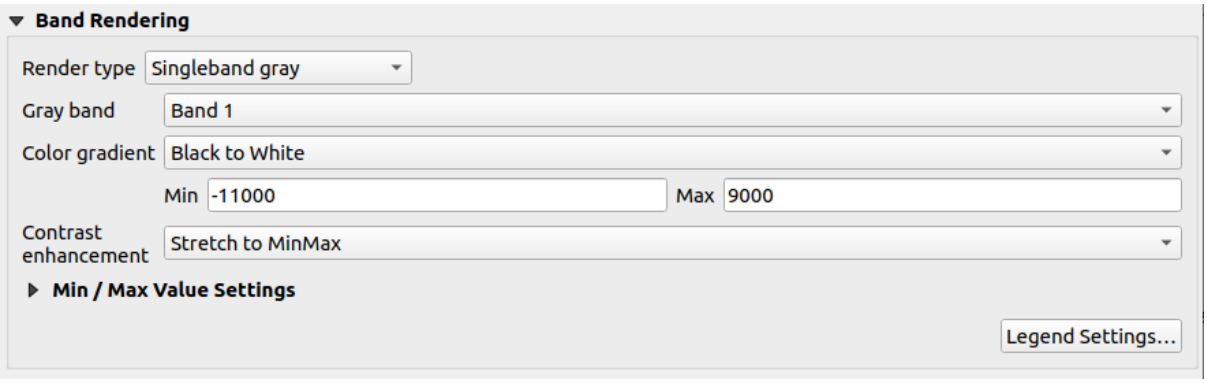
Слика13.3: Raster Symbology - Paletted unique value rendering

The pulldown menu, that opens when clicking the … (Advanced options) button below the color map to the right, offers color map loading (*Load Color Map from File…*) and exporting (*Export Color Map to File…*), and loading of classes (*Load Classes from Layer*).

Singleband gray

This renderer allows you to render a layer using only one band with a *Color gradient*: 'Black to white' or 'White to black'. You can change the range of values to color (*Min* and *Max*) in the *Min/Max Value Settings*.

A *Contrast enhancement* method can be applied to the values: 'No enhancement', 'Stretch to MinMax', 'Stretch and clip to MinMax' and 'Clip to min max'.

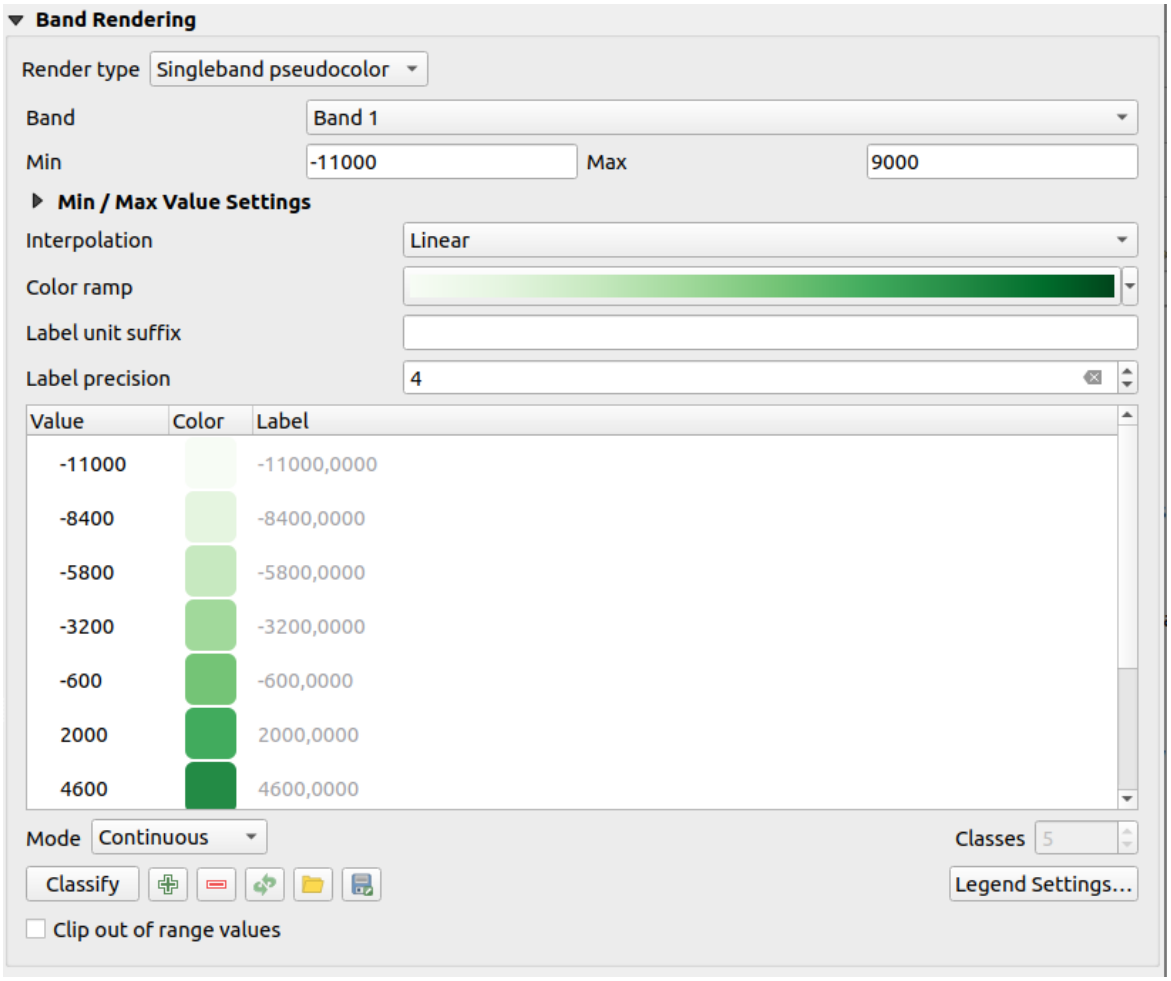


Слика13.4: Raster Symbology - Singleband gray rendering

Pixels are assigned a color based on the selected color gradient and the layer's legend (in the *Layers* panel and the layout *legend item*) is displayed using a continuous color ramp. Press *Legend settings...* if you wish to tweak the settings. More details at *Customize raster legend*.

Singleband pseudocolor

This is a render option for single-band files that include a continuous palette. You can also create color maps for a band of a multiband raster.



Слика13.5: Raster Symbology - Singleband pseudocolor rendering

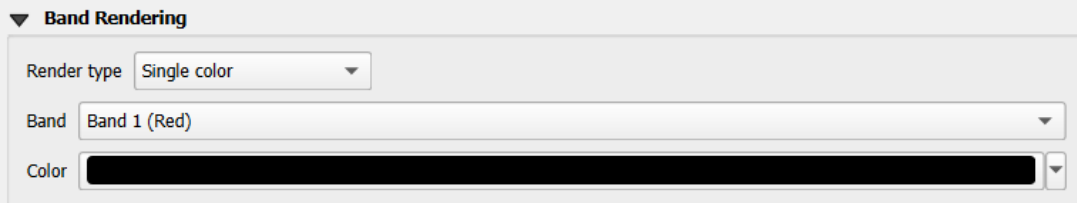
Using a *Band* of the layer and a *values range*, you can now interpolate and assign representation color to pixels within classes. More at *Color ramp shader classification*.

Pixels are assigned a color based on the selected color ramp and the layer's legend (in the *Layers* panel and the layout *legend item*) is displayed using a continuous color ramp. Press *Legend settings...* if you wish to tweak the settings or instead use a legend with separated classes (and colors). More details at *Customize raster legend*.

Single color

This renderer allows you to render a raster layer using *Single color*. This type of renderer is useful when you want to display a raster layer uniformly, without any variation in color based on pixel values.

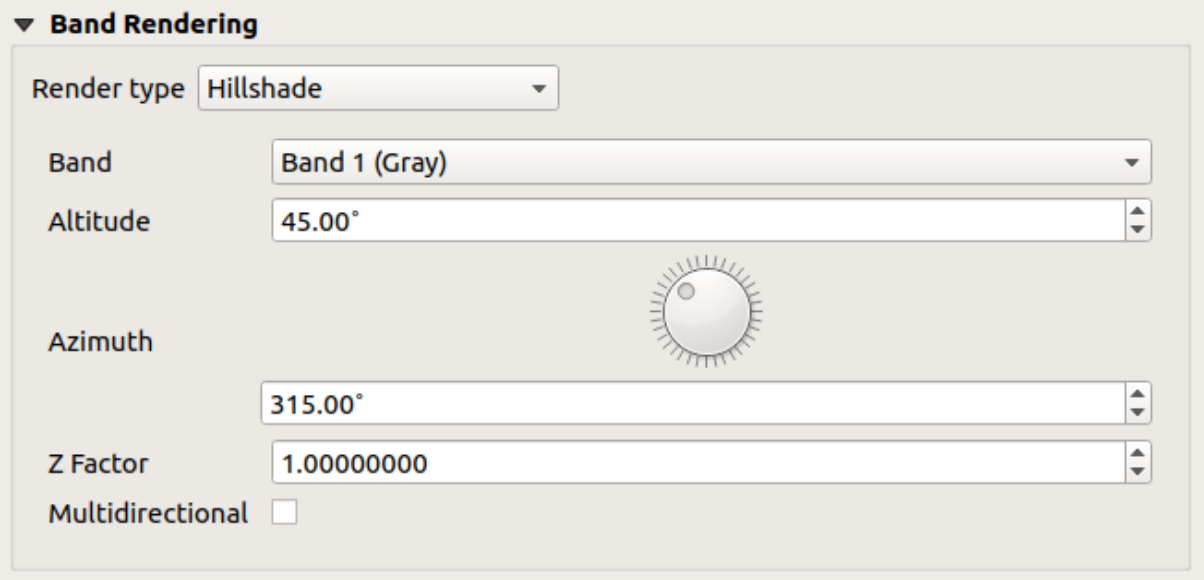
The single color renderer can be used with both single-band and multiband raster layers. When used with multiband rasters, you can select which band to apply the single color to, effectively displaying that specific band uniformly across the entire layer.



Слика13.6: Raster Symbology - Single color rendering

Osenčeni reljef

Render a band of the raster layer using hillshading.



Слика13.7: Raster Symbology - Hillshade rendering

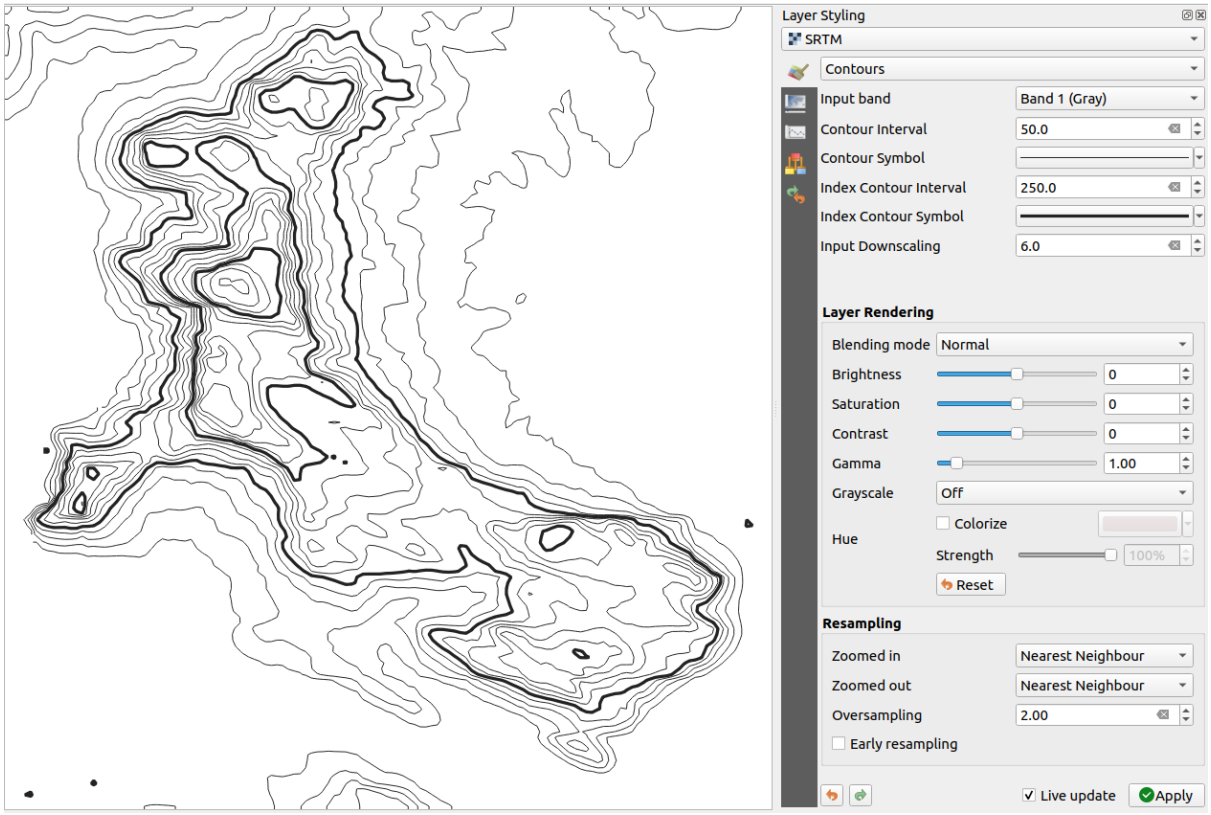
Opcije:

- *Band*: The raster band to use.
- *Altitude*: The elevation angle of the light source (default is `45°`).
- *Azimuth*: The azimuth of the light source (default is `315°`).
- *Z Factor*: Scaling factor for the values of the raster band (default is `1`).

- *Multidirectional*: Specify if multidirectional hillshading is to be used (default is `off`).

Contours

This renderer draws contour lines that are calculated on the fly from the source raster band.



Слика13.8: Raster Symbology - Contours rendering

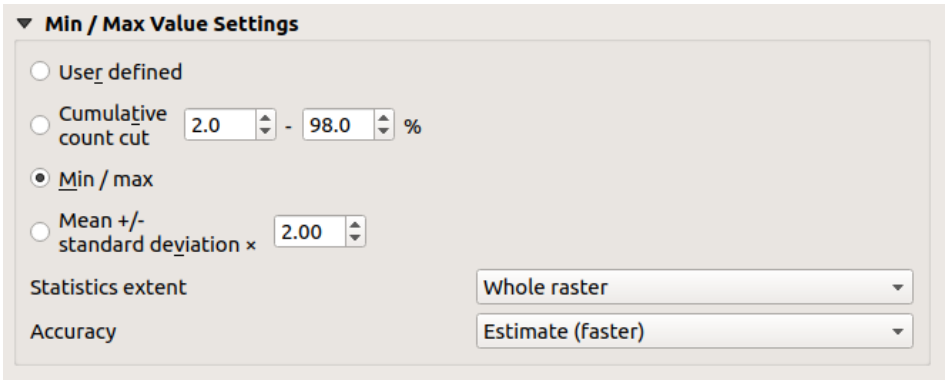
Opcije:

- *Input band*: the raster band to use.
- *Contour interval*: the distance between two consecutive contour lines
- *Contour symbol*: the *symbol* to apply to the common contour lines.
- *Index contour interval*: the distance between two consecutive **index contours**, that is the lines shown in a distinctive manner for ease of identification, being commonly printed more heavily than other contour lines and generally labeled with a value along its course.
- *Index contour symbol*: the symbol to apply to the index contour lines
- *Input downscaling*: Indicates by how much the renderer will scale down the request to the data provider (default is `4.0`).

  For example, if you generate contour lines on input raster block with the same size as the output raster block, the generated lines would contain too much detail. This detail can be reduced by the „downscale" factor, requesting lower resolution of the source raster. For a raster block 1000x500 with downscale 10, the renderer will request raster 100x50 from provider. Higher downscale makes contour lines more simplified (at the expense of losing some detail).

Setting the min and max values

By default, QGIS reports the *Min* and *Max* values of the band(s) of the raster. A few very low and/or high values can have a negative impact on the rendering of the raster. The *Min/Max Value Settings* frame helps you control the rendering.



Слика13.9: Raster Symbology - Min and Max Value Settings

Available options are:

- *User defined*: The default *Min* and *Max* values of the band(s) can be overridden
- *Cumulative count cut*: Removes outliers. The standard range of values is `2%` to `98%`, but it can be adapted manually.
- *Min / max*: Uses the whole range of values in the image band.
- *Mean +/- standard deviation x*: Creates a color table that only considers values within the standard deviation or within multiple standard deviations. This is useful when you have one or two cells with abnormally high values in a raster layer that impact the rendering of the raster negatively.

Calculations of the min and max values of the bands are made based on the:

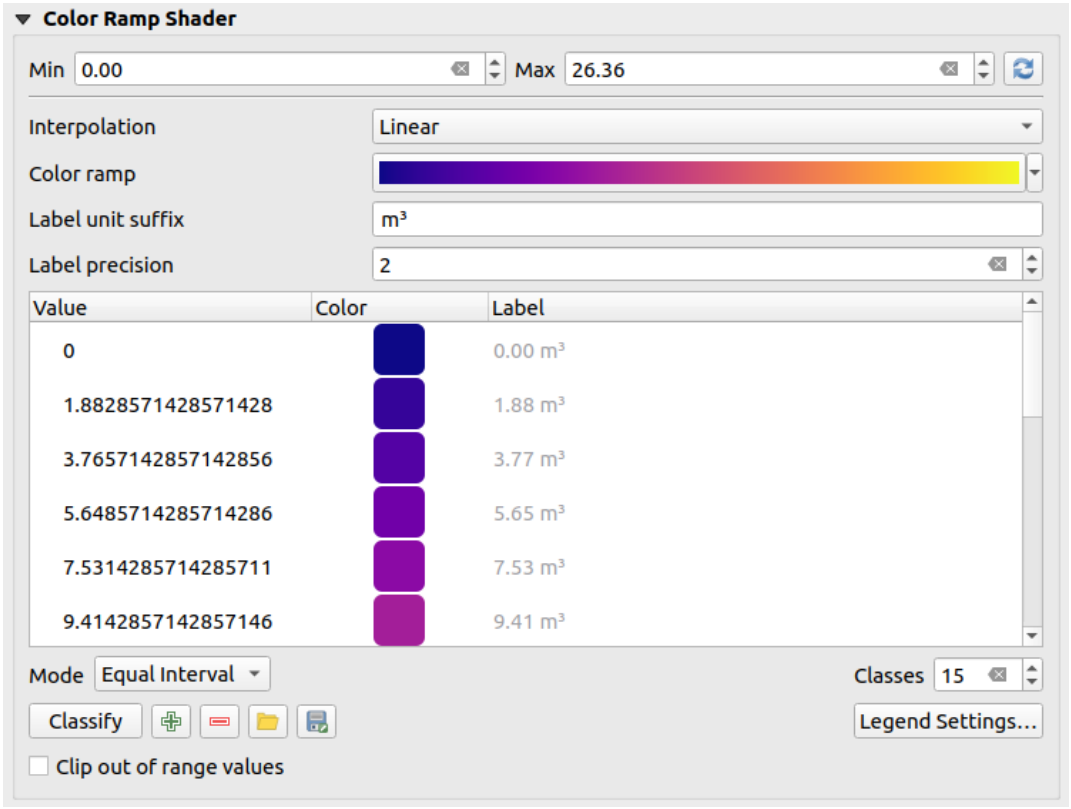
- *Statistics extent*: it can be *Whole raster*, *Current canvas* or *Updated canvas*. *Updated canvas* means that min/max values used for the rendering will change with the canvas extent (dynamic stretching).
- *Accuracy*, which can be either *Estimate (faster)* or *Actual (slower)*.

Белешка

For some settings, you may need to press the *Apply* button of the layer properties dialog in order to display the actual min and max values in the widgets.

Color ramp shader classification

This method can be used to classify and represent scalar dataset (raster or mesh contour) based on their values. Given a *color ramp* and a number of classes, it generates intermediate color map entries for class limits. Each color is mapped with a value interpolated from a range of values and according to a classification mode. The scalar dataset elements are then assigned their color based on their class.



Слика13.10: Classifying a dataset with a color ramp shader

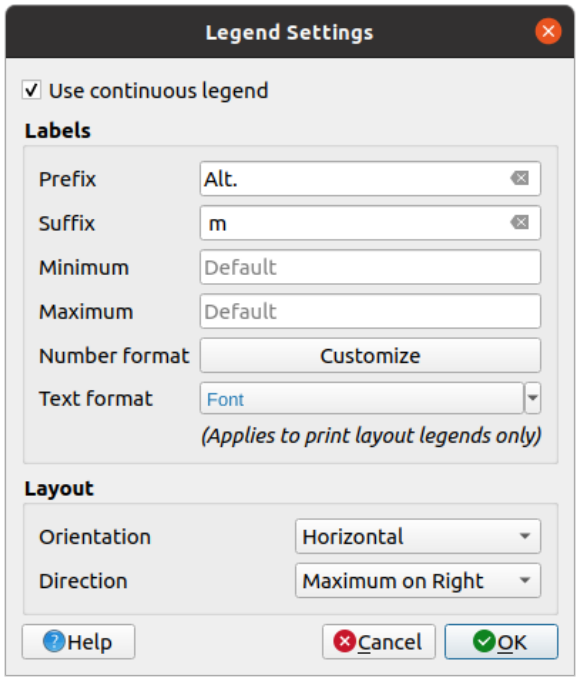
1. A *Min* and *Max* values must be defined and used to interpolate classes bounds. By default QGIS detects them from the dataset but they can be modified.
2. The *Interpolation* entry defines how scalar elements are assigned their color :
   - *Discrete* (a <= symbol appears in the header of the *Value* column): The color is taken from the closest color map entry with equal or higher value
   - *Linear*: The color is linearly interpolated from the color map entries above and below the pixel value, meaning that to each dataset value corresponds a unique color
   - *Exact* (a = symbol appears in the header of the *Value* column): Only pixels with value equal to a color map entry are applied a color; others are not rendered.
3. The *Color ramp* widget helps you select the color ramp to assign to the dataset. As usual with *this widget*, you can create a new one and edit or save the currently selected one. The name of the color ramp will be saved in the configuration.
4. The *Label unit suffix* adds a label after the value in the legend, and the *Label precision* controls the number of decimals to display.

5. The classification *Mode* helps you define how values are distributed across the classes:
   - *Equal interval*: Provided the *Number of classes*, limits values are defined so that the classes all have the same magnitude.
   - *Continuous*: Classes number and color are fetched from the color ramp stops; limits values are set following stops distribution in the color ramp.
   - *Quantile*: Provided the *Number of classes*, limits values are defined so that the classes have the same number of elements. Not available with *mesh layers*.
6. You can then *Classify* or tweak the classes:
   - The button  Add values manually adds a value to the table.
   - The button  Remove selected row deletes selected values from the table.
   - Double clicking in the *Value* column lets you modify the class value.
   - Double clicking in the *Color* column opens the dialog *Change color*, where you can select a color to apply for that value.
   - Double clicking in the *Label* column to modify the label of the class, but this value won't be displayed when you use the identify feature tool.
   - Right-clicking over selected rows in the color table shows a contextual menu to *Change Color…* and *Change Opacity…* for the selection.

   You can use the buttons  Load color map from file or  Export color map to file to load an existing color table or to save the color table for later use.
7. With linear *Interpolation*, you can also configure:
   -  *Clip out of range values*: By default, the linear method assigns the first class (respectively the last class) color to values in the dataset that are lower than the set *Min* (respectively greater than the set *Max*) value. Check this setting if you do not want to render those values.
   - *Legend settings*, for display in the *Layers* panel and the layout *legend item*. More details at *Customize raster legend*.

Customize raster legend

When applying a color ramp to a raster or a mesh layer, you may want to display a legend showing the classification. By default, QGIS displays a continuous color ramp with min and max values in the *Layers* panel and the layout *legend item*. This can be customized using the *Legend settings* button in the classification widget.



Слика13.11: Modifying a raster legend

In this dialog, you can set whether to *Use continuous legend*: if unchecked, the legend displays separated colors corresponding to the different classes applied. This option is not available for raster *singleband gray* symbology.

Checking the *Use continuous legend* allows you to configure both the labels and layout properties of the legend.

Labels

- Add a *Prefix* and a *Suffix* to the labels
- Modify the *Minimum* and a *Maximum* values to show in the legend
- *Customize* the *Number format*
- *Customize* the *Text format* to use in the print layout legend.

Layout

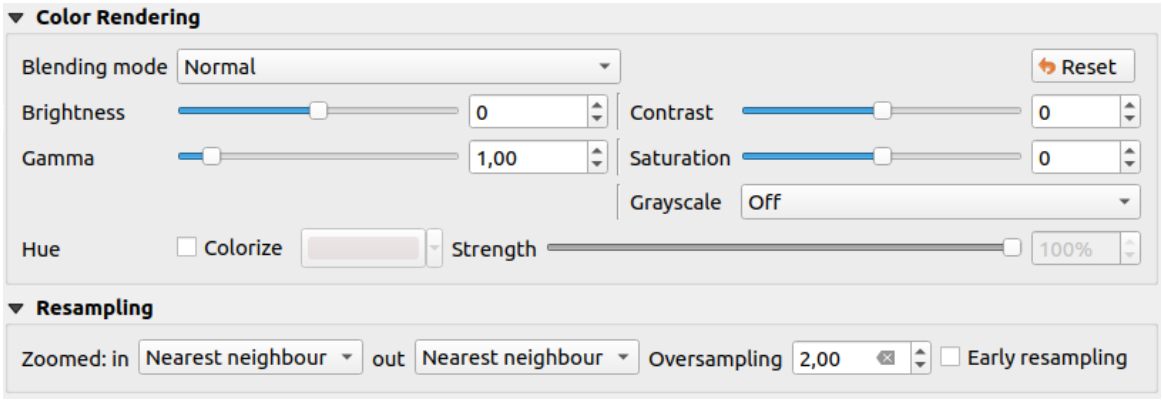
- Control the *Orientation* of the legend color ramp; it can be **Vertical** or **Horizontal**
- Control the *Direction* of the values depending on the orientation:
  - If vertical, you can display the **Maximum on top** or the **Minimum on top**
  - If horizontal, you can display the **Maximum on right** or the **Minimum on right**

Layer rendering

Over the symbology type applied to the layer band(s), you can achieve special rendering effects for the whole raster file(s):

- Use one of the blending modes (see *Blending Modes*)
- Set custom *Brightness*, *Saturation*, *Gamma* and *Contrast* to colors.
- With the *Invert colors*, the layer is rendered with opposite colors. Handy, for example, to switch out-of-the box OpenStreetMap tiles to dark mode.
- Turn the layer to *Grayscale* option either 'By lightness', 'By luminosity' or 'By average'.
- *Colorize* and adjust the *Strength* of *Hue* in the color table

Press *Reset* to remove any custom changes to the layer rendering.



Слика13.12: Raster Symbology - Layer rendering and Resampling settings

Resampling

The *Resampling* option has effect when you zoom in and out of an image. Resampling modes can optimize the appearance of the map. They calculate a new gray value matrix through a geometric transformation.

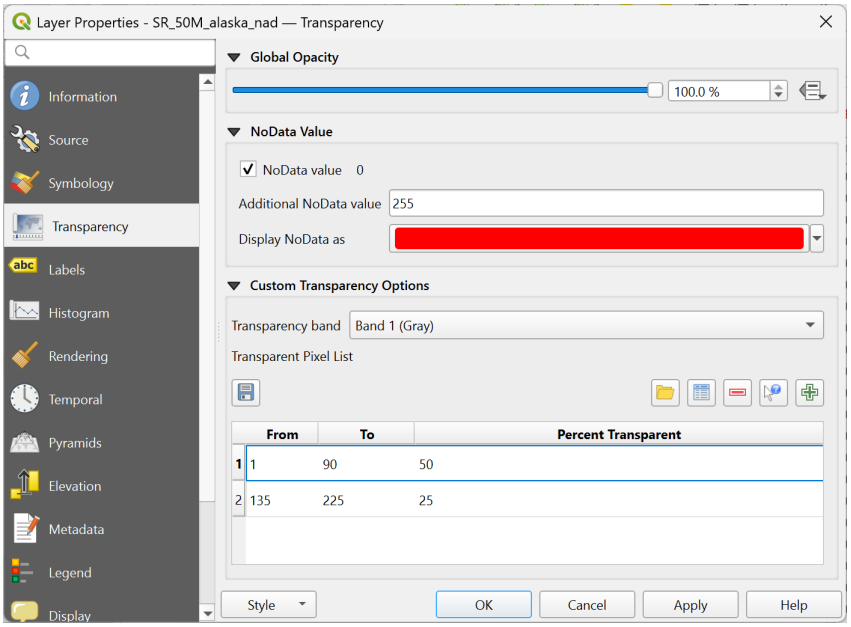
When applying the 'Nearest neighbour' method, the map can get a pixelated structure when zooming in. This appearance can be improved by using the 'Bilinear (2x2 kernel)' or 'Cubic (4x4 kernel)' method, which cause sharp edges to be blurred. The effect is a smoother image. This method can be applied to for instance digital topographic raster maps.

Early resampling: allows to calculate the raster rendering at the provider level where the resolution of the source is known, and ensures a better zoom in rendering with QGIS custom styling. Really convenient for tile rasters loaded using an *interpretation method*.

13.1.4 Transparency Properties

QGIS provides capabilities to set the *Transparency* level of a raster layer.

Use the *Global opacity* slider to set to what extent the underlying layers (if any) should be visible through the current raster layer. This is very useful if you overlay raster layers (e.g., a shaded relief map overlaid by a classified raster map). This will make the look of the map more three dimensional. The opacity of the raster can be data-defined, and vary e.g. depending on the visibility of another layer, by temporal variables, on different pages of an atlas, …



Слика13.13: Raster Transparency

With *No data value* QGIS reports the original source no data value (if defined) which you can consider as is in the rendering. Additionally, you can enter a raster value that should be treated as an *Additional no data value*. The *Display no data as* color selector allows you to apply a custom color to no data pixels, instead of the default transparent rendering.

An even more flexible way to customize the transparency is available in the *Custom transparency options* section:

- Use *Transparency band* to apply transparency for an entire band.
- Provide a list of pixels to make transparent with corresponding levels of transparency:

  1. Click the Add values manually button. A new row will appear in the pixel list.
  2. For single-band based symbology (e.g. DEMs), enter the **From** and **To** values and adjust the **Percent Transparent** to apply.
  3. For multiband based symbology (e.g. RGB images) enter the **Red**, **Green** and **Blue** values of the pixel and adjust the **Percent Transparent** to apply. QGIS supports **Tolerance** for pixel values, when defining transparency. This means that pixels with color close to the specified RGB values can also be made transparent. Note that this feature applies only to multiband rasters.
  4. Alternatively, you can fetch the pixel values directly from the raster using the Add values from display button. Then enter the transparency value.
  5. Repeat the steps to adjust more values with custom transparency.
  6. Press the *Apply* button and have a look at the map.

  As you can see, it is quite easy to set custom transparency, but it can be quite a lot of work. Therefore, you can use the button Export to file to save your transparency list to a file. The button Import from file loads your transparency settings and applies them to the current raster layer.

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.

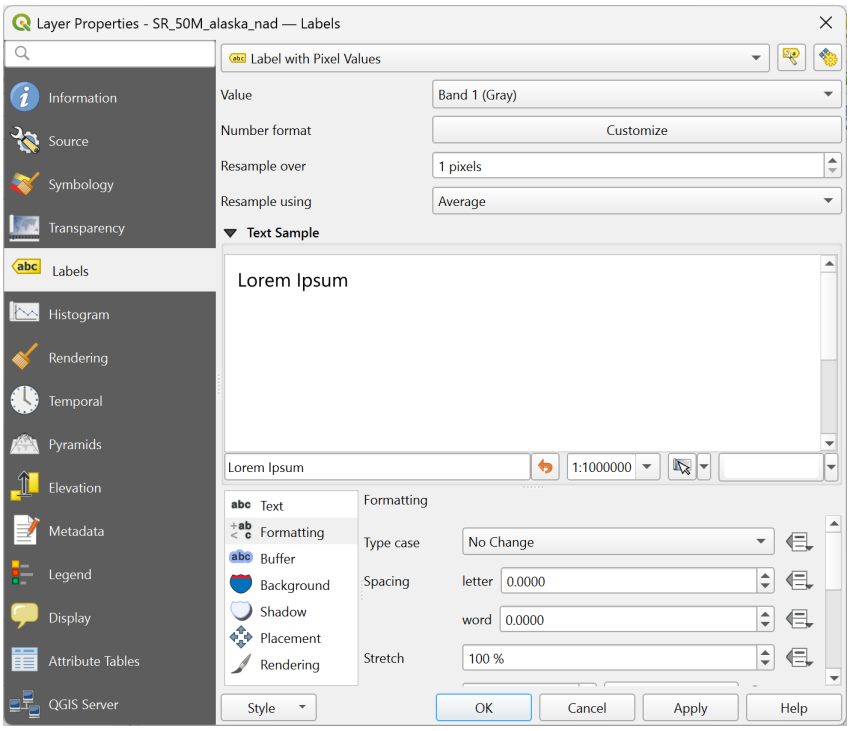
At the top of the dialog, you have:

- 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 *Configuring project labeling rules*.
- the Automated placement settings (applies to all layers) button: configure general properties on label placement and conflicts resolution. More details at *Setting the automated placement engine*.

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.

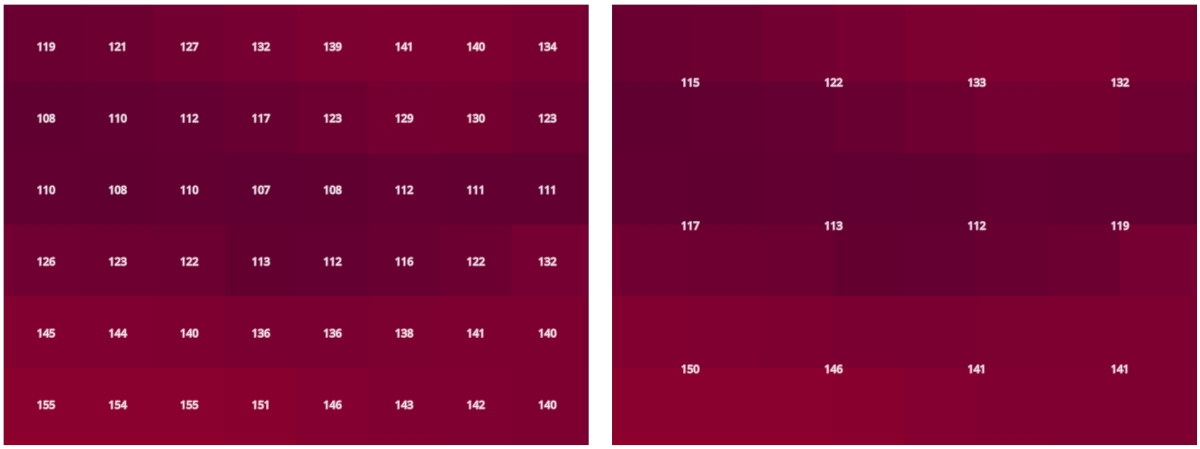
Using the *Label with pixel values* option, the following dialog opens.



Слика13.14: 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.



Слика13.15: 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*

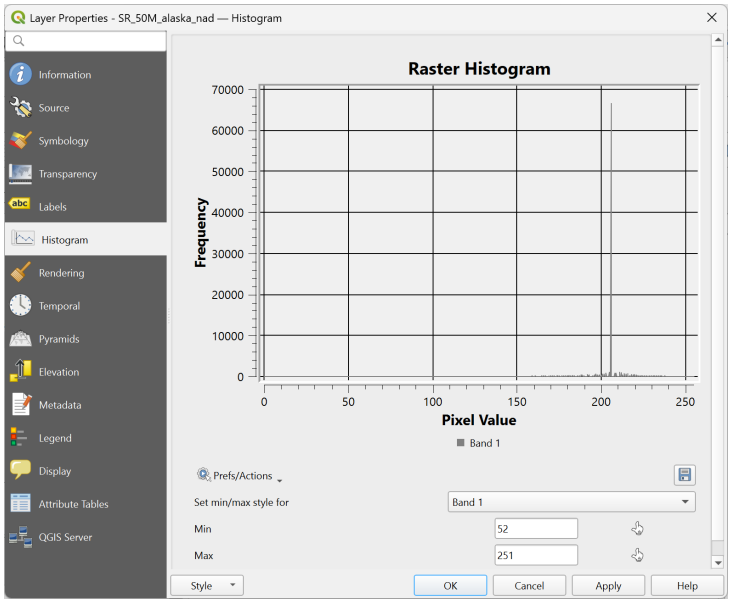
Description of how to set each property is exposed at *Setting a label*.

13.1.6 Histogram Properties

The *Histogram* tab allows you to view the distribution of the values in your raster. The histogram is generated when you press the *Compute Histogram* button. All existing bands will be displayed together. You can save the histogram as an image with the button.

At the bottom of the histogram, you can select a raster band in the drop-down menu and *Set min/max style for* it. The *Prefs/Actions* drop-down menu gives you advanced options to customize the histogram:

- With the *Visibility* option, you can display histograms for individual bands. You will need to select the option *Show selected band*.
- The *Min/max options* allow you to 'Always show min/max markers', to 'Zoom to min/max' and to 'Update style to min/max'.
- The *Actions* option allows you to 'Reset' or 'Recompute histogram' after you have changed the min or max values of the band(s).



Слика13.16: Raster Histogram

13.1.7 Rendering Properties

In the *Rendering* tab, it's possible to:

- set *Scale dependent visibility* for the layer: You can set the *Maximum (inclusive)* and *Minimum (exclusive)* scales, defining a range of scales in which the layer will be visible. It will be hidden outside this range. The Set to current canvas scale button helps you use the current map canvas scale as a boundary. See *Visibility Scale Selector* for more information.

  Белешка

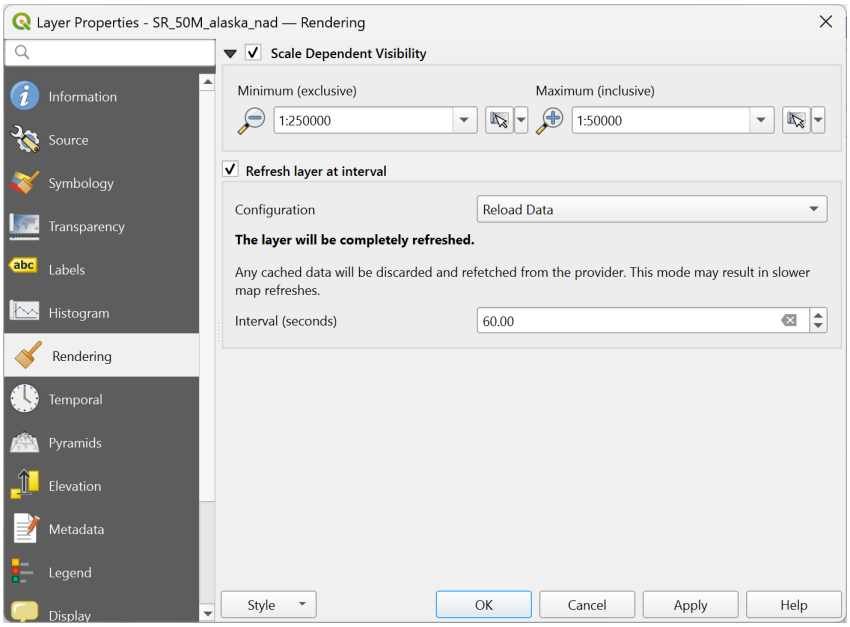
  You can also activate scale dependent visibility on a layer from within the *Layers* panel: right-click on the layer and in the contextual menu, select *Set Layer Scale Visibility*.

- *Refresh layer at interval*: controls whether and how regular a layer can be refreshed. Available *Configuration* options are:

  – *Reload data*: the layer will be completely refreshed. Any cached data will be discarded and refetched from the provider. This mode may result in slower map refreshes.

  – *Redraw layer only*: this mode is useful for animation or when the layer's style will be updated at regular intervals. Canvas updates are deferred in order to avoid refreshing multiple times if more than one layer has an auto update interval set.

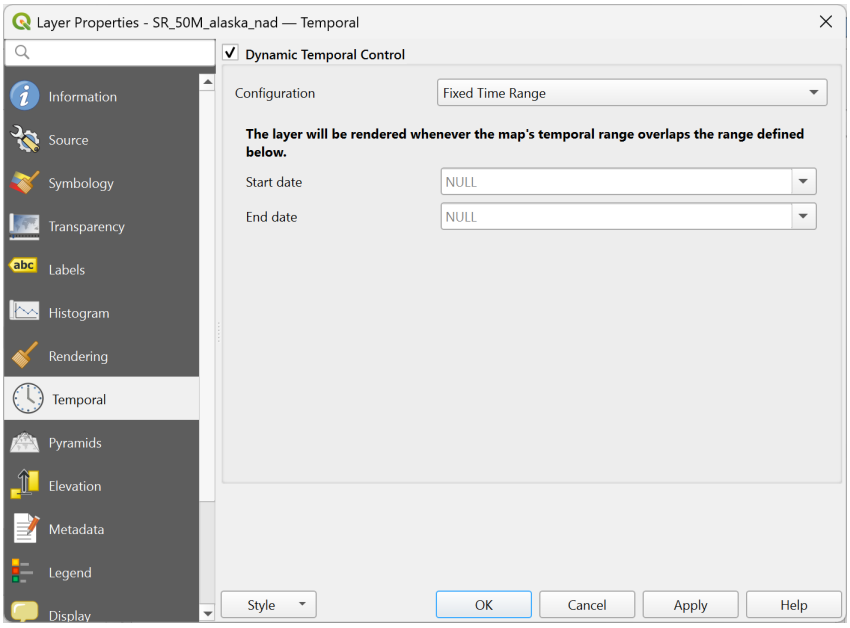
  It is also possible to set the *Interval (seconds)* between consecutive refreshments.



Слика13.17: Raster Rendering Properties

13.1.8 Temporal Properties

The *Temporal* tab provides options to control the rendering of the layer over time. Such dynamic rendering requires the *temporal navigation* to be enabled over the map canvas.



Слика13.18: Raster Temporal Properties

Check the *Dynamic Temporal Control* option and set whether the layer redraw should be:

- *Automatic*: the rendering is controlled by the underlying data provider if it supports 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.
- *Fixed Time Range Per Band*: only shows a band when the current animation time is between its *Begin* and *End* date range. This option allows you to either manually set these time ranges for each band or use the button to automatically generate datetime values, enabling detailed temporal analysis and visualization. This mode is particularly useful for working with raster layers where each band corresponds to a specific time period, such as NetCDF files.
- *Represents Temporal Values*: interprets each pixel in the raster layer as a datetime value. When this temporal mode is active, pixels that do not fall within the temporal range specified in the render context will be hidden, ensuring that only temporally relevant data is displayed. This mode is effective for:
  - Analyzing land use changes, like observing deforestation patterns.
  - Studying flooding by comparing water coverage across different times.
  - Evaluating movement costs in terrain analysis, for example, using GRASS GIS's r.walk tool to calculate travel costs across a landscape.
- *Redraw layer only*: the layer is redrawn at each new animation frame. It's useful when the layer uses time-based expression values for renderer settings (e.g. data-defined renderer opacity, to fade in/out a raster layer).

13.1.9 Pyramids Properties

High resolution raster layers can slow navigation in QGIS. By creating lower resolution copies of the data (pyramids), performance can be considerably improved, as QGIS selects the most suitable resolution to use depending on the zoom level.

You must have write access in the directory where the original data is stored to build pyramids.

From the *Resolutions* list, select resolutions at which you want to create pyramid levels by clicking on them.

If you choose **Internal (if possible)** from the *Overview format* drop-down menu, QGIS tries to build pyramids internally.

Белешка

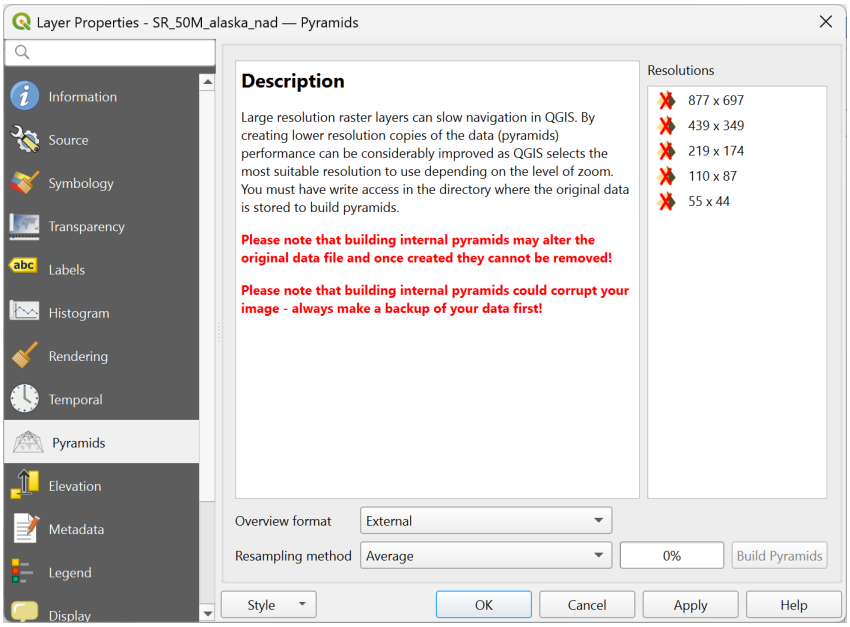
Please note that building pyramids may alter the original data file, and once created they cannot be removed. If you wish to preserve a 'non-pyramided' version of your raster, make a backup copy prior to pyramid building.

If you choose **External** and **External (Erdas Imagine)** the pyramids will be created in a file next to the original raster with the same name and a `.ovr` extension.

Several *Resampling methods* can be used for pyramid calculation:

- Nearest Neighbour
- Average
- Gauss
- Cubic
- Cubic Spline
- Laczos
- Mode
- None

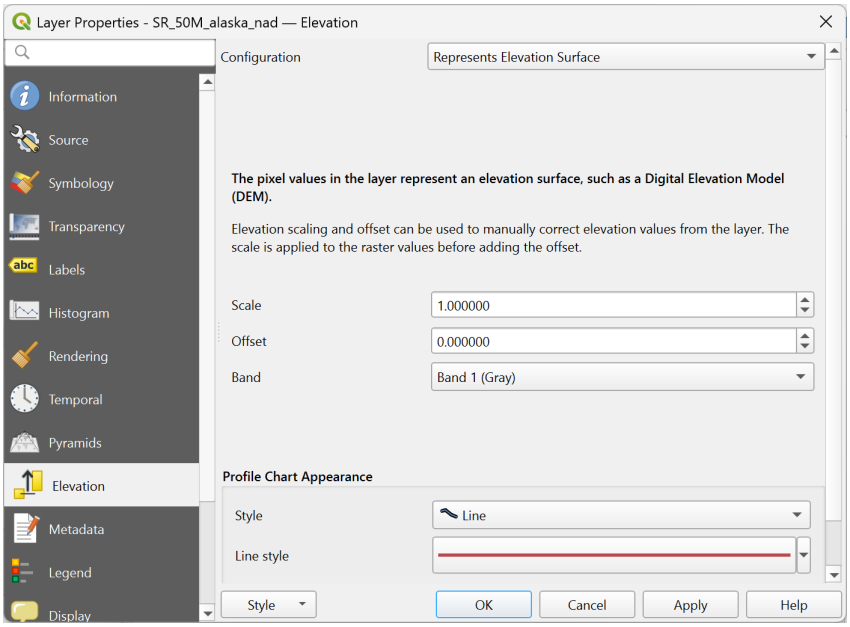
Finally, click *Build Pyramids* to start the process.



Слика13.19: Raster Pyramids

13.1.10 Elevation Properties

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:



Слика13.20: Raster Elevation Properties

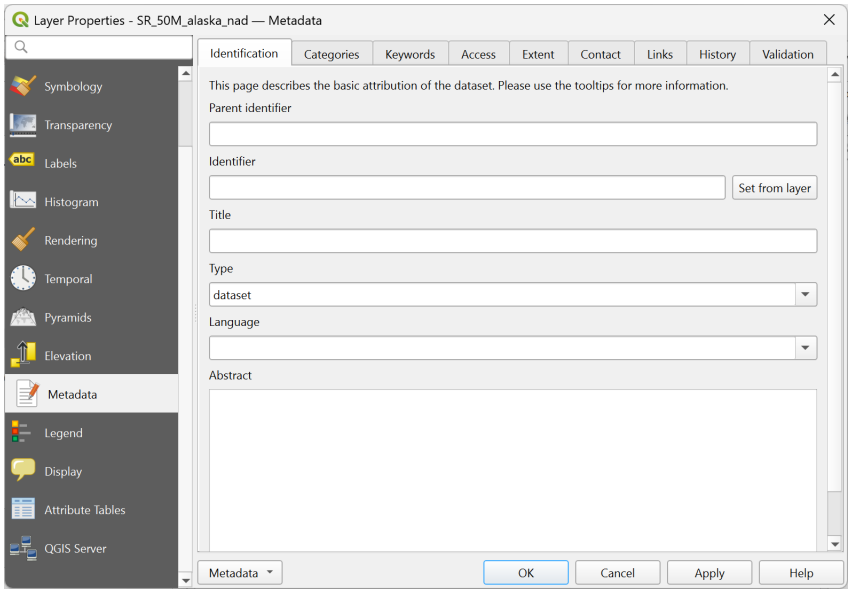
- *Represents Elevation Surface*: whether the raster layer represents a height surface (e.g DEM) and the pixel values should be interpreted as elevations. Choose this option if you want to display a raster in an *elevation profile view*. You will also need to fill in the *Band* to pick values from and 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.
- *Profile Chart Appearance*: controls the rendering of the raster elevation data in the profile chart. The profile *Style* can be set as:
  - a *Line* with a specific *Line style*
  - an elevation surface rendered using a fill symbol either above (*Fill above*) or below (*Fill below*) the elevation curve line. The surface symbology is represented using:
    - a *Fill style*
    - and a *Limit*: the maximum (respectively minimum) altitude determining how high the fill surface will be

13.1.11 Metadata Properties

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



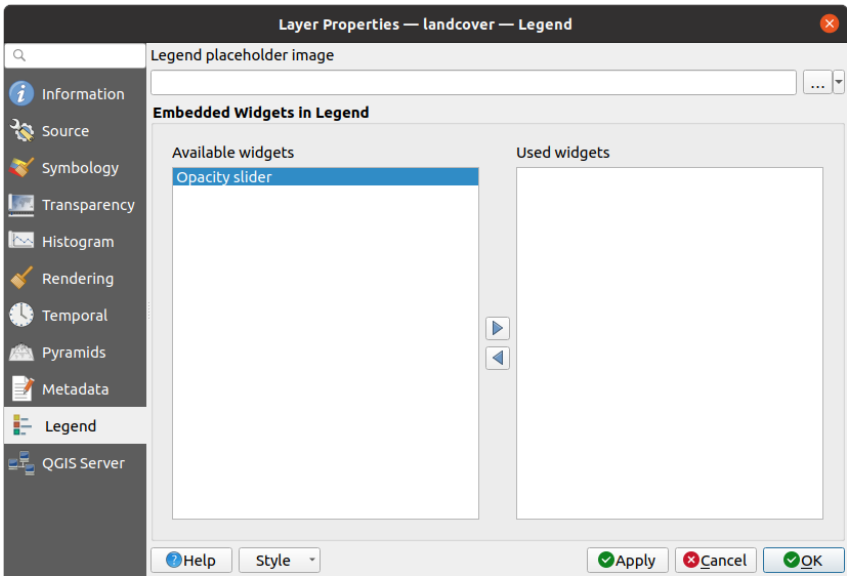
Слика13.21: Raster Metadata

13.1.12 Legend Properties

The *Legend* tab provides you with advanced settings for the *Layers panel* and/or the *print layout legend*. These options include:

- Depending on the symbology applied to the layer, you may end up with several entries in the legend, not necessarily readable/useful to display. The *Legend placeholder image* helps you *select an image* for replacement, displayed both in the *Layers* panel and the print layout legend.
- The *Embedded widgets in Legend* provides you with a list of widgets you can embed within the layer tree in the Layers panel. The idea is to have a way to quickly access some actions that are often used with the layer (setup transparency, filtering, selection, style or other stuff…).

  By default, QGIS provides a transparency widget but this can be extended by plugins that register their own widgets and assign custom actions to layers they manage.



Слика13.22: Raster Legend

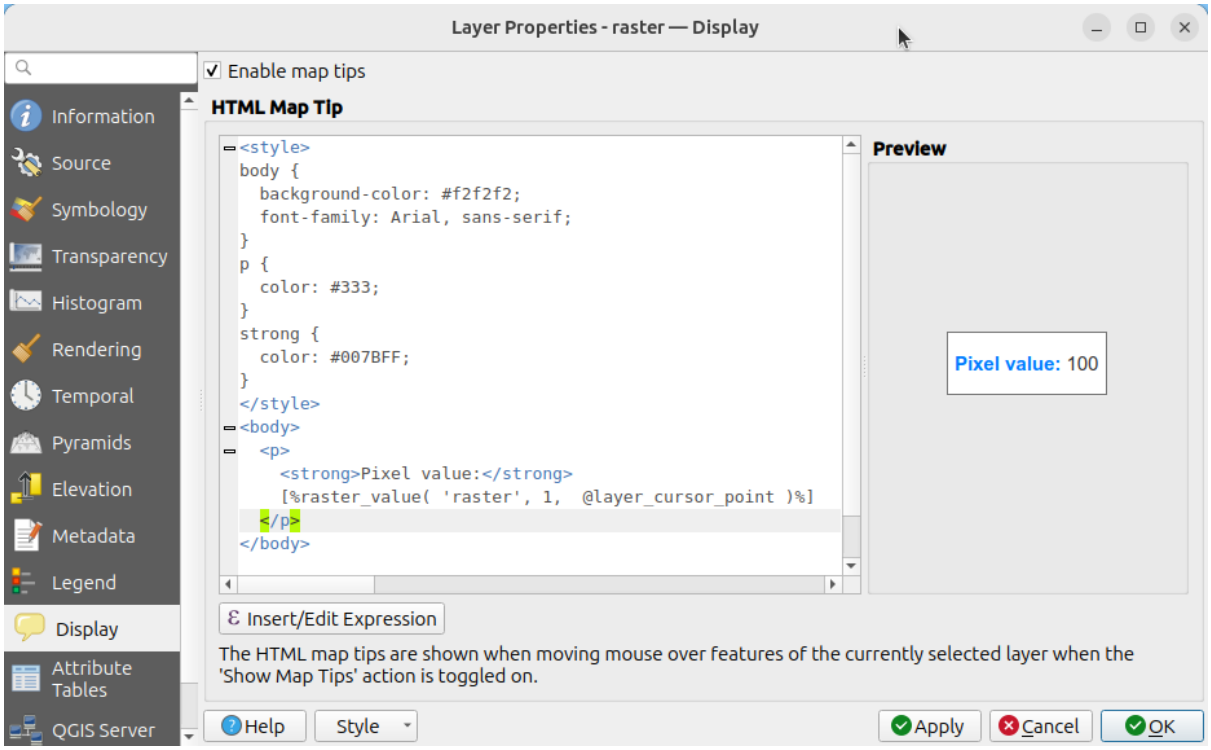
13.1.13 Display Properties

The *Display* tab helps you configure HTML map tips to use for pixels identification:

- *Enable Map Tips* controls whether to display map tips for the layer
- 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.

> **Белешка**
>
> **Understanding the *Insert/Edit Expression* button behavior**
>
> If you select some text within an expression (between „[%” and „%]”), or if no text is selected but the cursor is inside an expression, the whole expression will be automatically selected for editing. If the cursor or a selected text is outside an expression, the dialog opens with the selection.



Слика13.23: Map tips with raster layer

To display map tips:

1. Select the menu option *View* ► *Show Map Tips* or click on the Show Map Tips icon of the *Attributes Toolbar*.
2. Make sure that the layer you target is active and has the *Enable Map Tips* property checked.
3. Move over a pixel, and the corresponding information will be displayed over.

Map tip is a cross-layer feature meaning that once activated, it stays on and applies to any map tip enabled layer in the project until it is toggled off.

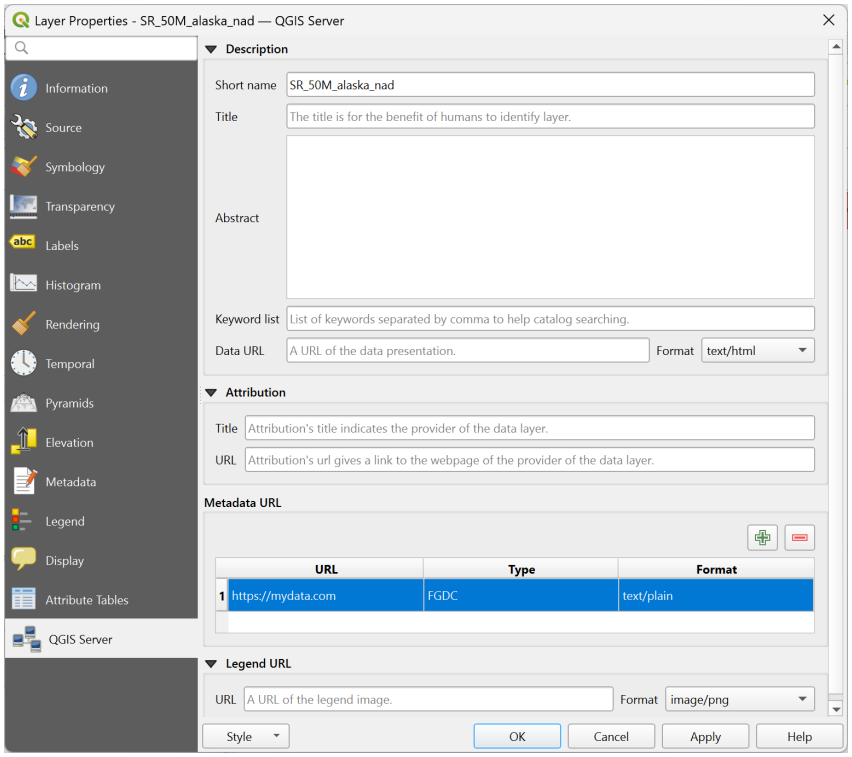
13.1.14 QGIS Server Properties

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*

Белешка

When the raster layer you want to publish is already provided by a web service, further *properties* are available for setting.



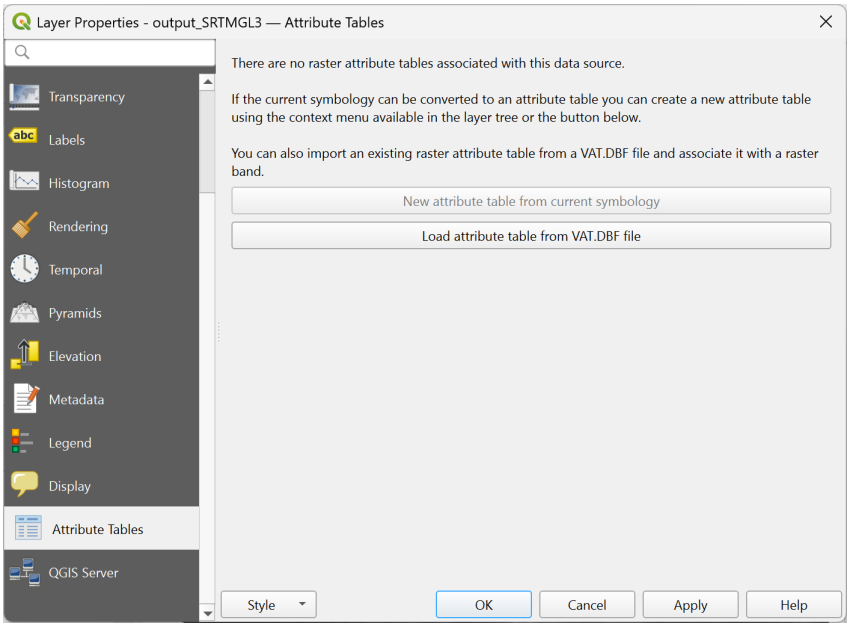
Слика13.24: QGIS Server in Raster Properties

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.

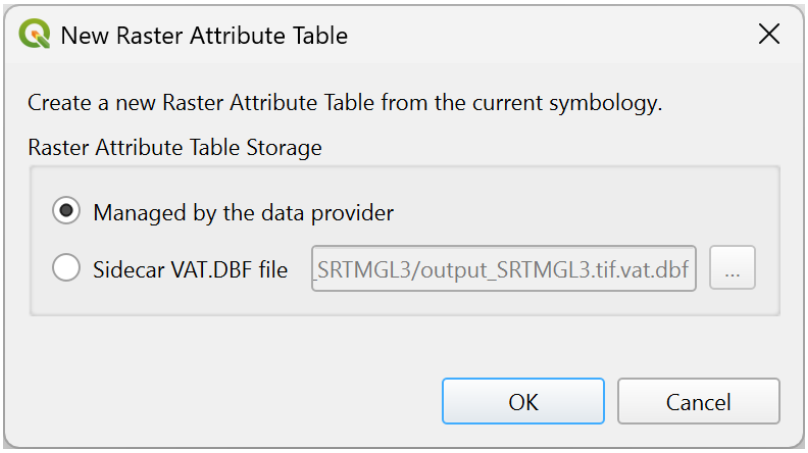
The *Attribute Tables* tab allows you to view and edit raster attribute data. To access the attribute table of a raster layer, right-click the layer in the layers panel, open *Properties…* and select *Attribute Tables*.



Слика13.25: 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:

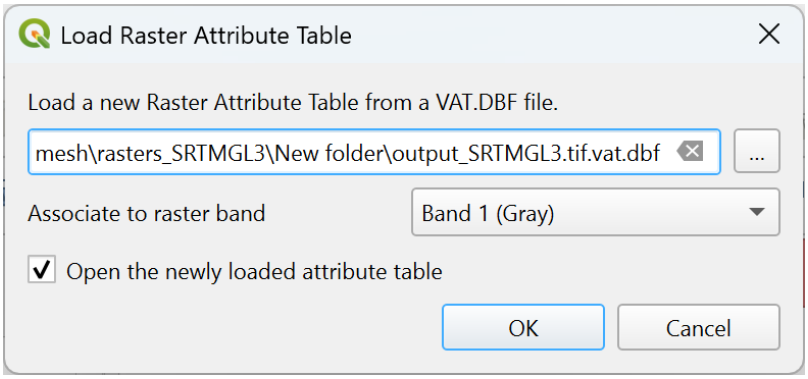
- *New attribute table from current symbology*: will appear if your raster layer has a classified symbology applied. Otherwise, this option is disabled. Applying classified symbology also triggers an option in the layer context menu to  *Create Raster Attribute Table*. Click this button and set the following 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.



Слика13.26: 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.



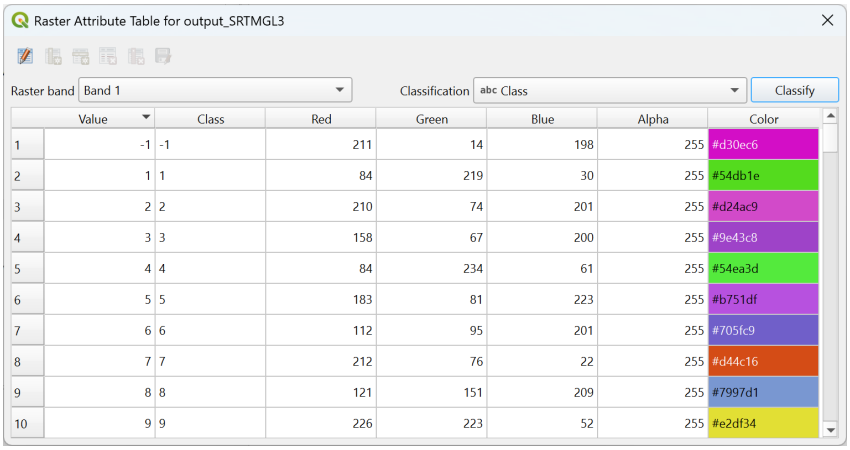
Слика13.27: Load attribute table from VAT.DBF file

Editing raster attribute tables

When set, a raster attribute table can be accessed from the *Attribute tables* tab of the layer properties dialog or by selecting *Open raster attribute table* from the layer contextual menu in *Layers* panel. This dialog shows a table whose rows correspond to the different pixel values or value ranges classified in a given band of the raster layer. Depending on the type of classification, a row is described by:

- for an exact values classification (e.g., paletted/unique values): the band name and the applied *Color*. Additional fields of the *Red*, *Green*, *Blue* and *Opacity* of the color are displayed.
- for range values classification (e.g., singleband pseudocolor): the range of values defined by their *Min*, *Max*, the associated label (*Class*) and color ramp interpolated over the values (*Color*). Additional fields of the minimum and maximum of *Red*, *Green*, *Blue* and *Opacity* of the color ramp are displayed.

You can also customize the raster attribute table by adding new fields for classification and group different value ranges accordingly, modifying value ranges and their associated colors or labels, adding new rows, … The new attribute table can be used to update the current layer symbology.



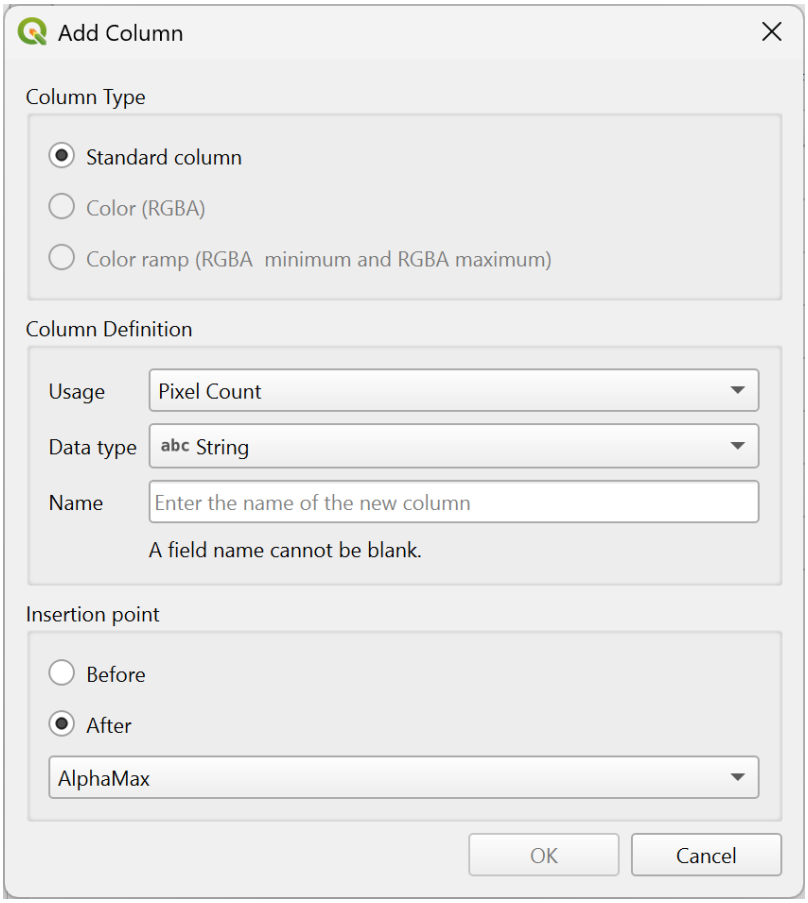
	Value	Class	Red	Green	Blue	Alpha	Color
1	-1	-1	211	14	198	255	#d30ec6
2	1	1	84	219	30	255	#54db1e
3	2	2	210	74	201	255	#d24ac9
4	3	3	158	67	200	255	#9e43c8
5	4	4	84	234	61	255	#54ea3d
6	5	5	183	81	223	255	#b751df
7	6	6	112	95	201	255	#705fc9
8	7	7	212	76	22	255	#d44c16
9	8	8	121	151	209	255	#7997d1
10	9	9	226	223	52	255	#e2df34

Слика13.28: 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.



Слика13.29: 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 Identify raster cells

The *identify features* tool allows you to get information about specific points in a raster layer.

To use the *Identify features* tool:

1. Select the raster layer in the Layers panel.
2. Kliknite na alatku *Identifikuj objekte* u paleti alatki ili pritisnite `Ctrl+Shift+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:

- a `Table` view - organizes the information about the identified features and their values in a table.
- a `Graph` view - organizes the information about the identified features and their values in a graph.

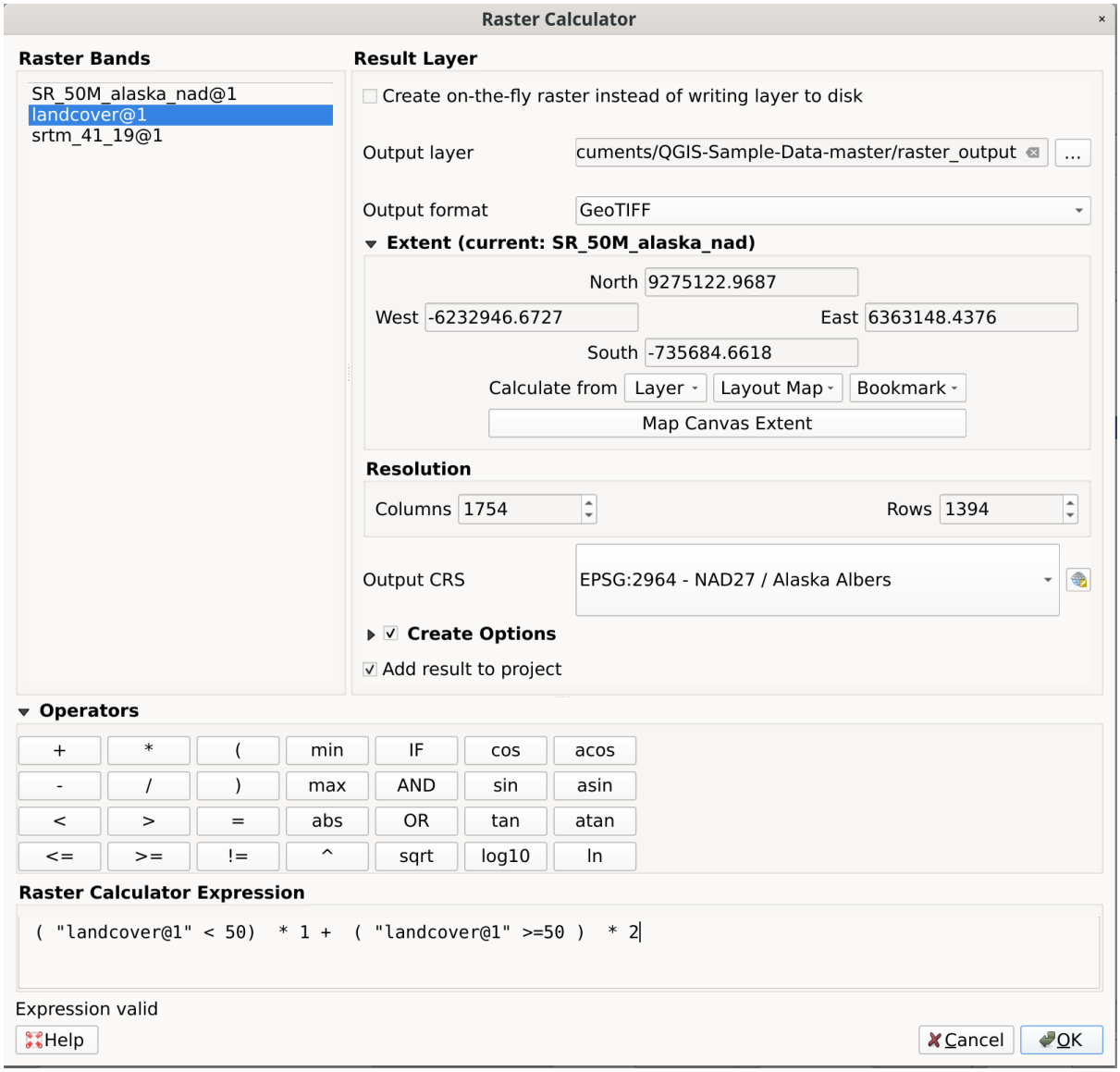
Under the pixel attributes, you will find the *Derived* information, such as:

- `X` and `Y` coordinate values of the point clicked
- Column and row of the point clicked (pixel) when compatible

13.2 Raster Analysis

13.2.1 Raster Calculator

The *Raster Calculator* in the *Raster* menu allows you to perform calculations on the basis of existing raster pixel values (see Сл.13.30). The results are written to a new raster layer in a GDAL-supported format.



Слика13.30: Raster Calculator

The *Raster bands* list contains all loaded raster layers that can be used. To add a raster to the raster calculator expression field, double click its name in the Fields list. You can then use the operators to construct calculation expressions, or you can just type them into the box.

In the *Result layer* section, you will need to define an output layer. You can:

- *Create on-the-fly raster instead of writing layer to disk*:
  - If unchecked, the output is stored on the disk as a new plain file. An *Output layer* path and an *Output format* are required.
  - If checked, a virtual raster layer, i.e. a raster layer defined by its URI and whose pixels are calculated on-the-fly, is created. 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.
- Define the *Spatial extent* of the calculation using the *extent selector* widget.
- Set the *Resolution* of the layer using columns and rows number. If the input layer has a different resolution,

the values will be resampled with the nearest neighbor algorithm.

- 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*.
- 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 for virtual rasters.

The *Operators* section contains all available operators. To add an operator to the raster calculator expression box, click the appropriate button. Mathematical calculations (`+`, `-`, `*`, …) and trigonometric functions (`sin`, `cos`, `tan`, …) are available. Conditional expressions (`=`, `!=`, `<`, `>=`, …) return either 0 for false or 1 for true, and therefore can be used with other operators and functions.

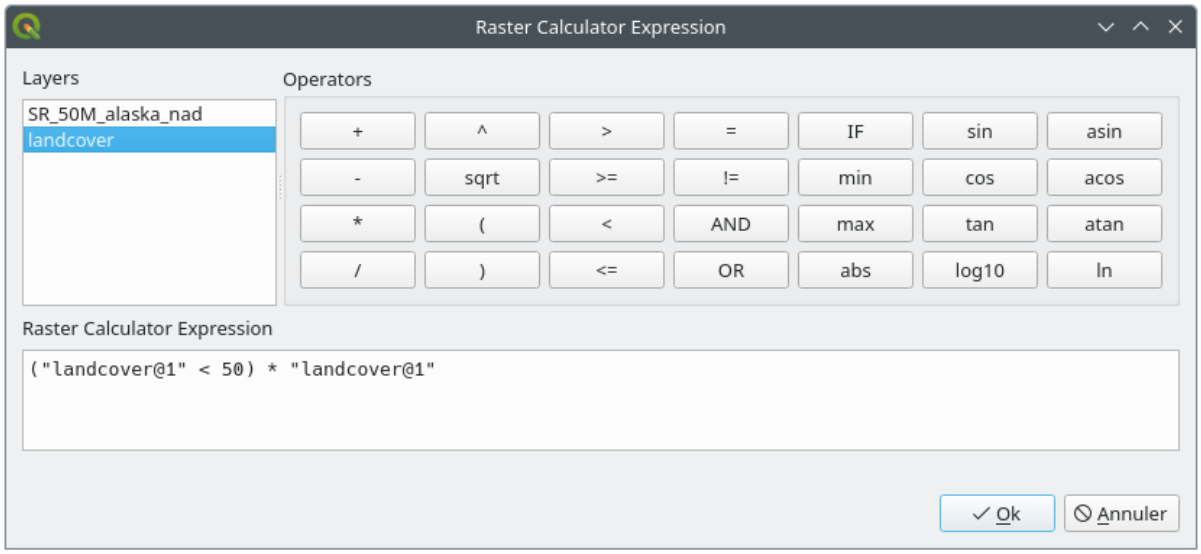
See also

Rasterski kalkulator and *Rasterski kalkulator (virtuelni)* algorithms

Raster calculator expression

The dialog

The *Raster calculator expression* dialog provides means to write expressions for pixels calculations between a set of raster layers.



Слика13.31: Raster Expression Calculator

- *Layers*: Shows the list of all raster layers loaded in the legend. These can be used to fill the expression box (double click to add). Raster layers are referred by their name and the number of the band: `layer_name@band_number`. For instance, the first band from a layer named `DEM` will be referred as `DEM@1`.
- *Operators*: contains a number of calculation operators for pixels manipulation:
  - Arithmetical: `+`, `-`, `*`, `sqrt`, `abs`, `ln`, …
  - Trigonometric: `sin`, `cos`, `tan`, …
  - Comparison: `=`, `!=`, `<`, `>=`, …
  - Logical: `IF`, `AND`, `OR`, `(`, `)`
  - Statistical: `min`, `max`

To add an operator to the raster calculator expression box, click the appropriate button.

- *Raster calculator expression* is the area in which the expression is composed

Primeri

Convert elevation values from meters to feet

Creating an elevation raster in feet from a raster in meters, you need to use the conversion factor for meters to feet: 3.28. The expression is:

```
"elevation@1" * 3.28
```

Using a mask

If you want to mask out parts of a raster – say, for instance, because you are only interested in elevations above 0 meters – you can use the following expression to create a mask and apply the result to a raster in one step.

```
("elevation@1" >= 0) * "elevation@1"
```

In other words, for every cell greater than or equal to 0 the conditional expression evaluates to 1, which keeps the original value by multiplying it by 1. Otherwise the conditional expression evaluates to 0, which sets the raster value to 0. This creates the mask on the fly.

Classify a Raster

If you want to classify a raster – say, for instance into two elevation classes, you can use the following expression to create a raster with two values 1 and 2 in one step.

```
("elevation@1" < 50) * 1 + ("elevation@1" >= 50) * 2
```

In other words, for every cell less than 50 set its value to 1. For every cell greater than or equal 50 set its value to 2.

Or you can use the `IF` operator.

```
if ( elevation@1 < 50 , 1 , 2 )
```

ГЛАВА 14

Working with Mesh Data

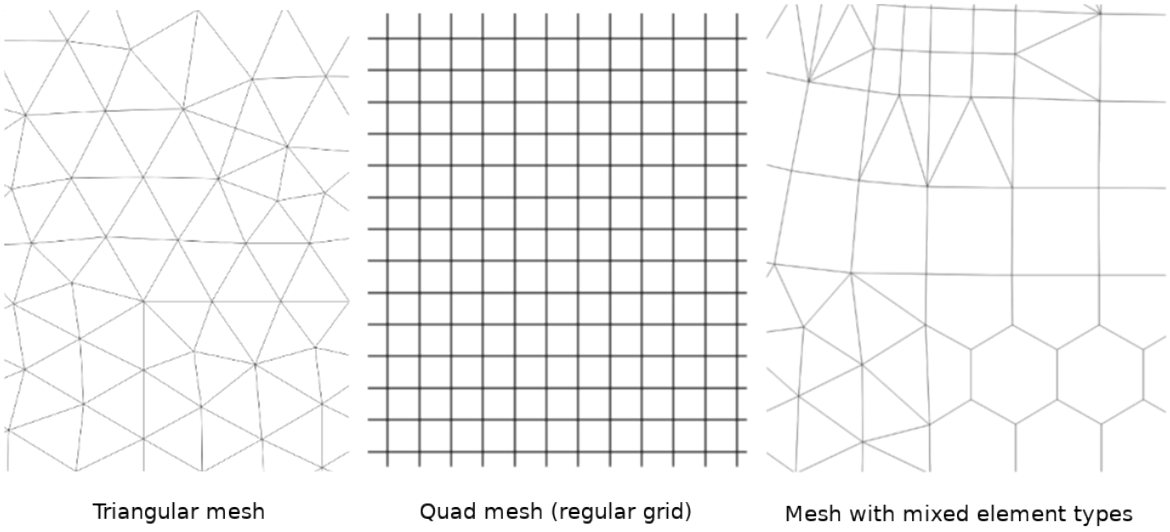
14.1 What's a mesh?

A mesh is an unstructured grid usually with temporal and other components. The spatial component contains a collection of vertices, edges and/or faces, in 2D or 3D space:

- **vertices** - XY(Z) points (in the layer's coordinate reference system)
- **edges** - connect pairs of vertices
- **faces** - a face is a set of edges forming a closed shape - typically a triangle or a quadrilateral (quad), rarely polygons with more vertices

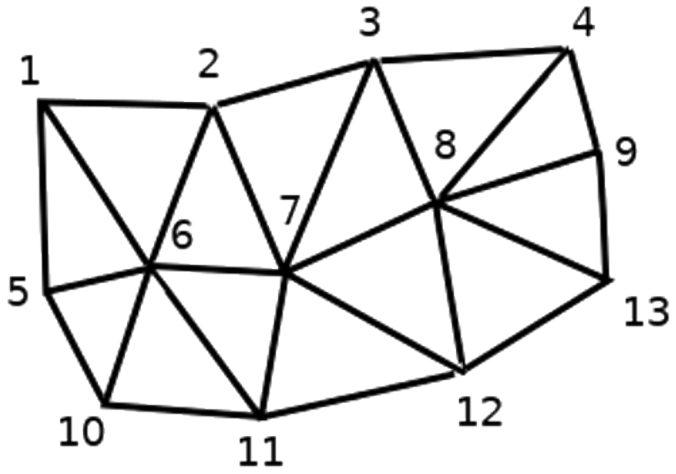
Relying on the above, mesh layers can thus have different types of structure:

- 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.



Слика14.1: 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:



Слика14.2: 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.

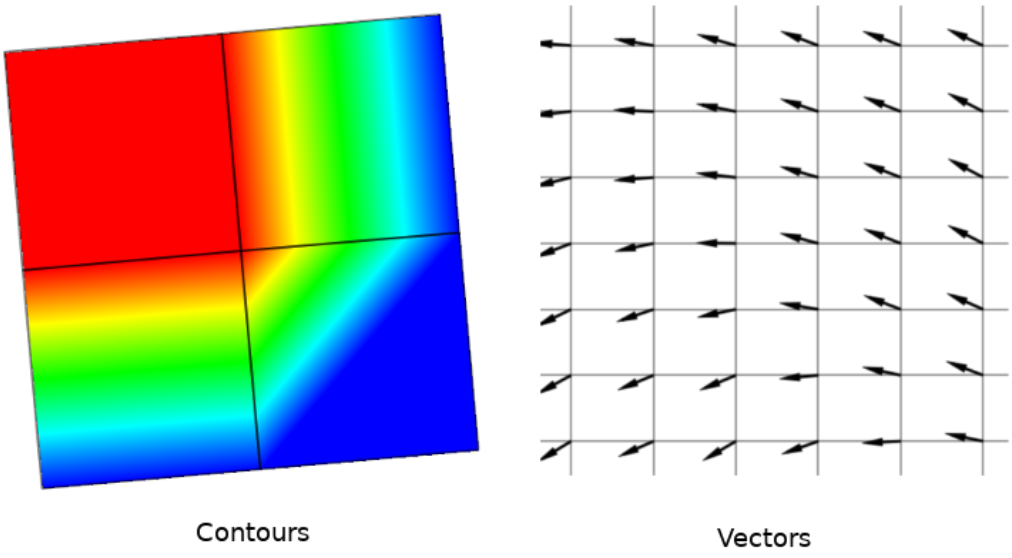
The following table gives an idea about the information that can be stored in mesh datasets. Table columns represent indices of mesh vertices, each row represents one dataset. Datasets can have different datatypes. In this case, it stores wind velocity at 10m at a particular moments in time (t1, t2, 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:

Табела14.1: 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.



Слика14.3: 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 *Loading a mesh layer* 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.

To access the *Layer Properties* dialog:

- In the *Layers* panel, double-click the layer or right-click and select *Properties…* from the pop-up menu;

- Go to *Layer* ► *Layer Properties…* menu when the layer is selected.

The mesh *Layer Properties* dialog provides the following sections:

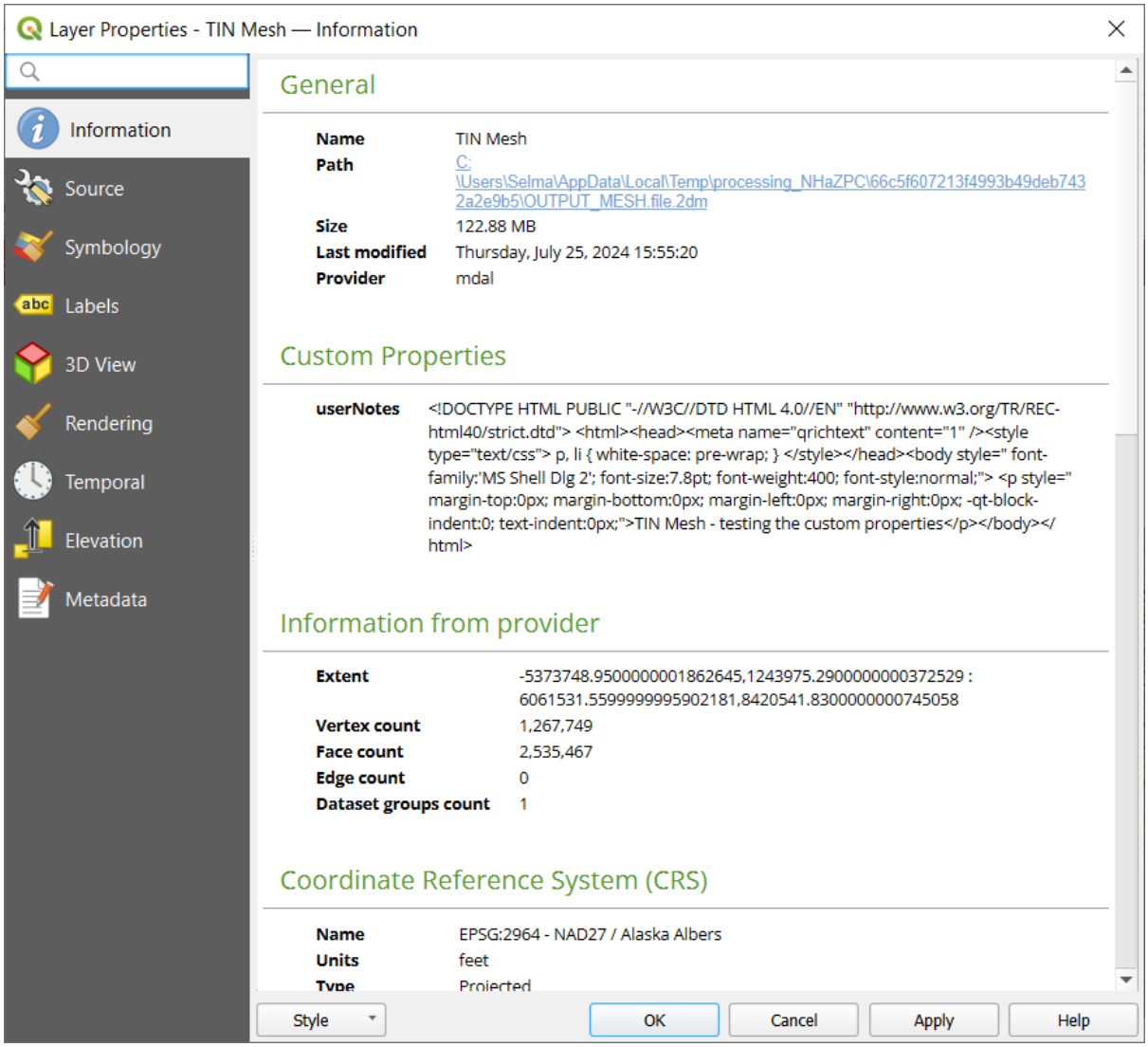
Табела14.2: Tabs of the Mesh Layer Properties

Information	*Source*	*Symbology* [1]
Labels [1]	*3D View* [1]	*Rendering*
Temporal	*Elevation*	*Metadata*

Белешка

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 *Managing Custom Styles*.

14.3.1 Information Properties



Слика14.4: Mesh Layer Information Properties

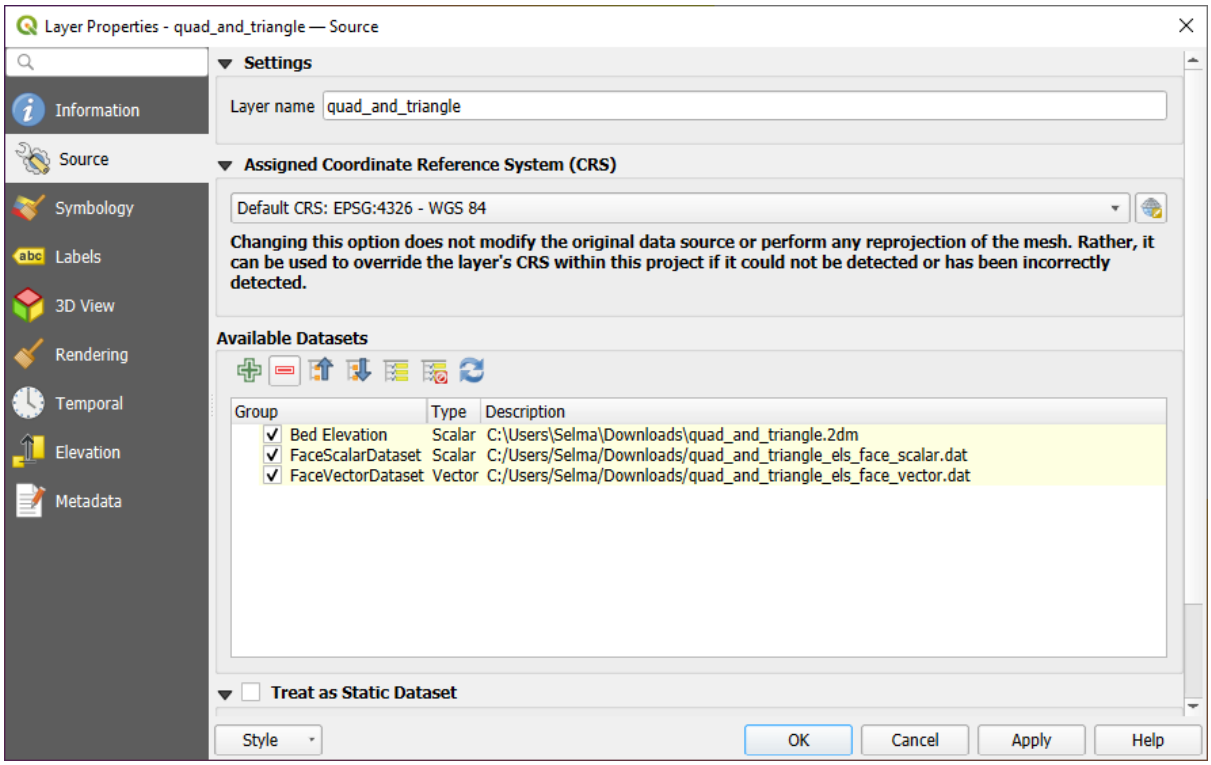
[1] Also available in the *Layer styling panel*

The *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata on the current layer. Provided information are:

- general such as name in the project, source path, list of auxiliary files, last save time and size, the used provider
- custom properties, used to store in the active project additional information about the layer. Default custom properties may include *layer notes*. More properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- based on the provider of the layer: extent, vertex, face, edges and/or dataset groups count
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e. whether it's static or dynamic)
- extracted from filled *metadata*: access, extents, links, contacts, history…

14.3.2 Source Properties

The *Source* tab displays basic information about the selected mesh, including:



Слика14.5: 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 *Coordinate Reference System Selector*). 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 Symbology Properties

Click the *Symbology* button to activate the dialog. Symbology properties are divided into several tabs:

- *Datasets*
- *Contours*
- *Vectors*
- *Rendering*
- *Stacked mesh averaging method*

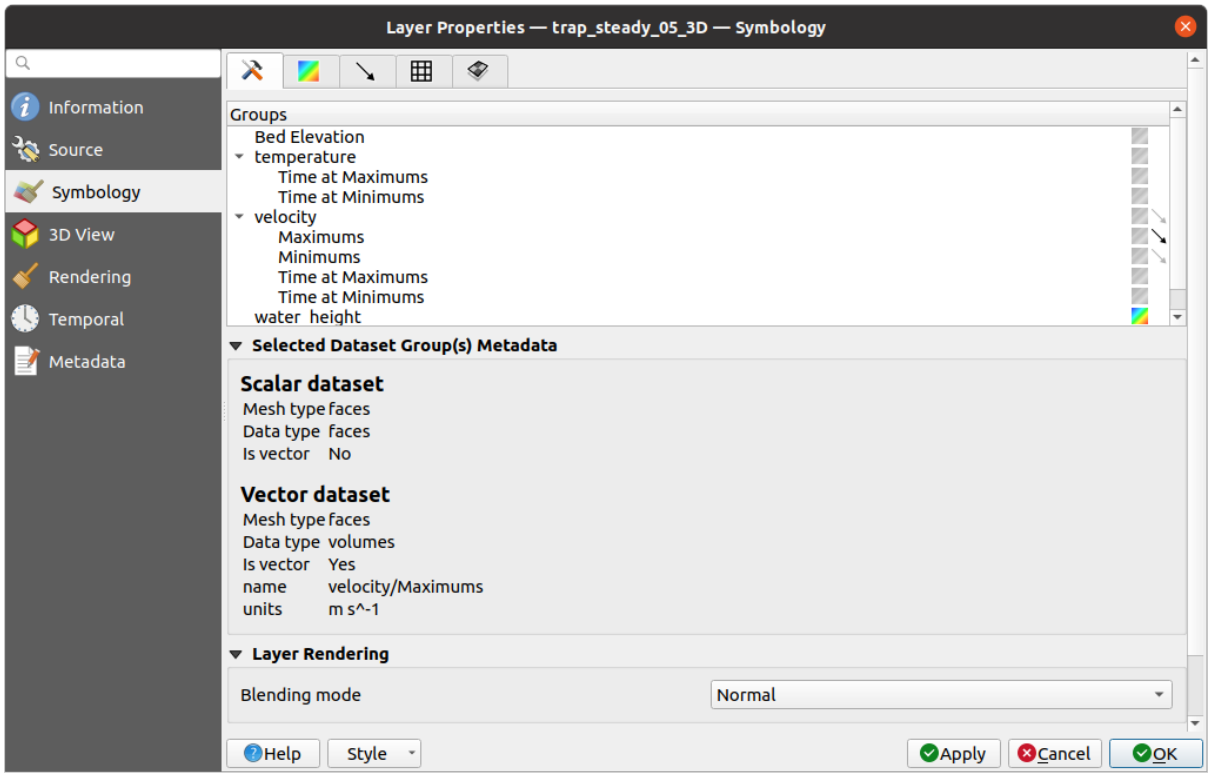
Datasets

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.



Слика14.6: Mesh Layer Datasets

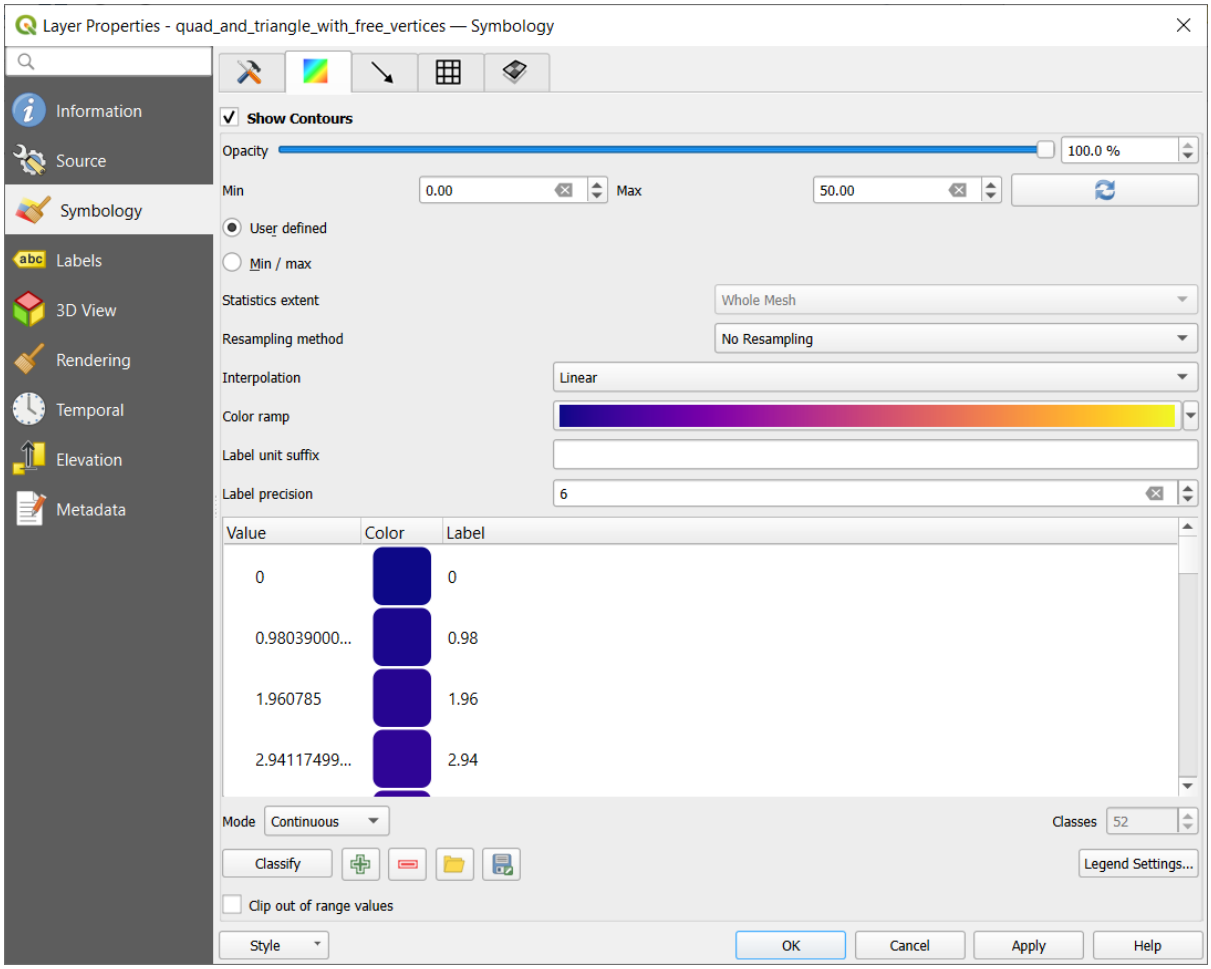
You can apply symbology to the selected vector and/or scalar group using the next tabs.

Contours Symbology

Белешка

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 Сл.14.7 below:



Слика14.7: 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.
- *Statistics 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.

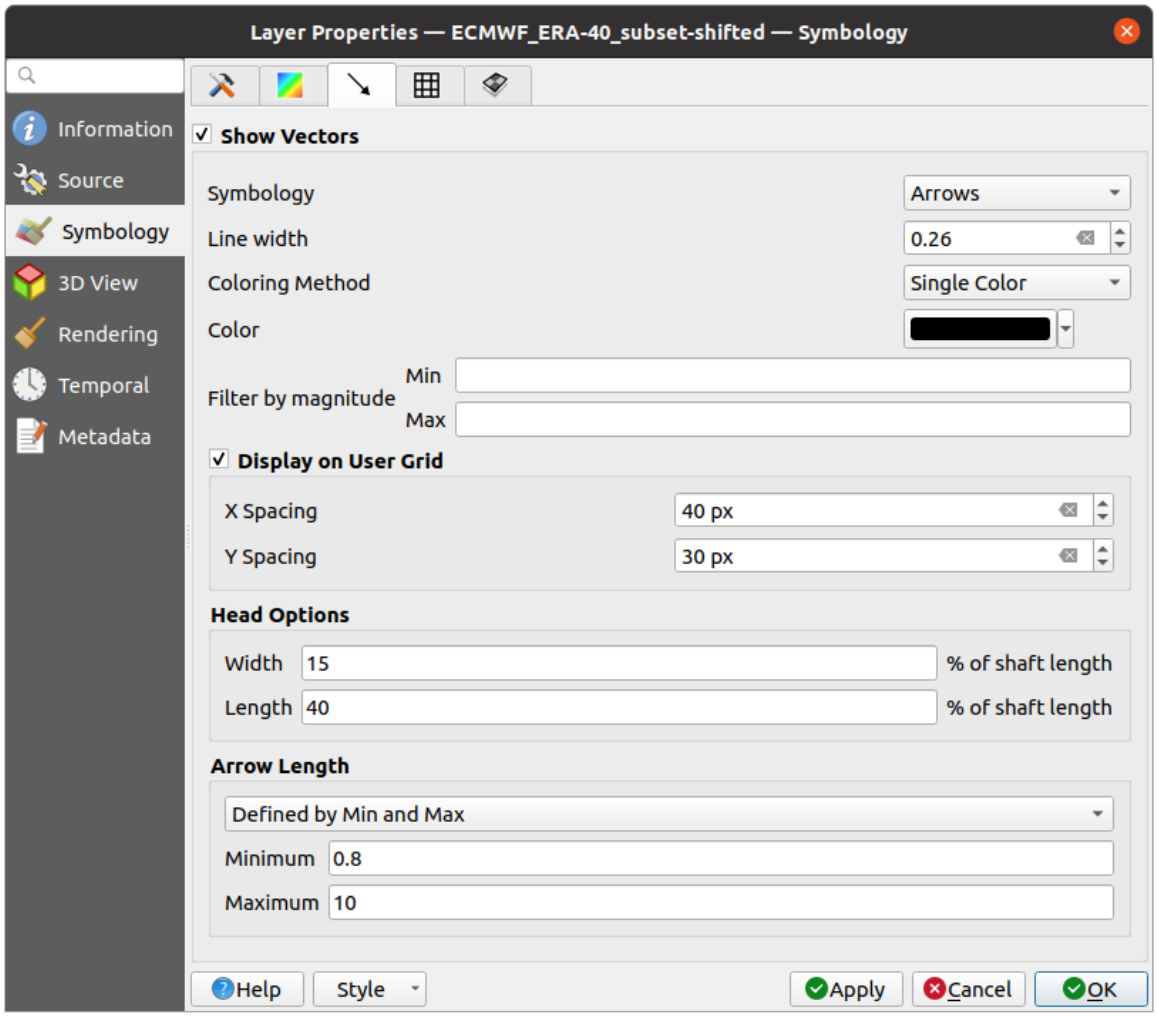
- Classify the dataset using the *color ramp shader* classification.

Vectors Symbology

Белешка

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 Сл.14.8:



Слика14.8: 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.

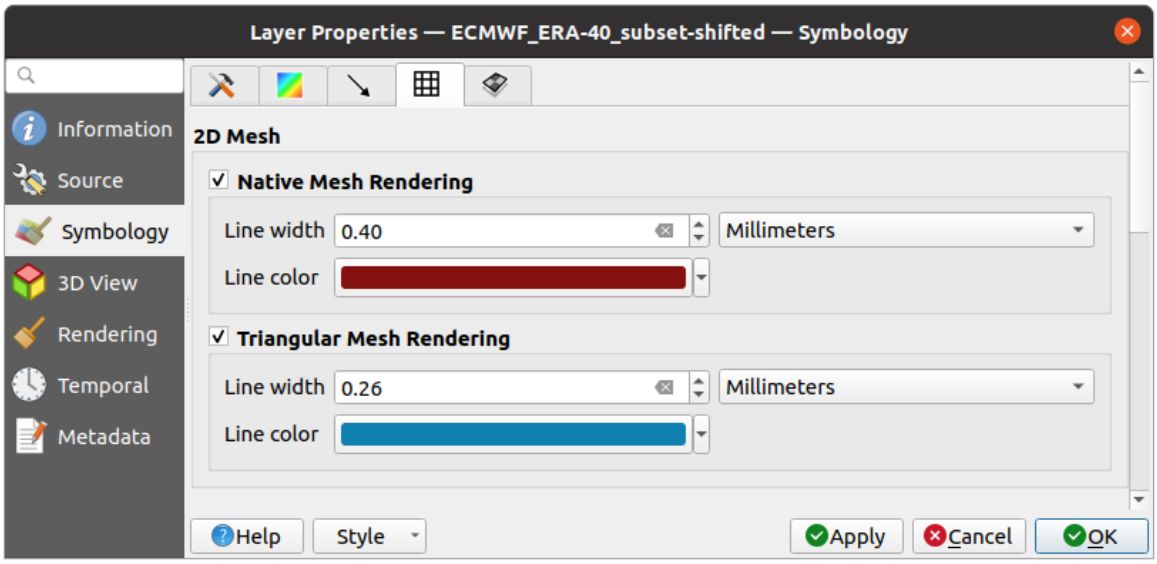
Табела14.3: Availability and meaning of the vectors symbology properties

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.				☑

Rendering

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



Слика14.9: 2D Mesh Rendering

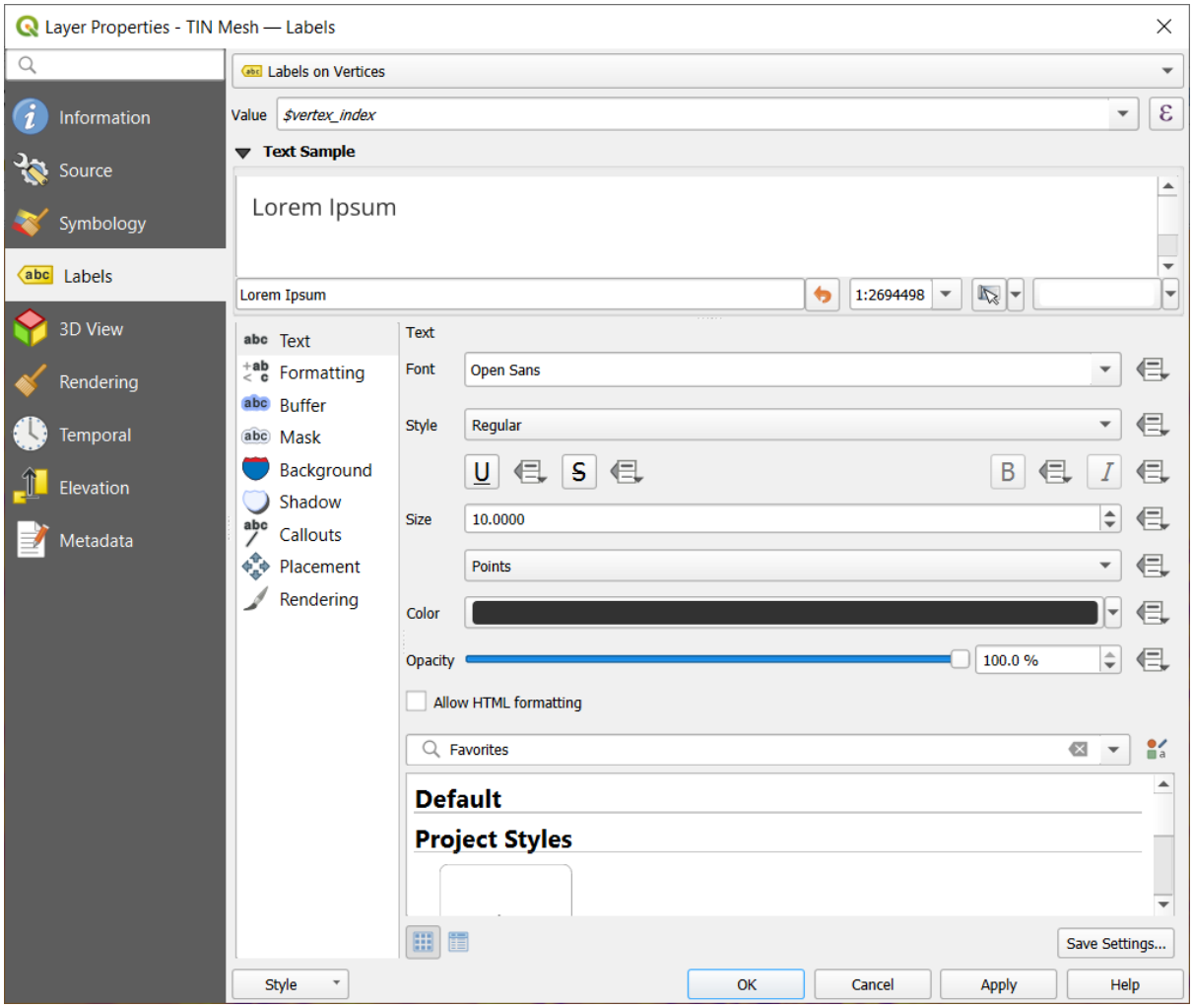
Stacked mesh averaging method

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 Labels Properties

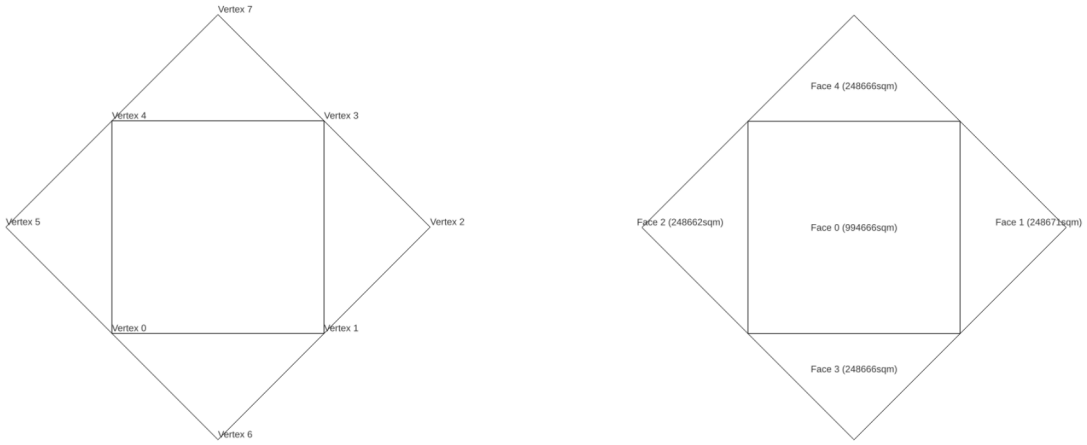
The *Labels* tab offers you dynamic labeling of mesh vertices and native mesh faces based on geometric properties and custom expressions. This dialog can also be accessed from the *Layer Styling* panel.



Слика14.10: 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.



Слика14.11: 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.

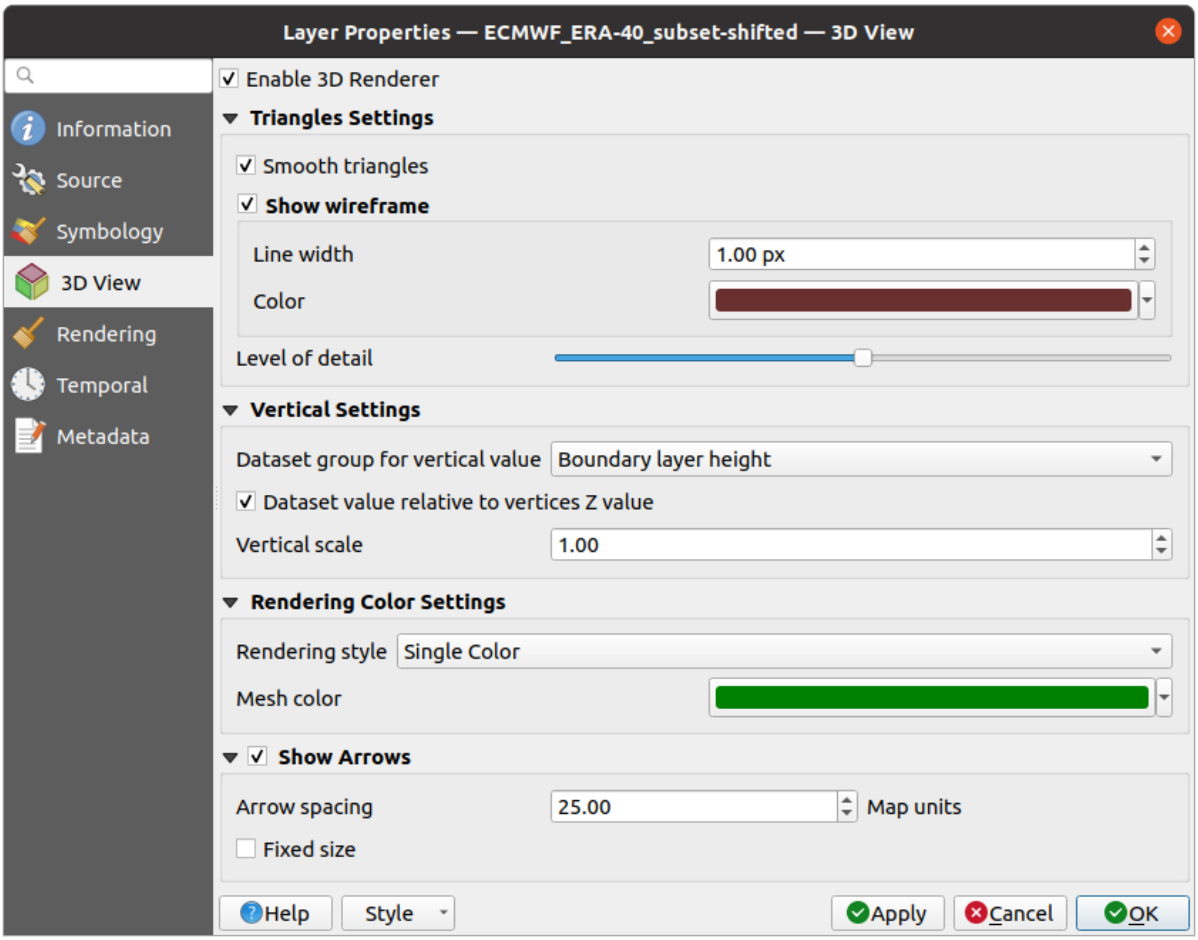
Below are displayed options to customize the labels, under various tabs:

- *Text*
- *Formatting*
- *Buffer*
- *Mask*
- *Background*
- *Shadow*
- *Callouts*
- *Placement*
- *Rendering*

Description of how to set each property is exposed at *Setting a label*.

14.3.5 3D View Properties

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).

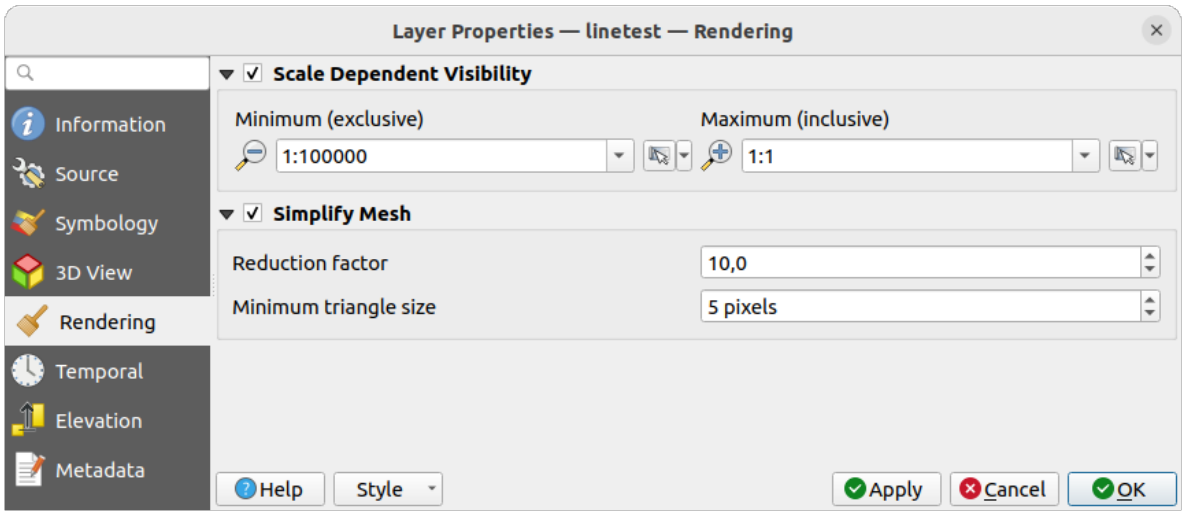


Слика14.12: Mesh dataset 3D properties

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 Rendering Properties



Слика14.13: 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 *Visibility Scale Selector* for more information.

Белешка

You can also activate scale dependent visibility on a layer from within the *Layers* panel: right-click on the layer and in the contextual menu, select *Set Layer Scale Visibility*.

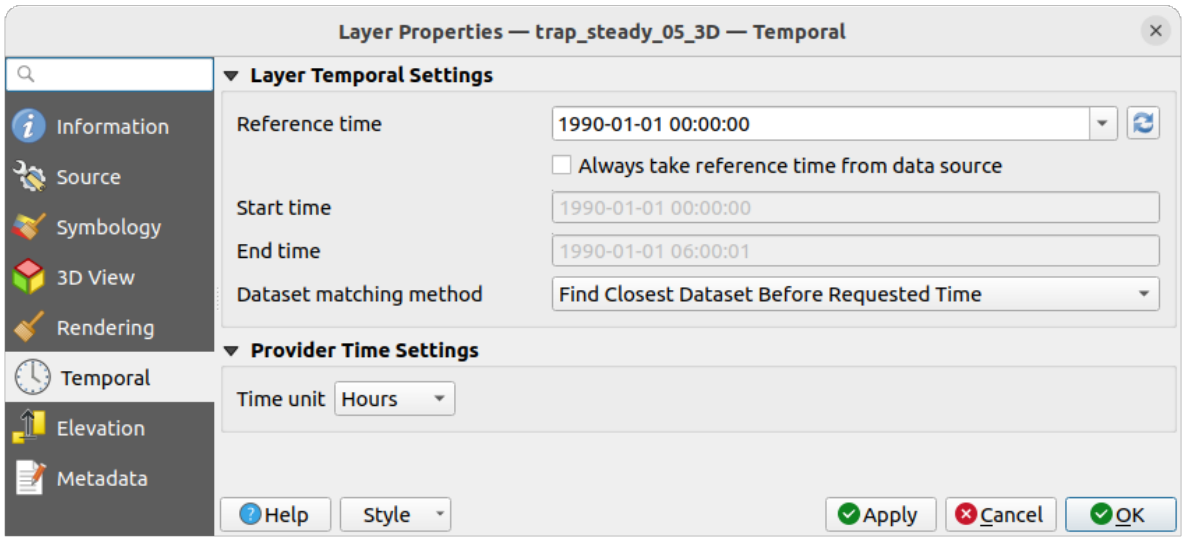
As mesh layers can have millions of faces, their rendering can sometimes be very slow, especially when all the faces are displayed in the view whereas they are too small to be viewed. To speed up the rendering, you can simplify the mesh layer, resulting in one or more meshes representing different *levels 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 Temporal Properties

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.



Слика14.14: 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.

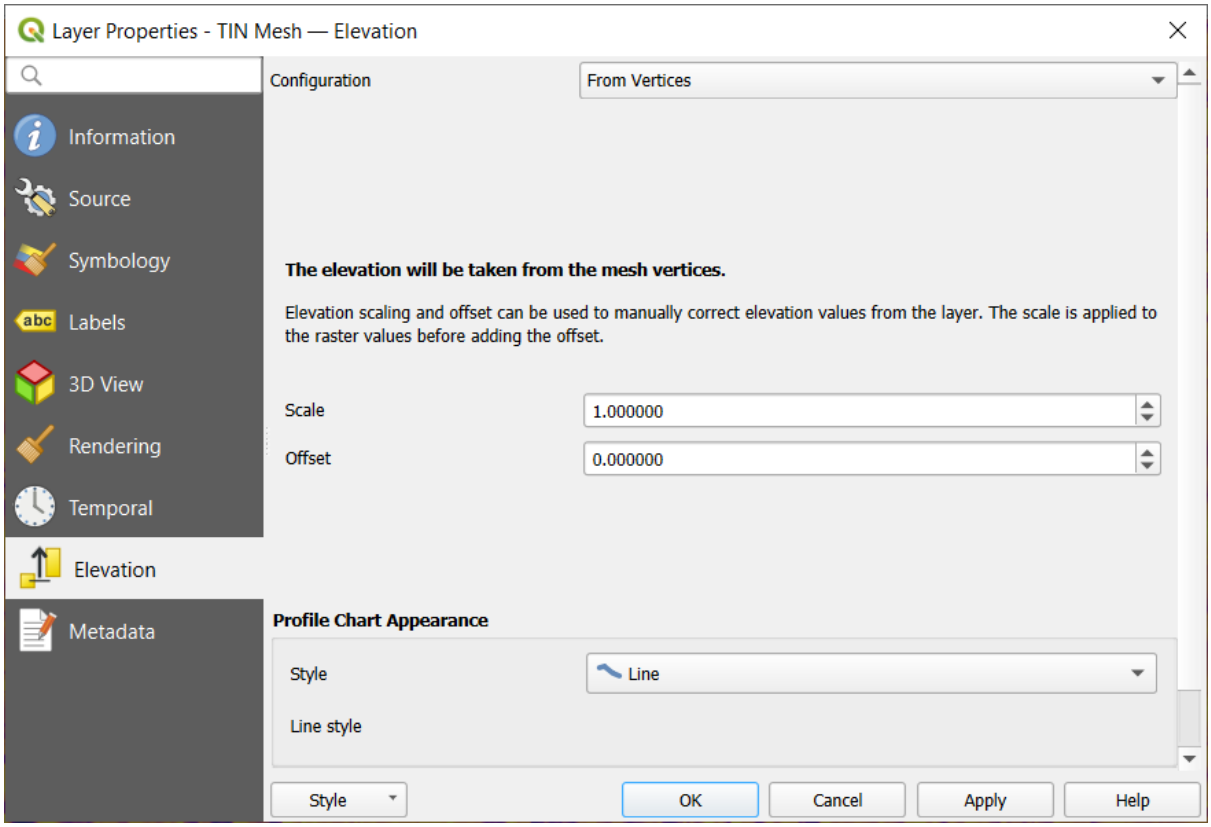
- *Dataset matching method*: determines the dataset to display at the given time. Options are *Find closest dataset before requested time* or *Find closest dataset from requested time (after or before)*.

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 Elevation Properties

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:



Слика14.15: 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 *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:
  - a *Line* with a specific *Line style*
  - an elevation surface rendered using a fill symbol either above (*Fill above*) or below (*Fill below*) the elevation curve line. The surface symbology is represented using:
    - a *Fill style*
    - and a *Limit*: the maximum (respectively minimum) altitude determining how high the fill surface will be

14.3.9 Metadata Properties

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

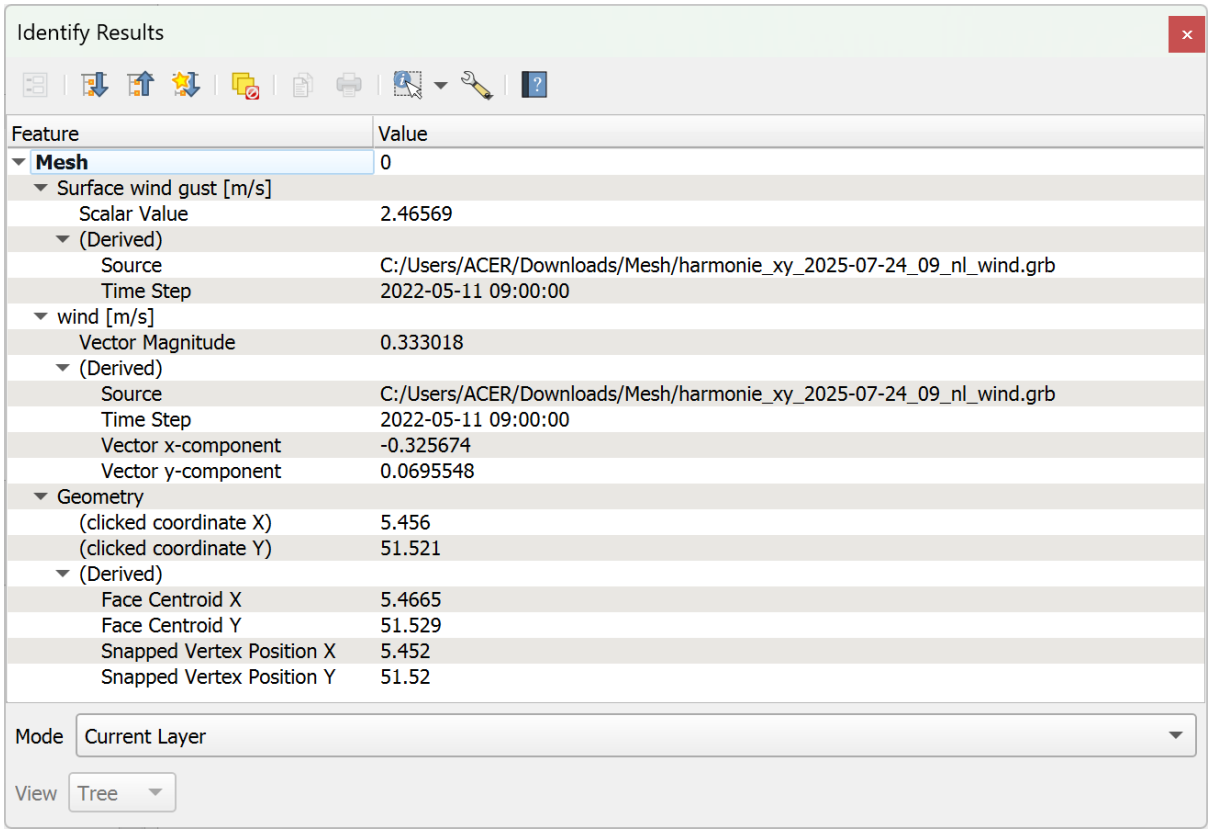
14.4 Exploring mesh datasets

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).



Слика14.16: 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.

Белешка

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.

Табела14.4: Tools for mesh digitizing

Oznaka	Svrha	Location
 Current Edits	Access to save, rollback or cancel changes in all or selected layers simultaneously	*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 tools	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 Polygon	Select vertices and faces overlapped by a drawn polygon	*Mesh Digitizing* toolbar
 Select Mesh Elements by Expressi	Select vertices and faces using an expression	*Mesh Digitizing* toolbar
 Transform Vertices Coordinates	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:

Табела14.5: Z value assignment methods

Method	Opis
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.

Табела14.6: 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	Ne	*Vertex Z value*	Default or user defined
		Advanced Digitizing Panel (if *z* widget is in Locked state)	*z* widget if in Locked state
	Da	*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

Белешка

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.

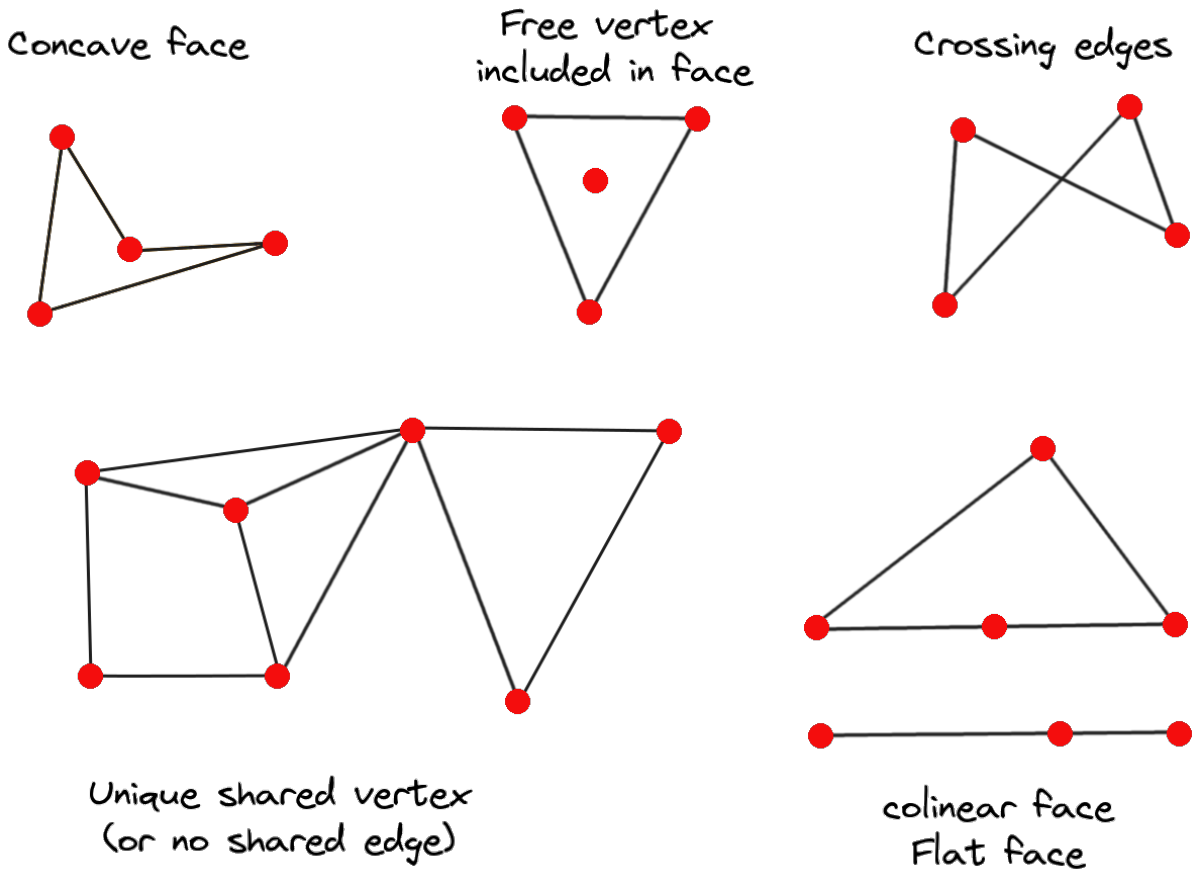
     With the *Refine neighboring faces when adding vertices* option enabled in the Digitize mesh elements button drop-down menu, a check is applied to triangular faces that share at least one vertex with the face the new vertex is added to. If their edges do not satisfy Delaunay triangulation rules, they 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.



Слика14.17: 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

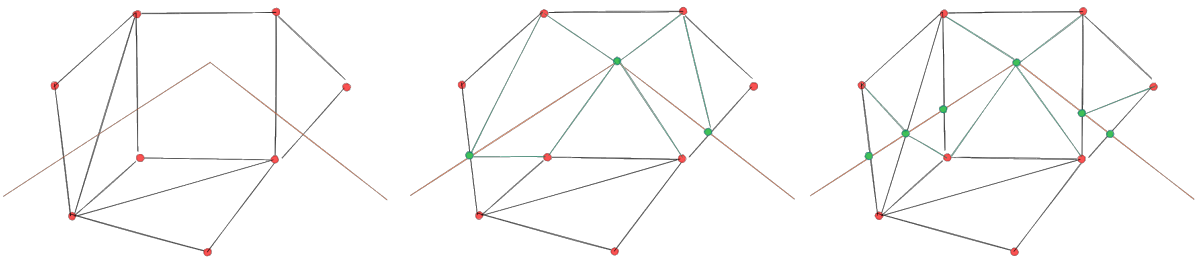
The Force by Selected Geometries tool provides advanced ways to apply break lines using lines geometry. A break line will force the mesh to have edges along the line. Note that the break line will not be considered persistent once the operation is done; resulting edges will not act as constraints anymore 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).



Слика14.18: 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 *Unit Selector*).

14.5.5 Reindexing meshes

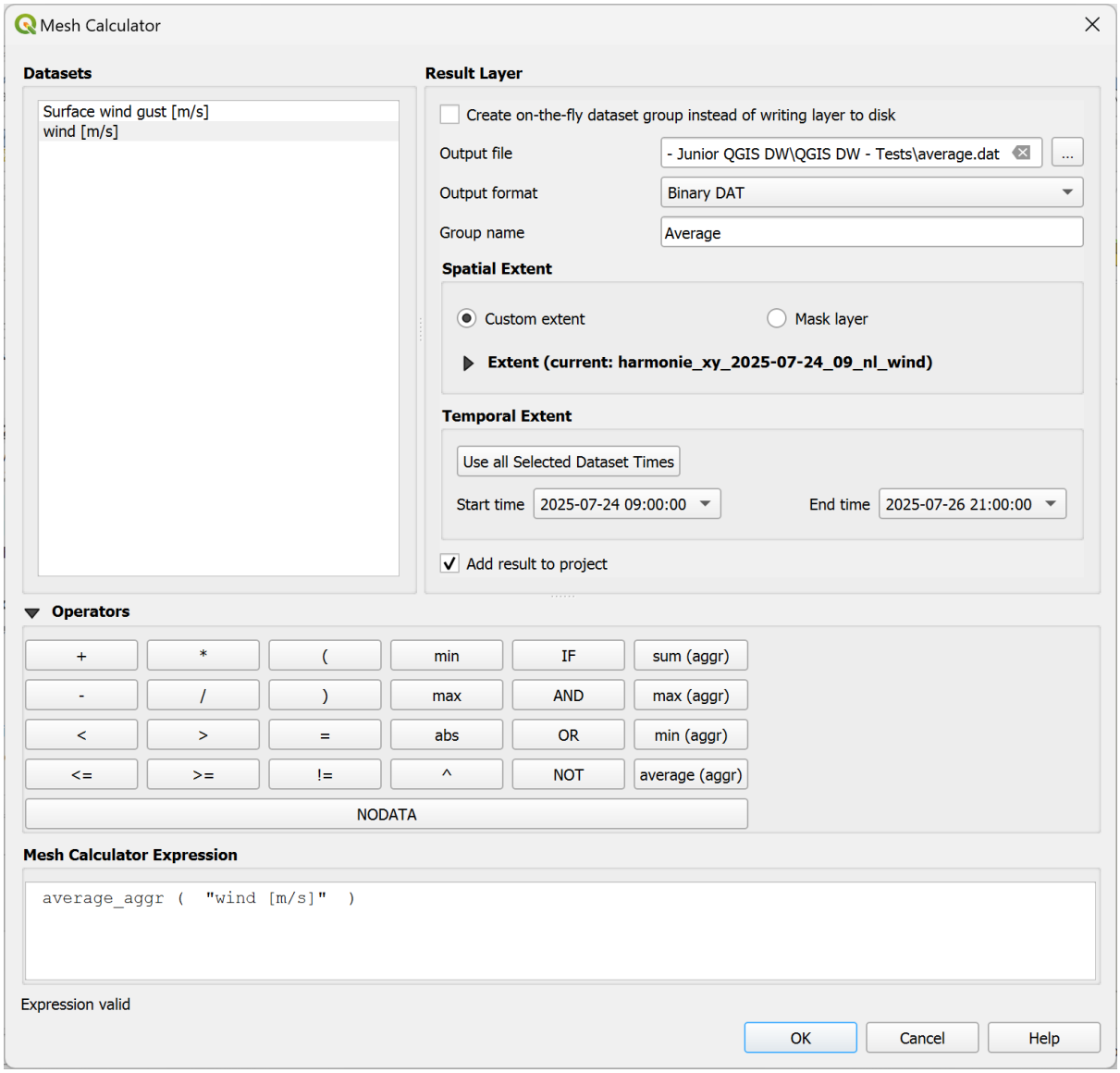
During edit, and in order to allow quick undo/redo operations, QGIS keeps empty places for deleted elements, which may lead to growing memory use and inefficient mesh structuring. The *Mesh* ► *Reindex Faces and Vertices* tool is designed to remove these holes and renumber the indices of faces and vertices so that they are continuous and somewhat reasonably ordered. This optimizes relation between faces and vertices and increases the efficiency of calculation.

Белешка

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 Сл.14.19).



Слика14.19: 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.

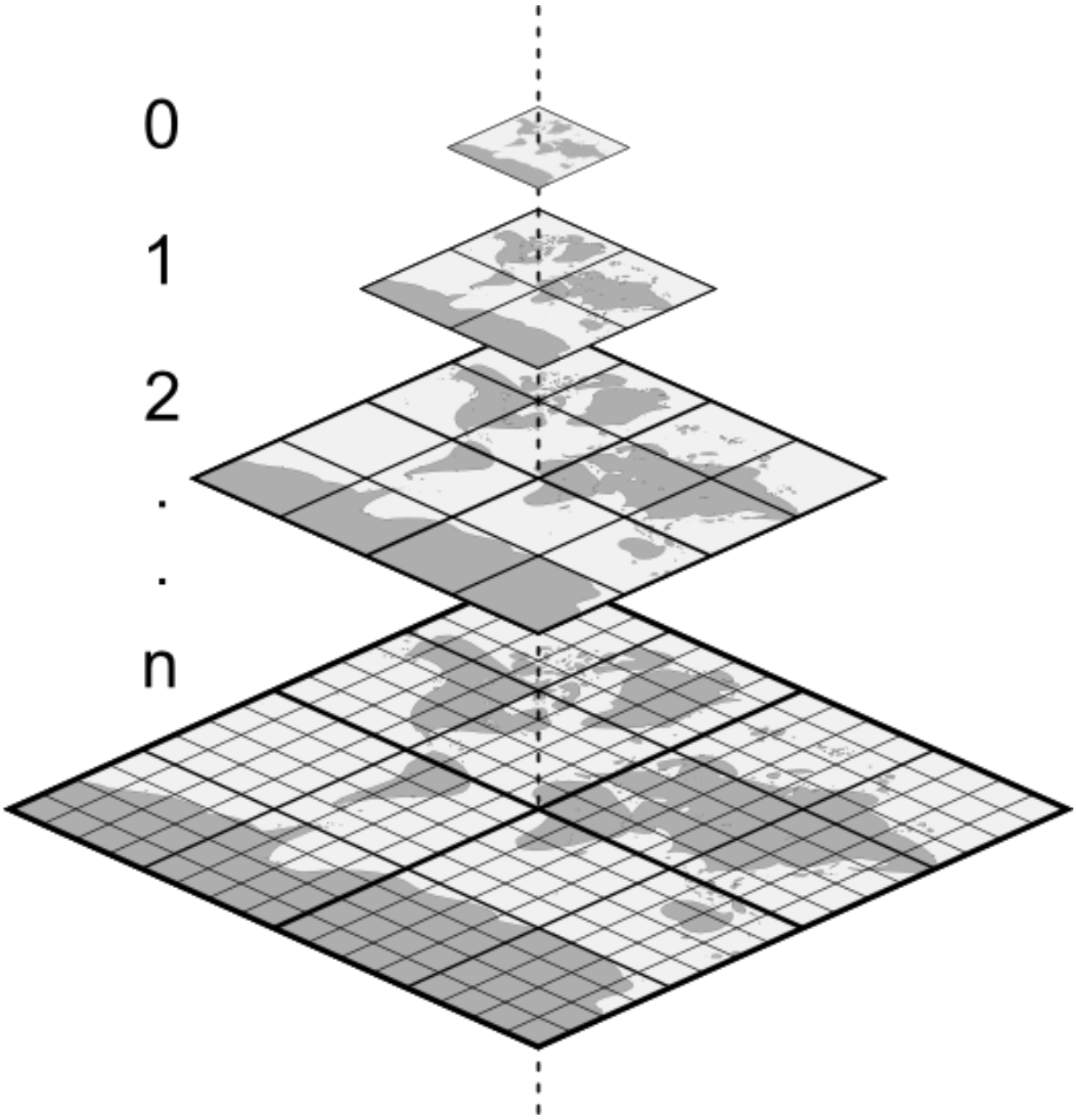
The *Mesh Calculator Expression* widget shows and lets you edit the expression to execute.

ГЛАВА 15

Working with Vector Tiles

15.1 What are Vector Tiles?

Vector tiles are packets of geographic data, packaged into pre-defined roughly-square shaped „tiles” for transfer over the web. They combine pre-rendered raster map tiles and vector map tiles. The vector tile server returns vector map data, which has been clipped to the boundaries of each tile, instead of a pre-rendered map image. The clipped tiles represent the zoom-levels of the vector tile service, derived from a pyramid approach. Using this structure, the data-transfer is reduced in comparison to un-tiled vector maps. Only data within the current map view, and at the current zoom level need to be transferred. Also, compared to a tiled raster map, data transfer is also greatly reduced, as vector data is typically much smaller than a rendered bitmap. Vector tiles do not have any styling information assigned so QGIS needs to apply a cartographic style in order to display the data.



Слика15.1: Pyramid structure of vector tiles with zoom-levels

15.2 Supported Formats

There is support for vector tiles through:

- remote sources (HTTP/S) - with XYZ template - `type=xyz&url=http://example.com/{z}/{x}/{y}.pbf`
- local files - with XYZ template - e.g. `type=xyz&url=file:///path/to/tiles/{z}/{x}/{y}.pbf`
- local MBTiles database - e.g. `type=mbtiles&url=file:///path/to/file.mbtiles`

To load a vector tiles dataset into QGIS, use the *Vector Tile* tab in the *Data Source Manager* dialog. Read *Using Vector Tiles services* for more details.

15.3 Vector Tiles Dataset Properties

The vector tiles *Layer Properties* dialog provides the following sections:



15.3.1 Information Properties

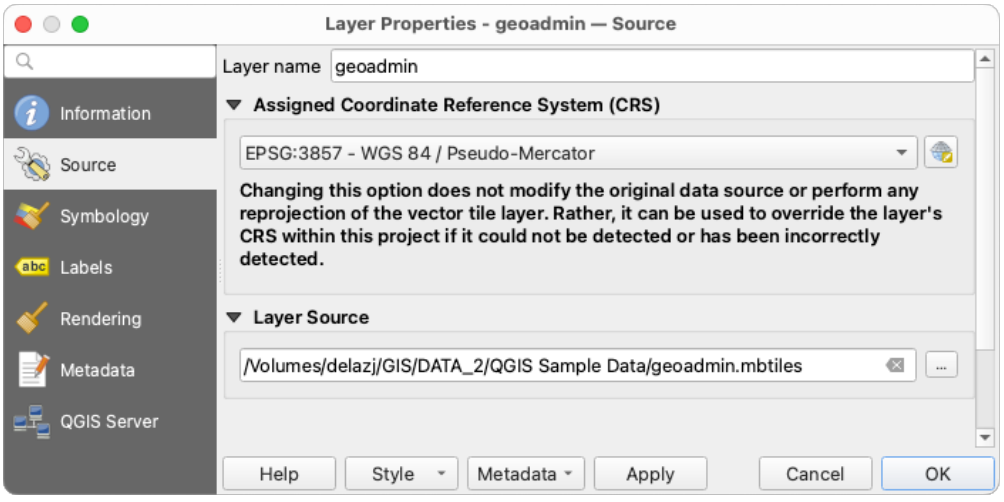
The *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata on the current layer. Provided information are:

- based on the provider of the layer: name, URI, source type and path, number of zoom levels
- custom properties, used to store in the active project additional information about the layer. More properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e., whether it is static or dynamic)
- picked from the *filled metadata*: access, extents, links, contacts, history…

15.3.2 Source Properties

The *Source* tab displays basic information about the selected vector tile, including:

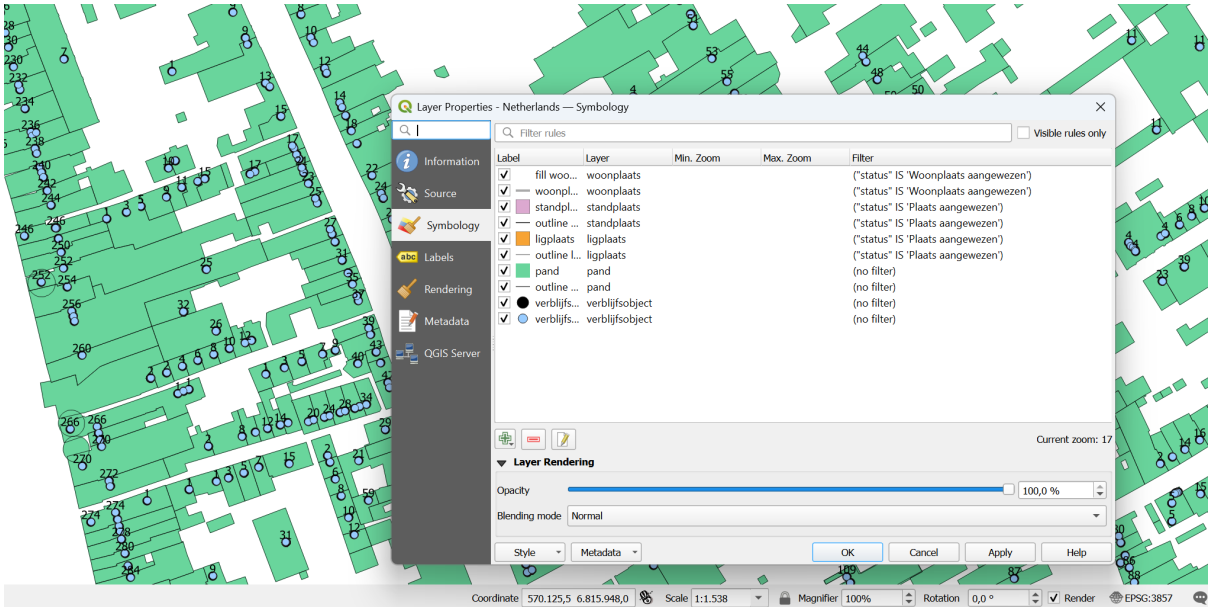
- the *Layer name* to display in the *Layers Panel*;
- the *Coordinate Reference System*: Displays the layer's *Coordinate Reference System (CRS)*. You can change the layer's CRS, by selecting a recently used one in the drop-down list or clicking on the Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the layer CRS is wrong or not specified.
- 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.



Слика15.2: Vector Tiles Properties - Source Dialog

[1] Also available in the *Layer styling panel*

15.3.3 Symbology and Labels Properties



Слика15.3: Vector Tile Layer Symbology

Setting rules

As vector tiles consist of point, line and polygon geometries, the respective symbols are available. To apply a cartographic style (with symbology and/or labels), you can either:

- Use a *Style URL* when creating the *Vector Tiles Connection*. The symbology will be shown immediately in the *Symbology* tab after the layer is loaded in QGIS.
- Or build your own symbology and labeling in the corresponding tabs of the layer properties. By default, QGIS assigns an identical symbol to the features based on their geometry type.

In both cases, setting a style for a vector tile relies on a set of *rules* applied to the features, indicating:

- a vector *symbol* or *label*
- a *Label*, a title for comprehensive identification of the rule
- the name of a particular *Layer* the rule should apply to, if not applied to `(all layers)`
- a *Min. Zoom* and a *Max. Zoom*, for the range of display. Symbology and labeling can be dependent on the zoom level.
- a *Filter*, a QGIS expression to identify the features to apply the style to

Each rule is added pressing the Add rule button and selecting the type of symbols (*Marker*, *Line*, *Fill*) corresponding to the features geometry type. You can as well Remove selected rules or Edit current rule.

At the bottom the *Current Zoom* is shown. Check the *Visible rules only* option at the top of the dialog to filter the list of rules to only those that are visible at the current zoom level. This makes it easier to work with complex vector styling and to locate troublesome rules. The *Filter rules* text box also helps you easily find a rule, by searching the *Label*, *Layer* and *Filter* fields.

In Сл.15.3 we set up style for the OpenStreetMap `landuse` layer. For better visibility most of the rules are deselected.

Layer rendering

From the *Symbology* tab, you can also set some options that invariably act on all features of the layer:

- *Opacity*: You can make the underlying layer in the map canvas visible with this tool. Use the slider to adapt the visibility of your vector layer to your needs. You can also make a precise definition of the percentage of visibility in the menu beside the slider.
- *Blending mode*: You can achieve special rendering effects with these tools that you may previously only know from graphics programs. The pixels of your overlaying and underlying layers are mixed through the settings described in *Blending Modes*.

Stilovi

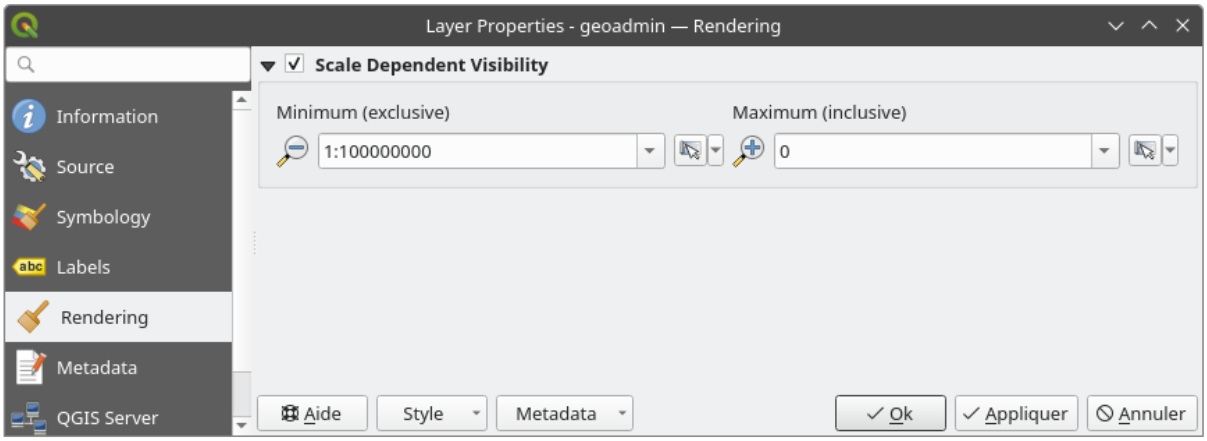
Available at the bottom of most of the tabs, the *Styles* ► menu provides shortcuts to save, load, create, switch styles to apply to the vector tiles. Vector tiles can have their style saved from QGIS as *QML* files and they can be imported as:

- *QML* files (*QML - The QGIS Style File Format*)
- *MapBox GL Json* style configuration files

More details at *Save and Share Layer Properties*.

15.3.4 Rendering Properties

Under *Scale dependent visibility*, you can set the *Maximum (inclusive)* and *Minimum (exclusive)* scales, defining a range of scales in which features 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 *Visibility Scale Selector* for more information.



Слика15.4: Vector Tiles Properties - Rendering Dialog

15.3.5 Metadata Properties

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

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*

ГЛАВА 16

Working with Point Clouds

16.1 Introduction to Point Clouds

What is A Point Cloud?

A point cloud is a three-dimensional image of a space made up of many individual of data points (up to billions, even trillions). Each of the points has an x, y and z coordinate. Depending on the capture method, point clouds usually also have additional attributes that came from the capture, such as color values or intensity. These attributes can be used, for instance, to display point clouds in different colors. In QGIS, a point cloud can be used to generate a three-dimensional image of the landscape (or of another space).

Supported Formats

QGIS supports the data formats Entwine Point Tile (EPT) and LAS/LAZ. To work with point clouds, QGIS always saves the data in EPT. EPT is a storage format that consists of several files stored in a common folder. To allow quick access to the data, EPT uses indexing. For more information on the EPT format, see entwine homepage

If the data is in LAS or LAZ format, QGIS will convert it to EPT when it is loaded for the first time. Depending on the size of the file, this may take some time. In this process, a subfolder is created in the folder in which the LAS/LAZ file is located according to the scheme `ept_` + `name_LAS/LAZ_file`. If such a subfolder already exists, QGIS loads the EPT immediately (which leads to a reduced loading time).

Worth Knowing

In QGIS it is not (yet) possible to edit point clouds. If you want to manipulate your point cloud, you can use CloudCompare, an open source point cloud processing tool. Also the Point Data Abstraction Library (PDAL - similar to GDAL) offers you options to edit point clouds (PDAL is command line only).

Due to the large number of data points, it is not possible to display an attribute table of point clouds in QGIS. However, the  *Identify tool* supports point clouds, so you can display all attributes, even of a single data point.

If you want to create a new layer, in the same or another supported format, from an existing point cloud layer, see *Creating new layers from an existing layer*.

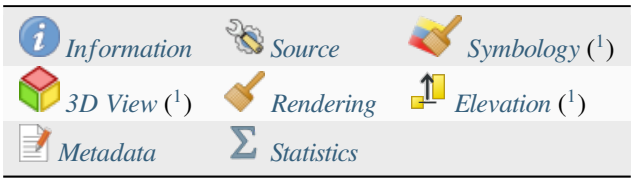
16.2 Point Clouds Properties

The *Layer Properties* dialog for a point cloud layer offers general settings for the layer and its rendering. It also provides information about the layer.

To access the *Layer Properties* dialog:

- In the *Layers* panel, double-click the layer or right-click and select *Properties…* from the context menu;
- Go to *Layer* ► *Layer Properties…* menu when the layer is selected.

The point cloud *Layer Properties* dialog provides the following sections:



Белешка

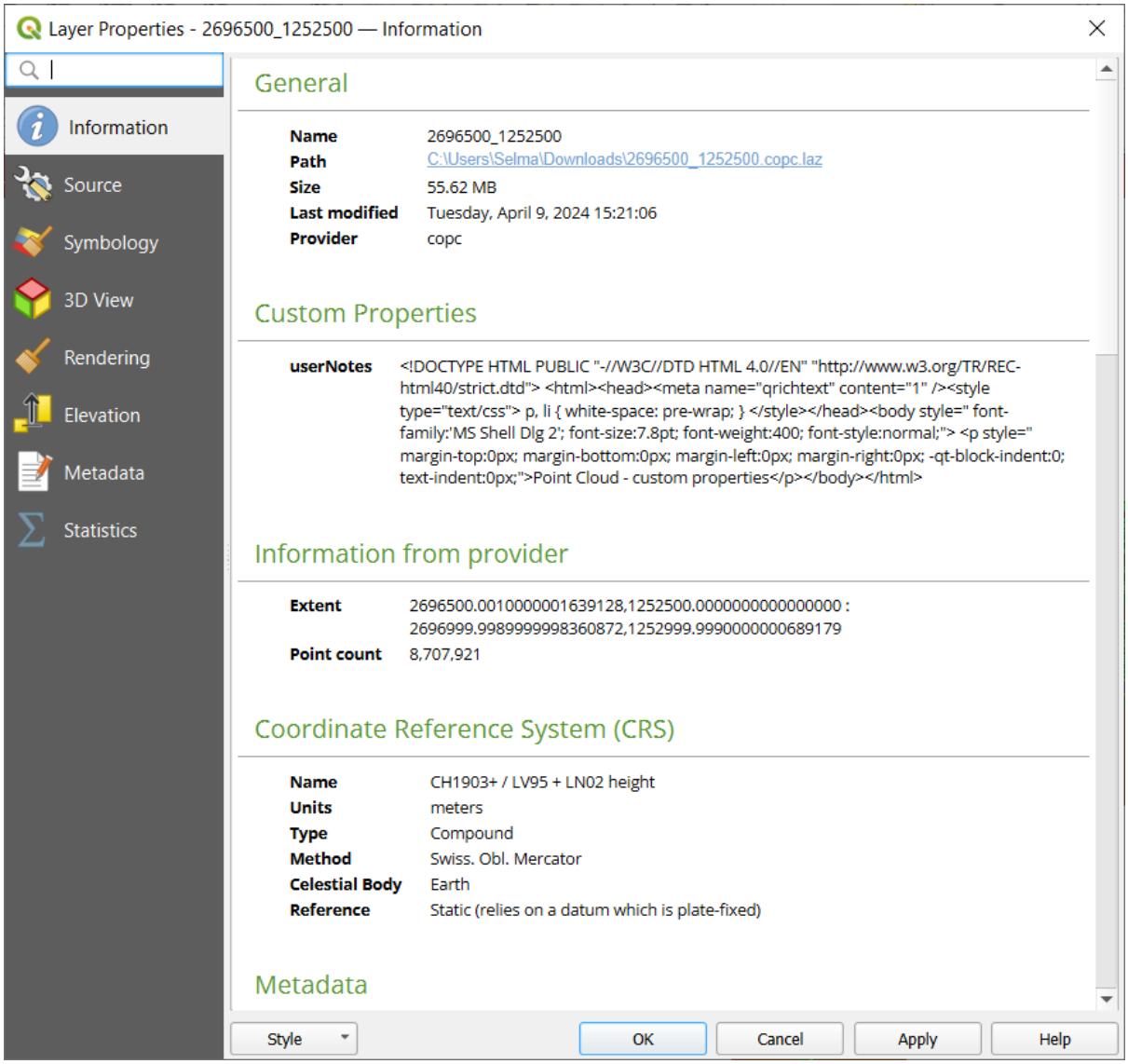
Most of the properties of a point cloud layer can be saved to or loaded from a `.qml` file using the *Style* menu at the bottom of the properties dialog. More details at *Save and Share Layer Properties*

16.2.1 Information Properties

The *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata on the current layer. Provided information are:

- General such as name in the project, source path, last save time and size, the used provider
- custom properties, used to store in the active project additional information about the layer. Default custom properties may include *layer notes*. More properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- Based on the provider of the layer: extent and number of points
- The Coordinate Reference System: name, units, method, accuracy, reference (i.e. whether it's static or dynamic)
- Metadata delivered by the provider: creation date, version, data format, scale X/Y/Z, …
- Picked from the *Metadata* tab (where they can be edited): access, extents, links, contacts, history…

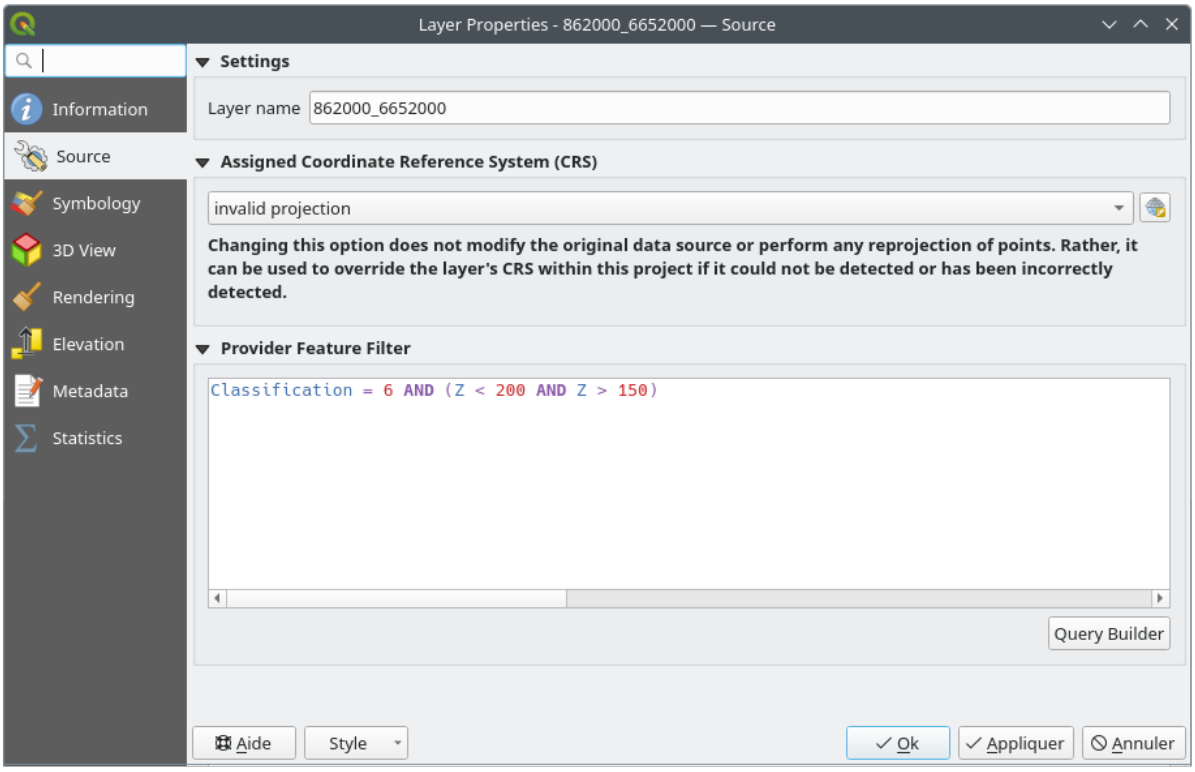
[1] Also available in the *Layer styling panel*



Слика16.1: Point cloud information tab

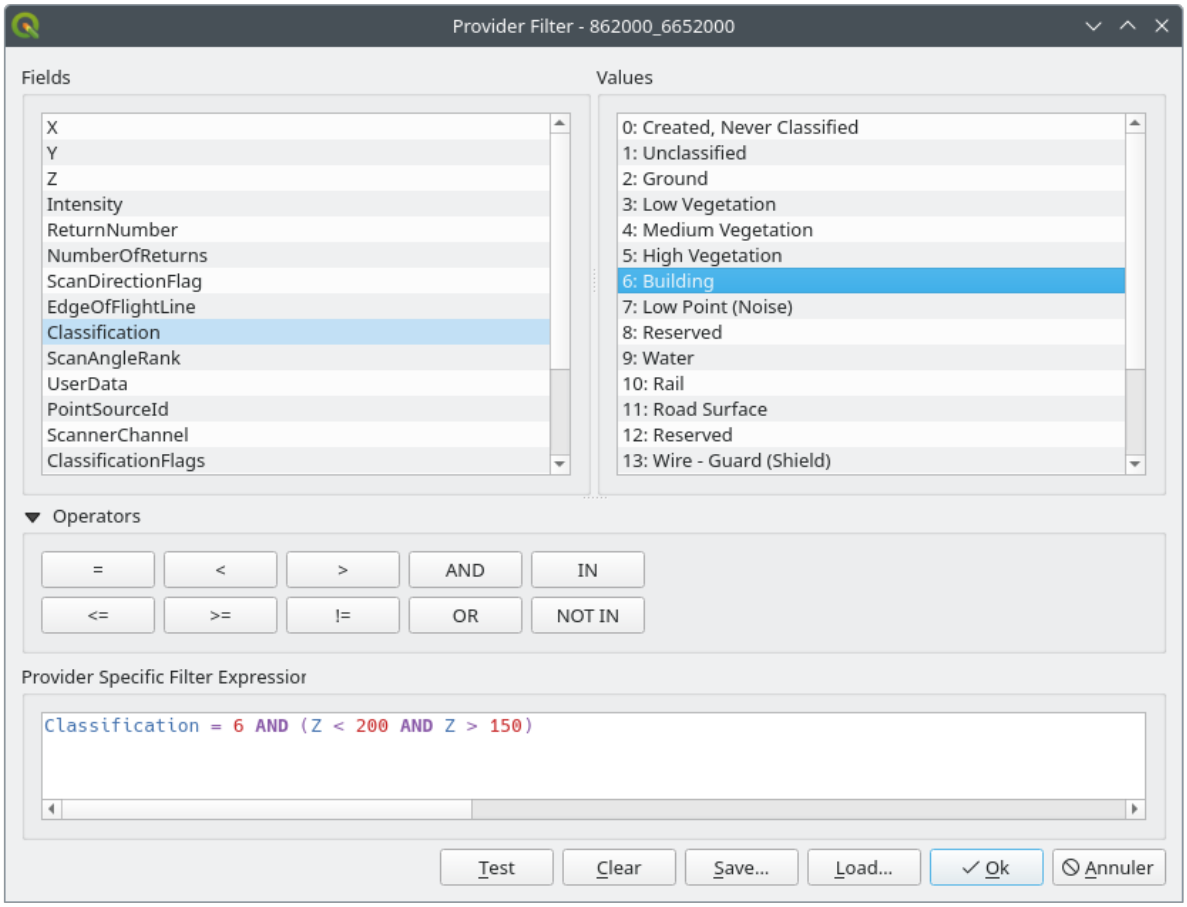
16.2.2 Source Properties

In the  *Source* tab you can see and edit basic information about the point cloud layer:



Слика16.2: Point cloud source tab

- *Settings*: Set a Layer name different from the layer filename that will be used to identify the layer in the project (in the Layers Panel, with expressions, in print layout legend,…)
- *Assigned Coordinate Reference System (CRS)*: Here you can change the layer's assigned *Coordinate Reference System*, selecting a recently used one in the drop-down list or clicking on set Projection Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the CRS applied to the layer is a wrong one or if none was applied.
- *Provider Feature Filter*: allows to restrict data to use in the current project to a subset, using functions and expression at the PDAL data provider level. Press the *Query Builder* button at the bottom to start setting the filter.



Слика16.3: Filtering point cloud layer to buildings at specific altitude

In the bottom part of the dialog, you can type a *Provider specific filter expression*. Such an expression can be built thanks to:

- *Fields*: the list contains all the attributes of the point cloud layer. To add an attribute to the expression, double-click its name or just type it into the text box.
- *Values*: frame lists values or statistics of the currently selected attribute, as a `key:value` pair. To add a value to the expression field, double click its name in the list: depending on the type of attribute, either the key or the value will be added to the expression. Or type the value in the expression text box.
- *Operators*: this toolbar contains all usable operators. To add an operator to the expression field, click the appropriate button. Relational operators (= , > , …) and logical operators (`AND`, `OR`, …) are available.

The *Test* button helps you check the syntax of your query. Use the *Clear* button to wipe the query and revert the layer to its original state (i.e., fully load all the points in the layer). It is possible to *Save…* the query as a `.QQF` file, or *Load…* the query from a file into the dialog.

When a filter is applied, QGIS treats the resulting subset as if it were the entire layer. For example if you applied the *filter above* for filtering buildings, you can not e.g. display, query, save or edit points that are of vegetation classification because they are not part of the subset.

Савет

Filtered layers are indicated in the Layers Panel

In the *Layers* panel, a filtered layer is listed with a  Filter icon next to it indicating the query used when the mouse hovers over the icon. Double-click the icon to open the *Query Builder* dialog for edit. This can

also be achieved through the *Layer* ► *Filter…* menu.

16.2.3 Symbology Properties

In the  *Symbology* tab the settings for the rendering of the point cloud are made. In the upper part, the settings of the different feature renderers can be found. In the lower part, there are sections with which general settings for the entire layer can be made and which apply over feature renderers.

Feature Rendering types

There are different options for rendering point clouds that can be selected using the drop-down menu at the top of the *Symbology* tab (see Сл.16.4):

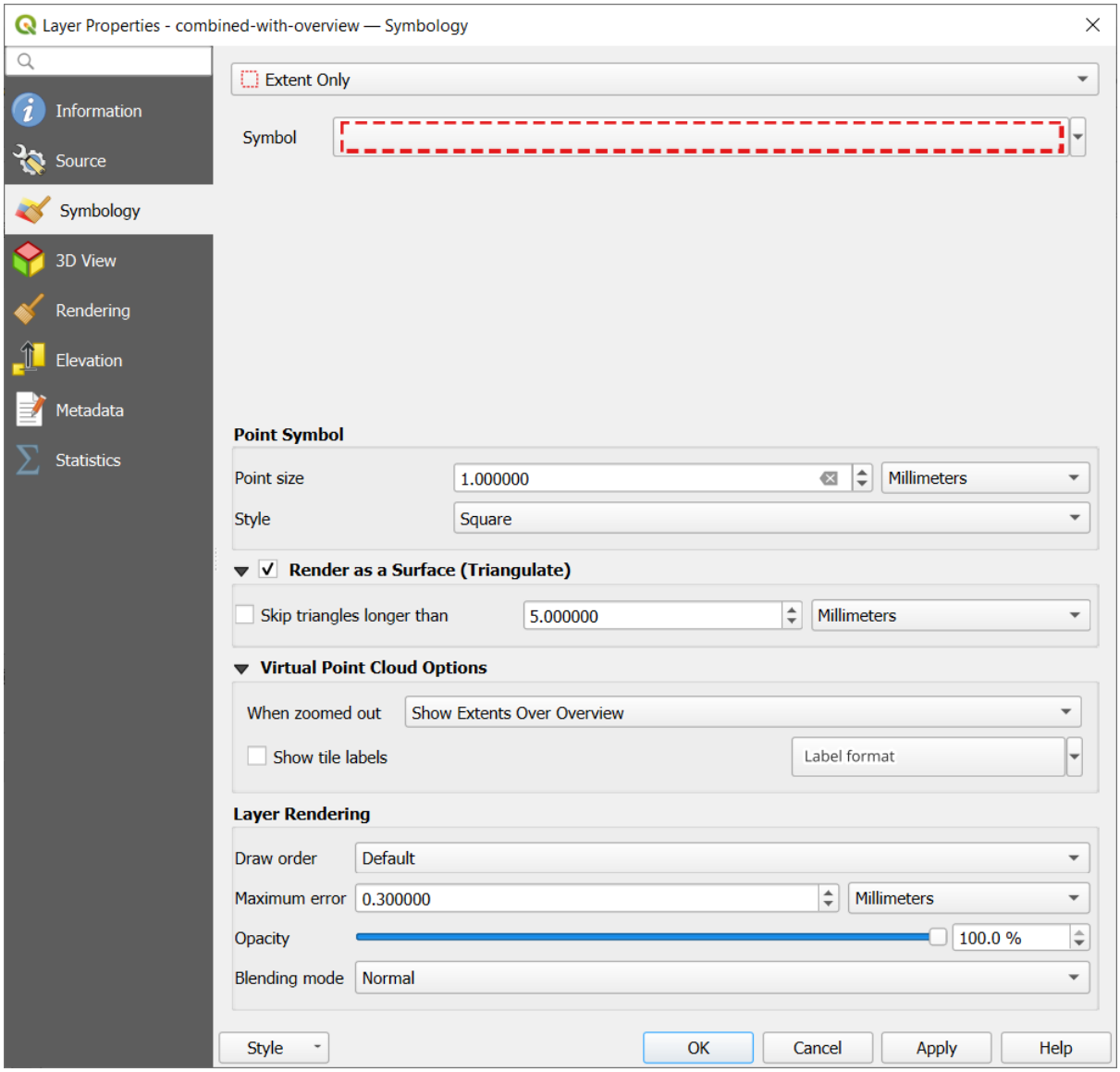
-  *Extent Only*: Only a bounding box of the extent of the data is displayed; convenient for overviewing the data extent. As usual, the *Symbol widget* helps you configure any properties (color, stroke, opacity, sub-layers, …) you'd like for the box.
-  *Attribute by Ramp*: The data is drawn over a color gradient. See *Attribute by Ramp Renderer*
-  *RGB*: Draw the data using red, green and blue color values. See *RGB Renderer*
-  *Classification*: The data is drawn using different colors for different classes. See *Classification Renderer*

When a point cloud is loaded, QGIS follows a logic to select the best renderer:

- if the dataset contains color information (red, green, blue attributes), the RGB renderer will be used
- else if the dataset contains a `Classification` attribute, the classified renderer will be used
- else it will fall back to rendering based on Z attribute

If you do not know the attributes of the point cloud, the  *Statistics tab* provides a good overview of which attributes are contained in the point cloud and in which ranges the values are located.

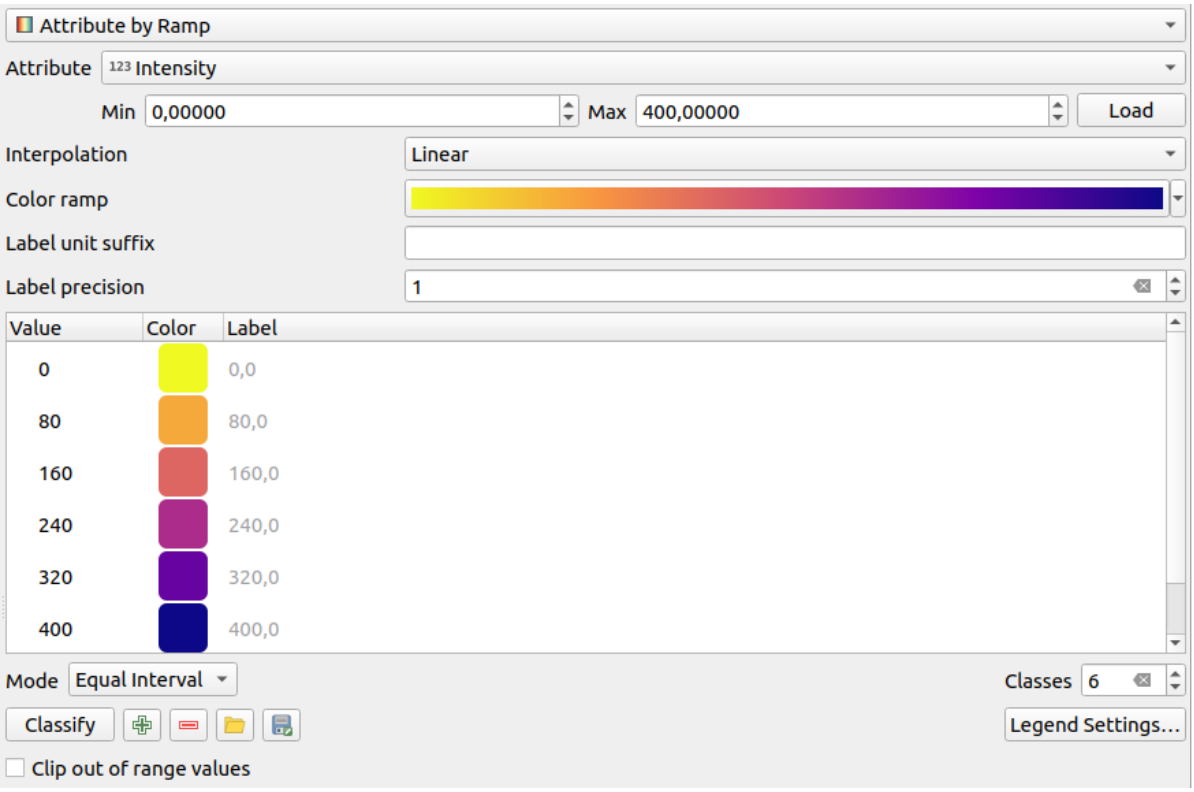
For each renderer, you can improve the data display adjusting the *point symbol size* or enabling *surface triangulation*.



Слика16.4: Point cloud symbology tab

Attribute by Ramp Renderer

With *Attribute by Ramp*, the data can be displayed by numerical values over a color gradient. Such numerical values can be, for example, an existing intensity attribute or the Z-value. Depending on a minimum and a maximum value, the other values are spread to the color gradient via interpolation. The distinct values and their assignment to a certain color are called „color map” and are shown in the table. There are various setting options, which are described below the figure.



Слика16.5: Point cloud symbology tab: Attribute by Ramp

- *Min* and *Max* define the range that is applied to the color ramp: the *Min* value represents the left, the *Max* value the right end of the color ramp, the values in between are interpolated. By default QGIS detects the minimum and the maximum from the selected attribute but they can be modified. Once you have changed the values, you can restore the defaults by clicking on the *Load* button.
- The *Interpolation* entry defines how values are assigned their color:
  - *Discrete* (a <= symbol appears in the header of the *Value* column): The color is taken from the closest color map entry with equal or higher value
  - *Linear* The color is linearly interpolated from the color map entries above and below the pixel value, meaning that to each dataset value corresponds a unique color
  - *Exact* (a = symbol appears in the header of the *Value* column): Only pixels with value equal to a color map entry are applied a color; others are not rendered.
- The *Color ramp* widget helps you select the color ramp to assign to the dataset. As usual with *this widget*, you can create a new one and edit or save the currently selected one.
- The *Label unit suffix* adds a label after the value in the legend, and the *Label precision* controls the number of decimals to display.

The classification *Mode* helps you define how values are distributed across the classes:

- *Continuous*: Classes number and color are fetched from the color ramp stops; limits values are set following stops distribution in the color ramp (you can find more information on stops in *Setting a Color Ramp*).
- *Equal interval*: The number of classes is set by the *Classes* field at the end of the line; limits values are defined so that the classes all have the same magnitude.

The classes are determined automatically and shown in the color map table. But you can also edit these classes manually:

- Double clicking in a *Value* in the table lets you modify the class value

- Double clicking in the *Color* column opens the *Color Selector* widget, where you can select a color to apply for that value
- Double clicking in the *Label* column to modify the label of the class
- Right-clicking over selected rows in the color table shows a contextual menu to *Change Color…* and *Change Opacity…* for the selection

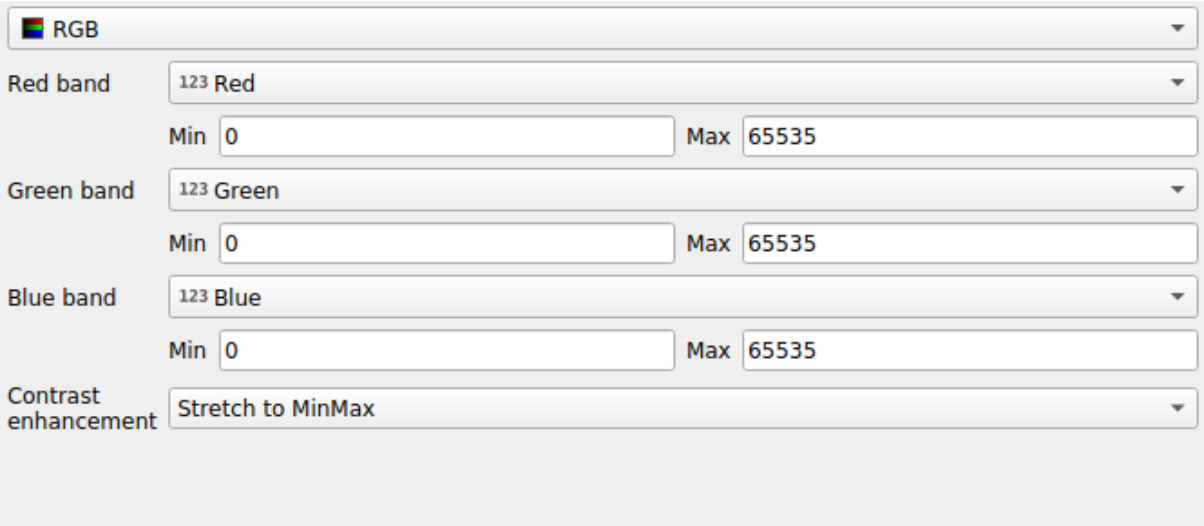
Below the table there are the options to restore the default classes with *Classify* or to manually Add values or Delete selected values from the table.

Since a customized color map can be very complex, there is also the option to Load an existing color map or to Save it for use in other layers (as a `txt` file).

If you have selected *Linear* for *Interpolation*, you can also configure:

- *Clip out of range values* By default, the linear method assigns the first class (respectively the last class) color to values in the dataset that are lower than the set *Min* (respectively greater than the set *Max*) value. Check this setting if you do not want to render those values.
- *Legend settings*, for display in the *Layers* panel and in the *layout legend*. Customization works the same way as with a raster layer (find more details at *Customize raster legend*).

RGB Renderer



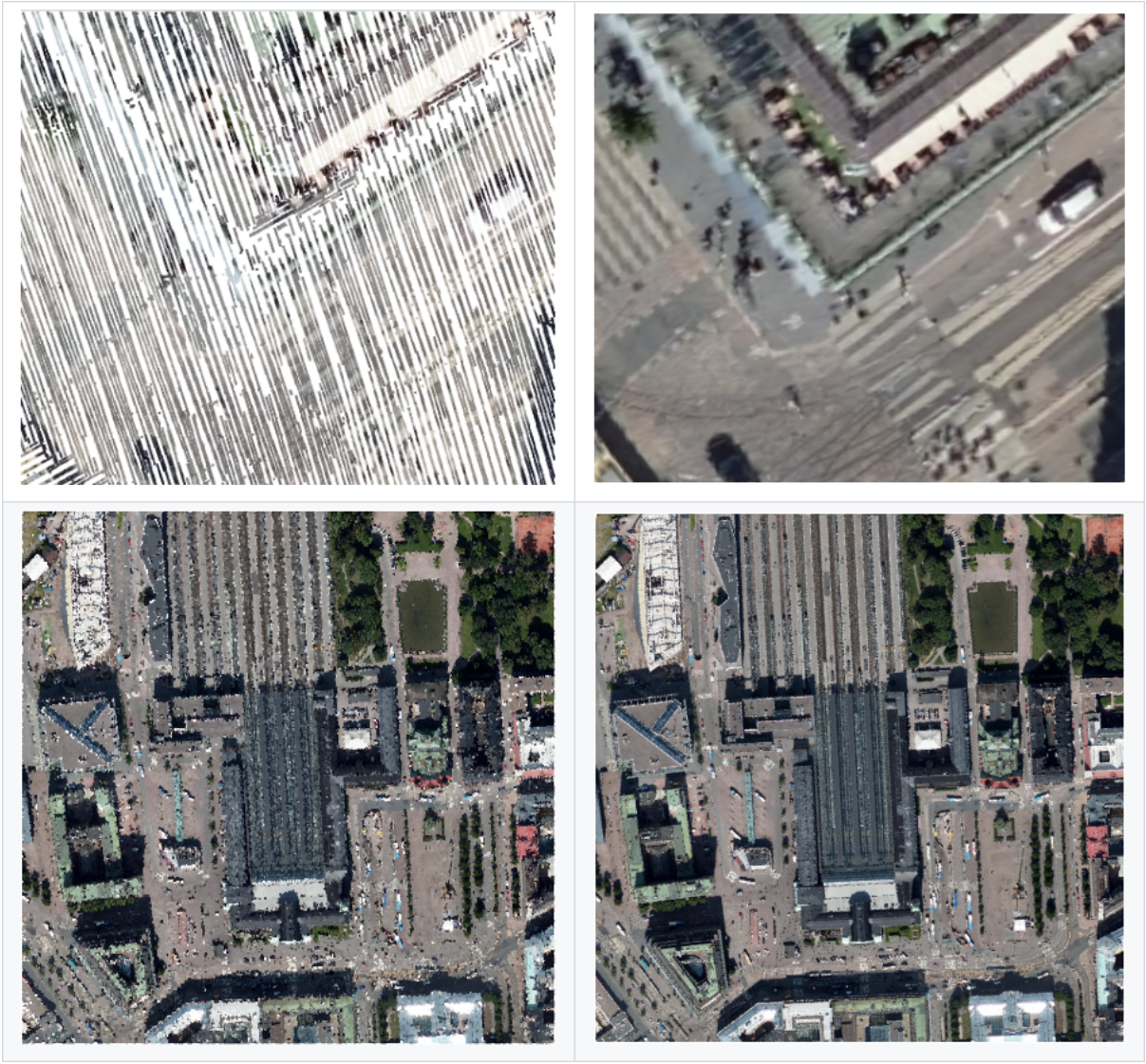
Слика16.6: The point cloud RGB renderer

With the *RGB* renderer, three selected attributes from the point cloud will be used as the red, green and blue component. If the attributes are named accordingly, QGIS selects them automatically and fetches *Min* and *Max* values for each band and scales the coloring accordingly. However, it is also possible to modify the values manually.

A *Contrast enhancement* method can be applied to the values: *No Enhancement*, *Stretch to MinMax*, *Stretch and Clip to MinMax* and *Clip to MinMax*

Белешка

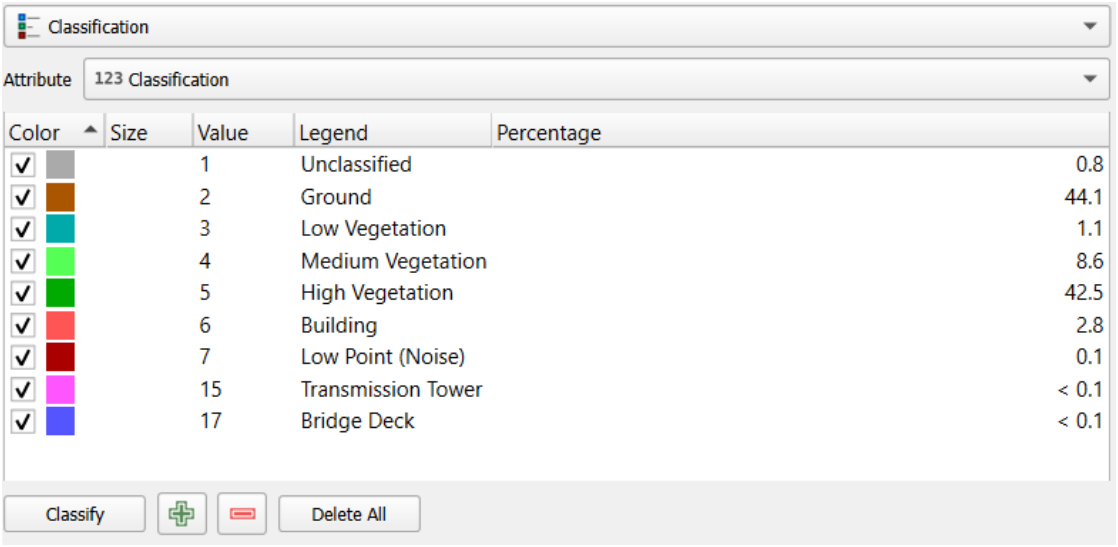
The *Contrast enhancement* tool is still under development. If you have problems with it, you should use the default setting *Stretch to MinMax*.



Слика16.7: Example of RGB renderer (left) combined with surface triangulation option (right)

Classification Renderer

In the *Classification* rendering, the point cloud is shown differentiated by color on the basis of an attribute. Any type of attribute can be used (numeric, string, …). Point cloud data often includes a field called `Classification`. This usually contains data determined automatically by post-processing, e.g. about vegetation. With *Attribute* you can select the field from the attribute table that will be used for the classification. By default, QGIS uses the definitions of the LAS specification (see table 'ASPRS Standard Point Classes' in the PDF on ASPRS home page). However, the data may deviate from this schema; in case of doubt, you have to ask the person or institution from which you received the data for the definitions.



Слика16.8: The point cloud classification renderer

In the table all used values are displayed with the corresponding color and legend. At the beginning of each row there is a check box; if it is unchecked, this value is no longer shown on the map. With double click in the table, you can modify:

- the *Color*: the *Color Selector* widget opens
- the *Size*: assigning a size of `0` to a category will revert it to use the default *point size* set for the layer
- the *Value*
- the *Legend*
- the *Percentage* column will show you the representation of the category within the layer.

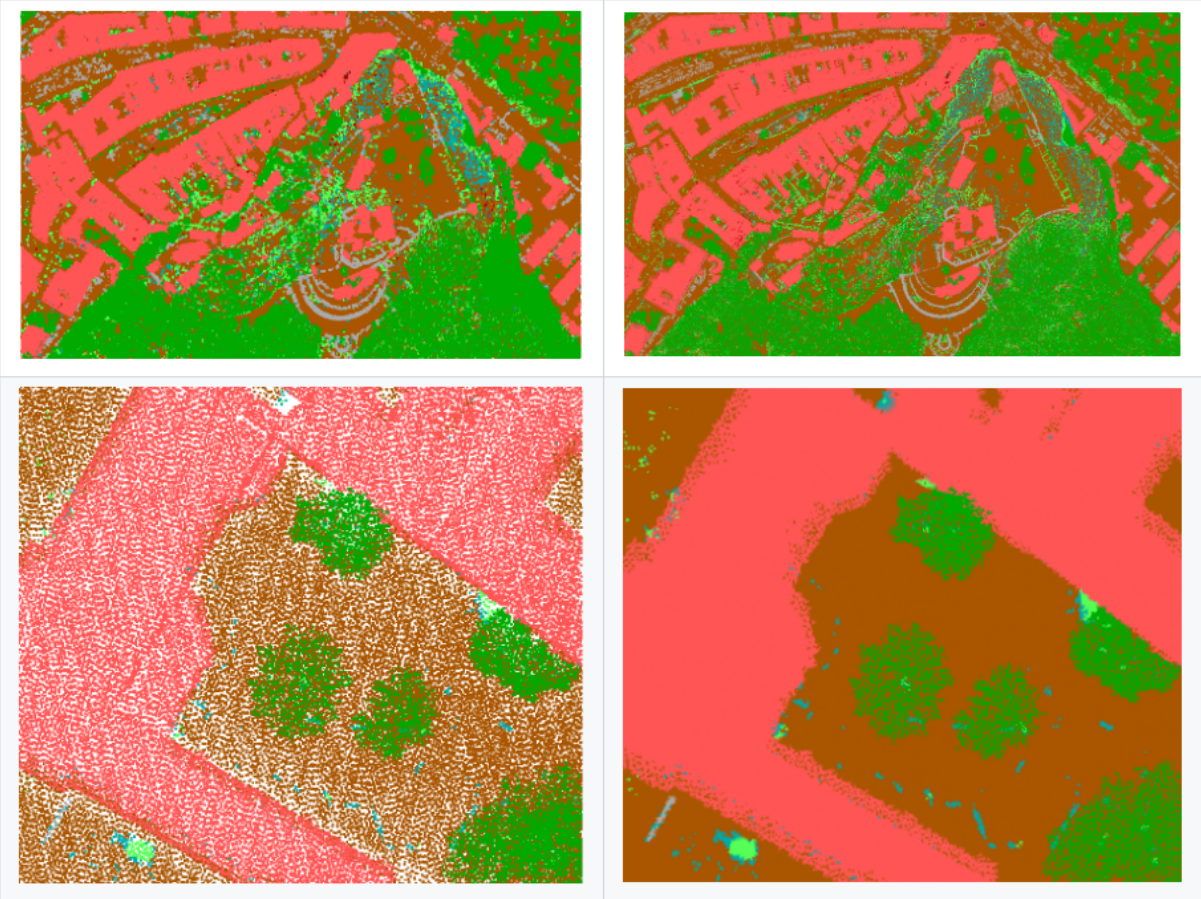
You can also right-click one or more rows to open a context menu with the options to change the color, opacity or point size.

Below the table there are buttons with which you can change the default classes generated by QGIS:

- With the *Classify* button the data can be classified automatically: all values that occur in the attributes and are not yet present in the table are added
- With Add and Delete, values can be added or removed manually
- *Delete All* removes all values from the table

Путоказ

In the *Layers* panel, you can right-click over a class leaf entry of a layer to quickly configure visibility of the corresponding features.



Слика16.9: Example of classification renderer (left) combined with surface triangulation option (right)

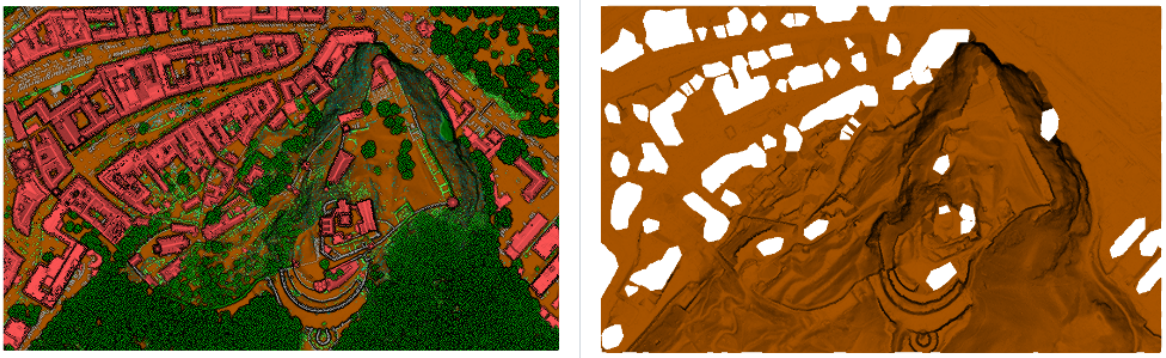
Point Symbol

Under *Point Symbol*, the size and the unit (e.g. millimeters, pixels, inches) with which each data point is displayed can be set. Either *Circle* or *Square* can be selected as the style for the points.

Render as a surface (Triangulate)

Check *Render as surface (Triangulate)* to enable the triangulation of the point cloud layer in the 2D view. This option allows rendering triangles instead of points. Each point keeps its color for interpolation in the triangle. You can control the horizontal length of computed triangles:

By checking the *Skip triangles longer than* option and setting up the threshold value, you can control the maximum length of a side of the triangles to consider in the horizontal plan. This can be particularly useful if you want to identify actual holes in the data.



Слика16.10: Rendering data as a surface with map shading (left) and with map shading, filtering large triangles (right)

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

In the *Layer Rendering* section you have the following options to modify the rendering of the layer:

- *Draw order*: allows to control whether point clouds rendering order on 2d map canvas should rely on their Z value. It is possible to render :
  - with the *Default* order in which the points are stored in the layer,
  - from *Bottom to top* (points with larger Z values cover lower points giving the looks of a true ortho photo),
  - or from *Top to bottom* where the scene appears as viewed from below.
- *Maximum error*: Point clouds usually contains more points than are needed for the display. By this option you set how dense or sparse the display of the point cloud will be (this can also be understood as 'maximum allowed gap between points'). If you set a large number (e.g. 5 mm), there will be visible gaps between points. Low value (e.g. 0.1 mm) could force rendering of unnecessary amount of points, making rendering slower (different units can be selected).
- *Opacity*: You can make the underlying layer in the map canvas visible with this tool. Use the slider to adapt the visibility of your layer to your needs. You can also make a precise definition of the percentage of visibility in the menu beside the slider.
- *Blending mode*: You can achieve special rendering effects with this tool. The pixels of your overlaying and underlying layers are mixed through the settings described in *Blending Modes*.
- *Eye dome lighting*: this applies shading effects to the map canvas for a better depth rendering. Rendering quality depends on the *draw order* property; the *Default* draw order may give sub-optimal results. Following parameters can be controlled:
  - *Strength*: increases the contrast, allowing for better depth perception
  - *Distance*: represents the distance of the used pixels off the center pixel and has the effect of making edges thicker.

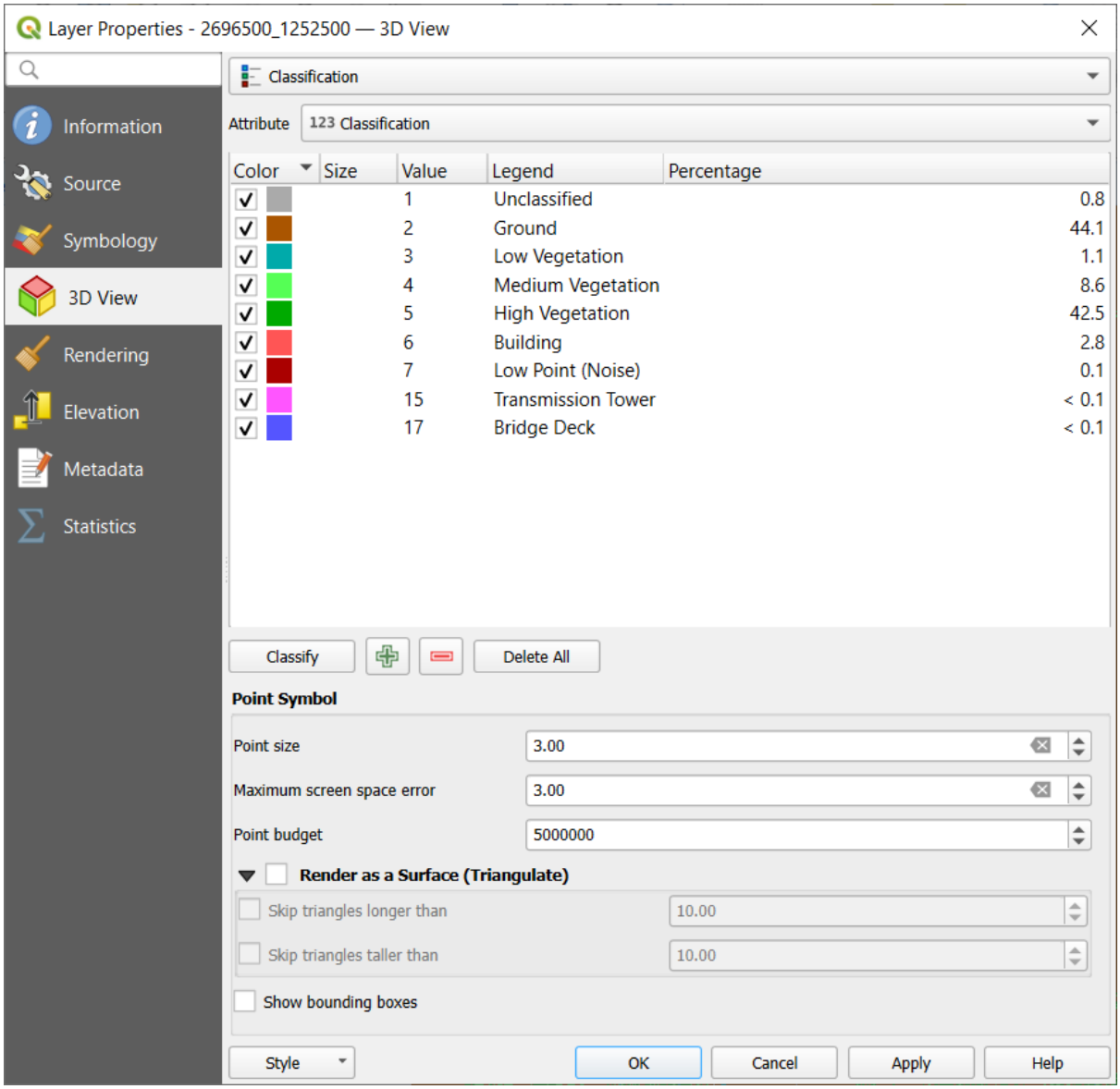
16.2.4 3D View Properties

In the  *3D View* tab you can make the settings for the rendering of the point cloud in 3D maps.

3D Rendering modes

Following options can be selected from the drop down menu at the top of the tab:

- *No Rendering*: Data are not displayed
- *Follow 2D Symbology*: Syncs features rendering in 3D with *symbology assigned in 2D*
-  *Single Color*: All points are displayed in the same *color* regardless of attributes
-  *Attribute by Ramp*: Interpolates a given attribute over a color ramp and assigns to features their matching color. See *Attribute by Ramp Renderer*.
-  *RGB*: Use different attributes of the features to set the Red, Green and Blue color components to assign to them. See *RGB Renderer*.
-  *Classification*: differentiates points by color on the basis of an attribute. See *Classification Renderer*.



Слика16.11: The point cloud 3D view tab with the classification renderer

3D Point Symbol

In the lower part of the *3D View* tab you can find the *Point Symbol* section. Here you can make general settings for the entire layer which are the same for all renderers. There are the following options:

- *Point size*: The size (in pixels) with which each data point is displayed can be set
- *Maximum screen space error*: By this option you set how dense or sparse the display of the point cloud will be (in pixels). If you set a large number (e.g. 10), there will be visible gaps between points; low value (e.g. 0) could force rendering of unnecessary amount of points, making rendering slower (you can find more details at *Symbology* *Maximum error*).
- *Point budget*: To avoid long rendering, you can set the maximum number of points that will be rendered
- Check *Render as surface (Triangulate)* to render the point cloud layer in the 3D view with a solid surface obtained by triangulation. You can control dimensions of the computed triangles:
  - *Skip triangles longer than* a threshold value: sets in the horizontal plan, the maximum length of a side of the triangles to consider

  - *Skip triangles taller than* a threshold value: sets in the vertical plan, the maximum height of a side of the triangles to consider

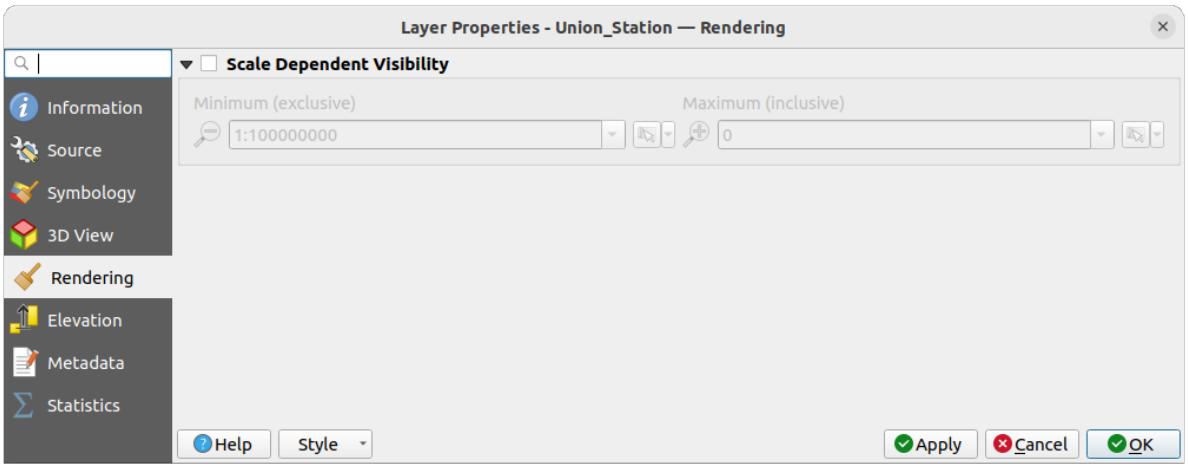
- *Show bounding boxes*: Especially useful for debugging, shows bounding boxes of nodes in hierarchy

16.2.5 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 features 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 *Visibility Scale Selector* for more information.

Белешка

You can also activate scale dependent visibility on a layer from within the *Layers* panel: right-click on the layer and in the contextual menu, select *Set Layer Scale Visibility*.



Слика16.12: The point cloud rendering tab

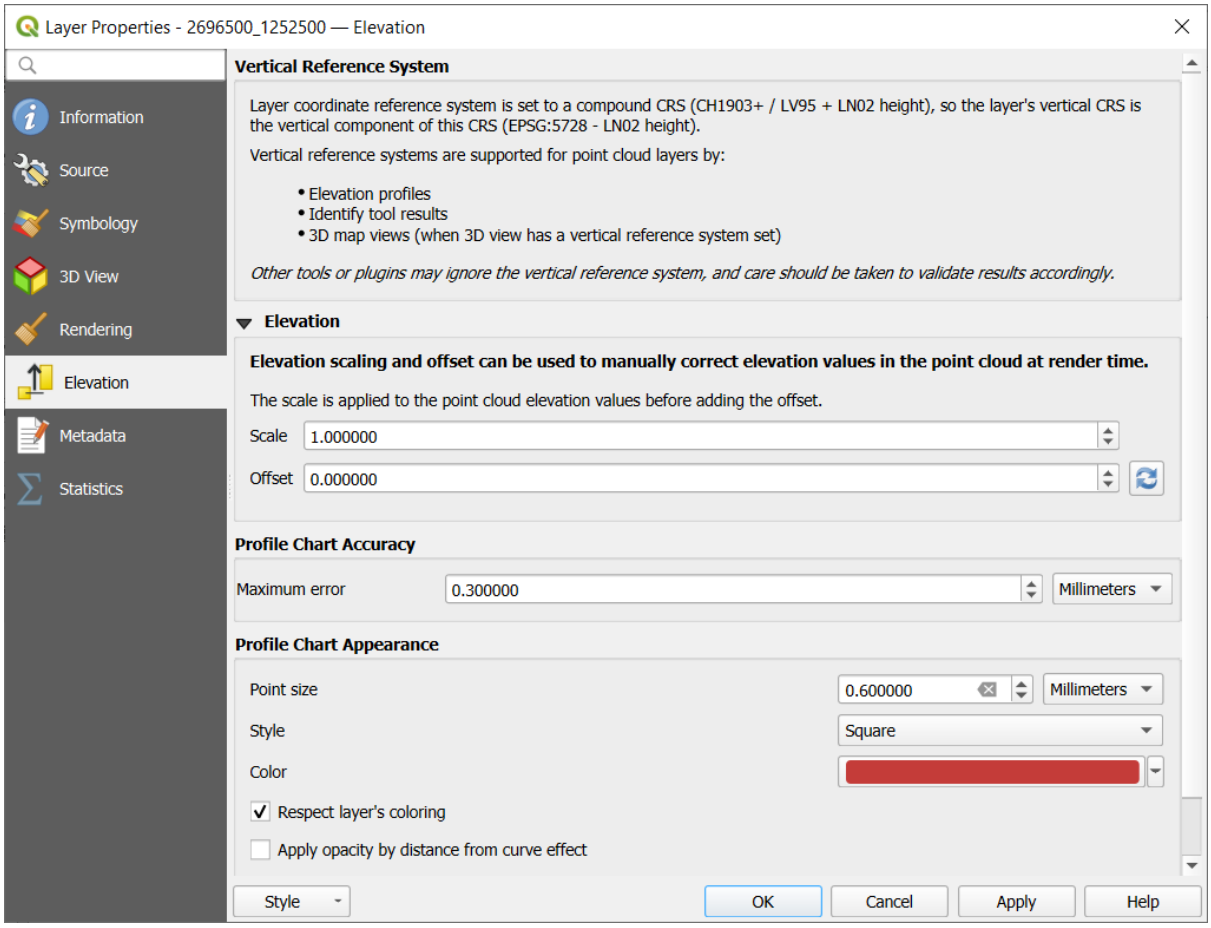
16.2.6 Elevation Properties

In the *Elevation* tab, you can set corrections for the Z-values of the data. This may be necessary to adjust the elevation of the data in 3D maps and its appearance in the *profile tool charts*. There are following options:

- *Vertical Reference System*: If the CRS of your point cloud layer is a compound one (including a Z dimension), then the vertical CRS used for the layer will be automatically derived from the vertical component of the layer's CRS. In this case, you cannot manually set a different vertical CRS, and the option to change it will be disabled. If your point cloud layer uses a horizontal (2D) CRS (though uncommon), you can manually select a specific vertical CRS by clicking on the Select CRS. Vertical reference systems are supported for point cloud layers in:
  - *Elevation profiles*
  - *Identify Tool Results*
  - *3D map views*
- Under *Elevation* group:
  - You can set a *Scale*: If `10` is entered here, a point that has a value Z = `5` is displayed at a height of `50`.

  - An *offset* to the z-level can be entered. This is useful to match different data sources in its height to each other. By default, the lowest z-value contained in the data is used as this value. This value can also be restored with the Refresh button at the end of the line.

- Under *Profile Chart Accuracy*, the *Maximum error* helps you control how dense or sparse the points will be rendered in the elevation profile. Larger values result in a faster generation with less points included.
- Under *Profile Chart Appearance*, you can control the point display:
  - *Point size*: the size to render the points with, in supported units (millimeters, map units, pixels, …)
  - *Style*: whether to render the points as *Circle* or *Square*
  - Apply a single *Color* to all the points visible in the profile view
  - Check *Respect layer's coloring* to instead show the points with the color assigned via their *2D symbology*
  - *Apply opacity by distance from curve effect*, reducing the opacity of points which are further from the profile curve



Слика16.13: The point cloud elevation tab

16.2.7 Metadata Properties

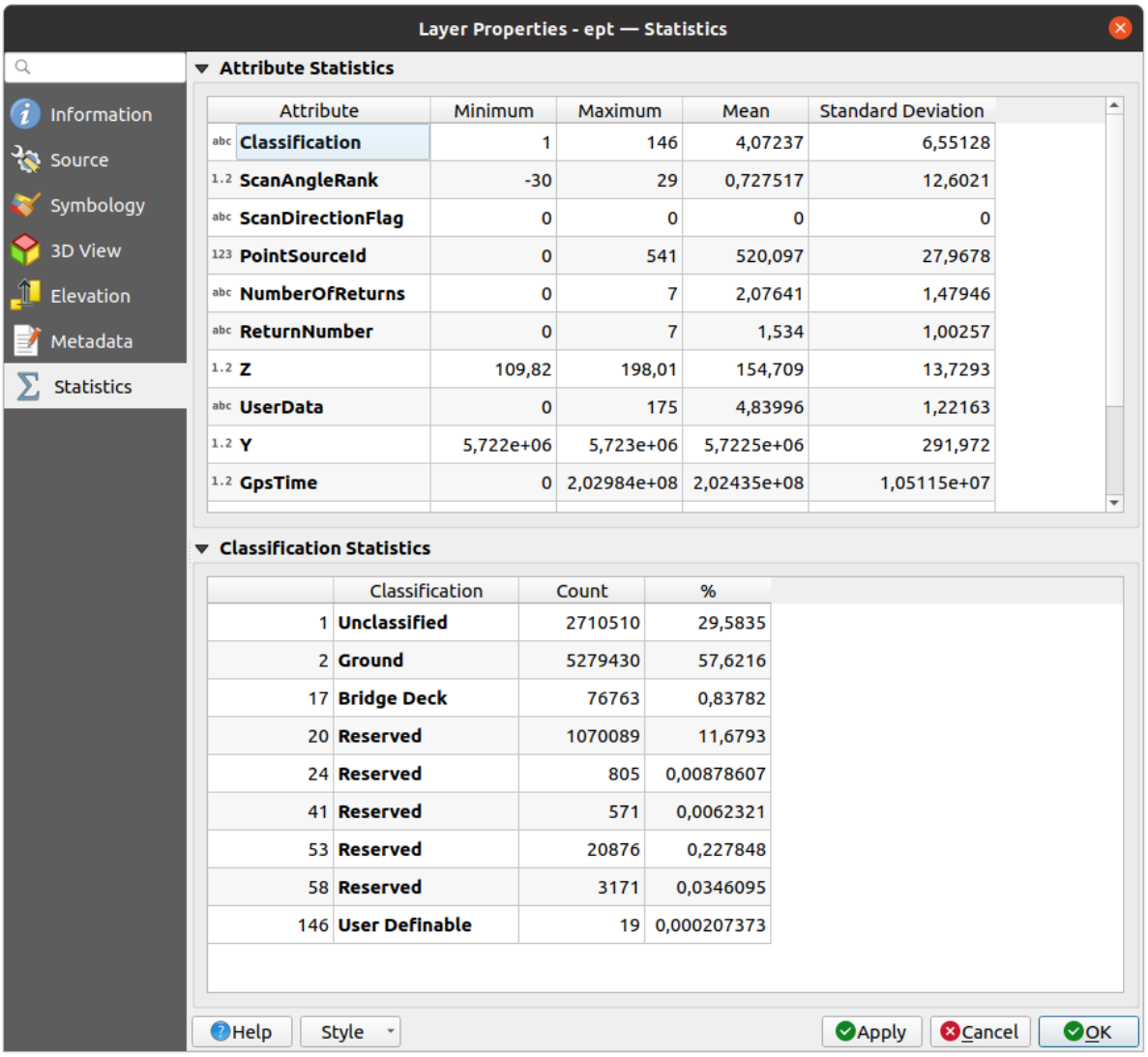
The *Metadata* tab provides you with options to create and edit a metadata report on your layer. See *Metadata* for more information.

16.2.8 Statistics Properties

In the Σ *Statistics* tab you can get an overview of the attributes of your point cloud and their distribution.

At the top you will find the section *Attribute Statistics*. Here all attributes contained in the point cloud are listed, as well as some of their statistical values: *Minimum*, *Maximum*, *Mean*, *Standard Deviation*

If there is an attribute *Classification*, then there is another table in the lower section. Here all values contained in the attribute are listed, as well as their absolute *Count* and relative % abundance.



Слика16.14: The point cloud statistics tab

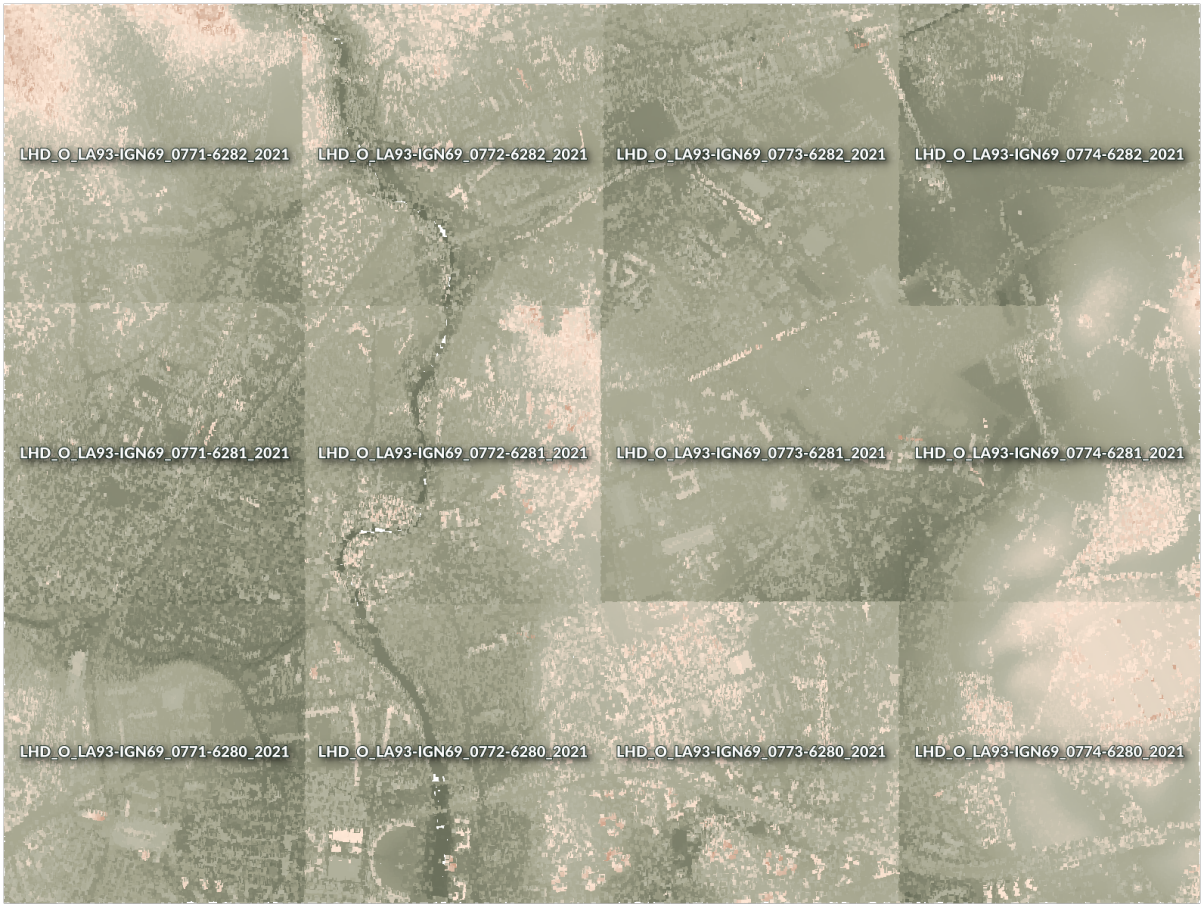
16.3 Virtual point cloud

Lidar surveys of larger areas are often multi-terabyte datasets with many billions of points. Having such large datasets represented as a single point cloud file is not practical due to the difficulties of storage, transfer, display and analysis. Point cloud data are therefore typically stored and distributed split into square tiles (e.g. `1km x 1km`), each tile having a more manageable file size (e.g. ~200 MB when compressed).

Tiling of data solves the problems with size of data, but it introduces issues when processing or viewing an area of interest that does not fit entirely into a single tile. Users need to develop workflows that take into account multiple tiles and special care needs to be taken to deal with data near edges of tiles to avoid unwanted artefacts in outputs.

Similarly, when viewing point cloud data, it becomes cumbersome to load many individual files and apply the same symbology.

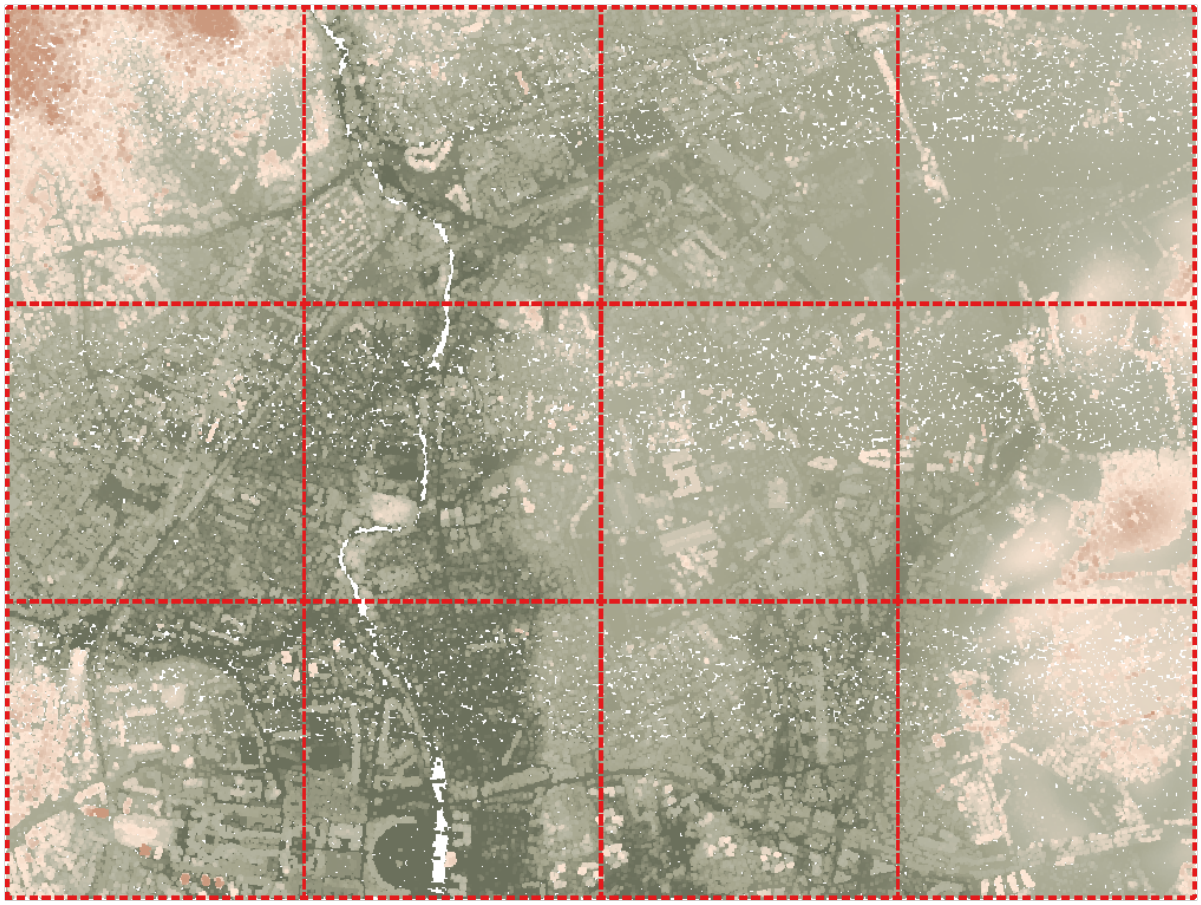
Here is an example of several point cloud tiles loaded in QGIS. Each tile is styled based on min/max Z values of the tile, creating visible artefacts on tile edges. The styling has to be adjusted for each layer separately:



Слика16.15: Individual point cloud tiles loaded, with artefacts on edges

In the GIS world, many users are familiar with the concept of virtual rasters. A virtual raster is a file that simply references other raster files with actual data. In this way, GIS software then treats the whole dataset comprising many files as a single raster layer, making the display and analysis of all the rasters listed in the virtual file much easier.

Borrowing the concept of virtual rasters from GDAL, **virtual point cloud (VPC)** is a file format that references other point cloud files. Software supporting virtual point clouds handles the whole tiled dataset as a single data source.



Слика16.16: The virtual point cloud

Displaying and manipulating virtual point cloud is much more fluent and easy.

At the core, a virtual point cloud file is a simple JSON file with `.vpc` extension, containing references to actual data files (e.g. `.LAS`, `.LAZ` or `.COPC` files) and additional metadata extracted from the files. Even though it is possible to write VPC files by hand, it is strongly recommended to create them using an automated tool, such as:

- The Processing *Napravi virtuelni oblak tačaka (VPC)* algorithm
- The `build_vpc` command of PDAL wrench tool

For more details, please refer to the VPC specification that also contains best practices and optional extensions (such as overviews).

ГЛАВА 17

Working with 3D Tiles

17.1 What are 3D Tiles?

3D tiles are specification for streaming and rendering large-scale 3D geospatial datasets. They use a hierarchical structure to efficiently manage and display 3D content, optimising performance by dynamically loading appropriate levels of detail. This technology is widely used in urban planning, architecture, simulation, gaming, and virtual reality, providing a standardised and interoperable solution for visualising complex geographical data.



Слика17.1: Example of 3D tiles

See *Using 3D tiled scene services* for instructions on how to add these data sources to QGIS.

17.1.1 State of 3D Tiles Support

Currently, QGIS supports two formats of 3D tiles:

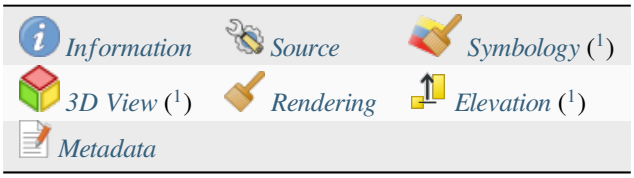
- „Cesium" 3D Tiles are used primarily for complex 3D models of buildings or whole cities. Such datasets can be provided by cloud-based platforms such as Cesium Ion or Google (*Photorealistic 3D Tiles*).
- Quantized Mesh tiles are used for terrain elevation data.

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 Properties

The 3D tiles *Layer Properties* dialog provides the following sections:



17.2.1 Information Properties

The *Information* tab is read-only and represents an interesting place to quickly grab summarized information and metadata on the current layer. Provided information are:

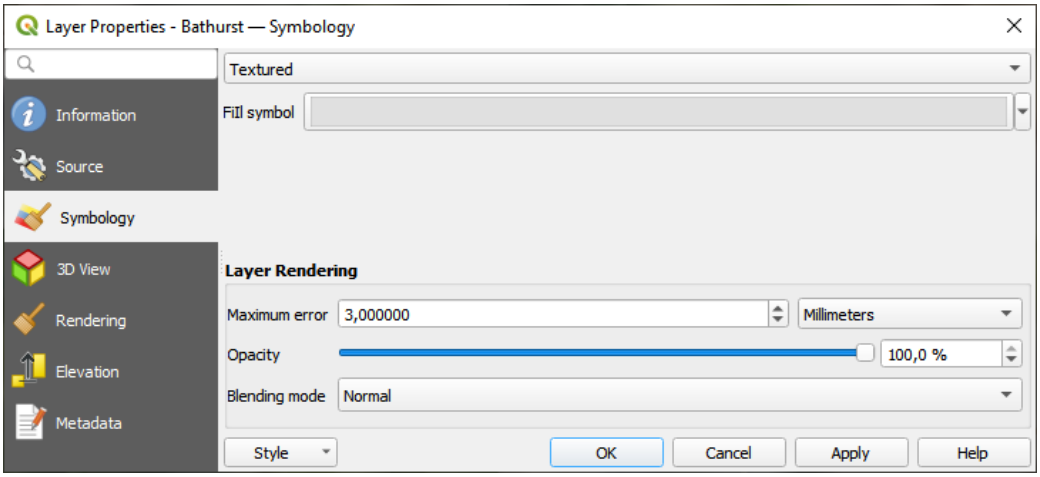
- based on the provider of the layer: name, URL, source type and path, number of zoom levels
- custom properties, used to store in the active project additional information about the layer. More properties can be created and managed using PyQGIS, specifically through the setCustomProperty() method.
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e. whether it's static or dynamic)
- picked from the *filled metadata*: access, extents, links, contacts, history…

17.2.2 Source Properties

The *Source* tab displays basic information about the selected 3D tile, including:

- the *Layer name* to display in the *Layers Panel*;
- the *Coordinate Reference System*: Displays the layer's *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 the Select CRS button (see *Coordinate Reference System Selector*). Use this process only if the layer CRS is wrong or not specified.

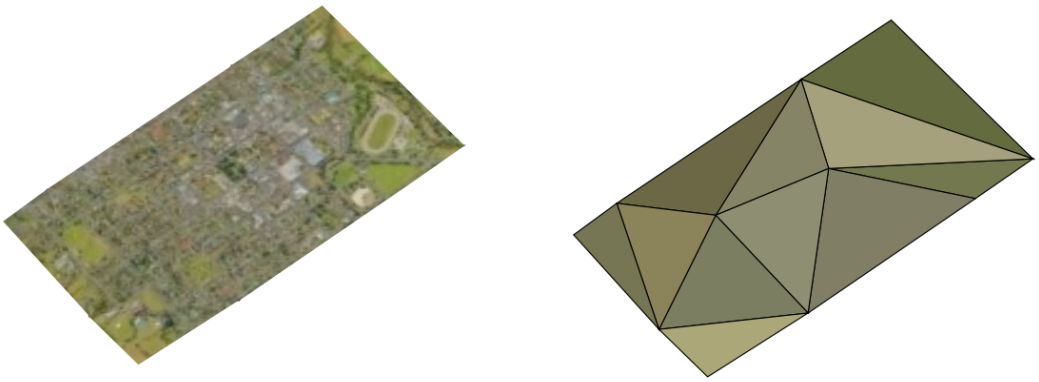
17.2.3 Symbology Properties



Слика17.2: 3D Tile Layer Symbology

By default, the layer is styled using texture, but you can change it to see the wireframe mesh behind the scene by choosing *Wireframe* in the drop-down menu. You can also, change the mesh fill and line symbols similar to the vector polygons. Checking *Use texture colors* will render each mesh element with the average value of the full texture. This is a good option to try when dealing with a large dataset and want to get a quick overview of the data.

[1] Also available in the *Layer styling panel*



Слика17.3: 3D Tiles - textured and wireframe

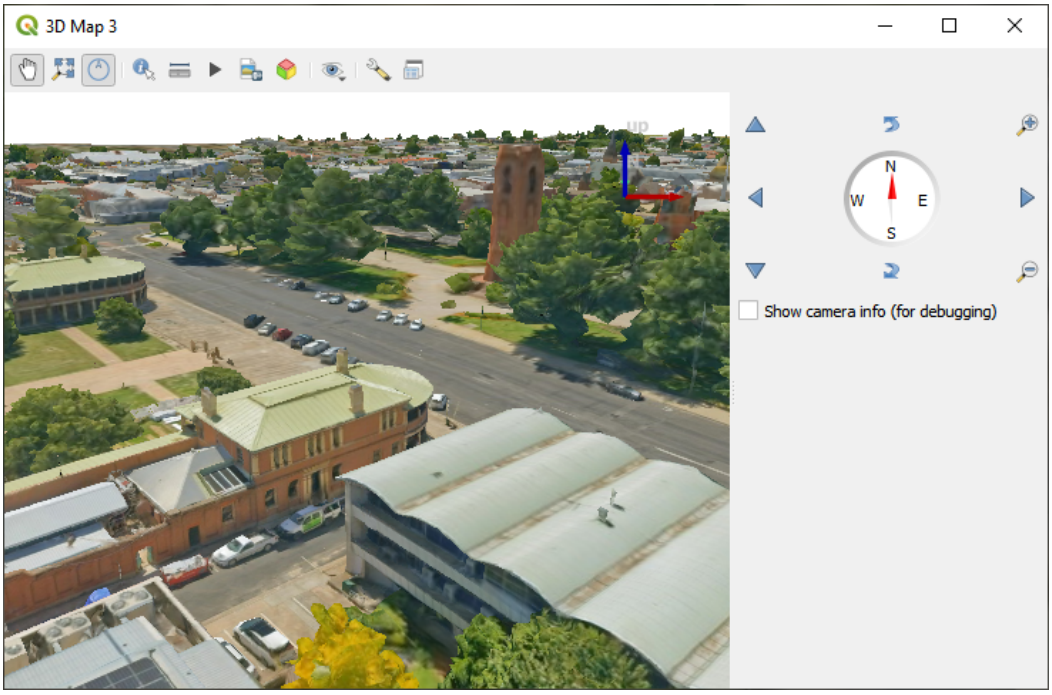
From the *Symbology* tab, you can also set some options that invariably act on all features of the layer:

- *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.
- *Blending mode*: You can achieve special rendering effects with these tools that you may previously only know from graphics programs. The pixels of your overlaying and underlying layers are mixed through the settings described in *Blending Modes*.

17.2.4 3D View Properties

- *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*.



Слика17.4: 3D Map View

17.2.5 Rendering Properties

Under *Scale dependent visibility*, you can set the *Maximum (inclusive)* and *Minimum (exclusive)* scales, defining a range of scales in which features 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 *Visibility Scale Selector* for more information.

17.2.6 Elevation Properties

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 Metadata Properties

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

ГЛАВА 18

Postavljanje mapa

Pomoću „Print Layouts and Reports“ možete kreirati mape i atlase i štampati ih ili sačuvati kao slike, PDF ili SVG datoteke.

18.1 Pregled rasporeda za štampu

Raspored za štampu pruža sve bogatije mogućnosti raspoređivanja i štampanja. Omogućava vam da dodate elemente kao što su QGIS 2D ili 3D platno karte, tekstualni natpisi, slike, legende, razmernici, osnovni oblici, strelice, atributske i jednostavne tabele, visinski profili i HTML okviri… Svaki element možete dimenzionisati, grupisati, poravnati, pozicionirati i rotirati i podesiti njegova svojstva da biste napravili svoj raspored.

Raspored se može odštampati ili izvesti u formate slike, PostScript, PDF ili SVG. Raspored možete sačuvati kao šablon i ponovo ga učitati u drugoj sesiji. Najzad, generisanje više karata na osnovu šablona može se obaviti pomoću generatora atlasa.

18.1.1 Primer sesije za početnike

Pre nego što počnete da radite sa rasporedom za štampu, potrebno je da učitate neke rasterske ili vektorske slojeve u QGIS platno karte i prilagodite njihova svojstva prema sopstvenim potrebama. Kada je sve iscrtano i simbolizovano po vašoj želji, kliknite na ikonu Novi raspored za štampu u paleti alatki *Projekat* ili izaberite *Projekat* ► *Novi raspored za štampu*. Biće vam zatraženo da izaberete naslov za novi raspored.

Da bismo pokazali kako se pravi karta, pratite sledeća uputstva.

1. Sa leve strane izaberite dugme palete alatki Dodaj kartu i nacrtajte pravougaonik na platnu držeći pritisnut levi taster miša. Unutar nacrtanog pravougaonika prikazaće se QGIS prikaz karte na platnu.

2. Izaberite dugme palete alatki Dodaj razmernik i kliknite levim tasterom miša na platno rasporeda za štampu. Razmernik će biti dodat na platno.

3. Izaberite dugme palete alatki Dodaj legendu i nacrtajte pravougaonik na platnu držeći pritisnut levi taster miša. Unutar nacrtanog pravougaonika biće iscrtana legenda.

4. Izaberite ikonu Izaberi/pomeri stavku da biste izabrali kartu na platnu i malo je pomerili.

5. Dok je stavka karte i dalje izabrana, možete promeniti i veličinu stavke karte. Kliknite držeći pritisnut levi taster miša na beli pravougaonik u jednom od uglova stavke karte i prevucite ga na novu lokaciju da biste promenili njenu veličinu.
6. Kliknite na panel *Svojstva stavke* u donjem levom delu i pronađite podešavanje za orijentaciju. Promenite vrednost podešavanja *Orijentacija karte* na „15.00°“. Trebalo bi da vidite kako se orijentacija stavke karte menja.
7. Sada možete odštampati ili izvesti svoj raspored za štampu u formate slike, PDF ili SVG pomoću alatki za izvoz u meniju *Raspored*.
8. Najzad, možete sačuvati svoj raspored za štampu unutar datoteke projekta pomoću dugmeta Sačuvaj projekat.

Možete dodati više elemenata u raspored za štampu. Takođe je moguće imati više od jednog prikaza karte ili legende ili razmernika na platnu rasporeda za štampu, na jednoj ili više strana. Svaki element ima svoja svojstva i, u slučaju karte, svoj obuhvat. Ako želite da uklonite bilo koje elemente sa platna rasporeda, to možete uraditi tasterom `Delete` ili `Backspace`.

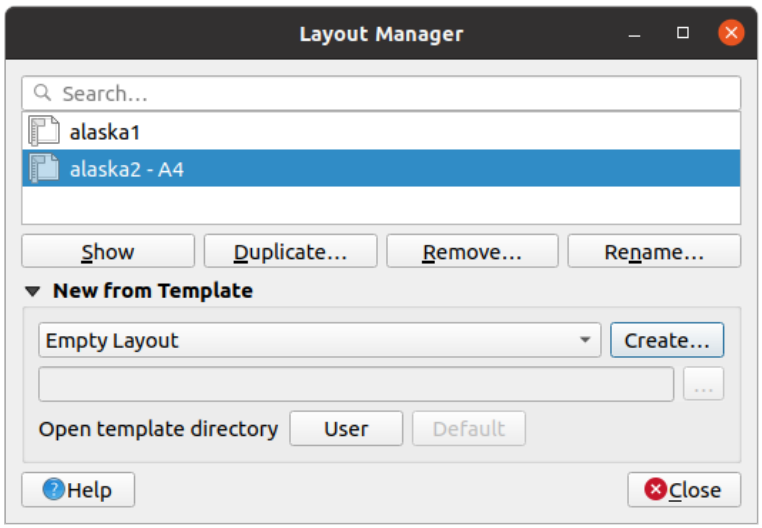
18.1.2 Menadžer rasporeda

Prozor *Menadžer rasporeda* je glavni prozor za upravljanje rasporedima za štampu u projektu. Daje vam pregled postojećih rasporeda za štampu i izveštaja u projektu i nudi alatke za:

- pretragu rasporeda;
- dodavanje novog rasporeda za štampu ili novog izveštaja od nule, iz šablona ili dupliranjem postojećeg;
- preimenovanje ili brisanje bilo kog od njih;
- njihovo otvaranje u projektu.

Da biste otvorili dijalog menadžera rasporeda:

- iz glavnog QGIS dijaloga izaberite meni *Projekat* ► *Menadžer rasporeda…* ili kliknite na dugme Menadžer rasporeda u *paleti alatki Projekat*;
- iz dijaloga rasporeda za štampu ili izveštaja izaberite meni *Raspored* ► *Menadžer rasporeda…* ili kliknite na dugme Menadžer rasporeda u *paleti alatki Raspored*.



Слика18.1: Menadžer rasporeda za štampu

Menadžer rasporeda u svom gornjem delu navodi sve dostupne rasporede za štampu ili izveštaje u projektu, sa alatkama za:

- prikaz izbora: možete izabrati više izveštaja i/ili rasporeda za štampu i otvoriti ih jednim klikom. Dvoklik na naziv takođe ga otvara;
- dupliranje izabranog rasporeda za štampu ili izveštaja (dostupno samo ako je izabrana jedna stavka): pravi novi dijalog koristeći izabrani kao šablon. Biće vam zatraženo da izaberete novi naslov za novi raspored;
- preimenovanje izveštaja ili rasporeda (dostupno samo ako je izabrana jedna stavka): biće vam zatraženo da izaberete novi naslov za raspored;
- uklanjanje rasporeda: izabrani rasporedi za štampu biće obrisani iz projekta.

U donjem delu moguće je praviti nove rasporede za štampu ili izveštaje od nule ili iz šablona. Podrazumevano će QGIS tražiti šablone u folderu *korisničkog profila* i u direktorijumima šablona aplikacije (dostupnim pomoću dva dugmeta na dnu okvira), ali i u bilo kom folderu deklarisanom kao *Putanja(e) za traženje dodatnih šablona za štampu* u *Podešavanja* ► *Opcije* ► *Rasporedi*. Pronađeni šabloni se navode u padajućoj listi. Izaberite stavku i pritisnite dugme *Napravi* da biste generisali novi izveštaj ili raspored za štampu.

Možete koristiti i šablone rasporeda iz prilagođenog foldera; u tom slučaju izaberite *specific* u padajućoj listi šablona, pronađite šablon i pritisnite *Napravi*.

Савет

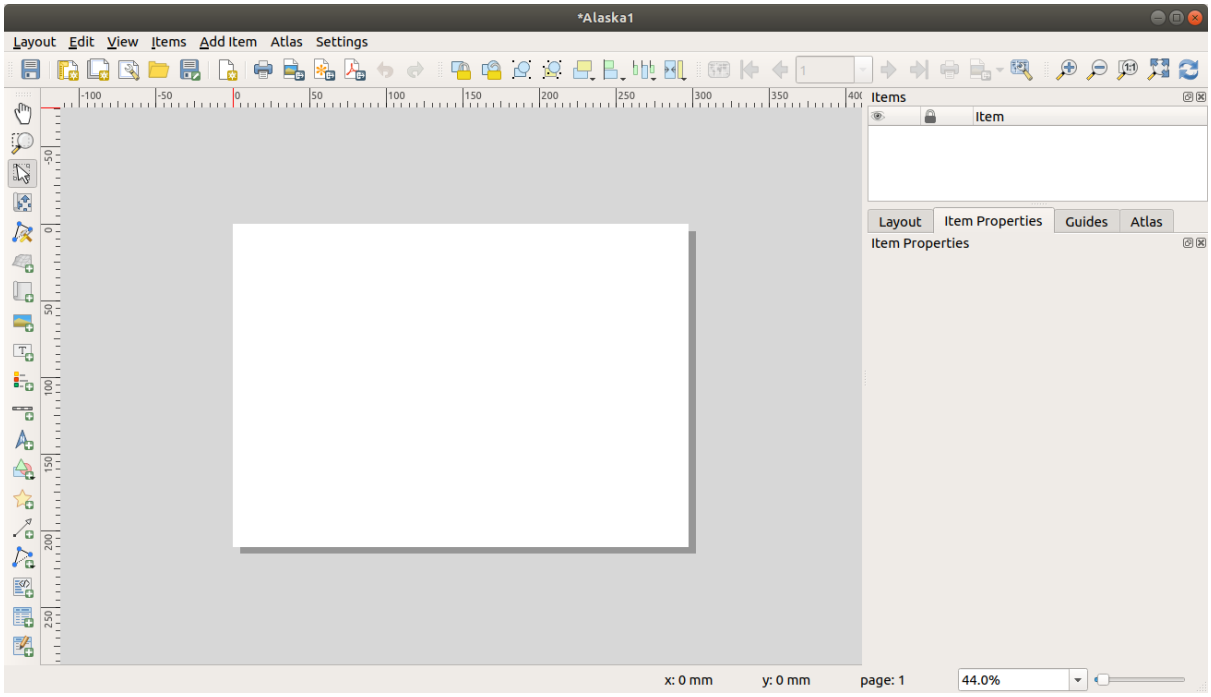
Pravljenje rasporeda za štampu na osnovu šablona iz panela Pregledač

Prevlačenje datoteke šablona rasporeda za štampu `.qpt` iz bilo kog pregledača datoteka na platno karte ili dvoklik na nju u *panelu Pregledač* generiše novi raspored za štampu iz šablona.

18.1.3 Meniji, alatke i paneli rasporeda za štampu

Otvaranje rasporeda za štampu pruža vam prazno platno koje predstavlja površinu papira pri korišćenju opcije štampanja. Isprva sa leve strane pored platna nalazite dugmad za dodavanje stavki rasporeda za štampu: 2D ili 3D platna karte, tekstualni natpisi, slike, legende, razmernici, osnovni oblici, strelice, atributske ili jednostavne tabele, HTML okviri, visinski profili… U ovoj paleti alatki takođe nalazite dugmad za navigaciju, uveličavanje oblasti i pomeranje prikaza na rasporedu, kao i dugmad za izbor bilo koje stavke rasporeda i za pomeranje obuhvata stavki karte.

Сл.18.2 prikazuje početni prikaz rasporeda za štampu pre nego što se dodaju bilo koji elementi.



Слика18.2: Raspored za štampu

Sa desne strane pored platna nalazite dva skupa panela. Gornji sadrži panele *Stavke* i *Istorija opoziva*, a donji sadrži panele *Raspored*, *Svojstva stavke* i *Generisanje atlasa*.

- Panel *Stavke* pruža listu svih stavki rasporeda za štampu dodatih na platno i načine za globalnu interakciju sa njima (za više informacija pogledajte *Panel Stavke*).
- Panel *Istorija opoziva* prikazuje istoriju svih izmena primenjenih na raspored. Klikom miša moguće je opozvati i ponoviti korake rasporeda napred i nazad do određenog stanja.
- Panel *Raspored* omogućava vam da postavite opšte parametre koji se primenjuju na raspored pri izvozu ili radu unutar njega (za više detalja pogledajte *Panel Raspored*);
- Panel *Svojstva stavke* prikazuje svojstva izabrane stavke. Kliknite na ikonu  Izaberi/pomeri stavku da biste izabrali stavku (npr. legendu, razmernik ili natpis) na platnu. Zatim kliknite na panel *Svojstva stavke* i prilagodite podešavanja za izabranu stavku (za detaljne informacije o podešavanjima svake stavke pogledajte *Stavke rasporeda*).
- Panel *Atlas* omogućava vam da omogućite generisanje atlasa za trenutni raspored i daje pristup njegovim parametrima (za detaljne informacije o korišćenju generisanja atlasa pogledajte *Generate an Atlas*).

U donjem delu prozora rasporeda za štampu nalazite statusnu traku sa položajem miša, brojem trenutne strane, padajućom listom za postavljanje nivoa uvećanja, brojem izabranih stavki ako je primenljivo i, u slučaju generisanja atlasa, brojem objekata.

U gornjem delu prozora rasporeda za štampu nalazite menije i druge palete alatki. Sve alatke rasporeda za štampu dostupne su u menijima i kao ikone u paleti alatki.

Palete alatki i paneli mogu se isključiti i uključiti desnim tasterom miša iznad bilo koje palete alatki, ili preko *Prikaz* ► *Palete alatki* ► ili *Prikaz* ► *Paneli* ►.

Meniji i alatke

Meni Raspored

Raspored pruža radnje za upravljanje rasporedom:

- Sačuvajte datoteku projekta direktno iz prozora rasporeda za štampu.

- Napravite novi i prazan raspored za štampu pomoću *Novi raspored…*.
- *Dupliraj raspored…* : Napravite novi raspored za štampu dupliranjem trenutnog.
- Uklonite trenutni raspored pomoću *Obriši raspored…*.
- Otvorite *Menadžer rasporeda…*.
- *Rasporedi* ► : Otvorite postojeći raspored za štampu.

Kada je raspored napravljen, pomoću ikona *Sačuvaj kao šablon* i *Dodaj stavke iz šablona*, možete sačuvati trenutno stanje sesije rasporeda za štampu kao datoteku šablona `.qpt` i ponovo učitati njene stavke u drugoj sesiji/rasporedu za štampu.

U meniju *Raspored* postoje i moćni načini za deljenje geografskih informacija napravljenih pomoću QGIS-a koje se mogu uključiti u izveštaje ili objaviti. Ove alatke su *Izvezi kao sliku…*, *Izvezi kao PDF…*, *Izvezi kao SVG…* i *Štampaj…*.

Ispod je lista svih dostupnih alatki u ovom meniju sa nekim korisnim informacijama.

Табела18.1: Alatke menija „Raspored“

Alatka	Prečica	Traka sa alatkama	Referenca
Sačuvaj projekat	`Ctrl+S`	*Raspored*	*Introducing QGIS projects*
Novi raspored	`Ctrl+N`	*Raspored*	*Menadžer rasporeda*
Dupliraj raspored		*Raspored*	*Menadžer rasporeda*
Obriši raspored			
Menadžer prikaza…		*Raspored*	*Menadžer rasporeda*
Prikazi ►			
Svojstva rasporeda…			*Panel Raspored*
Preimenuj raspored…			
Svojstva strane…			*Rad sa svojstvima strane*
Dodaj strane…		*Raspored*	*Rad sa svojstvima strane*
Dodaj stavke iz šablona		*Raspored*	*Pravljenje stavke rasporeda*
Sačuvaj kao šablon…		*Raspored*	*Menadžer rasporeda*
Izvezi kao sliku…		*Raspored*	*Export as Image*
Izvezi kao SVG…		*Raspored*	*Export as SVG*
Izvezi kao PDF…		*Raspored*	*Export as PDF*
Podešavanje strane štampača…	`Ctrl+Shift+P`		
Štampaj…	`Ctrl+P`	*Raspored*	*Creating an Output*
Zatvori	`Ctrl+Q`		

Meni Uređivanje

Meni *Uređivanje* nudi alatke za manipulaciju stavkama rasporeda za štampu. Uključuje uobičajene radnje kao što su alatke za izbor, Kopiraj/Iseci/Nalepi i funkcionalnost opoziva/ponavljanja (pogledajte *Panel Istorija opoziva: radnje opoziva i ponavljanja*) za stavke u rasporedu.

Pri korišćenju radnje Nalepi, elementi će biti nalepljeni prema trenutnom položaju miša. Korišćenje radnje *Uređivanje* ► *Nalepi na mesto* ili pritiskanje `Ctrl+Shift+V` nalepiće stavke u trenutnu stranu, na isti položaj na kojem su bile na svojoj početnoj strani. Time se obezbeđuje kopiranje/lepljenje stavki na isto mesto, sa strane na stranu.

Ispod je lista svih dostupnih alatki u ovom meniju sa nekim korisnim informacijama.

Табела18.2: Alatke menija „Uređivanje“

Alatka	Prečica	Traka sa alatka-ma	Referenca
Opozovi (poslednju izmenu)	`Ctrl+Z`	*Raspored*	*Panel Istorija opoziva: radnje opoziva i ponavljanja*
Ponovi (poslednju opozvanu izmenu)	`Ctrl+Y`	*Raspored*	*Panel Istorija opoziva: radnje opoziva i ponavljanja*
Obriši	`Del`		
Iseci	`Ctrl+X`		
Kopiraj	`Ctrl+C`		
Nalepi	`Ctrl+V`		
Nalepi na mesto	`Ctrl+Shift+V`		
Izaberi sve	`Ctrl+A`		
Poništi ceo izbor	`Ctrl+Shift+A`		
Obrni izbor			
Izaberi sledeću stavku ispod	`Ctrl+Alt+[`		
Izaberi sledeću stavku iznad	`Ctrl+Alt+]`		
Pomeri raspored	`P`	*Kutija sa alatkama*	
Uveličaj	`Z`	*Kutija sa alatkama*	
Izaberi/pomeri stavku	`V`	*Kutija sa alatkama*	*Interakcija sa stavkama rasporeda*
Pomeri sadržaj	`C`	*Kutija sa alatkama*	*The Map Item*
Uredi čvorove stavke		*Kutija sa alatkama*	*The Node-Based Shape Items*

Meni Prikaz

Meni *Prikaz* daje pristup alatkama za navigaciju i pomaže u konfigurisanju opšteg ponašanja rasporeda za štampu. Pored uobičajenih alatki za uveličavanje, imate načine da:

- Osveži prikaz (ako zateknete prikaz u nedoslednom stanju);
- omogućite *mrežu* za koju biste mogli lepiti stavke pri pomeranju ili pravljenju. Podešavanje mreže vrši se u *Podešavanja* ► *Opcije rasporeda…* ili u *panelu Raspored*;
- omogućite *vodilje* za koje biste mogli lepiti stavke pri pomeranju ili pravljenju. Vodilje su crvene linije koje možete napraviti klikom na lenjir (iznad ili sa leve strane rasporeda) i prevlačenjem na željenu lokaciju;
- *Pametne vodilje*: koristi druge stavke rasporeda kao vodilje za dinamičko lepljenje dok pomerate ili menjate oblik stavke;

- *Očisti vodilje* za uklanjanje svih trenutnih vodilja;
- *Prikaži granični pravougaonik* oko stavki radi lakšeg prepoznavanja vašeg izbora;
- *Prikaži lenjire* oko rasporeda;
- *Prikaži strane* ili podesite strane na prozirne. Raspored se često koristi za pravljenje rasporeda koji nisu za štampu, npr. za uključivanje u prezentacije ili druge dokumente, i poželjno je izvesti kompoziciju sa potpuno prozirnom pozadinom. U drugim programima za uređivanje ovo se ponekad naziva „beskonačno platno“.

U rasporedu za štampu možete promeniti nivo uvećanja pomoću točkića miša ili klizača i padajuće liste u statusnoj traci. Ako je potrebno da pređete u režim pomeranja dok radite u oblasti rasporeda, možete držati `Spacebar` ili točkić miša. Pomoću `Ctrl+Spacebar` možete privremeno preći u režim Uveličaj, a pomoću `Ctrl+Alt+Spacebar` u režim Umanji.

Paneli i palete alatki mogu se omogućiti iz menija *Prikaz* ►. Da biste maksimizovali prostor dostupan za interakciju sa kompozicijom, možete označiti opciju *Prikaz* ► *Uključi/isključi vidljivost panela* ili pritisnuti `Ctrl+Tab`; svi paneli se sakrivaju, a samo prethodno vidljivi paneli se vraćaju kada se opcija poništi.

Takođe je moguće preći u režim celog ekrana radi više prostora za interakciju pritiskom na `F11` ili korišćenjem *Prikaz* ► *Uključi/isključi ceo ekran*.

Таблеа18.3: Alatke menija „Prikaz“

Alatka	Prečica	Traka sa alatkama	Referenca
Osveži	`F5`	*Navigacija*	
Pregled ►			
Uvećaj	`Ctrl++`	*Navigacija*	
Umanji	`Ctrl+-`	*Navigacija*	
Uveličaj na 100%	`Ctrl+1`	*Navigacija*	
Uvećaj na pun obuhvat	`Ctrl+0`	*Navigacija*	
Uveličaj na širinu			
Prikaži mrežu	`Ctrl+'`		*Vodilje i mreža*
Lepi za mrežu	`Ctrl+Shift+'`		*Vodilje i mreža*
Prikaži vodilje	`Ctrl+;`		*Vodilje i mreža*
Lepi za vodilje	`Ctrl+Shift+;`		*Vodilje i mreža*
Pametne vodilje	`Ctrl+Alt+;`		
Upravljaj vodiljama…			*Panel Vodilje*
Očisti vodilje			*Panel Vodilje*
Prikaži lenjire	`Ctrl+R`		
Prikaži granične pravougaonike	`Ctrl+Shift+B`		
Prikaži strane			
Trake sa alatkama ►			*Paneli i trake sa alatkama*
Paneli ►			*Paneli i trake sa alatkama*
Uključi/isključi ceo ekran	`F11`		*Prikaz*
Uključi/isključi vidljivost panela	`Ctrl+Tab`		*Prikaz*

Meni Stavke

Meni *Stavke* pomaže vam da konfigurišete položaj stavki u rasporedu i odnose između njih (pogledajte *Interakcija sa stavkama rasporeda*).

Табела18.4: Alatke menija „Stavke“

Alatka	Prečica	Traka sa alatkama	Referenca
Grupiši	`Ctrl+G`	*Radnje*	*Grupisanje stavki*
Razgrupiši	`Ctrl+Shift+G`	*Radnje*	*Grupisanje stavki*
Podigni	`Ctrl+]`	*Radnje*	*Poravnanje i raspoređivanje*
Spusti	`Ctrl+[`	*Radnje*	*Poravnanje i raspoređivanje*
Dovedi u prvi plan	`Ctrl+Shift+]`	*Radnje*	*Poravnanje i raspoređivanje*
Pošalji u pozadinu	`Ctrl+Shift+[`	*Radnje*	*Poravnanje i raspoređivanje*
Zaključaj izabrane stavke	`Ctrl+L`	*Radnje*	*Zaključavanje stavki*
Otključaj sve	`Ctrl+Shift+L`	*Radnje*	*Zaključavanje stavki*
Poravnaj stavke ►		*Radnje*	*Poravnanje i raspoređivanje*
Rasporedi stavke ►		*Radnje*	*Pomeranje i promena veličine stavki*
Promeni veličinu ►		*Radnje*	*Pomeranje i promena veličine stavki*

Meni Dodaj stavku

Ovo su alatke za pravljenje stavki rasporeda. Svaka od njih detaljno je opisana u poglavlju *Stavke rasporeda*.

Табела18.5: Alatke menija „Dodaj stavku“

Alatka	Traka sa alatkama	Referenca
Dodaj kartu	*Kutija sa alatkama*	*The Map Item*
Dodaj 3D kartu	*Kutija sa alatkama*	*The 3D Map Item*
Dodaj sliku	*Kutija sa alatkama*	*The Picture Item*
Dodaj natpis	*Kutija sa alatkama*	*The Label Item*
Dodaj dinamički tekst ►		*Stavka Natpis*
Dodaj legendu	*Kutija sa alatkama*	*The Legend Item*
Dodaj razmernik	*Kutija sa alatkama*	*The Scale Bar Item*
Dodaj strelicu severa	*Kutija sa alatkama*	*The North Arrow Item*
Dodaj oblik ►	*Kutija sa alatkama*	*The Regular Shape Item*
► *Dodaj pravougaonik*	*Kutija sa alatkama*	*The Regular Shape Item*

наставак на следећој страни

Табела 18.5 – настављено са претходне стране

Alatka	Traka sa alatkama	Referenca
► *Dodaj elipsu*	*Kutija sa alatkama*	*The Regular Shape Item*
► *Dodaj trougao*	*Kutija sa alatkama*	*The Regular Shape Item*
Dodaj marker	*Kutija sa alatkama*	*The Marker Item*
Dodaj strelicu	*Kutija sa alatkama*	*The Arrow Item*
Dodaj stavku sa čvorovima ►	*Kutija sa alatkama*	*The Node-Based Shape Items*
► *Dodaj poligon*	*Kutija sa alatkama*	*The Node-Based Shape Items*
► *Dodaj poliliniju*	*Kutija sa alatkama*	*The Node-Based Shape Items*
Dodaj HTML	*Kutija sa alatkama*	*The HTML Frame Item*
Dodaj atributsku tabelu	*Kutija sa alatkama*	*The attribute table item*
Dodaj fiksnu tabelu	*Kutija sa alatkama*	*The fixed table item*
Dodaj visinski profil	*Kutija sa alatkama*	*The Elevation Profile Item*

Meni Atlas

Табела18.6: Alatke menija „Atlas“

Alatka	Prečica	Traka sa alatkama	Referenca
Pregled atlasa	`Ctrl+ALt+/`	*Atlas*	*Preview and generate an atlas*
Prvi objekat	`Ctrl+<`	*Atlas*	*Preview and generate an atlas*
Prethodni objekat	`Ctrl+,`	*Atlas*	*Preview and generate an atlas*
Sledeći objekat	`Ctrl+.`	*Atlas*	*Preview and generate an atlas*
Poslednji objekat	`Ctrl+>`	*Atlas*	*Preview and generate an atlas*
Štampaj atlas…		*Atlas*	*Preview and generate an atlas*
Izvezi atlas kao slike…		*Atlas*	*Preview and generate an atlas*
Izvezi atlas kao SVG…		*Atlas*	*Preview and generate an atlas*
Izvezi atlas kao PDF…		*Atlas*	*Preview and generate an atlas*
Podešavanja atlasa		*Atlas*	*Generate an Atlas*

Meni Podešavanja

Meni *Podešavanja* ► *Opcije rasporeda…* je prečica do menija *Podešavanja* ► *Opcije* ► *Rasporedi* glavnog QGIS platna. Ovde možete postaviti neke opcije koje će se koristiti kao podrazumevane na svakom novom rasporedu za štampu:

- *Podrazumevane vrednosti rasporeda* omogućavaju vam da zadate podrazumevani font koji se koristi;
- Pomoću *Izgled mreže* možete postaviti stil mreže i njenu boju. Postoje tri tipa mreže: **Tačke**, **Pune** linije i **Krstići**;
- *Podrazumevane vrednosti mreže i vodilja* definišu razmak, pomeraj i toleranciju mreže (za više detalja pogledajte *Vodilje i mreža*);
- *Putanje rasporeda*: za upravljanje listom prilagođenih putanja za traženje šablona za štampu.

Meni *Podešavanja* ► *Prečice na tastaturi…* omogućava vam da koristite *menadžer prečica* u interfejsu rasporeda za štampu.

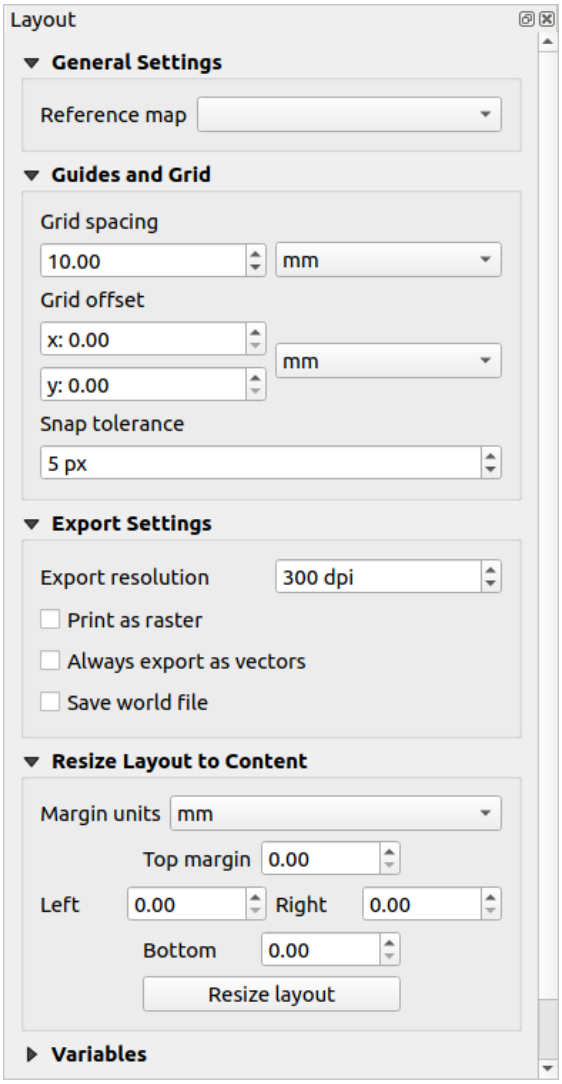
Kontekstni meniji

U zavisnosti od toga gde kliknete desnim tasterom u dijalogu rasporeda za štampu, otvarate kontekstni meni sa raznim mogućnostima:

- Kliknite desnim tasterom na traku menija ili bilo koju paletu alatki i dobijate listu panela i paleta alatki rasporeda koje možete omogućiti ili onemogućiti jednim klikom.
- Kliknite desnim tasterom iznad lenjira i možete *Prikaži vodilje*, *Lepi za vodilje*, *Upravljaj vodiljama…* čime se otvara *panel Vodilje* ili *Očisti vodilje*. Takođe je moguće sakriti lenjire.
- Kliknite desnim tasterom na platno rasporeda za štampu i:
  - Moći ćete da *Opozovete* i *Ponovite* nedavne izmene, ili *Nalepite* bilo koju kopiranu stavku (dostupno samo ako nijedna stavka nije izabrana).
  - Ako kliknete iznad strane, dodatno možete pristupiti panelu trenutnih *Svojstava strane* ili *Ukloni stranu*.
  - Ako kliknete na izabranu stavku, možete je iseći ili kopirati, kao i otvoriti panel *Svojstva stavke*.
  - Ako je izabrano više od jedne stavke, možete ih grupisati i/ili razgrupisati ako je bar jedna grupa već u izboru.
- Klik desnim tasterom unutar tekstualnog polja ili vidžeta brojača bilo kog panela rasporeda pruža opcije uređivanja za manipulaciju njegovim sadržajem.

Panel Raspored

U panelu *Raspored* možete definisati globalna podešavanja svog rasporeda za štampu.



Слика18.3: Podešavanja rasporeda u rasporedu za štampu

Opšta podešavanja

U rasporedu za štampu možete koristiti više od jedne stavke karte. *Referentna karta* predstavlja stavku karte koja se koristi kao glavna karta rasporeda. Dodeljena je sve dok u rasporedu postoji stavka karte. Raspored će koristiti ovu kartu u bilo kom od svojih svojstava i promenljivih pri računanju jedinica ili razmere. Ovo uključuje izvoz rasporeda za štampu u georeferencirane formate.

Osim toga, nove stavke rasporeda kao što su razmernik, legenda ili strelica severa podrazumevano imaju svoja podešavanja (orijentacija, prikazani slojevi, razmera, …) vezana za stavku karte iznad koje su iscrtane, a vraćaju se na referentnu kartu ako nema preklapajuće karte.

Vodilje i mreža

Na svoj list papira možete postaviti neke referentne oznake koje vam pomažu da precizno postavite neke stavke. Ove oznake mogu biti:

- jednostavne horizontalne ili vertikalne linije (nazvane **Vodilje**) postavljene na položaj koji želite (za pravljenje vodilja pogledajte *Panel Vodilje*).
- ili pravilna **Mreža**: mreža horizontalnih i vertikalnih linija naneta preko rasporeda.

Podešavanja kao što su *Razmak mreže* ili *Pomeraj mreže* mogu se prilagoditi u ovoj grupi, kao i *Tolerancija lepljenja* koja se koristi za stavke. Tolerancija je maksimalno rastojanje ispod kojeg se pokazivač miša lepi za mrežu ili

vodilju, tokom pomeranja, promene veličine ili pravljenja stavke.

Da li mrežu ili vodilje treba prikazati postavlja se u meniju *Prikaz*. Tamo takođe možete odlučiti da li se mogu koristiti za lepljenje stavki rasporeda. Kada su i linija mreže i linija vodilje unutar tolerancije neke tačke, vodilje će uvek imati prednost - pošto su ručno postavljene (dakle, pretpostavka je da su namerno postavljene na veoma poželjne lokacije za lepljenje i da ih treba izabrati umesto opšte mreže).

Белешка

U meniju *Podešavanja* ► *Opcije rasporeda* takođe možete postaviti parametre mreže i vodilja gore izložene. Međutim, ove opcije će se primeniti samo kao podrazumevane na nove rasporede za štampu.

Podešavanja izvoza

Možete definisati rezoluciju koja se koristi za sve izvezene karte u *Rezolucija izvoza*. Ovo podešavanje se zatim može zameniti svaki put kada izvozite kartu.

Zbog nekih naprednih opcija iscrtavanja (*režim stapanja*, *efekti*…), stavka rasporeda može zahtevati rasterizaciju da bi bila ispravno izvezena. QGIS će je pojedinačno rasterizovati bez prisiljavanja svake druge stavke da takođe bude rasterizovana. Time se pri štampanju ili čuvanju kao PostScript ili PDF stavke zadržavaju što je više moguće kao vektori, npr. stavka karte sa neprozirnošću sloja neće prisiliti natpise, razmernike itd. da takođe budu rasterizovani. Međutim, možete:

- prisiliti sve stavke da budu rasterizovane označavanjem polja *Štampaj kao raster*;
- ili koristiti suprotnu opciju, tj. *Uvek izvezi kao vektore*, da biste prisilili izvoz da zadrži stavke kao vektore pri izvozu u kompatibilan format. Imajte u vidu da u nekim slučajevima to može učiniti da izlaz izgleda drugačije od rasporeda.

Gde format to omogućava (npr. `.TIF`, `.PDF`), izvoz rasporeda za štampu podrazumevano daje georeferenciranu datoteku (na osnovu stavke *Referentna karta* u grupi *Opšta podešavanja*). Za druge formate georeferencirani izlaz zahteva da generišete world datoteku označavanjem *Sačuvaj world datoteku*. World datoteka se pravi pored izvezene karte(a), ima naziv izlaza strane sa stavkom referentne karte i sadrži informacije za njeno lako georeferenciranje.

Promeni veličinu rasporeda prema sadržaju

Pomoću alatke *Promeni veličinu strane* u ovoj grupi pravite jedinstvenu kompoziciju strane čiji obuhvat pokriva trenutni sadržaj rasporeda za štampu (sa nekim opcionim *marginama* oko isečenih granica).

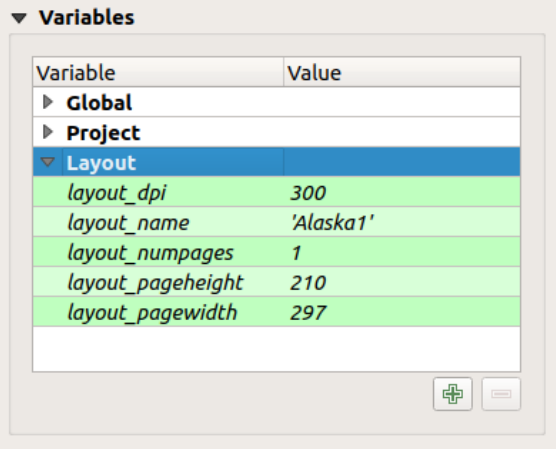
Imajte u vidu da se ovo ponašanje razlikuje od opcije *iseci prema sadržaju* po tome što se sve stavke postavljaju na stvarnu i jedinstvenu stranu koja zamenjuje sve postojeće strane.

Promenljive

Promenljive navode sve promenljive dostupne na nivou rasporeda (što uključuje sve globalne i projektne promenljive).

Takođe omogućava korisniku da upravlja promenljivama na nivou rasporeda. Kliknite na dugme da biste dodali novu prilagođenu promenljivu na nivou rasporeda. Isto tako, izaberite prilagođenu promenljivu na nivou rasporeda iz liste i kliknite na dugme da biste je uklonili.

Više informacija o korišćenju promenljivih u odeljku *Opšte alatke*.



Слика18.4: Уређивач промењивих у распореду за штампу

Рад са својствима стране

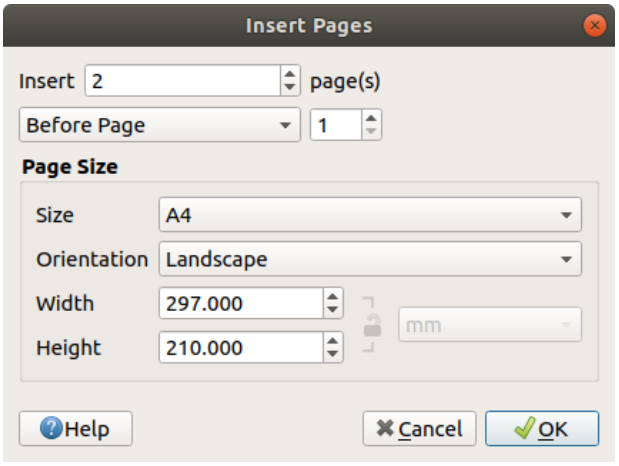
Распоред се може састојати од више страна. На пример, прва страна може приказивати платно карте, друга страна атрибутску табелу повезану са слојем, док трећа приказује HTML оквир који води ка сајту ваше организације. Или на сваку страну можете додати много типова ставки.

Додавање нове стране

Осим тога, распоред се може направити помоћу различитих величина и/или оријентација страна. Да бисте додали страну, изаберите алатку  *Додај стране…* из менија *Распоред* или *палете алатки Распоред*. Отвара се дијалог *Уметни стране* и тражи се од вас да попуните:

- број страна које се уметћу;
- положај страна: пре или после задате стране или на крају распореда за штампу;
- *Величина стране*: може бити унапред задати формат стране (`A4`, `B0`, `Legal`, `Letter`, `ANSI A`, `Arch A` и њихове изведенице, као и тип резолуције, као што је `1920x1080` или `1024x768`) са припадајућом *Оријентацијом* (Усправно или Положено).

  Величина стране може бити и `custom` формата; у том случају биста морали да унесете њену *Ширину* и *Висину* (са закључаним односом величине ако је потребно) и изаберете јединицу која се користи међу `mm`, `cm`, `px`, `pt`, `in`, `ft`… Претварање унетих вредности се аутоматски примењује при прелазку из једне јединице у другу.



Слика18.5: Прављење нове стране у распореду за штампу

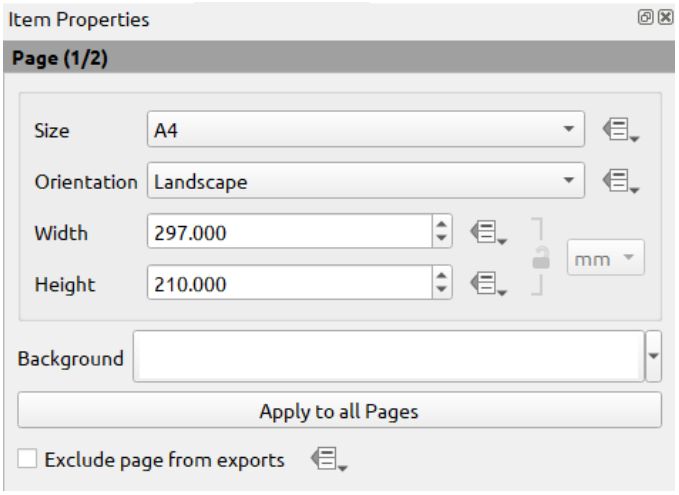
Ažuriranje svojstava strane

Bilo koja strana može se kasnije prilagoditi kroz panel Svojstva stavke za stranu. Da biste pristupili svojstvima strane, ili:

- Kliknite levim tasterom na prazan deo strane
- Kliknite desnim tasterom na stranu i izaberite *Svojstva strane…*.
- Idite na *Raspored* ► *Svojstva strane…* dok je pokazivač iznad strane

Otvara se panel *Svojstva stavke* sa podešavanjima kao što su:

- za rasporede sa više strana, prikazuje broj aktivne strane i broj dostupnih strana
- okvir *Veličina strane* opisan gore. Svako svojstvo možete izmeniti pomoću opcija zamene definisane podacima (za slučaj upotrebe pogledajte *Explore Data-defined override buttons with atlas*);
- *Pozadina* trenutne strane pomoću *boje* ili *simbola* koje želite.
- dugme *Primeni na sve strane* ažurira svojstva trenutne strane na druge postojeće strane u rasporedu;
- *Izostavi stranu iz izvoza* za kontrolu da li trenutnu stranu sa njenim sadržajem treba uključiti u *izlaz rasporeda*;

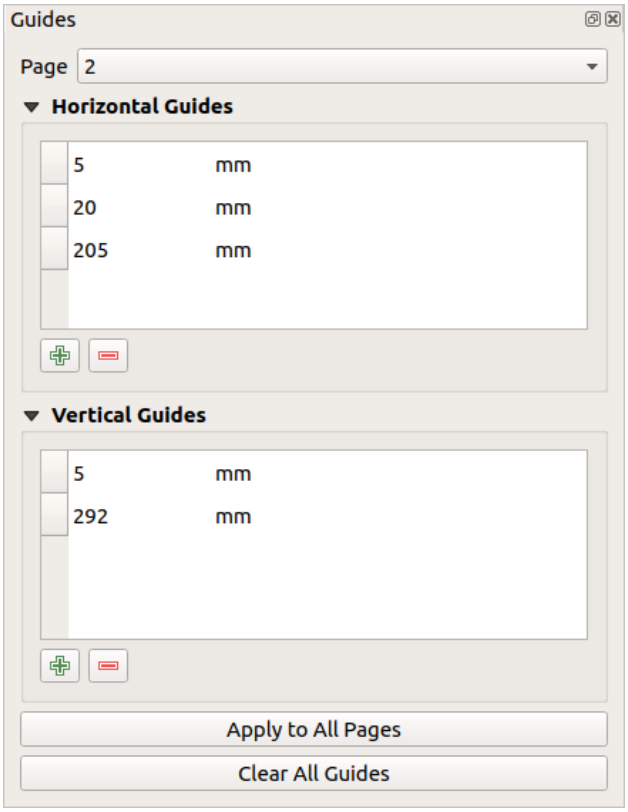


Слика18.6: Dijalog svojstava strane

Panel Vodilje

Vodilje su vertikalne ili horizontalne linijske reference koje možete postaviti na stranu rasporeda da vam pomognu pri postavljanju stavki, pri njihovom pravljenju, pomeranju ili promeni veličine. Da bi bile aktivne, vodilje zahtevaju da opcije *Prikaz* ► *Prikaži vodilje* i *Prikaz* ► *Lepi za vodilje* budu označene. Za pravljenje vodilje postoje dva različita metoda:

- ako je postavljena opcija *Prikaz* ► *Prikaži lenjire*, izvucite lenjir i otpustite taster miša unutar oblasti strane, na željenom položaju.
- za veću preciznost, koristite panel *Vodilje* iz *Prikaz* ► *Kutija sa alatkama* ► ili izborom *Upravljaj vodiljama za stranu…* iz kontekstnog menija strane.



Слика18.7: Panel Vodilje

Panel *Vodilje* omogućava pravljenje linija za lepljenje na određenim lokacijama:

1. Izaberite *Stranu* na koju želite da dodate vodilje
2. Kliknite na dugme Dodaj novu vodilju i unesite koordinate horizontalne ili vertikalne linije. Ishodište je u gornjem levom uglu. Za ovo su dostupne različite jedinice.

   Panel takođe omogućava prilagođavanje položaja postojećih vodilja na tačne koordinate: dvokliknite i zamenite vrednost.
3. Panel *Vodilje* navodi samo stavke za trenutnu stranu. Omogućava pravljenje ili uklanjanje vodilja samo na trenutnoj strani. Međutim, možete koristiti dugme *Primeni na sve strane* da biste replikovali konfiguraciju vodilja trenutne strane na druge strane u rasporedu.
4. Da biste obrisali vodilju, izaberite je i pritisnite dugme Ukloni izabranu vodilju. Koristite *Očisti sve vodilje* da biste uklonili sve vodilje na trenutnoj strani.

Савет

Lepljenje za postojeće stavke rasporeda

Osim vodilja i mreža, postojeće stavke možete koristiti kao reference za lepljenje pri pomeranju, promeni veličine ili pravljenju novih stavki; one se nazivaju **pametne vodilje** i zahtevaju da opcija *Prikaz* ► *Pametne vodilje* bude označena. Svaki put kada je pokazivač miša blizu granice neke stavke, pojavljuje se krstić za lepljenje.

Panel Stavke

Panel *Stavke* nudi neke opcije za upravljanje izborom i vidljivošću stavki. Sve stavke dodate na platno rasporeda za štampu (uključujući *grupu stavki*) prikazane su u listi, a izbor stavke čini odgovarajući red izabranim u listi, kao što i izbor reda bira odgovarajuću stavku na platnu rasporeda za štampu. Ovo je stoga zgodan način za izbor stavke postavljene iza druge. Imajte u vidu da je izabran red prikazan podebljano. Izbor više stavki takođe je moguć držanjem `Shift` za susedne stavke i tastera `Ctrl` za nesusedne.

Za bilo koju dostupnu stavku možete:

- postaviti je vidljivom ili ne;
- zaključati ili otključati njen položaj;
- sortirati njen Z položaj. Svaku stavku u listi možete pomeriti gore i dole klikom i prevlačenjem. Gornja stavka u listi biće dovedena u prvi plan na platnu rasporeda za štampu. Podrazumevano se novonapravljena stavka postavlja u prvi plan.
- promeniti ID stavke dvoklikom na tekst;
- kliknuti desnim tasterom na stavku i izabrati da li da je kopirate ili obrišete ili otvorite njen *panel svojstava*.

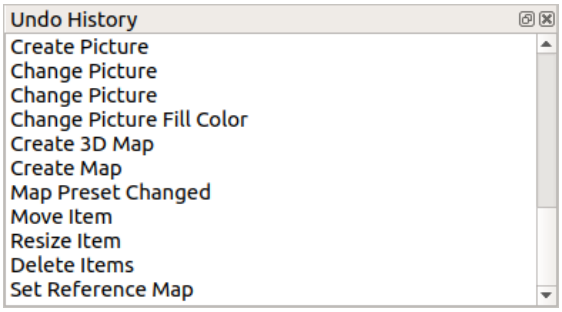
Kada pronađete ispravan položaj za stavku, možete je zaključati označavanjem polja u koloni . Zaključane stavke **nisu** izborljive na platnu. Zaključane stavke mogu se otključati izborom stavke u panelu *Stavke* i poništavanjem polja ili možete koristiti ikone u paleti alatki.

Panel Istorija opoziva: radnje opoziva i ponavljanja

Tokom procesa rasporeda moguće je opozvati i ponoviti izmene. To se može uraditi pomoću alatki za opoziv i ponavljanje dostupnih u meniju *Uređivanje*, paleti alatki *Raspored* ili kontekstnom meniju svaki put kada kliknete desnim tasterom u oblasti rasporeda za štampu:

- Opozovi poslednju izmenu
- Ponovi poslednju izmenu

To se takođe može uraditi klikom miša unutar panela *Istorija opoziva* (pogledajte Сл.18.8). Panel Istorija navodi poslednje radnje obavljene unutar rasporeda za štampu. Izaberite tačku na koju želite da se vratite i kada obavite novu radnju, sve radnje obavljene posle izabrane biće uklonjene.



Слика18.8: Istorija opoziva u rasporedu za štampu

18.2 Stavke rasporeda

18.2.1 Zajedničke opcije stavki rasporeda

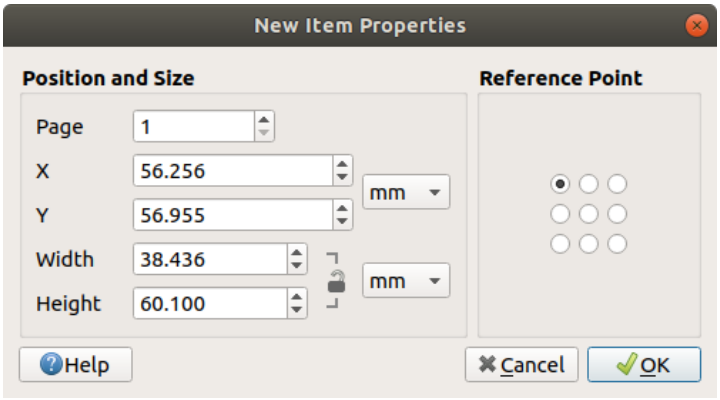
QGIS pruža veliki skup stavki za raspoređivanje karte. Mogu biti tipa karte, legende, razmernika, slike, tabele, strelice severa, slike… Međutim, dele neke zajedničke opcije i ponašanje koji su izloženi ispod.

Pravljenje stavke rasporeda

Stavke se mogu praviti pomoću različitih alatki, bilo od nule ili na osnovu postojećih stavki.

Da biste napravili stavku rasporeda od nule:

1. Izaberite odgovarajuću alatku iz menija *Dodaj stavku* ili trake *Kutija sa alatkama*.
2. Zatim:
   - Kliknite na stranu i popunite informacije o veličini i položaju zatražene u dijalogu *Svojstva nove stavke* koji iskoči (za detalje pogledajte *Položaj i veličina*);



Слика18.9: Dijalog Svojstva nove stavke

   - Ili kliknite i prevucite da biste definisali početnu veličinu i položaj stavke. Za bolji položaj možete se osloniti na lepljenje za *mreže i vodilje*.

Белешка

Pošto mogu imati posebne oblike, crtanje stavki sa čvorovima ili strelica ne radi metodima jednog klika ni klika i prevlačenja; potrebno je da kliknete i postavite svaki čvor stavke. Za više detalja pogledajte *The Node-Based Shape Items*.

Takođe možete:

1. Izabrati postojeću stavku dugmetom  Izaberi/pomeri stavku iz palete alatki *Kutija sa alatkama*
2. Koristiti kontekstni meni ili alatke menija *Uređivanje* da biste kopirali/isekli stavku i nalepili je na položaj miša kao novu stavku.

   Takođe možete koristiti komandu *Nalepi na mesto* (`Ctrl+Shift+V`) da biste duplirali stavku sa jedne strane na drugu i postavili je na novu stranu na istim koordinatama kao original.

Osim toga, stavke možete praviti pomoću šablona rasporeda za štampu (za detalje pogledajte *Menadžer rasporeda*) komandom *Raspored* ► *Dodaj stavke iz šablona…*.

Савет

Dodavanje stavki rasporeda pomoću pregledača datoteka

Iz svog pregledača datoteka ili pomoću panela *Pregledač*, prevucite šablon rasporeda za štampu (`.qpt` datoteku) na dijalog rasporeda za štampu, a QGIS automatski dodaje sve stavke iz tog šablona u raspored.

Interakcija sa stavkama rasporeda

Svaka stavka unutar rasporeda za štampu može se pomeriti i promeniti veličina radi pravljenja savršenog rasporeda.

Za obe operacije prvi korak je aktiviranje alatke  Izaberi/pomeri stavku i klik na stavku.

Možete izabrati više stavki dugmetom  Izaberi/pomeri stavku: kliknite i prevucite preko stavki ili držite dugme `Shift` i kliknite na svaku od stavki koje želite. Da biste poništili izbor stavke, kliknite na nju držeći dugme `Shift`.

Svaki put kada postoji izbor, broj izabranih stavki prikazuje se u statusnoj traci. Unutar menija *Uređivanje* možete pronaći radnje za izbor svih stavki, poništavanje svih izbora, obrtanje trenutnog izbora i još mnogo toga…

Pomeranje i promena veličine stavki

Osim ako je opcija *Prikaz* ► *Prikaži granične pravougaonike* poništena, izabrana stavka će prikazati kvadratiće na svojim granicama; pomeranje jednog od njih mišem promeniće veličinu stavke u odgovarajućem pravcu. Tokom promene veličine, držanje `Shift` održaće odnos stranica. Držanje `Alt` će promeniti veličinu iz centra stavke.

Da biste pomerili stavku rasporeda, izaberite je mišem i pomerajte držeći levi taster. Ako je potrebno da ograničite kretanje na horizontalnu ili vertikalnu osu, držite dugme `Shift` na tastaturi dok pomerate miša. Izabranu stavku možete pomeriti i pomoću `Arrow keys` na tastaturi; ako je kretanje presporo, možete ga ubrzati držanjem `Shift`. Ako vam je potrebna veća preciznost, koristite svojstva *Položaj i veličina*, ili lepljenje za mrežu/vodilje kako je objašnjeno gore za pravljenje stavke.

Promena veličine ili pomeranje više stavki odjednom vrši se na isti način kao za jednu stavku. QGIS, međutim, pruža neke napredne alatke za automatsku promenu veličine izbora stavki prema različitim pravilima:

- visina svake stavke odgovara  najvišoj ili  najnižoj izabranoj stavki;
- širina svake stavke odgovara  najširoj ili  najužoj izabranoj stavki;
- menja veličinu stavki na  kvadrate: svaka stavka se uvećava da bi dobila oblik kvadrata.

Isto tako, dostupne su *alatke* za organizovanje položaja više stavki jednakim rasporedom:

- ivica (leve, desne, gornje ili donje) stavki;
- centara stavki horizontalno ili vertikalno;
- razmaka između stavki horizontalno ili vertikalno.

Grupisanje stavki

Grupisanje stavki omogućava vam da manipulišete skupom stavki kao jednom: možete lako promeniti veličinu, pomeriti, obrisati, kopirati stavke kao celinu.

Da biste napravili grupu stavki, izaberite više od jedne i pritisnite dugme  *Grupiši* u meniju *Stavke* ili paleti alatki *Radnje*, ili iz menija desnog klika. Red pod nazivom `Group` se dodaje u panel *Stavke* i može se zaključati ili sakriti kao bilo koji drugi *objekat panela Stavke*. Grupisane stavke **nisu pojedinačno** izborljive na platnu; koristite panel Stavke za direktan izbor i pristup panelu svojstava stavke.

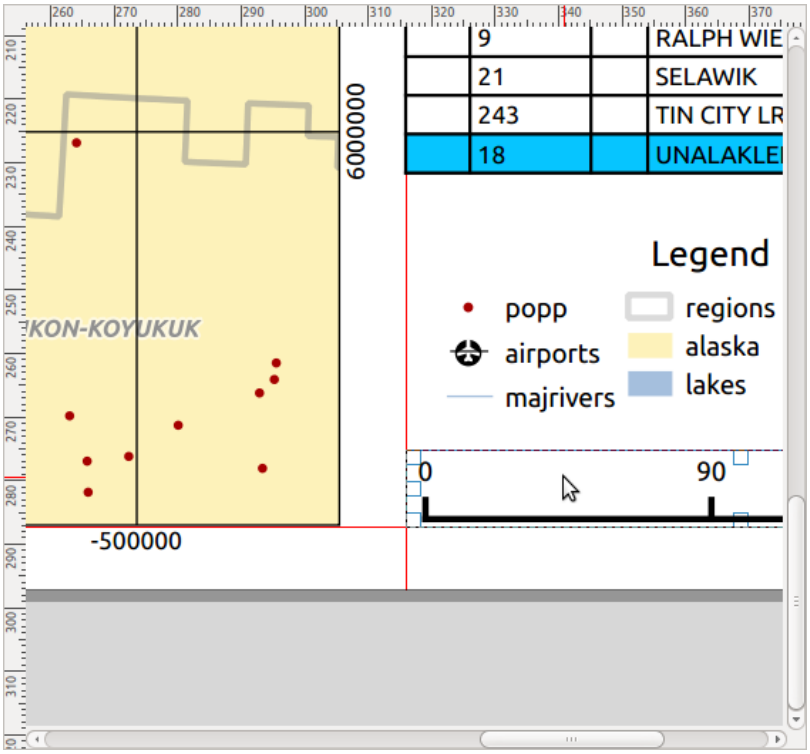
Zaključavanje stavki

Kada pronađete ispravan položaj za stavku, možete je zaključati pomoću dugmeta *Zaključaj izabrane stavke* u meniju *Stavke* ili paleti alatki *Radnje*, ili označavanjem polja pored stavke u panelu *Stavke*. Zaključane stavke **nisu** izborljive na platnu.

Zaključane stavke mogu se otključati izborom stavke u panelu *Stavke* i poništavanjem polja ili možete koristiti ikone u paleti alatki.

Poravnanje i raspoređivanje

Alatke za podizanje ili spuštanje Z položaja stavki u rasporedu nalaze se u padajućem meniju Podigni izabrane stavke. Izaberite element na platnu rasporeda za štampu i izaberite odgovarajuću funkcionalnost za podizanje ili spuštanje izabranog elementa iznad drugih elemenata. Ovaj redosled se prikazuje u panelu *Stavke*. Objekte možete podizati ili spuštati i u panelu *Stavke* klikom i prevlačenjem natpisa objekta u ovoj listi.



Слика18.10: Pomoćne linije poravnanja u rasporedu za štampu

U padajućem meniju Poravnaj izabrane stavke dostupno je nekoliko opcija poravnanja (pogledajte Сл.18.10). Da biste koristili funkciju poravnanja, prvo izaberite elemente, a zatim kliknite na jednu od ikona poravnanja:

- *Poravnaj levo* ili *Poravnaj desno*;
- *Poravnaj gore* ili *Poravnaj dole*;
- *Poravnaj po centru* horizontalno ili *Poravnaj po centru vertikalno*.

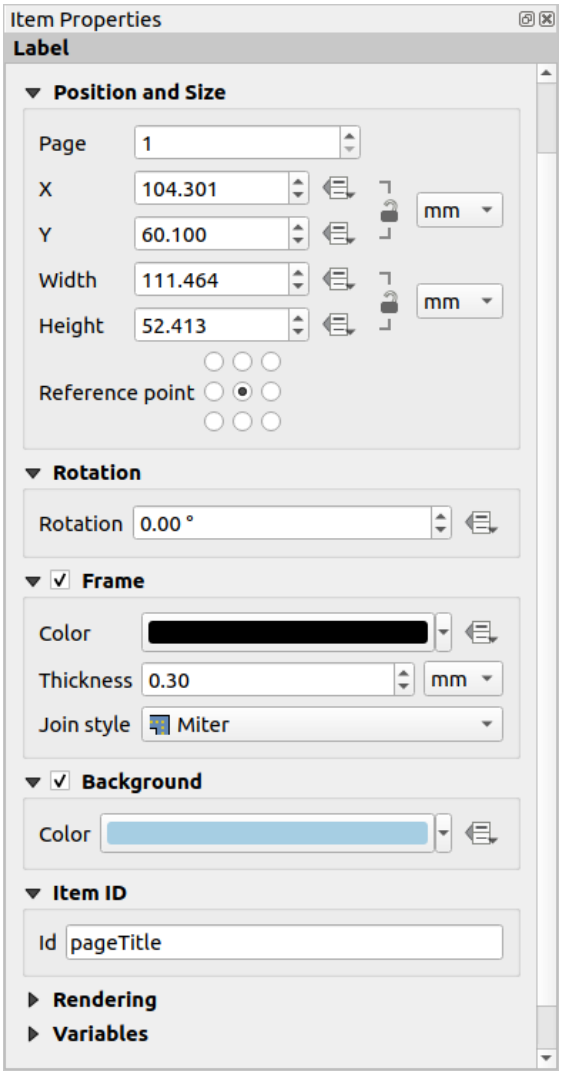
Svi izabrani elementi biće tada poravnati prema svom zajedničkom graničnom pravougaoniku. Pri pomeranju stavki na platnu rasporeda, pomoćne linije poravnanja se pojavljuju kada su ivice, centri ili uglovi poravnati.

Još jedan način poboljšanja položaja stavki rasporeda jeste prilagođavanje razmaka između njih, preko strane rasporeda. To se može uraditi izborom stavki i pritiskom na padajući meni *Rasporedi leve ivice* da biste:

- *Rasporedi leve ivice* ili *Rasporedi desne ivice* stavki jednako
- *Rasporedi gornje ivice* ili *Rasporedi donje ivice* stavki jednako
- *Rasporedi horizontalne centre* ili *Rasporedi vertikalne centre* stavki jednako
- Dodaj jednak razmak između stavki: *Rasporedi horizontalni razmak jednako* ili *Rasporedi vertikalni razmak jednako*

Zajednička svojstva stavki

Stavke rasporeda imaju skup zajedničkih svojstava koja ćete pronaći na dnu panela *Svojstva stavke*: Položaj i veličina, Rotacija, Okvir, Pozadina, ID stavke, Promenljive i Iscrtavanje (pogledajte Сл.18.11).



Слика18.11: Grupe zajedničkih svojstava stavke

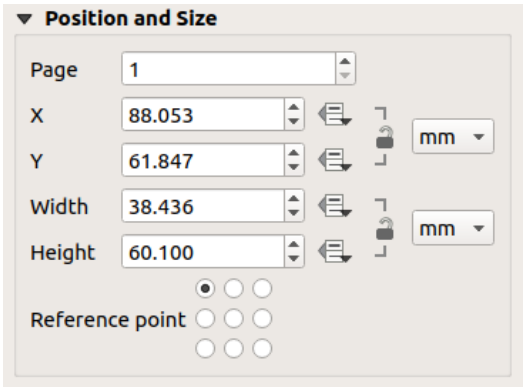
Белешка

Ikona Zamena definisana podacima pored većine opcija znači da to svojstvo možete povezati sa slojem, atributima objekata, geometrijom ili sa bilo kojim drugim svojstvom stavke rasporeda, pomoću *izraza* ili *promenljivih*. Za više informacija pogledajte *Data defined override setup*.

- Grupa *Položaj i veličina* omogućava vam da definišete veličinu i položaj okvira koji sadrži stavku (za više informacija pogledajte *Položaj i veličina*).
- *Rotacija* postavlja rotaciju elementa (u stepenima).
- *Okvir* prikazuje ili sakriva okvir oko stavke. Koristite vidžete *Boja*, *Debljina* i *Stil spoja* da biste prilagodili ta svojstva.
- U grupi *Pozadina* možete napraviti ili izabrati *Boju* pozadine. Prozirnost se može prilagoditi izmenom podešavanja polja alfa.
- Koristite *ID stavke* da biste napravili odnos sa drugim stavkama rasporeda za štampu. Ovo se koristi sa QGIS serverom i drugim potencijalnim veb klijentima. Možete postaviti ID na stavku (na primer, kartu ili natpis), a zatim veb klijent može poslati podatke za postavljanje svojstva (npr. teksta natpisa) za tu konkretnu stavku. Komanda GetProjectSettings navešće stavke i ID-jeve dostupne u rasporedu.
- Režim *Iscrtavanje* pomaže vam da postavite da li i kako stavka može biti prikazana: možete, na primer, primeniti *režim stapanja*, prilagoditi neprozirnost stavke ili *Izostaviti stavku iz izvoza*.

Položaj i veličina

Proširujući mogućnosti dijaloga *Svojstva nove stavke* sposobnostima definisanim podacima, ova grupa omogućava vam da stavke postavite precizno.

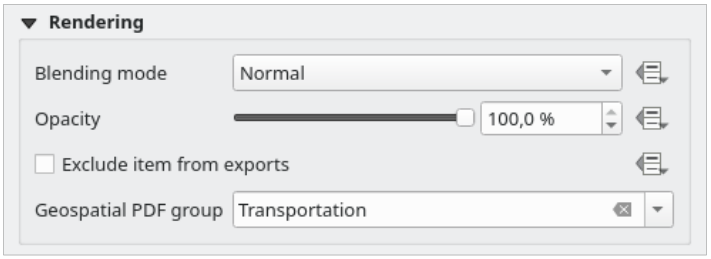


Слика18.12: Položaj i veličina

- stvarni broj strane na koju se stavka postavlja;
- referentnu tačku stavke;
- *X* i *Y* koordinate *Referentne tačke* stavke na izabranoj strani. Odnos između ovih vrednosti može se zaključati klikom na dugme . Izmene napravljene na vrednosti pomoću vidžeta ili alatke Izaberi/pomeri stavku odraziće se na obe;
- *Širinu* i *Visinu* graničnog pravougaonika stavke. Kao i za koordinate, odnos između širine i visine može se zaključati.

Režim iscrtavanja

QGIS omogućava napredno iscrtavanje stavki rasporeda baš kao za vektorske i rasterske slojeve.



Слика18.13: Režim iscrtavanja

- *Režim stapanja*: Ovom alatkom možete postići efekte koji bi se inače postigli samo pomoću softvera za grafičko iscrtavanje. Pikseli vaših preklapajućih i podležećih stavki mogu se mešati prema postavljenom režimu (za opis svakog efekta pogledajte *Blending Modes*).
- *Neprozirnost*: Ovom alatkom možete učiniti podležeću stavku u rasporedu vidljivom. Koristite klizač da biste prilagodili vidljivost svoje stavke svojim potrebama. Takođe možete precizno definisati procenat vidljivosti u meniju pored klizača.
- *Izostavi stavku iz izvoza*: U kombinaciji sa izrazima definisanim podacima, možete odlučiti da stavku učinite nevidljivom u nekim izvozima. Nakon aktiviranja ovog polja, stavka neće biti uključena u izvoz u PDF, štampanje itd.
- *Grupa geoprostornog PDF-a*: Kada je postavljeno, odgovarajuća grupa stabla slojeva biće napravljena u izvezenom geoprostornom PDF-u, a stavka će biti vidljiva samo kada je ta grupa označena. Time se sadržaj može selektivno prikazivati kao grupa onima koji pregledaju geoprostorni PDF. Npr. može omogućiti da se dodatni sadržaj rasporeda kao što su opisni natpisi ili legende prikaže samo kada su slojevi iz grupe vidljivi, čineći izvoz datoteke mnogo fleksibilnijim.

Promenljive

Promenljive navode sve promenljive dostupne na nivou stavke rasporeda (što uključuje sve globalne, projektne i promenljive rasporeda). Stavke karte u rasporedu takođe imaju dodatni odeljak *Podešavanja karte* za promenljive koje pružaju lak pristup vrednostima kao što su razmera karte, obuhvat i tako dalje.

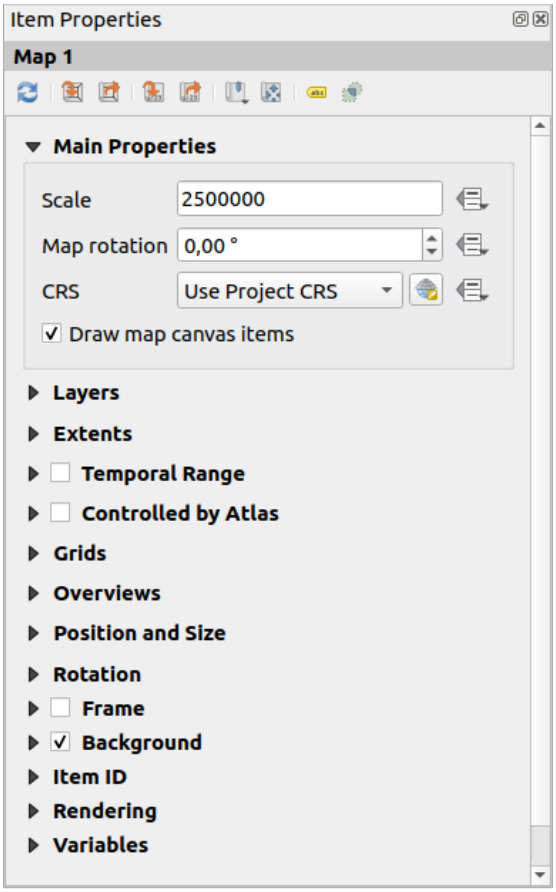
U *Promenljive* takođe je moguće upravljati promenljivama na nivou stavke rasporeda. Kliknite na dugme da biste dodali novu prilagođenu promenljivu. Isto tako, izaberite bilo koju prilagođenu promenljivu na nivou stavke iz liste i kliknite na dugme da biste je uklonili.

Više informacija o korišćenju promenljivih u odeljku *Storing values in Variables*.

18.2.2 The Map Item

The map item is the main frame that displays the map you've designed in the map canvas. Use the *Add Map* tool following *items creation instructions* to add a new map item that you can later manipulate the same way as exposed in *Interakcija sa stavkama rasporeda*.

By default, a new map item shows the current status of the *map canvas* with its extent and visible layers. You can customize it thanks to the *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities:



Слика18.14: Map Item Properties Panel

The Toolbar

The Map *Item Properties* panel embeds a toolbar with the following functionalities:

- Update map preview
- Set map canvas to match main canvas extent
- View current map extent in main canvas
- Set map scale to match main canvas scale
- Set main canvas to match current map scale
- Bookmarks: set the map item extent to match an existing spatial bookmark
- Interactively edit map extent: pan and zoom interactively within the map item
- Labeling settings: control feature label behaviour (placement, visibility…) in the layout map item extent:
  - set a *Margin from map edges*, a data definable distance from the map item's limits inside which no label should be displayed
  - *Allow truncated labels on edges of map*: controls whether labels which fall partially outside of the map item allowed extent should be rendered. If checked, these labels will be shown (when there's no way to place them fully within the visible area). If unchecked then partially visible labels will be skipped.
  - *Label blocking items*: allows other layout items (such as scalebars, north arrows, inset maps, etc) to be marked as a blockers for the map labels in the **active** map item. This prevents any map labels from

being placed under those items - causing the labeling engine to either try alternative placement for these labels or discard them altogether.

If a *Margin from map edges* is set, the map labels are not placed closer than the specified distance from the checked layout items.

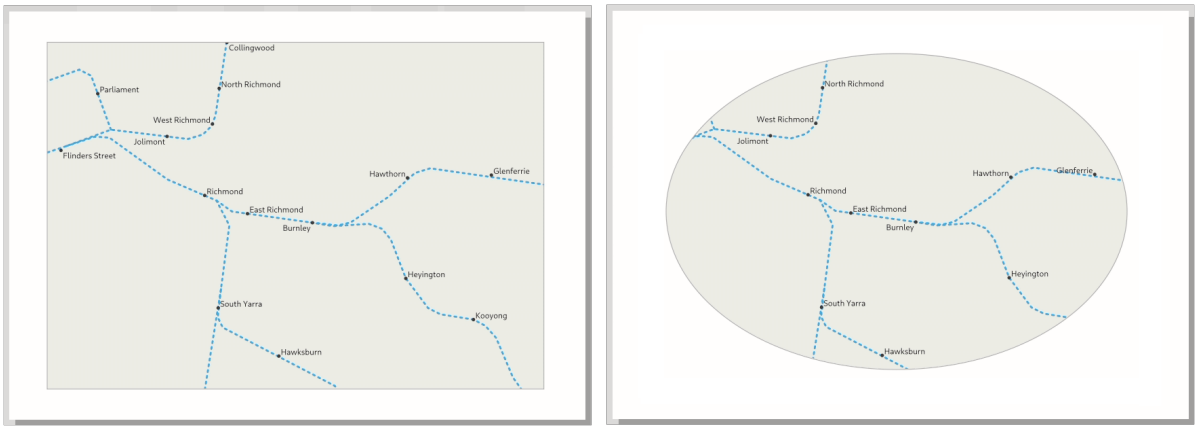
- *Show unplaced labels*: can be used to determine whether labels are missing from the layout map (e.g. due to conflicts with other map labels or due to insufficient space to place the label) by highlighting them in a *predefined color*.

- Clipping settings: allows to clip the map item to the atlas feature and to shape and polygon items:
  - *Clip to atlas feature*: you can determine that the layout map item will be clipped automatically to the current *atlas feature*.

    There are different clipping modes available:

    * *Clip During Render Only*: applies a painter based clip, so that portions of vector features which sit outside the atlas feature become invisible
    * *Clip Feature Before Render*: applies the clip before rendering features, so borders of features which fall partially outside the atlas feature will still be visible on the boundary of the atlas feature
    * *Render Intersecting Features Unchanged*: renders all features which intersect the current atlas feature, but without clipping their their geometry.

    You can *Force labels inside atlas feature*. If you don't want to *Clip all layers* to the atlas feature you can use the *Clip selected layers* option.
  - *Clip to item*: it is possible to change the shape of the map item by using a *shape* or *polygon* item from the print layout. When you enable this option the map will be automatically clipped to the selected shape in the combobox. Again, the above mentioned clipping modes are available and labels can be forced to display only inside the clipping shape.



Слика18.15: Clipping a layout map item to shapes

Main properties

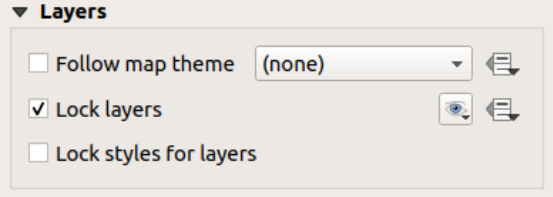
In the *Main properties* group (see Сл.18.14) of the map *Item Properties* panel, available options are:

- The *Update Preview* button to refresh the map item rendering if the view in map canvas has been modified. Note that most of the time, the map item refresh is automatically triggered by the changes;
- The *Scale* to manually set the map item scale;
- The *Map rotation* allows you to rotate the map item content clockwise in degrees. The rotation of the map canvas can be imitated here;

- The *CRS* allows you to display the map item content in any *CRS*. It defaults to `Use project CRS`;
- *Draw map canvas items* lets you show in the print layout *annotations* that are placed on the main map canvas.

Layers

By default, map item appearance is synced with the map canvas rendering meaning that toggling visibility of the layers or modifying their style in the *Layers Panel* is automatically applied to the map item. Because, like any other item, you may want to add multiple map items to a print layout, there's a need to break this synchronization in order to allow showing different areas, layer combinations, at different scales… The *Layers* properties group (see Сл.18.16) helps you do that.



Слика18.16: Map Layers group

If you want to keep the map item consistent with an existing *map theme*, check *Follow map theme* and select the desired theme in the drop-down list. Any changes applied to the theme in QGIS' main window (using the replace theme function) will automatically affect the map item. If a map theme is selected, the *Lock styles for layers* option is disabled because *Follow map theme* also updates the style (symbology, labels, diagrams) of the layers.

To lock the layers shown in a map item to the current map canvas visibility, check *Lock layers*. When this option is enabled, any changes on the layers' visibility in QGIS' main window will not affect the layout's map item. Nevertheless, style and labels of locked layers are still refreshed according to QGIS' main window. You can prevent this by using *Lock styles for layers*.

Instead of using the current map canvas, you can also lock the layers of the map item to those of an existing map theme: select a map theme from the Set layer list from a map theme drop-down button, and the *Lock layers* is activated. The set of visible layers in the map theme is from now on used for the map item until you select another map theme or uncheck the *Lock layers* option. You then may need to refresh the view using the Refresh view button of the *Navigation* toolbar or the *Update Preview* button seen above.

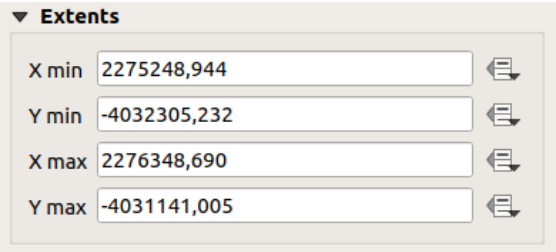
Note that, unlike the *Follow map theme* option, if the *Lock layers* option is enabled and set to a map theme, the layers in the map item will not be refreshed even if the map theme is updated (using the replace theme function) in QGIS' main window.

Locked layers in the map item can also be *data-defined*, using the icon beside the option. When used, this overrides the selection set in the drop-down list. You need to pass a list of layers separated by `|` character. The following example locks the map item to use only layers `layer 1` and `layer 2`:

```
concat ('layer 1', '|', 'layer 2')
```

Extents

The *Extents* group of the map item properties panel provides the following functionalities (see Сл.18.17):



Слика18.17: Map Extents group

The **Extents** area displays `X` and `Y` coordinates of the area shown in the map item. Each of these values can be manually replaced, modifying the map canvas area displayed and/or map item size. The extent can also be modified using tools at the top of the map item panel such as:

- Set map canvas to match main canvas extent
- Set map scale to match main canvas scale

You can also alter a map item extent using the Move item content tool: click-and-drag within the map item to modify its current view, keeping the same scale. With the tool enabled, use the mouse wheel to zoom in or out, modifying the scale of the shown map. Combine the movement with `Ctrl` key pressed to have a smaller zoom.

Elevation range

The *Elevation range* setting in the map item properties allows you to filter the content of certain layers based on a specific elevation range. While all layers remain visible, those that support elevation filtering (currently point clouds and raster DEMs) will have their data filtered. Only the parts of these layers that fall within the elevation range set by the *Lower* and *Upper* values will be displayed.

The *Elevation range* can be data-defined. This means you can set different elevation ranges for different features in the *atlas* or *report*.

Temporal range

The *Temporal range* group of the map item properties panel provides the options to control layers rendering in the map item based on a temporal range. Only layers whose temporal properties overlap with the time range set by the *Start* and *End* dates are displayed in the map item.

The associated data-defined widgets help make the time range dynamic, and allow outputting temporal *atlases*, i.e. automated maps with fixed spatial extent and whose contents vary based on time. For example, using as coverage layer a csv file with a start and end pair of fields and a number of rows representing date ranges, enable both the temporal range and control by atlas in the map item properties and hit atlas export.

Controlled by atlas

The *Controlled by atlas* group properties is available only if an *atlas* is active in the print layout. Check this option if you want the map item being ruled by the atlas; when iterating over the coverage layer, the map item extent is panned/zoomed to the atlas feature following:

- *Margin around features*: zooms to the feature at the best scale, keeping around each a margin representing a percentage of the map item width or height. The margin can be the same for all features or *set variable*, e.g., depending on map scale;
- *Predefined scale (best fit)*: zooms to the feature at the project *predefined scale* where the atlas feature best fits;

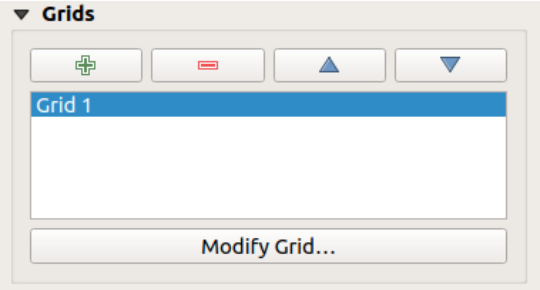
- *Fixed scale*: atlas features are panned from one to another, keeping the same scale of the map item. Ideal when working with features of same size (e.g., a grid) or willing to highlight size differences among atlas features.

Grids

With grids, you can add, over your map, information relative to its extent or coordinates, either in the map item projection or a different one. The *Grids* group provides the possibility to add several grids to a map item.

- With the and buttons you can add or remove a selected grid;
- With the and buttons you can move up and down a grid in the list, hence move it on top or bottom of another one, over the map item.

Double-click the added grid to rename it.



Слика18.18: Map Grids Dialog

To modify a grid, select it and press the *Modify Grid…* button to open the *Map Grid Properties* panel and access its configuration options.

Grid Appearance

In the *Map Grid Properties* panel, check *Grid enabled* to show the grid on the map item.

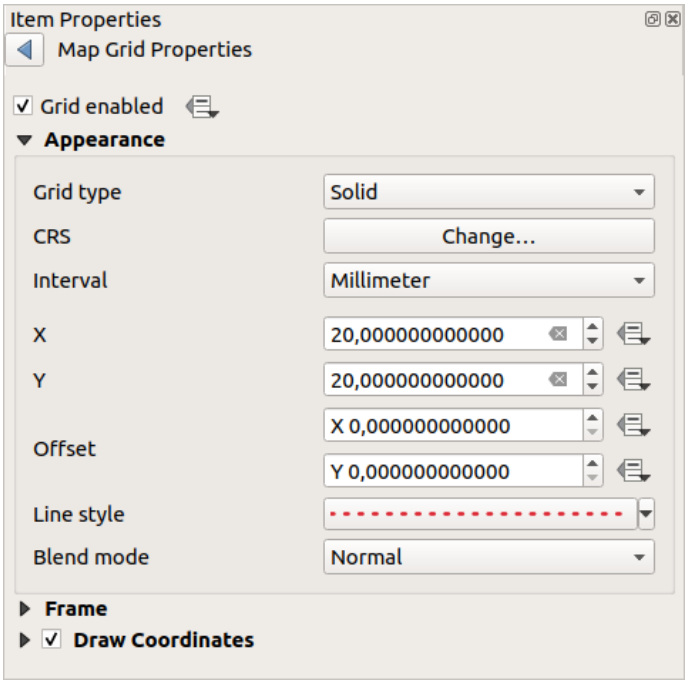
As grid type, you can specify to use a:

- **Solid**: shows a line across the grid frame. The *Line style* can be customized using *color* and *symbol* selector widget;
- **Cross**: displays segment at the grid lines intersection for which you can set the *Line style* and the *Cross width*;
- **Markers**: only displays customizable markers symbol at grid lines intersection;
- or **Frame and annotations only**.

Other than the grid type, you can define:

- the *CRS* of the grid: by default, it will follow the map item CRS. Press Select CRS button to set it to a different CRS.
- the *Interval* type to use for the grid references:
  - **Map Units**: you set a distance within the map (in the unit of the grid CRS) between consecutive grid references in the *X* and *Y* directions. The number of grid ticks will vary depending on the map scale.
  - choosing **Fit Segment Width** will dynamically select the grid interval based on the map extent to a „pretty” interval. That optimal interval is calculated within a range of distances whose *Minimum* and *Maximum* values can be customized.
  - With **Millimeters** or **Centimeters**, you set a distance on the paper between consecutive grid references in the *X* and *Y* directions. The number of grid ticks will be the same whatever the map scale.

- the *Offset* from the map item edges, in the *X* and/or the *Y* direction
- and the *Blend mode* of the grid (see *Blending Modes*) when compatible.



Слика18.19: Grid Appearance Dialog

Grid Frame

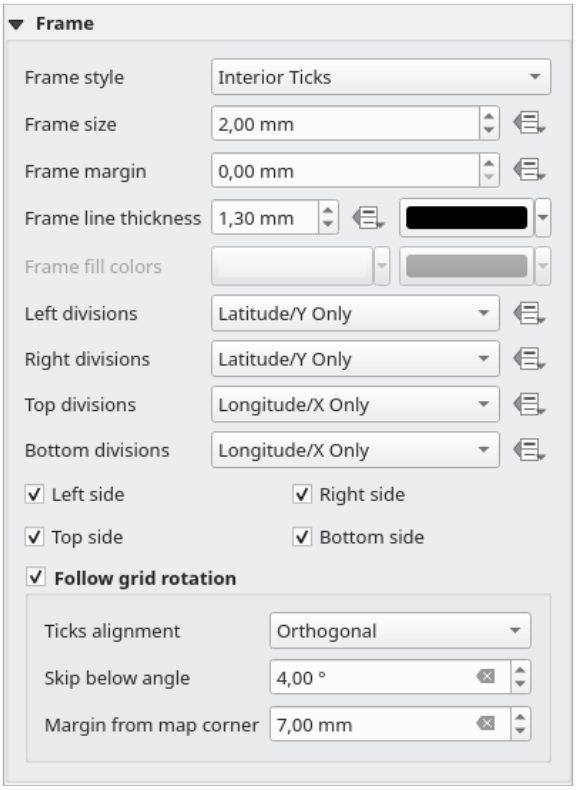
There are different options to style the frame that holds the map. The following options are available: `No Frame`, `Zebra`, `Zebra (nautical)`, `Interior ticks`, `Exterior ticks`, `Interior and Exterior ticks`, `Line border` and `Line border (nautical)`.

When compatible, it's possible to set the *Frame size*, a *Frame margin*, the *Frame line thickness* with associated color and the *Frame fill colors*.

Using `Latitude/Y only` and `Longitude/X only` values in the divisions section you can prevent a mix of latitude/Y and longitude/X coordinates showing on each side when working with rotated maps or reprojected grids. Also you can choose to set visible or not each side of the grid frame.

When the map item extent is rotated (from the *Main properties* group) or the grid has a different CRS applied, grid lines may not be orthogonal to the map item sides. This can result in bad looking of the grid when styled with interior and/or exterior ticks. Checking *Follow grid rotation* will align the ticks with grid lines. Moreover, you can adjust some more properties:

- *Ticks alignment*: The interior and/or exterior ticks will be parallel to their corresponding grid line. Their alignment can be:
  - **Orthogonal**: ticks on the same side end at one line, parallel to the side. This can result e.g. in some ticks getting longer when with a low angle to the frame.
  - **Fixed length**: all ticks have the same length, so they may not align
- *Skip below angle*: prevents displaying ticks for grid lines intersecting the frame border below a specified threshold
- *Margin from map corner*: prevents displaying ticks too close to the map corners, because they could overlap and/or be out of bounds.



Слика18.20: Grid Frame Dialog

Coordinates

The *Draw coordinates* checkbox allows you to add coordinates to the map frame. Displayed values relate to the chosen *grid interval* unit. You can choose the annotation numeric format, the options range from decimal to degrees, minute and seconds, with or without suffix, aligned or not and a custom format using the expression dialog.

For each of the *Left*, *Right*, *Top* and *Bottom* sides of the grid frame, you can indicate:

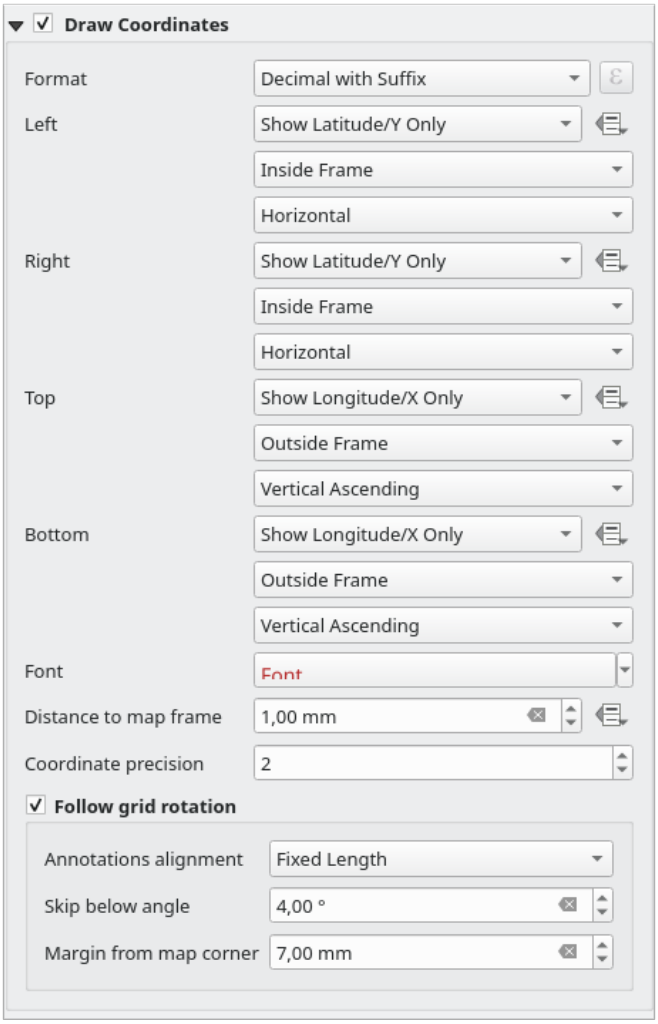
- whether to render the coordinates: **Show all**, **Show latitude/Y only**, **Show longitude/X only**, **Disabled**. Showing only Latitude/Y or Longitude/X values in the divisions helps prevent a mix of latitude/Y and longitude/X coordinates showing on each side when working with rotated maps or reprojected grids.
- the relative position of the text to the grid frame: **Outside frame** or **Inside frame**
- the placement and orientation of the annotation:
  - **Horizontal**
  - **Vertical ascending**, **Vertical descending**
  - **Boundary direction**
  - **Above tick**, **On tick**, **Under tick** when a tick-based frame is used

You can also define the *Font properties* (font, size, color, buffer,…) the *Distance to the map frame* and the *Coordinate precision* (number of decimals) for the drawn annotations.

Follow grid rotation: available when the map extent is rotated or the grid is reprojected, it helps you adjust the annotations placement. Depending on the selected placement mode, the annotations are also rotated:

- *Annotations alignment*: it can be **Orthogonal** or of **Fixed length**
- *Skip below angle*: prevents displaying annotations for grid lines intersecting the frame border below a specified threshold

- *Margin from map corner*: prevents displaying annotations too close to the map corners, because they could overlap and/or be out of bounds.

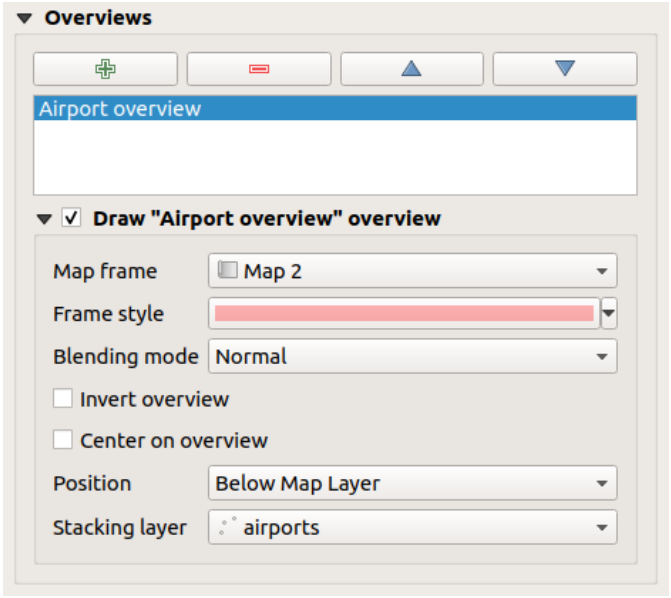


Слика18.21: Grid Draw Coordinates dialog

Overviews

Sometimes you may have more than one map in the print layout and would like to locate the study area of one map item on another one. This could be for example to help map readers identify the area in relation with its larger geographic context shown in the second map.

The *Overviews* group of the map panel helps you create the link between two different maps extent and provides the following functionalities:



Слика18.22: Map Overviews group

To create an overview, select the map item on which you want to show the other map item's extent and expand the *Overviews* option in the *Item Properties* panel. Then press the button to add an overview.

Initially this overview is named 'Overview 1' (see Сл.18.22). You can:

- Rename it with a double-click
- With the and buttons, add or remove overviews
- With the and buttons, move an overview up and down in the list, placing it above or below other overviews in the map item (when they are at the same *stack position*).

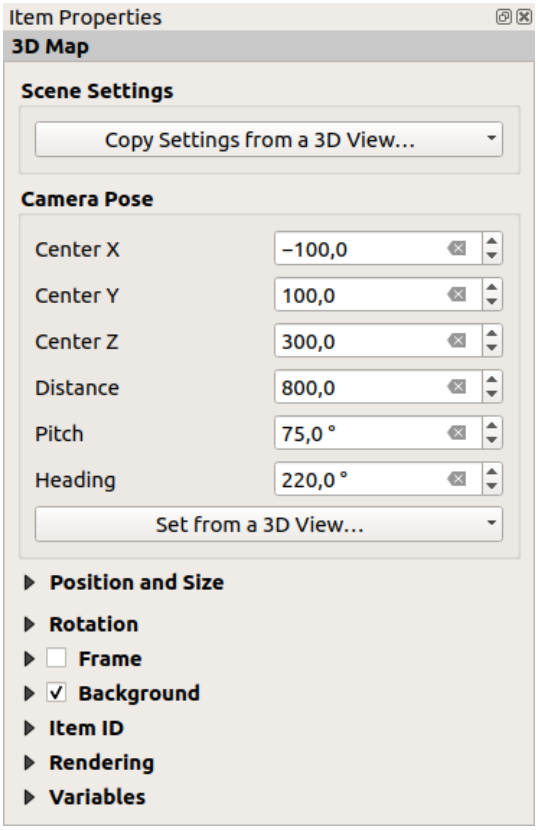
Then select the overview item in the list and check the *Draw „<name_overview>" overview* to enable the drawing of the overview on the selected map frame. You can customize it with:

- The *Map frame* selects the map item whose extents will be shown on the present map item.
- The *Frame Style* uses the *symbol properties* to render the overview frame.
- The *Blending mode* allows you to set different transparency blend modes.
- The *Invert overview* creates a mask around the extents when activated: the referenced map extents are shown clearly, whereas the rest of the map item is blended with the frame fill color (if a fill color is used).
- The *Center on overview* pans the map item content so that the overview frame is displayed at the center of the map. You can only use one overview item to center, when you have several overviews.
- The *Position* controls exactly where in the map item's layer stack the overview will be placed, e.g. allowing an overview extent to be drawn below some feature layers such as roads whilst drawing it above other background layers. Available options are:
  - *Below map*
  - *Below map layer* and *Above map layer*: place the overview frame below and above the geometries of a layer, respectively. The layer is selected in the *Stacking layer* option.
  - *Below map labels*: given that labels are always rendered above all the feature geometries in a map item, places the overview frame above all the geometries and below any label.
  - *Above map labels*: places the overview frame above all the geometries and labels in the map item.

18.2.3 The 3D Map Item

The 3D Map item is used to display a *3D map view*. Use the *Add 3D Map* button, and follow *items creation instructions* to add a new 3D Map item that you can later manipulate the same way as demonstrated in *Interakcija sa stavkama rasporeda*.

By default, a new 3D Map item is empty. You can set the properties of the 3D view and customize it in the *Item Properties* panel. In addition to the *common properties*, this feature has the following functionalities (Сл.18.23):



Слика18.23: 3D Map Item Properties

Scene settings

Press *Copy Settings from a 3D View…* to choose the 3D map view to display.

The 3D map view is rendered with its current configuration (layers, terrain, lights, camera position and angle…).

Camera pose

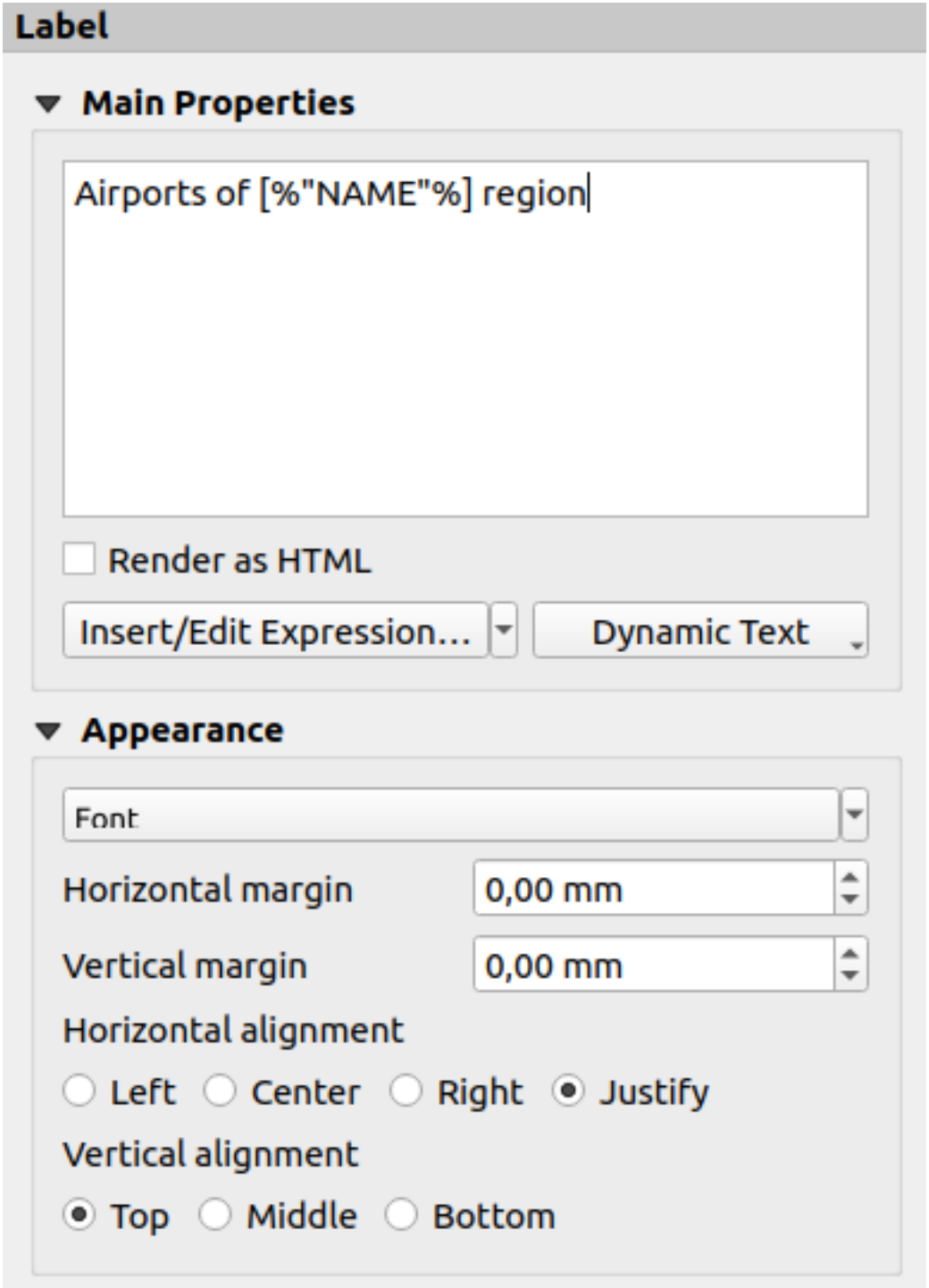
- *Center X* sets the X coordinate of the point the camera is pointing at
- *Center Y* sets the Y coordinate of the point the camera is pointing at
- *Center Z* sets the Z coordinate of the point the camera is pointing at
- *Distance* sets the distance from the camera center to the point the camera is pointing at
- *Pitch* sets the rotation of the camera around the X-axis (vertical rotation). Values from 0 to 360 (degrees). 0°: terrain seen straight from above; 90°: horizontal (from the side); 180°: straight from below; 270°: horizontal, upside down; 360°: straight from above.
- *Heading* sets the rotation of the camera around the Y-axis (horizontal rotation - 0 to 360 degrees). 0°/360°: north; 90°: west; 180°: south; 270°: east.

The *Set from a 3D View…* pull-down menu lets you populate the items with the parameters of a 3D View.

18.2.4 The Label Item

The *Label* item is a tool that helps decorate your map with texts that would help to understand it; it can be the title, author, data sources or any other information… You can add a label with the *Add Label* tool following *items creation instructions* and manipulate it the same way as exposed in *Interakcija sa stavkama rasporeda*.

By default, the label item provides a default text that you can customize using its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.24):



Слика18.24: Label Item Properties Panel

Main properties

The *Main properties* group is the place to provide the text of the label. The text can be static, dynamic with *expression* functions and variables, and/or formatted with HTML. Dynamic parts of a label need to be surrounded by `[%` and `%]` in order to be interpreted and evaluated as such.

- To use expressions in labels, you can click on *Insert/Edit Expression…* button, write your formula as usual and when the dialog is applied, QGIS automatically adds the surrounding characters.

  Путоказ

  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.

  Because maps are usually filled with some common textual information (date, author, title, page number, …), QGIS provides a direct access to the corresponding expressions or variables: press the *Dynamic text* button to select and insert them into your label.

  Савет

  The top menu *Add Item* ► *Add Dynamic Text* ► can be used to create a new label item filled with the selected predefined expression.

  It's possible to turn a dynamic label into static: press the drop-down arrow next to the *Insert/Edit Expression…* button and select *Convert to Static*. Any dynamic parts of the label's contents will be evaluated and replaced with their current values. You can then manually tweak the resulting text when needed.

- Labels can be interpreted as HTML code: check *Render as HTML*. You can now insert HTML tags or styles, URL, a clickable image that links to a web page, or something more complex…

The following code combines HTML rendering with expressions, for an advanced labeling and will output Сл.18.25:

```
<html>
 <head>
   <style>
      /* Define some custom styles, with attribute-based size */
      name {color:red; font-size: [% ID %]px; font-family: Verdana; text-shadow:␣
↪grey 1px 0 10px;}
      use {color:blue;}
   </style>
 </head>

 <body>
   <!-- Information to display -->
   <u>Feature Information</u>
   <ul style="list-style-type:disc">
     <li>Feature Id: [% ID %]</li>
     <li>Airport: <name>[% NAME %]</name></li>
     <li>Main use: <use>[% USE %]</use></li>
   </ul>
   Last check: [% concat( format_date( "control_date", 'yyyy-MM-dd'), ' by <b><i>',
↪ @user_full_name, '</i></b>' ) %]

   <!-- Insert an image -->
```

(наставак на следећој страни)

(настављено са претходне стране)

```
   <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>
```



Слика18.25: Leveraging a label with HTML styling

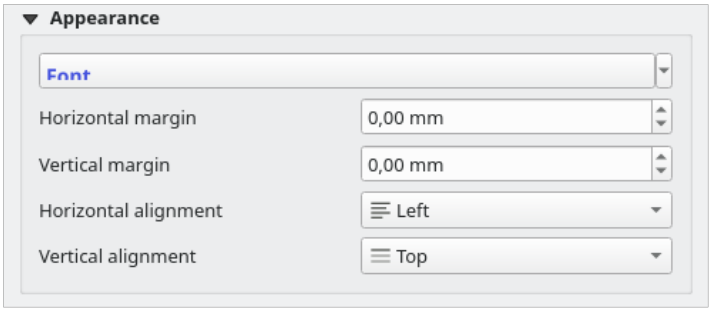
Appearance

- Define font and style of the text by clicking on the *Font* button. In the *Label Font* menu you can use some of the options for *Formatting the label text*.
- You can specify different horizontal and vertical margins in mm. This is the margin from the edge of the layout item. The label can be positioned outside the bounds of the label e.g. to align label items with other items. In this case you have to use negative values for the margin.
- Using the text alignment is another way to position your label. It can be:
  - *Left*, *Center*, *Right* or *Justify* for *Horizontal alignment*
  - and *Top*, *Vertical Center*, *Bottom* for *Vertical alignment*.

Савет

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.



Слика18.26: Label Item Appearance Properties

Exploring expressions in a label item

Below some examples of expressions you can use to populate the label item with interesting information - remember that the code, or at least the calculated part, should be surrounded by `[%` and `%]` in the *Main properties* frame:

- Display a title with the current atlas feature value in „field1”:

```
'This is the map for ' || "field1"
```

  or, written in the *Main properties* section:

```
This is the map for [% "field1" %]
```

- Add a pagination for processed atlas features (eg, `Page 1/10`):

```
concat( 'Page ', @atlas_featurenumber, '/', @atlas_totalfeatures )
```

- Return the name of the airports of the current atlas region feature, based on their common attributes:

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := fk_regionId = attribute( @atlas_feature, 'ID' ),
           concatenator := ', '
         )
```

  Or, if an *attributes relation* is set:

```
relation_aggregate( relation := 'airports_in_region_relation',
                    aggregate := 'concatenate',
                    expression := "NAME",
                    concatenator := ', '
                  )
```

- Return the name of the airports contained in the current atlas region feature, based on their spatial relationship:

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := contains( geometry( @parent ), $geometry ),
           concatenator := ', '
         )
```

  OR:

```
array_to_string( array:= overlay_contains( layer := 'airports',
                                           expression := "NAME" ),
                 delimiter:= ', '
               )
```

- Return the lower X coordinate of the `Map 1` item's extent:

```
x_min( map_get( item_variables( 'Map 1' ), 'map_extent' ) )
```

- Retrieve the name of the layers in the current layout `Map 1` item, and formats in one name by line:

```
array_to_string(
 array_foreach(
  map_get( item_variables( 'Map 1' ), 'map_layers' ), -- retrieve the layers␣
↪list
```

(наставак на следећој страни)

(настављено са претходне стране)

```
  layer_property( @element, 'name' ) -- retrieve each layer name
 ),
 '\n' -- converts the list to string separated by breaklines
)
```

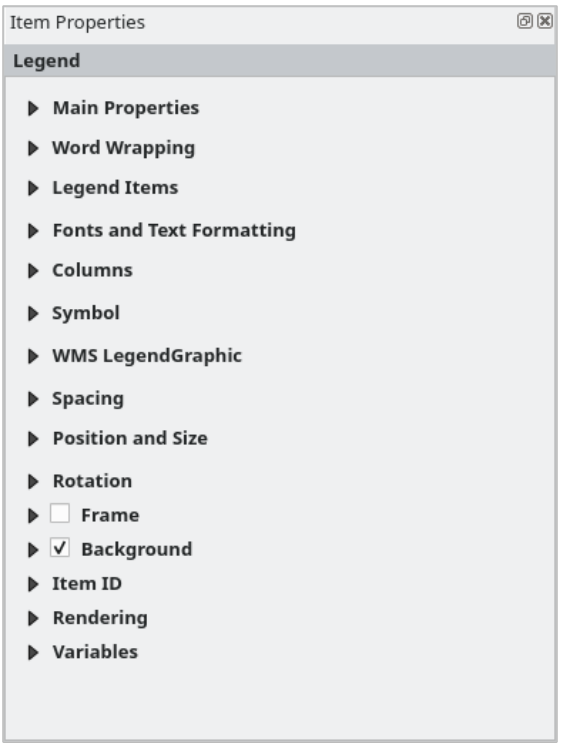
- Display the list of layers with their license strings (usage rights) in a layout `Map 1` item. You need to fill the layers' *Access metadata* properties first.

```
array_to_string( map_credits( 'Map 1', true ) )
```

18.2.5 The Legend Item

The *Legend* item is a box or a table that explains the meanings of the symbols used on the map. A legend is then bound to a map item. You can add a legend item with the *Add Legend* tool following *items creation instructions* and manipulate it the same way as exposed in *Interakcija sa stavkama rasporeda*.

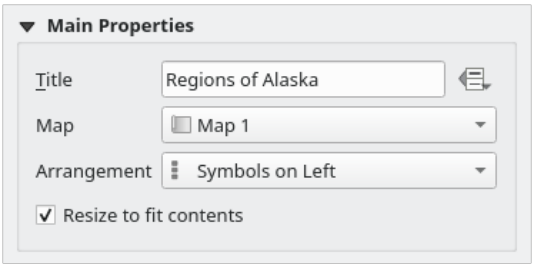
By default, the legend item displays all available layers and can be refined using its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.27):



Слика18.27: Legend Item Properties Panel

Main properties

The *Main properties* group of the legend *Item Properties* panel provides the following functionalities (see Сл.18.28):



Слика18.28: Legend Main properties group

In Main properties you can:

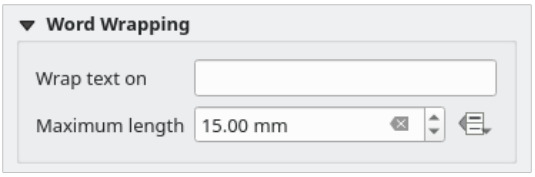
- Change the *Title* of the legend. It can be made dynamic using the *data-defined override* setting, useful for example when generating an atlas;
- Choose which *Map* item the current legend will refer to. By default, the map over which the legend item is drawn is picked. If none, then it falls back to the *reference map*.

> **Белешка**
>
> *Variables* of the linked map item (@map_id, @map_scale, @map_extent…) are also accessible from data-defined properties of the legend.

- Set the symbols and text placement in the legend: the *Arrangement* can be *Symbols on left* or *Symbols on right*. The default value depends on the locale in use (right-to-left based or not).
- Use  *Resize to fit contents* to control whether or not a legend should be automatically resized to fit its contents. If unchecked, then the legend will never resize and instead just stick to whatever size the user has set. Any content which doesn't fit the size is cropped out.

Word wrapping

The *Word wrapping* group of the legend *Item Properties* panel provides the following functionalities:



Слика18.29: Legend's word wrapping group

- Wrap the text of the legend on a given character: each time the character appears, it's replaced with a line break;
- 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

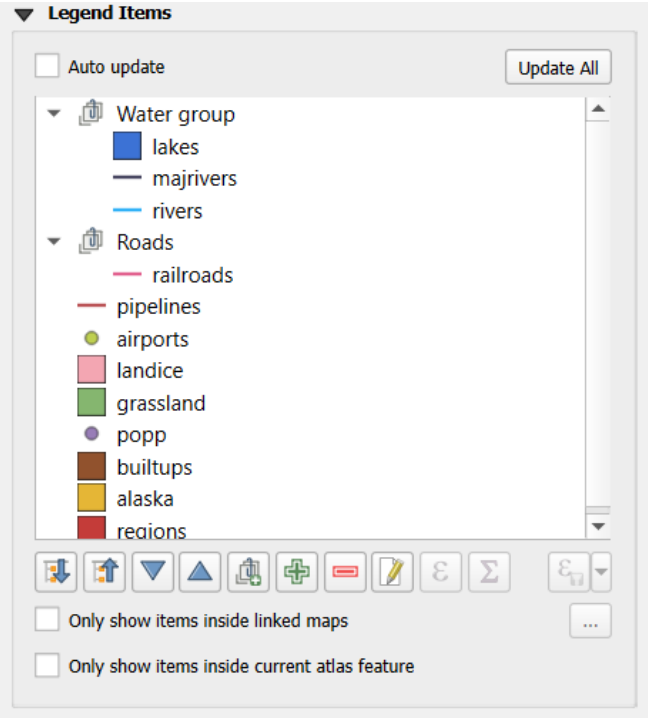
  ```
  IF( @layout_pageheight >= @layout_pagewidth, 35, 80 )
  ```

  – 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
```

Legend items

The *Legend items* group of the legend *Item Properties* panel provides the following functionalities (see Сл.18.30):



Слика18.30: Legend Items group

- The legend will be updated automatically if *Auto update* is checked. When *Auto update* is unchecked this will give you more control over the legend items. All the icons below the legend items list will be activated.
- The legend items window lists all legend items and allows you to change item order, group layers, remove and restore items in the list, edit layer names and symbology and add a filter.
  - Use the Expand all and Collapse all buttons to respectively expand or collapse all the groups and subgroups in the legend tree. Ensure that the *Auto update* is unchecked to use these buttons.
  - The item order can be changed using the and buttons or with 'drag-and-drop' functionality. The order can not be changed for WMS legend graphics.
  - Use the button to add a legend group.
  - Use the button to add layers and button to remove groups, layers or symbol classes.
  - The button is used to edit the layer, group name or title. First you need to select the legend item. Double-clicking the item also opens the text box to rename it.
  - The button uses expressions to customize each symbol label of the selected layer (see *Data-define the legend labels*)

  - The Σ button adds a feature count for each class of vector layer.
  - The Filter legend by expression helps you filter which of the legend items of a layer will be displayed, i.e. using a layer that has different legend items (e.g., from a rule-based or categorized symbology), you can specify a boolean expression to remove from the legend tree, styles that have no feature satisfying a condition. Note that the features are nevertheless kept and shown in the layout map item.

  While the default behavior of the legend item is to mimic the *Layers* panel tree, displaying the same groups, layers and classes of symbology, right-click any item offers you options to hide layer's name or raise it as a group or subgroup. In case you have made some changes to a layer, you can revert them by choosing *Reset to defaults* from the contextual menu of the legend entry.

  After changing the symbology in the QGIS main window, you can click on *Update All* to adapt the changes in the legend element of the print layout.

- With the *Only show items inside linked maps*, only the legend items visible in the linked map will be listed in the legend. If you have more than one map you can click on … and select other maps from your layout. This tool remains available when *Auto-update* is active.
- While generating an atlas with polygon features, you can filter out legend items that lie outside the current atlas feature. To do that, check the *Only show items inside current atlas feature* option.

Data-define the legend labels

ε allows you to add *expressions* to each symbol label of a given layer. New variables (`@symbol_label`, `@symbol_id` and `@symbol_count`) help you interact with the legend entry.

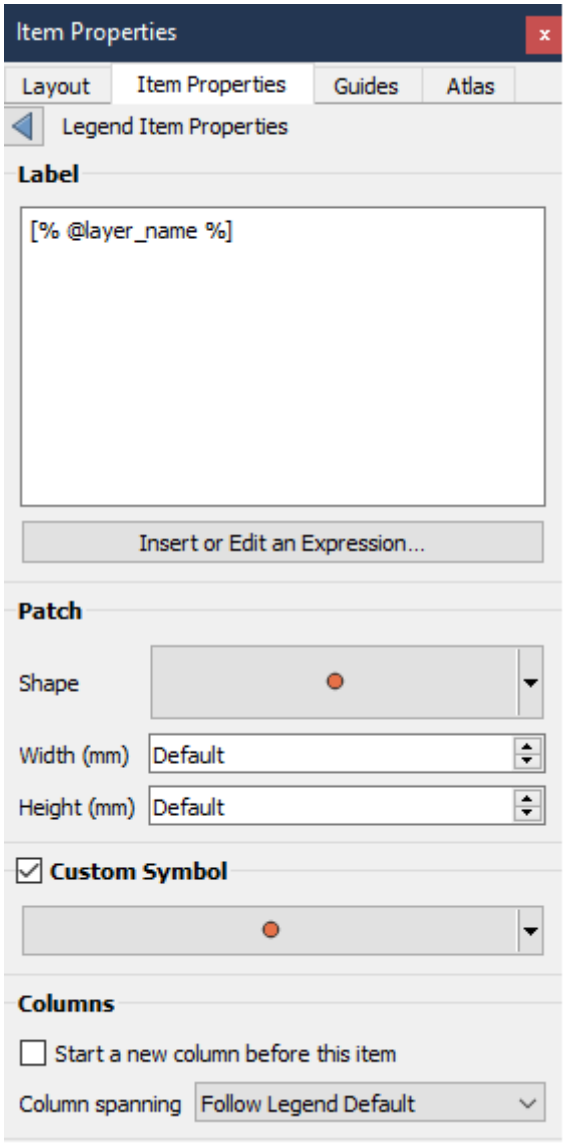
For example, given a `regions` layer categorized by its `type` field, you can append to each class in the legend their number of features and total area, e.g. `Borough (3) - 850ha`:

1. Select the layer entry in the legend tree
2. Press the ε button, opening the *Expression String Builder* dialog
3. Enter the following expression (*assuming symbol labels have not been edited*):

```
format( '%1 (%2) - %3ha',
        @symbol_label,
        @symbol_count,
        round( aggregate(@layer, 'sum', $area, filter:= "type"=@symbol_label)/
↪10000 )
      )
```

4. Pritisnite *OK*

Customizing legend items



Legend items can also be customized individually in the *Legend Items Properties*. But these customization can only be done with *Auto update* disabled.

Double-clicking on an item or pressing Edit selected item properties allows for further customization.

Natpis

For all item types it allows to modify the label text by typing in or by inserting expressions using the *Insert or Edit an Expression*. Expressions can also be added directly anywhere in the item's label by using the [% expression %] notation.

Columns

The Legend Item Property also allows you to control the column splitting behaviour by forcing the column split to occur after a specific item or all symbols of a layer. Automatic splitting of a layer and its child can also be allowed or blocked on a layer-basis in this widget.

Patch

For items with a symbol, the Legend Item Property allows you to specify the maximum height and width that a

symbol can occupy.

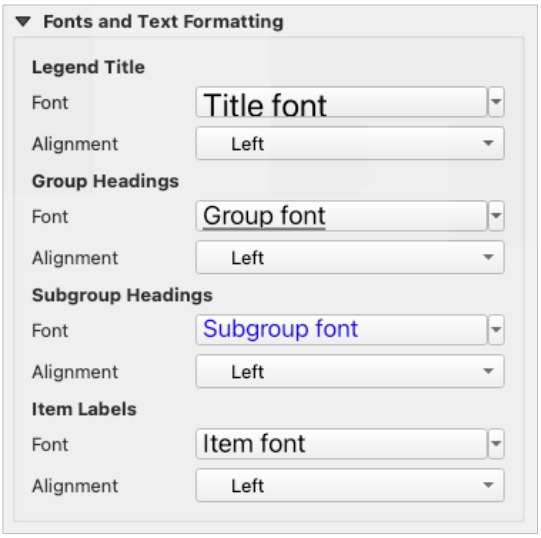
For vector symbols, you can specify a custom shape for the symbol. The shapes are usually defined by an expression to represent the geometry in a simple plane, but those symbols can also be saved in the style manager and imported later. The default symbol for each geometry type can also be controlled via the style manager.

Custom Symbol

A custom symbol can also be specified for vector symbols. This can be useful to tweak the render of a specific symbol, to enhance it in the legend or have a symbol independent from its true symbol preview. This custom symbol will override the legend symbol, but will take into account the symbol *Patch* specified.

Fonts and text formatting

The *Fonts and text formatting* group of the legend *Item Properties* panel provides the following functionalities:



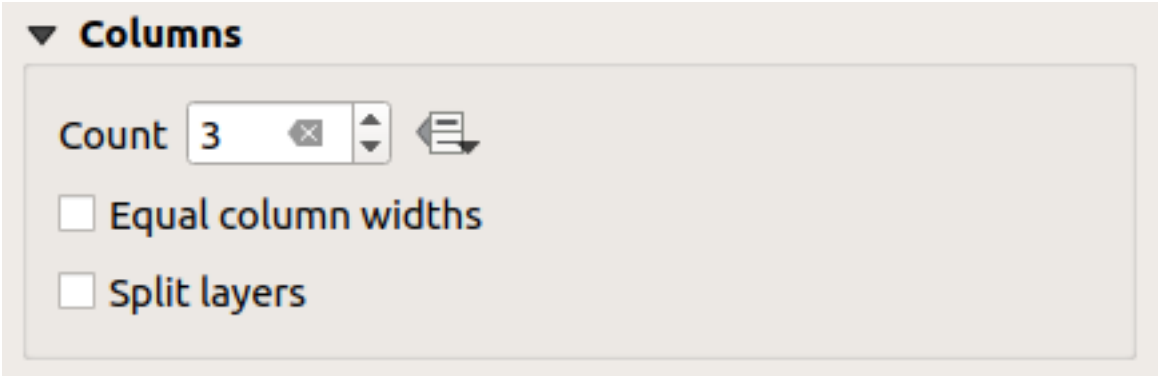
Слика18.31: Legend Fonts properties

- You can change the font of the legend title, group, subgroup and item (feature) in the legend item using the *font selector* widget, providing all the capabilities of *text formatting* (font spacing control, mixed HTML formatting, coloring, blending, background, text buffer, shadow, …)
- For each of these levels you can set the text *Alignment*: it can be *Left* (default for left-to-right based locales), *Center* or *Right* (default for right-to-left based locales).

Columns

Under the *Columns* group of the legend *Item Properties* panel, legend items can be arranged over several columns:

- Set the number of columns in the *Count* 1,00 field. This value can be made dynamic e.g., following atlas features, legend contents, the frame size…
- *Equal column widths* sets how legend columns should be adjusted.
- The *Split layers* option allows a categorized or a graduated layer legend to be divided between columns.

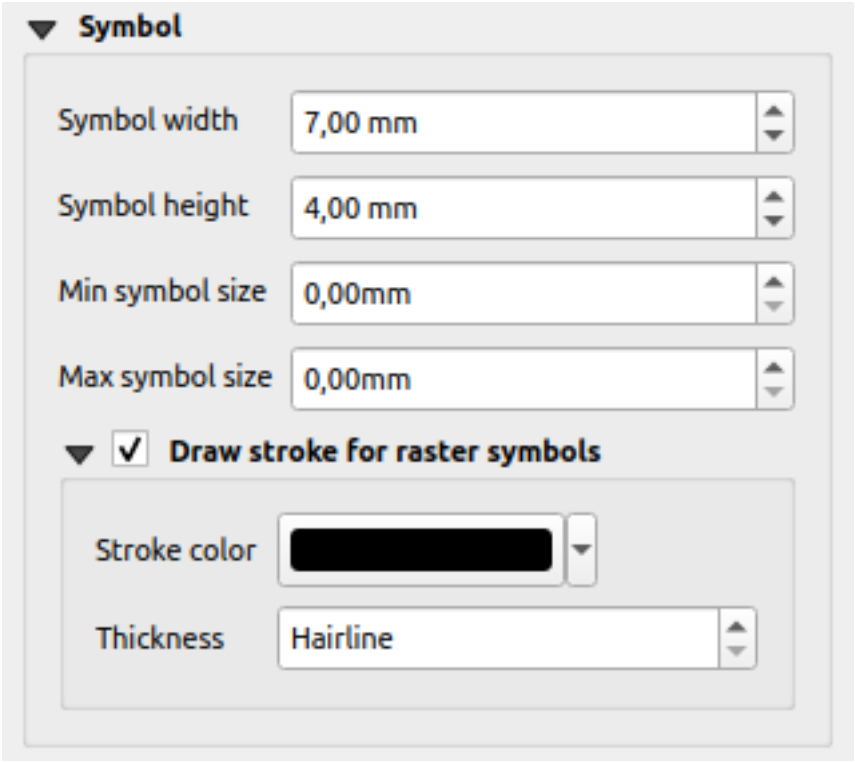


Слика18.32: Legend Columns settings

Symbol

The *Symbol* group of the legend *Item Properties* panel configures the size of symbols displayed next to the legend labels. You can:

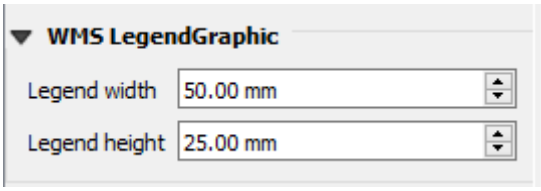
- Set the *Symbol width* and *Symbol height*
- Set the markers' *Min symbol size* and *Max symbol size*: `0.00mm` means there is no value set.
- *Draw stroke for raster symbols*: this adds an outline to the symbol representing the band color of the raster layer; you can set both the *Stroke color* and *Thickness*.



Слика18.33: Legend Symbol configuration

WMS LegendGraphic

The *WMS LegendGraphic* section of the legend *Item Properties* panel provide the following functionalities (see Сл.18.34):

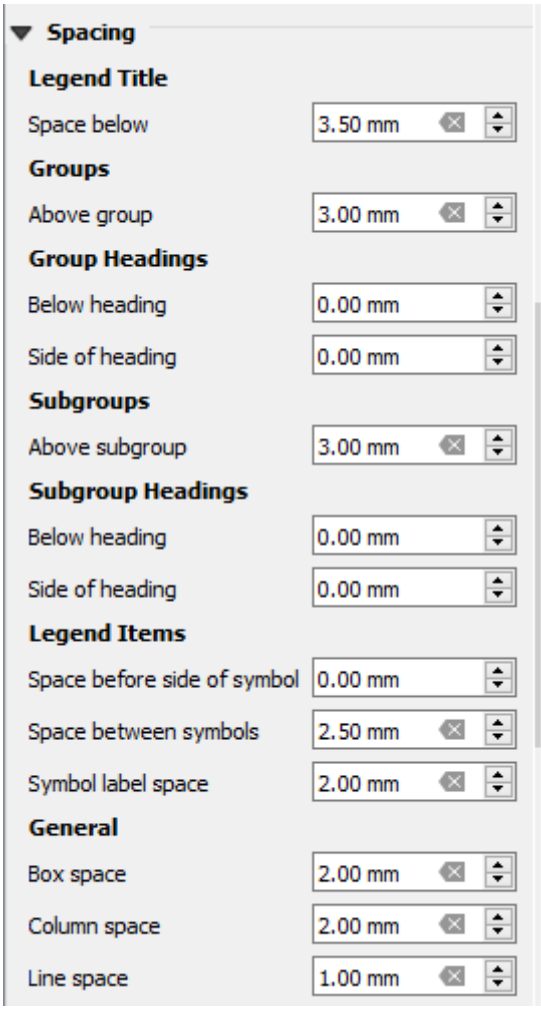


Слика18.34: WMS LegendGraphic

When you have added a WMS layer and you insert a legend item, a request will be sent to the WMS server to provide a WMS legend. This Legend will only be shown if the WMS server provides the GetLegendGraphic capability. The WMS legend content will be provided as a raster image.

WMS LegendGraphic is used to be able to adjust the *Legend width* and the *Legend height* of the WMS legend raster image.

Spacing



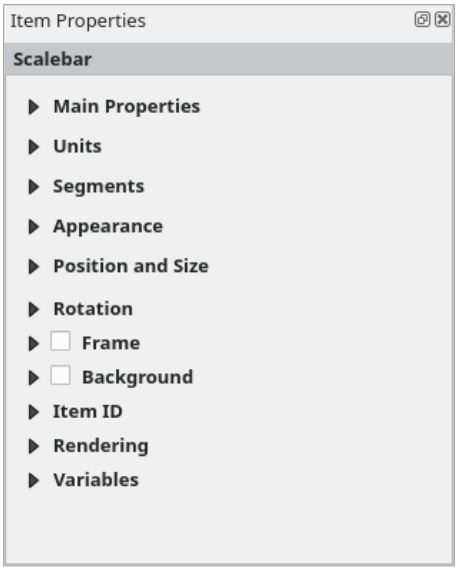
The *Spacing* section allows you to customize the spacing within the legend. Spacing can greatly help denote the groupement of items in the legend and their relation.

Spacing around and before title, groups, subgroups, symbols, labels, boxes, columns and lines can be customized through this dialog.

18.2.6 The Scale Bar Item

Scale bars provide a visual indication of the size of features, and distance between features, on the map item. A scale bar item requires a map item. Use the *Add Scale Bar* tool following *items creation instructions* to add a new scale bar item that you can later manipulate the same way as exposed in *Interakcija sa stavkama rasporeda*.

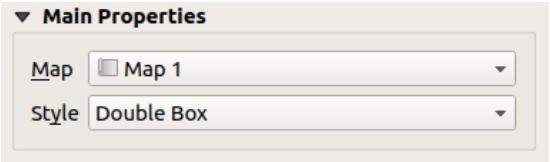
By default, a new scale bar item shows the scale of the map item over which it is drawn. If there is no map item below, the *reference map* is used. You can customize it in the *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.35):



Слика18.35: Scale Bar Item Properties Panel

Main properties

The *Main properties* group of the scale bar *Item Properties* panel provides the following functionalities (see Сл.18.36):

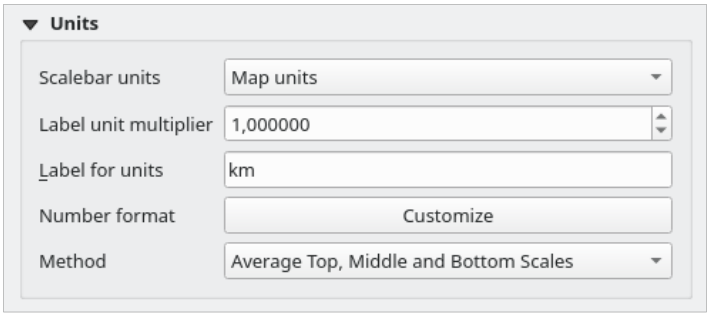


Слика18.36: Scale Bar Main properties group

1. First, choose the map the scale bar will be attached to
2. Then, choose the style of the scale bar. Available styles are:
   - **Single box** and **Double box** styles, which contain one or two lines of boxes alternating colors;
   - **Middle**, **Up** or **Down** line ticks;
   - **Stepped line** style that draws a stepped line representation of a scalebar
   - **Hollow** style that draws a single box with alternating color for the segments, with horizontal lines through alternating segments
   - **Numeric**, where the scale ratio is printed (e.g., `1:50000`).
3. Set properties as appropriate

Units

The *Units* group of the scale bar *Item Properties* panel provides the functionalities to set the units of display and some text formatting (see Сл.18.37):

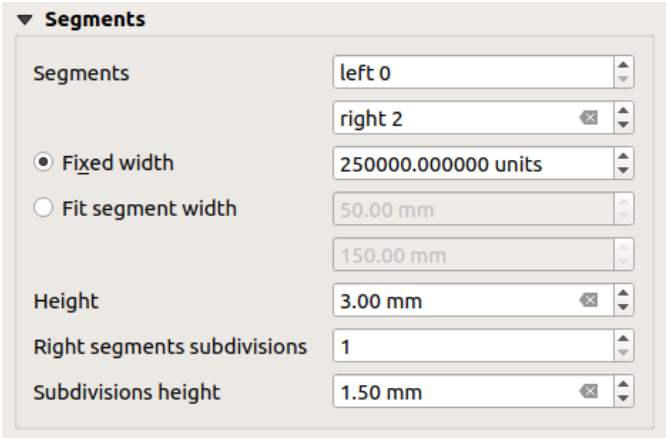


Слика18.37: Scale Bar Units group

- Select the units you want to use with *Scalebar units*. There are many possible choices: **Map Units** (the default one), **Meters**, **Feet**, **Miles** or **Nautical Miles**… and some derivatives. Units conversion is handled automatically.
- The *Label unit multiplier* specifies how many scale bar units per labeled unit. Eg, if your scale bar units are set to „meters", a multiplier of 1000 will result in the scale bar labels in „kilometers".
- The *Label for units* field defines the text used to describe the units of the scale bar, eg `m` or `km`. This should be matched to reflect the multiplier above.
- Press *Customize* next to *Number format* to have control over all the formatting properties for the numbers in the scale bar, including thousand separators, decimal places, scientific notation, etc. (see *Number Formatting* for more details). Very useful in the case of making maps for audiences outside of the current QGIS locale, or when you would like to vary the style from the locale defaults (e.g. adding thousands separators when the locale default is to hide them).
- 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.

Segments

The *Segments* group of the scale bar *Item Properties* panel provides the functionalities to configure the number and size of segments and subdivisions (see Сл.18.38):



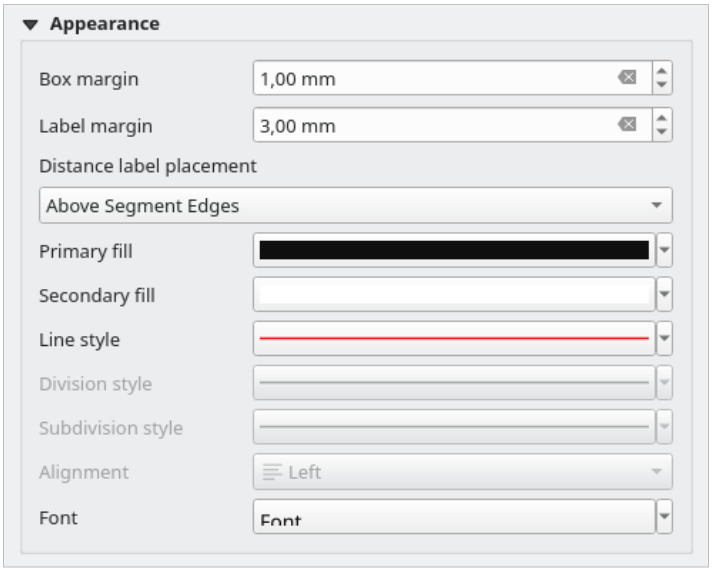
Слика18.38: Scale Bar Segments group

- You can define the number of *Segments* that will be drawn at the left and right sides of the `0` of the scale bar:

  - number of subdivisions of a unique segment on the *Left* side
  - number of segments on the *Right* side
- You can set the width of a segment or a range for the total length of the scale bar:
  - Set how long a segment will be in your scale bar units (*Fixed width*)
  - Or limit the total scale bar size in mm with *Fit segment width* option. In the latter case, each time the map scale changes, the scale bar is resized (and its label updated) to fit between the upper and lower range set.
- *Height* is used to define the height of the bar.
- *Right segment subdivisions* is used to define the number of sections the right-side segments of the scale bar can have (for *Line Ticks Down*, *Line Ticks Middle* and *Line Ticks Up* scale bar styles) .
- *Subdivision height* is used to define the height of the subdivision segment.

Appearance

The *Appearance* group of the scale bar *Item Properties* panel provides the following functionalities:



Слика18.39: Scale Bar Appearance group

You can define how the scale bar will be displayed in its frame.

- *Box margin* : space between text and frame borders
- *Label margin* : space between text and scale bar drawing
- *Distance label placement* defines the position of the text relative to the scale bar segments: *Above Segment Edges*, *Above Segment Centers*, *Below Segment Edges* or *Below Segment Centers*
- *Primary fill* and *Secondary fill* of the scale bar drawing using *fill symbols properties* (color, opacity, patterns, effects…) — for *Single Box*, *Double Box* and *Hollow* styles
- *Line style* of the scale bar drawing using *line symbols properties* (color, stroke, join, cap style, patterns, effects…) — for all but *Numeric* style
- *Division style* and *Subdivision style* respectively for division and subdivision segments in *Line Ticks Up*, *Line Ticks Middle* and *Line Ticks Down* scale bar styles using *line symbols properties* (color, stroke, join, cap style, patterns, effects…)
- *Alignment* puts text on the left, center or right side of the frame (only for *Numeric* scale bar style)

- *Font* to set the *properties* (size, font, color, letter spacing, shadow, background…) of the scale bar label.

Since most of the display properties of the scale bar rely on symbols whose properties can be data-defined, it's possible to render data-defined scale bars.

Example: The following code applied to the bold property of the scale labels will display numbers in bold when they are a multiple of 500:

```
-- returns True (or 1) if the value displayed on the bar
-- is a multiple of 500

@scale_value % 500 = 0
```

18.2.7 The Table Items

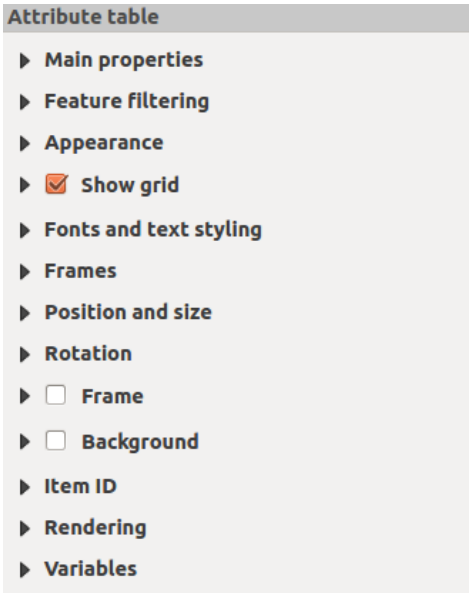
You can use table items to decorate and explain your map:

- *Attribute table*: automatically extracts a subset of the attributes of a layer, based on predefined rules
- *Fixed table*: allows for creation of tables with contents manually entered (i.e. spreadsheet style), so that you can create completely custom tables.

The attribute table item

Any layer in the project can have its attributes shown in the print layout. Use the  *Add Attribute Table* tool following *items creation instructions* to add a new table item that you can later manipulate the same way as exposed in *Interakcija sa stavkama rasporeda*.

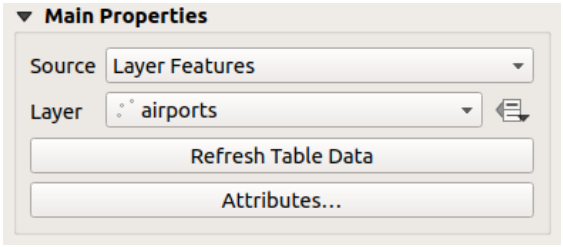
By default, a new attribute table item loads first rows of the first (alphabetically sorted) layer, with all the fields. You can however customize the table thanks to its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.40):



Слика18.40: Attribute table Item Properties Panel

Main properties

The *Main properties* group of the attribute table provides the following functionalities (see Сл.18.41):



Слика18.41: Attribute table Main properties Group

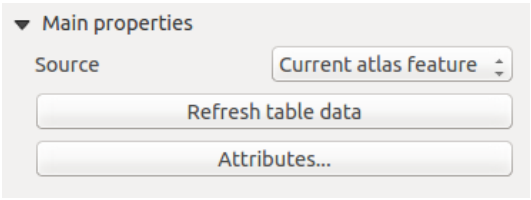
- For *Source* you can by default only select **Layer features** allowing you to select a *Layer* from the vector layers loaded in the project.

  The Data-defined override button near the layer list allows you to dynamically change the layer which is used to populate the table, e.g. you could fill the attribute table with different layer attributes per atlas page. Note that the table structure used (Сл.18.44) is the one of the layer shown in the *Layer* drop-down list and it is left intact, meaning that setting a data defined table to a layer with different field(s) will result in empty column(s) in the table.

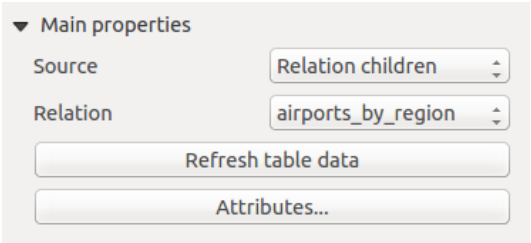
  In case you activate the *Generate an atlas* option in the *Atlas* panel (see *Generate an Atlas*), there are two additional *Source* possible:

  - **Current atlas feature** (see Сл.18.42): you won't see any option to choose the layer, and the table item will only show a row with the attributes from the current feature of the atlas coverage layer.
  - and **Relation children** (see Сл.18.43): an option with the relation names will show up. This feature can only be used if you have defined a *relation* using your atlas coverage layer as parent, and the table will show the children rows of the atlas coverage layer's current feature.

- The button *Refresh Table Data* can be used to refresh the table when the actual contents of the table has changed.

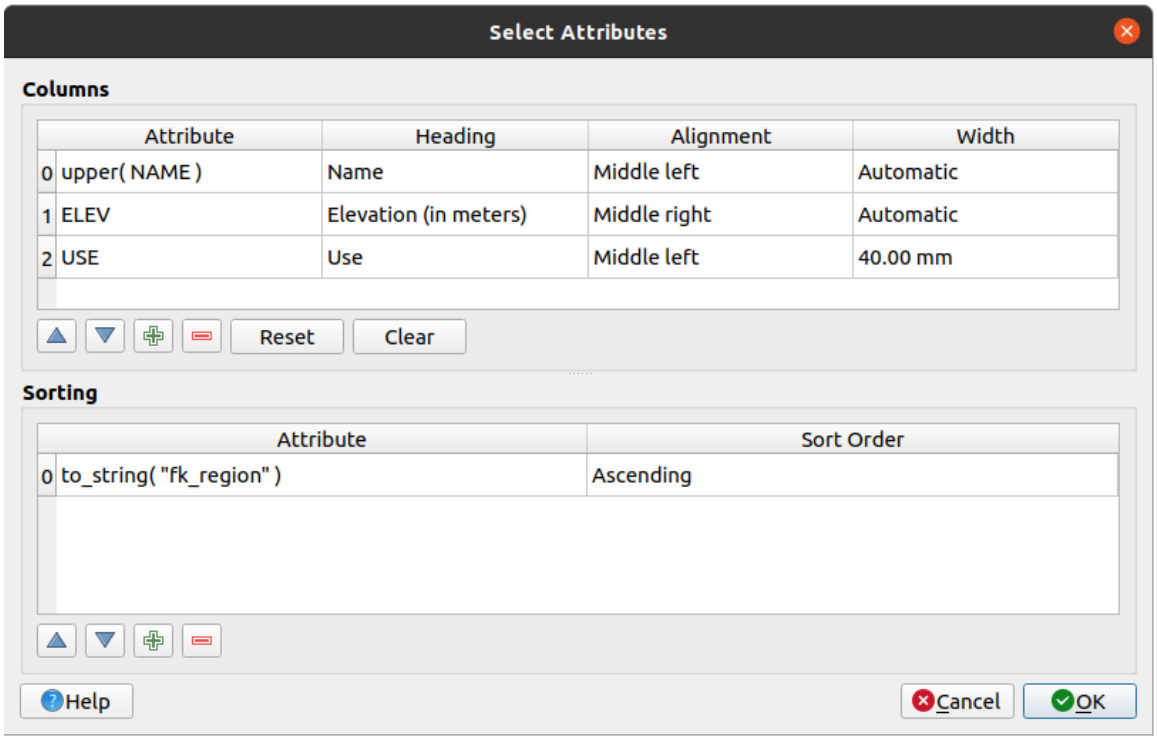


Слика18.42: Attribute table Main properties for 'Current atlas feature'



Слика18.43: Attribute table Main properties for 'Relation children'

- The button *Attributes…* starts the *Select Attributes* dialog, (see Сл.18.44) that can be used to change the visible contents of the table. The upper part of the window shows the list of the attributes to display and the lower part helps you sort the data.



Слика18.44: Attribute table Select attributes Dialog

In the *Columns* section you can:

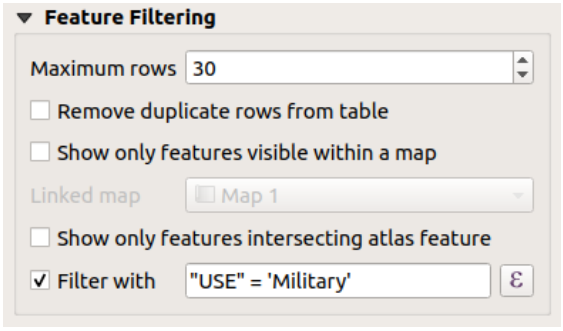
- Move attributes up or down the list by selecting the rows and then using the and buttons to shift the rows. Multiple rows can be selected and moved at any one time.
- Add an attribute with the button. This will add an empty row at the bottom of the table where you can select a field to be the attribute value or create an attribute via a regular expression.
- Remove an attribute with the button. Multiple rows can be selected and removed at any one time.
- Reset the attribute table back to its default state with the *Reset* button.
- Clear the table using the *Clear* button. This is useful when you have a large table but only want to show a small number of attributes. Instead of manually removing each row, it may be quicker to clear the table and add the rows needed.
- Cell headings can be altered by adding the custom text in the *Heading* column.
- Cell alignment can be managed with the *Alignment* column which will dictate the texts position within the table cell.
- Cell width can be manually managed by adding custom values to the *width* column.

In the *Sorting* section you can:

- Add an attribute to sort the table with: press the button and a new empty row is added. Insert a field or an expression in the *Attribute* column and set the *Sort order* to **Ascending** or **Descending**.
- Select a row in the list and use the and buttons to change the sort priority on attribute level. Selecting a cell in the *Sort Order* column helps you change the sorting order of the attribute field.
- Use the button to remove an attribute from the sorting list.

Feature filtering

The *Feature filtering* group of the attribute table provides the following functionalities (see Сл.18.45):



Слика18.45: Attribute table Feature filtering Group

You can:

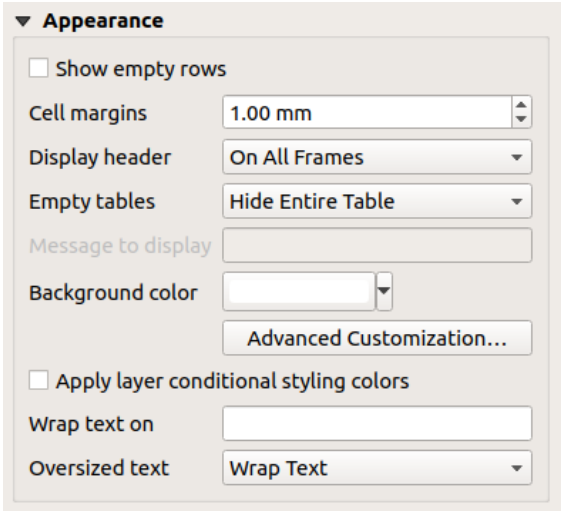
- Define the *Maximum rows* to be displayed.
- Activate *Remove duplicate rows from table* to show unique records only.
- Activate *Show only visible features within a map* and select the corresponding *Linked map* whose visible features attributes will be displayed.
- Activate *Show only features intersecting Atlas feature* is only available when *Generate an atlas* is activated. When activated it will show a table with only the features which intersect the current atlas feature.
- Activate *Filter with* and provide a filter by typing in the input line or insert a regular expression using the given expression button. A few examples of filtering statements you can use when you have loaded the airports layer from the Sample dataset:

  - `ELEV > 500`
  - `NAME = 'ANIAK'`
  - `NAME NOT LIKE 'AN%'`
  - `regexp_match( attribute( $currentfeature, 'USE' ) , '[i]')`

  The last regular expression will include only the airports that have a letter 'i' in the attribute field 'USE'.

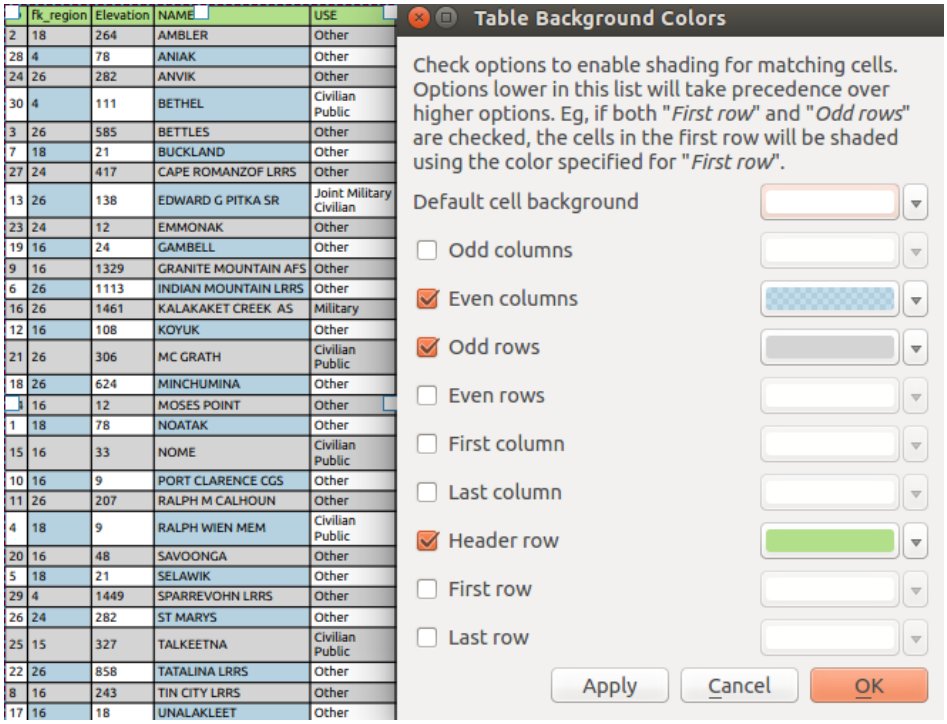
Appearance

The *Appearance* group of the attribute table provides the following functionalities (see Сл.18.46):



Слика18.46: Attribute table appearance Group

- Click *Show empty rows* to fill the attribute table with empty cells. This option can also be used to provide additional empty cells when you have a result to show!
- With *Cell margins* you can define the margin around text in each cell of the table.
- With *Display header* you can select from a list one of 'On first frame', 'On all frames' default option, or 'No header'.
- The option *Empty table* controls what will be displayed when the result selection is empty.
  - **Draw headers only**, will only draw the header except if you have chosen 'No header' for *Display header*.
  - **Hide entire table**, will only draw the background of the table. You can activate *Don't draw background if frame is empty* in *Frames* to completely hide the table.
  - **Show set message**, will draw the header and adds a cell spanning all columns and display a message like 'No result' that can be provided in the option *Message to display*
- The option *Message to display* is only activated when you have selected **Show set message** for *Empty table*. The message provided will be shown in the table in the first row, when the result is an empty table.
- With *Background color* you can set the background color of the table using the *color selector* widget. The *Advanced customization* option helps you define different background colors for each cell (see Сл.18.47)



Слика18.47: Attribute table Advanced Background Dialog

- *Apply layer conditional styling colors*: the *conditional table formatting* present in the layer is applied inside the layout attribute table (background color, font family and properties such as bold, italic, strikeout, underline, color, …). Conditional formatting rules take precedence over other layout table formatting settings, e.g. they will override other cell background color settings such as alternating row colors.
- With the *Wrap text on* option, you can define a character on which the cell content will be wrapped each time it is met
- With *Oversized text* you define the behavior when the width set for a column is smaller than its content's length. It can be **Wrap text** or **Truncate text**.

Белешка

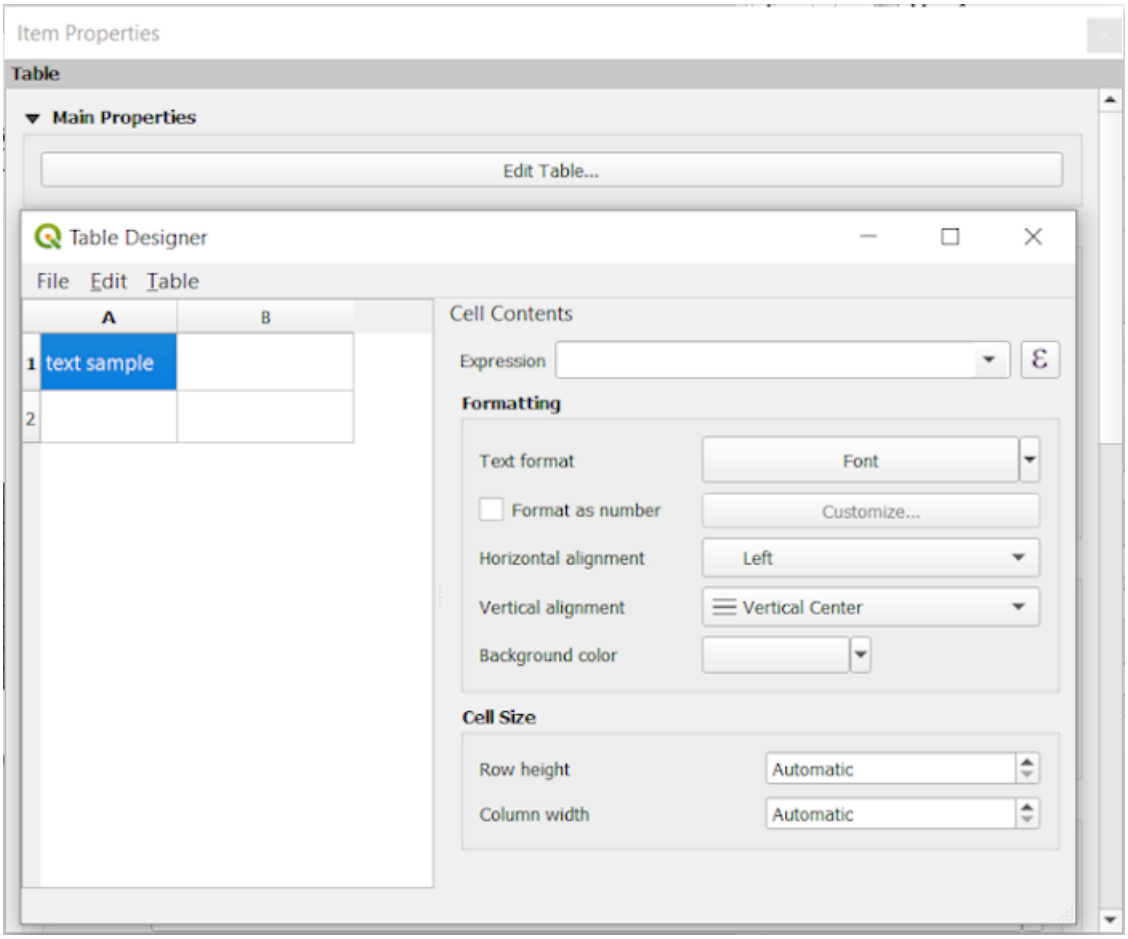
More properties of the attribute table item are described in the *Tables common functionalities* section.

The fixed table item

Additional information about the map can be inserted manually into a table by choosing *Add Fixed Table* and by following *items creation instructions* to add a new table item that you can later manipulate the same way as exposed in *Interakcija sa stavkama rasporeda*.

By default, an empty table with two minimized columns and rows appears in the map layout. You have to customize the table in the *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities:

Main properties



Слика18.48: Fixed table Item Properties Panel with Table designer

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.

- Through the menus on top of the dialog, it is possible to:
  - *File* ► *Import Content From Clipboard*: it overrides contents of the table.
  - *File* ► *Close editor*: it closes the *Table Designer* dialog.
  - From the *Edit* menu, you can work with selection functionalities for rows and columns:
    - *Select all* cells in the table
    - *Select columns* or *Select Rows*: when cells are selected in the table, you can extend the selection respectively to their column(s) or row(s)
    - *Clear Cells*: deletes the content of the selected cells
  - The *Table* menu is the place where you design the structure of the table. You can:
    - *Insert rows* ► above or below the selection
    - *Insert columns* ► before or after the selection
    - *Delete Rows* or *Delete Columns* of selected cells
    - *Merge selected cells*: select multiple cells in a rectangular shape and you can merge them and concatenate their contents into the top left cell (whose styling is also applied to the merged cell, unless there is a last row/column cell in the selection).
    - *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.
    - *Include Header Row* whose styling can only be controlled from the *Fonts and text styling* widget
- 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*:
    - by choosing the *Text format*
    - by *Format as number* (several formats are available)
    - by defining the *Horizontal alignment* and the *Vertical alignment*
    - by choosing a *Background color*
  - Define the *Cell Size* with *Row height* and *Column width*.

Appearance

The *Appearance* group of the fixed table provides the following functionalities:

- Click *Show empty rows* to fill the attribute table with empty cells.
- With *Cell margins* you can define the margin around text in each cell of the table.
- With *Display header* you can select from a list one of 'On first frame', 'On all frames' default option, or 'No header'.
- With *Background color* you can set the background color of the table using the *color selector* widget. The *Advanced customization* option helps you define different background colors for each cell.
- With *Oversized text* you define the behavior when the width set for a column is smaller than its content's length. It can be **Wrap text** or **Truncate text**.

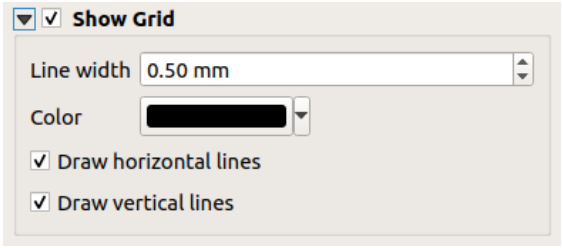
Белешка

More properties of the fixed table item are described in the *Tables common functionalities* section.

Tables common functionalities

Show grid

The *Show grid* group of the table items provides the following functionalities (see Сл.18.49):



Слика18.49: Attribute table Show grid Group

- Activate *Show grid* when you want to display the grid, the outlines of the table cells. You can also select to either *Draw horizontal lines* or *Draw vertical lines* or both.
- With *Line width* you can set the thickness of the lines used in the grid.
- The *Color* of the grid can be set using the color selection widget.

Fonts and text styling

The *Fonts and text styling* group of the table items provides the following functionalities (see Сл.18.50):

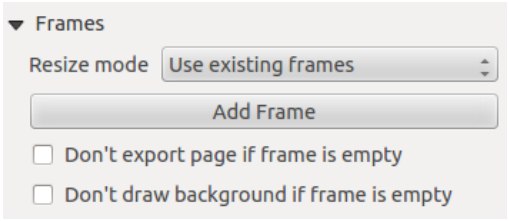


Слика18.50: Attribute table Fonts and text styling Group

- You can define *Font* properties for *Table heading* and *Table contents*, using the advanced *text settings* widget (with buffer, shadow, paint effects, transparence, background, coloring, …). Note that these changes do not affect the cells that have custom font assigned, either from the *Appearance* section or the *Table Designer* dialog. Only cells with the default rendering are overwritten.
- 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 Сл.18.44).

Frames

The *Frames* group of the table item properties provides the following functionalities (see Сл.18.51):



Слика18.51: Attribute table Frames Group

- With *Resize mode* you can select how to render the attribute table contents:
  - `Use existing frames` displays the result in the first frame and added frames only.
  - `Extend to next page` will create as many frames (and corresponding pages) as necessary to display the full selection of attribute table. Each frame can be moved around on the layout. If you resize a frame, the resulting table will be divided up between the other frames. The last frame will be trimmed to fit the table.
  - `Repeat until finished` will also create as many frames as the *Extend to next page* option, except all frames will have the same size.
- Use the *Add Frame* button to add another frame with the same size as selected frame. The result of the table that will not fit in the first frame will continue in the next frame when you use the Resize mode `Use existing frames`.
- Activate *Don't export page if frame is empty* prevents the page to be exported when the table frame has no contents. This means all other layout items, maps, scalebars, legends etc. will not be visible in the result.
- Activate *Don't draw background if frame is empty* prevents the background to be drawn when the table frame has no contents.

18.2.8 The Marker, Picture and North Arrow Items

- *The Picture Item*
  - *Main properties*
  - *Size and placement*
  - *Image rotation*
- *The North Arrow Item*
- *The Marker Item*

Along with the map or legend items in the print layout, you may want to decorate your realization with images or annotations. QGIS provides different tools to achieve this:

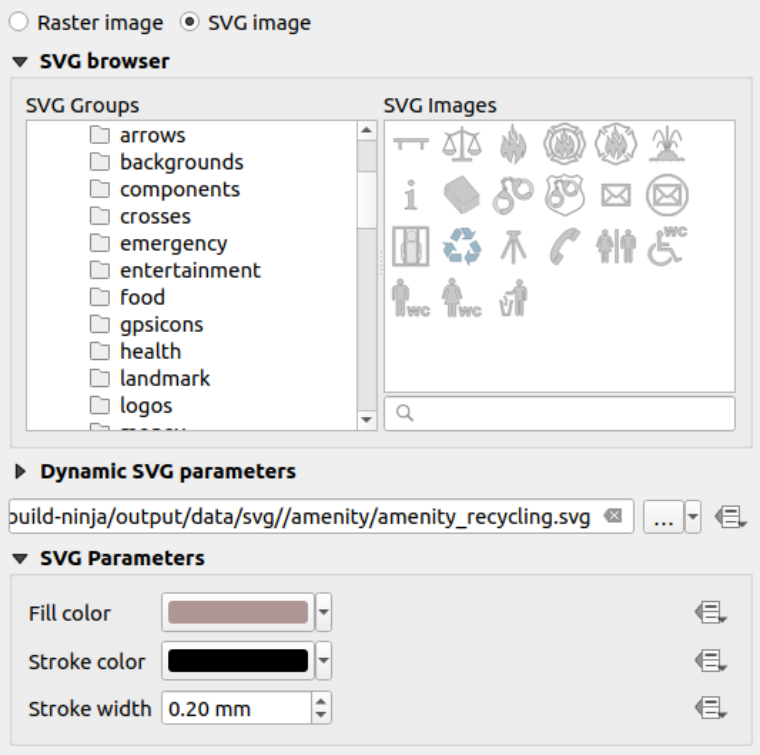
- the *picture item*: decorates the layout with an image raster or SVG file (e.g. logos, pictures, north arrows, …)
- the *north arrow item*: a picture item predefined with a north arrow image
- the *marker item*: decorates the layout with QGIS vector *symbols*. It can be used to place markers over a map item or for creation of advanced custom legends.

The Picture Item

You can add a picture by dragging it from your file manager onto the canvas, pasting it directly into the layout by using `Ctrl+V` or *Edit* ► *Paste* and by using the Add Picture, following *items creation instructions*. Then you can manipulate it, as explained in *Interakcija sa stavkama rasporeda*.

When using Add Picture, the picture item will be a blank frame that you can customize using its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities:

Main properties



Слика18.52: Picture Item Properties panel

The picture item supports two types of images:

- *Raster Image*: a file selector widget can be used to fetch the data. Use the … Browse button to select a file on your computer or enter the path directly in the text field. You can even provide a remote URL that points to a picture. The associated image can also be *embedded* in the layout.

  Use the data defined override button to set the image source from a feature attribute or using a regular expression.

- *SVG Image*: using by default the SVG libraries provided in *Settings* ► *Options* ► *System* ► *SVG Paths*. You can however use any other file, and the file selection follows the same rules as for the raster image. The SVG parameters can as well be set dynamic.

  The QGIS provided (default) `.SVG` files are customizable, meaning that you can easily apply other *Fill color*, *Stroke color* (including opacity) and *Stroke width* than the original, using their corresponding feature in the *SVG Parameters* group. These properties can also be *data-defined*.

  If you add an `.SVG` file that does not enable these properties, you may need to add the following tags to the file in order to add support e.g. for transparency:

  – *fill-opacity="param(fill-opacity)"*

– *stroke-opacity="param(outline-opacity)"*

More details at *Parametrizable SVG*.

Белешка

Drag-and-drop an image file (raster or SVG) into the layout page will create a layout picture item with corresponding settings.

Size and placement



Слика18.53: Layout pictures size and placement properties

With the *Resize mode* option, you can set how the image is displayed when the frame is resized:

- `Zoom`: enlarges/reduces the image to the frame while maintaining the aspect ratio of picture
- `Stretch`: stretches the image to fit inside the frame
- `Clip`: use this mode for raster images only, it sets the size of the image to the original image size without scaling, and the frame is used to clip the image. So only the part of the image that is inside the frame will be visible.
- `Zoom and resize frame`: enlarges the image to fit the frame, and then resizes frame to fit the resulting image dimensions
- `Resize frame to image size`: sets the size of the frame to match the original size of the image (no scaling)

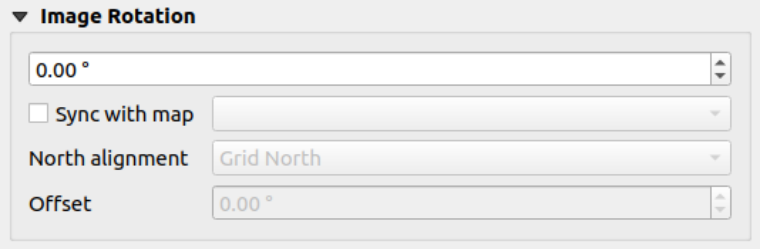
Depending on the selected *Resize mode*, the *Placement* and *Image rotation* options may be disabled. *Placement* lets you select the position of the image inside its frame (top/middle/bottom and left/center/right).

Image rotation

Images can be rotated with the *Image rotation* field. Activating the *Sync with map* checkbox synchronizes the rotation of the image with the rotation applied to the selected map item. This is a convenient feature to make any picture behave as a north arrow. The *North alignment* can be:

- **Grid north**: the direction of a grid line which is parallel to the central meridian of the national/local grid
- **True north**: direction of a meridian of longitude.

You can also apply a declination *Offset* to the picture rotation.



Слика18.54: Layout pictures image rotation properties

The North Arrow Item

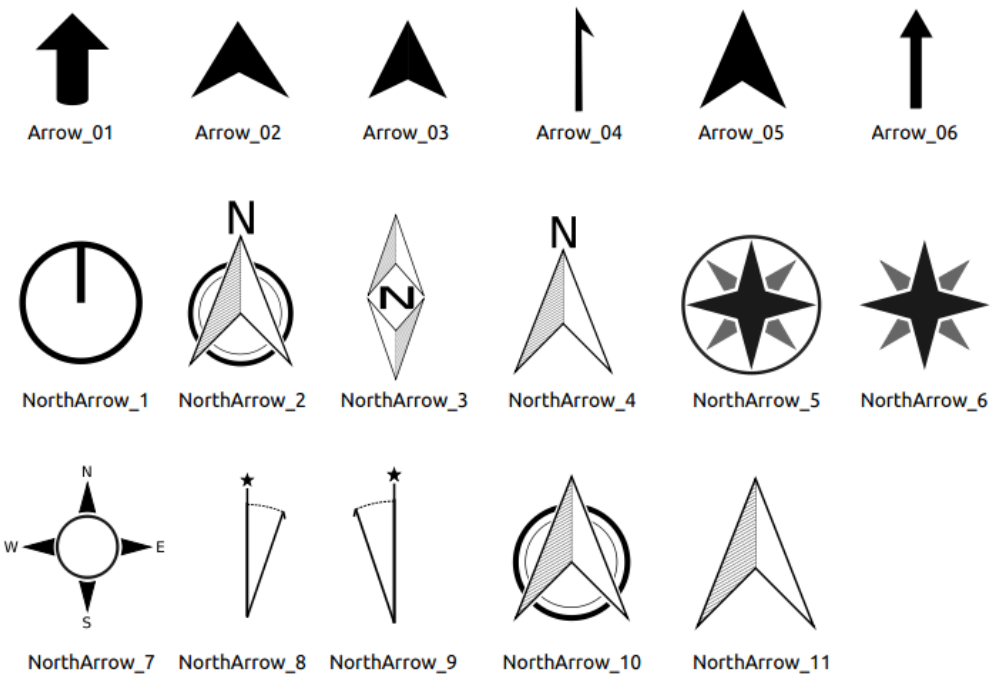
You can add a north arrow with the Add North Arrow button, following *items creation instructions* and manipulate it the same way as exposed in *Interakcija sa stavkama rasporeda*.

Since north arrows are images, the *North Arrow* item has the same properties as the *picture item*. The main differences are:

- A default north arrow is used when adding the item, instead of a blank frame
- The north arrow item is synced with a map item by default: the *Sync with map* property is the map over which the north arrow item is drawn. If none, it falls back to the *reference map*.

Белешка

Many of the north arrows do not have an 'N' added in the north arrow. This is done on purpose, since there are languages that do not use an 'N' for North.



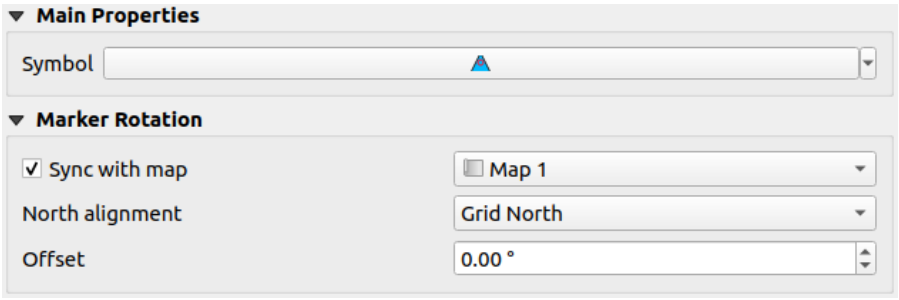
Слика18.55: North arrows available for selection in provided SVG library

The Marker Item

To add a marker item, select the  Add Marker button, and click on the page. A default point marker symbol is added. Then you can manipulate it, as explained in *Interakcija sa stavkama rasporeda*. But note that unlike most of the other items, you resize the item given that its size is controlled by the embedded symbols properties.

The marker item can be customized from the *Item Properties* panel. Other than the *items common properties*, you can also:

- modify the *Symbol*, relying on all the symbol *widget capabilities*
- sync the marker item rotation with the map's (see *Image rotation*), acting as a north arrow. The map rotation is added to any existing marker symbol level rotation (so .e.g if you have to rotate the triangle marker 90° to get it pointing straight up, it will still work nicely in north arrow mode!)

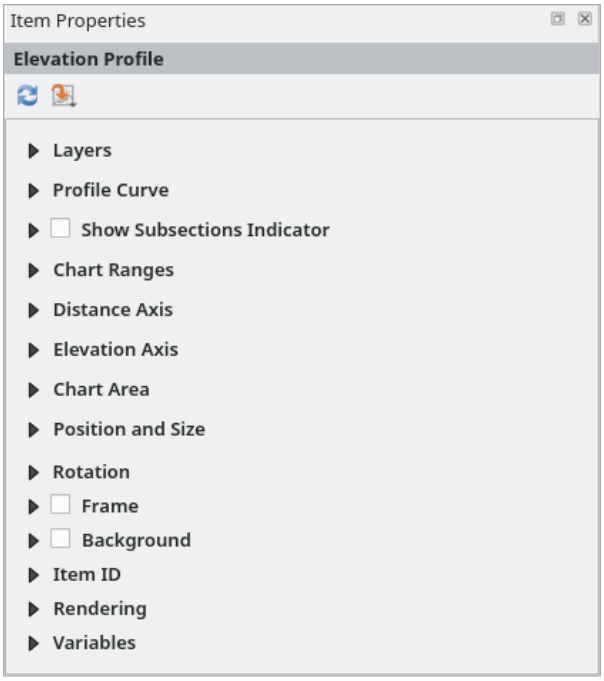


Слика18.56: The marker item custom properties

18.2.9 The Elevation Profile Item

The Elevation Profile item is used to display an *elevation profile view* in a layout. Use the  *Add Elevation Profile* button, and follow *items creation instructions* to add a new elevation profile item that you can later manipulate the same way as demonstrated in *Interakcija sa stavkama rasporeda*.

A new elevation profile item has default settings that make it render an empty chart. You can customize its properties in the *Item Properties* panel. In addition to the *common properties*, this feature has the following functionalities:



Слика18.57: Elevation Profile Item Properties

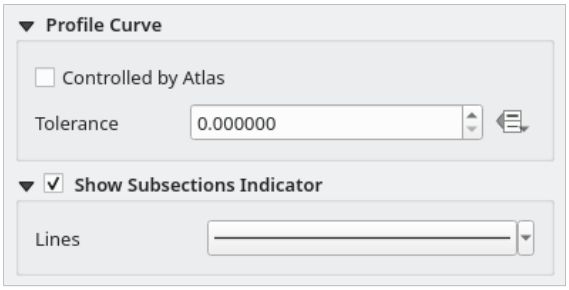
The Elevation profile *Item Properties* panel embeds a top toolbar with the following functionalities:

- Update elevation profile to refresh the item rendering
- Copy from elevation profile: a drop-down menu to select an elevation profile view from. The view settings are applied to the layout elevation profile item, and can be later modified.

Layers

Under the *Layers* group, check in the tree view the layers you would like to render in the profile item. Remember to properly configure the *Elevation* properties of the selected layers.

Profile curve and subsection indicators



Слика18.58: Elevation Profile Curve and Subsections Properties

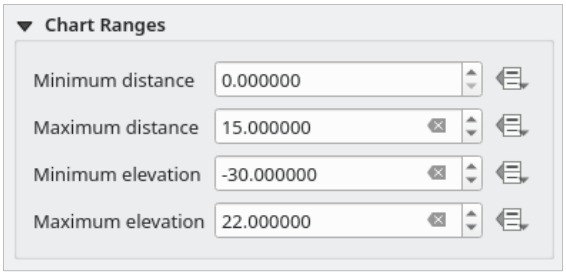
Under the *Profile curve* group, you can:

- Toggle *Controlled by atlas*: the *profile curve* will be taken from the current atlas feature and the elevation profile view updated as you walk through the atlas features. This is currently supported for an active layout atlas or report, using a line geometry type coverage layer.

- The *Tolerance* distance, which can be data-defined, helps you control how far a feature of the visible layers should be from the profile curve in order to display in the layout elevation view. Only point features are currently returned.

Under the *Show Subsections Indicator* group, It is possible to highlight the position of the vertices of the curve on the Elevation Profile. Activate *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.

Chart ranges

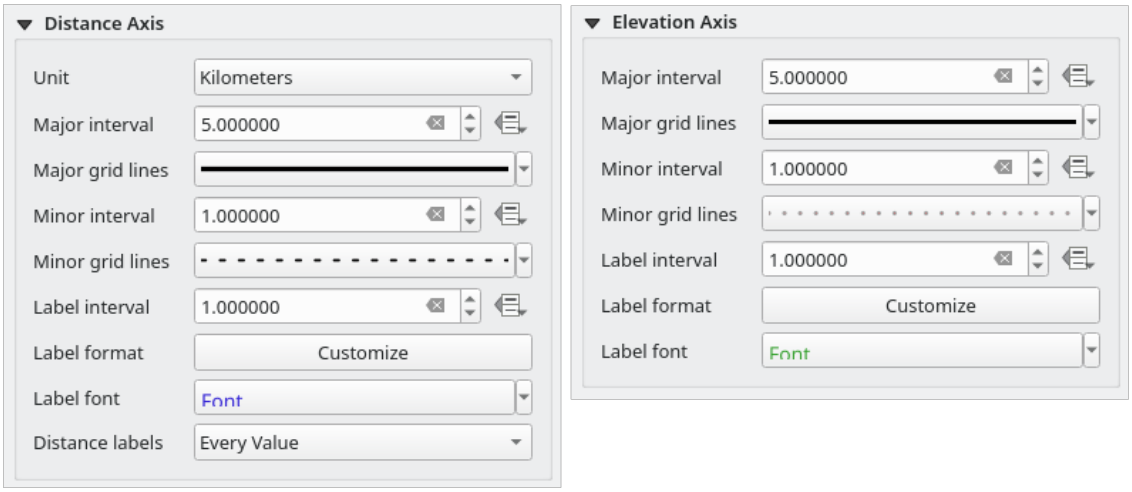


Слика18.59: Elevation Profile Chart Ranges Properties

A layout elevation profile item does not necessarily display the full extent of the elevation profile view it is based on. You can limit the area to render, providing:

- on the X axis, the *Minimum distance* and *Maximum distance* from the profile curve starting point
- On the Y axis, the *Minimum elevation* and *Maximum elevation*

Distance and elevation axes



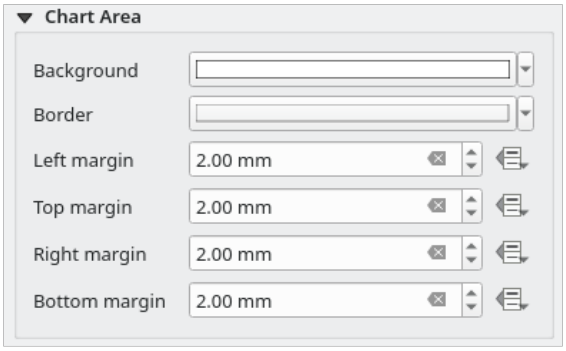
Слика18.60: Elevation Profile Distance and Elevation Axes Properties

The *Distance axis* and *Elevation axis* groups give options to tweak the grid over the elevation profile item, respectively on the X and Y axes:

- the distance display *Unit*, allowing to override the map canvas unit
- the graduation on the axis with both a *Major interval* and *Minor interval*
- the line symbols to apply to the corresponding *Major grid lines* and *Minor grid lines*

- how regular the graduation items should be labeled (*Label interval*) as well as their *Label format* and *Label font*
- the *Distance labels*: configures whether the distance unit symbol should be placed next to *Every value*, *First value*, *Last value*, *First and last values*, or skipped (*None*)

Chart area



Слика18.61: Elevation Profile Chart Area Properties

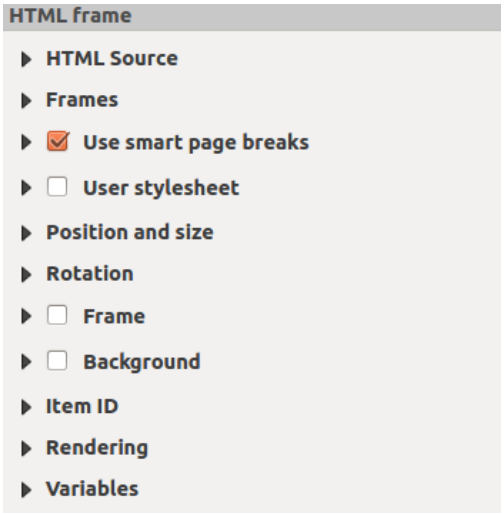
Under *Chart area*, you can configure the rendering of the area in which the elevation profile plot is actually displayed:

- a *Background* fill symbol
- a *Border* line symbol
- the margins from the elevation profile item border

18.2.10 The HTML Frame Item

It is possible to add a frame that displays the contents of a website or even create and style your own HTML page and display it! You can add a picture with the *Add HTML* following *items creation instructions* and manipulate it the same way as exposed in *Interakcija sa stavkama rasporeda*. Note that the HTML scale is controlled by the layout export resolution at the time the HTML frame is created.

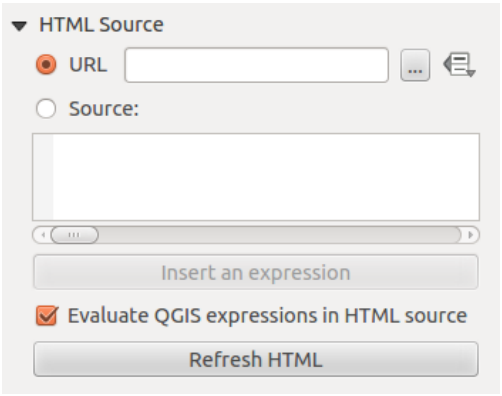
The HTML item can be customized using its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.62):



Слика18.62: HTML Frame, the Item Properties Panel

HTML Source

The *HTML Source* group of the HTML frame *Item Properties* panel provides the following functionalities (see Сл.18.63):

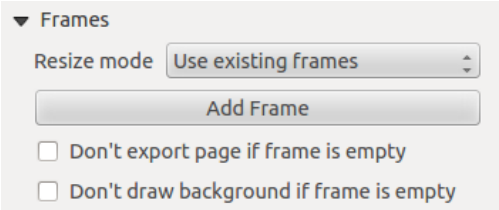


Слика18.63: HTML frame, the HTML Source properties

- In *URL* you can enter the URL of a webpage you copied from your Internet browser or select an HTML file using the … Browse button. There is also the option to use the Data-defined override button, to provide a URL from the contents of an attribute field of a table or using a regular expression.
- In *Source* you can enter text in the textbox with some HTML tags or provide a full HTML page.
- The *Insert or Edit an Expression…* button can be used to add an expression like `[%Year($now)%]` in the Source textbox to display the current year. This button is only activated when radiobutton *Source* is selected. After inserting the expression click somewhere in the textbox before refreshing the HTML frame, otherwise you will lose the expression.
- Activate *Evaluate QGIS expressions in HTML code* to see the result of the expression you have included, otherwise you will see the expression instead.
- Use the *Refresh HTML* button to refresh the HTML frame(s) and see the result of changes.

Frames

The *Frames* group of the HTML frame *Item Properties* panel provides the following functionalities (see Сл.18.64):

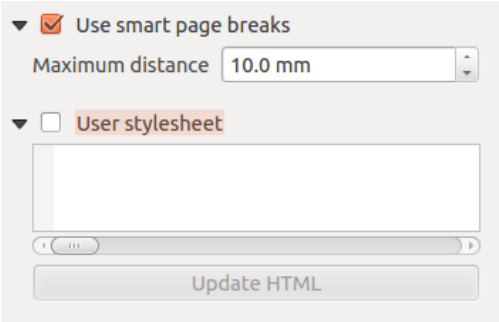


Слика18.64: HTML frame, the Frames properties

- With *Resize mode* you can select how to render the HTML contents:
  - `Use existing frames` displays the result in the first frame and added frames only.
  - `Extend to next page` will create as many frames (and corresponding pages) as necessary to render the height of the web page. Each frame can be moved around on the layout. If you resize a frame, the webpage will be divided up between the other frames. The last frame will be trimmed to fit the web page.
  - `Repeat on every page` will repeat the upper left of the web page on every page in frames of the same size.
  - `Repeat until finished` will also create as many frames as the `Extend to next page` option, except all frames will have the same size.
- Use the *Add Frame* button to add another frame with the same size as selected frame. If the HTML page does not fit in the first frame it will continue in the next frame when you use *Resize mode* or *Use existing frames*.
- Activate *Don't export page if frame is empty* prevents the page from being exported when the frame has no HTML contents. This means all other layout items, maps, scale bars, legends etc. will not be visible in the result.
- Activate *Don't draw background if frame is empty* prevents the HTML frame being drawn if the frame is empty.

Use smart page breaks and User style sheet

The *Use smart page breaks* dialog and *User style sheet* dialog of the HTML frame *Item Properties* panel provides the following functionalities (see Сл.18.65):



Слика18.65: HTML frame, Use smart page breaks and User style sheet properties

- Activate *Use smart page breaks* to prevent the html frame contents from breaking mid-way a line of text so it continues nice and smooth in the next frame.
- Set the *Maximum distance* allowed when calculating where to place page breaks in the html. This distance is the maximum amount of empty space allowed at the bottom of a frame after calculating the optimum break

location. Setting a larger value will result in better choice of page break location, but more wasted space at the bottom of frames. This is only used when *Use smart page breaks* is activated.

- Activate *User style sheet* to apply HTML styles that often is provided in cascading style sheets. An example of style code is provided below to set the color of `<h1>` header tag to green and set the font and font size of text included in paragraph tags `<p>`.

```
h1 {color: #00ff00;
}
p {font-family: "Times New Roman", Times, serif;
   font-size: 20px;
}
```

- Use the *Update HTML* button to see the result of the style sheet settings.

18.2.11 The Shape Items

QGIS provides a couple of tools to draw regular or more complex shapes over the print layout.

Белешка

Unlike other print layout items, you can not style the frame nor the background color of the shapes bounding frame (set to transparent by default).

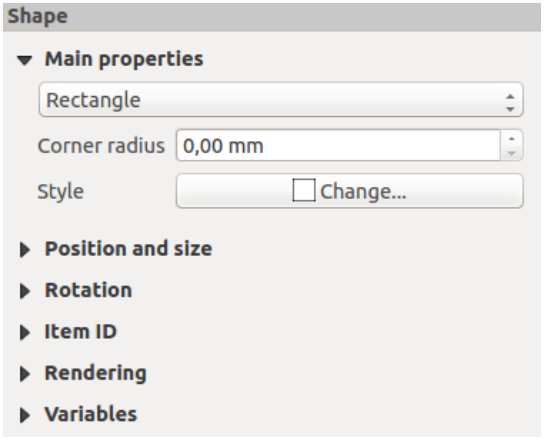
The Regular Shape Item

The *Shape* item is a tool that helps to decorate your map with regular shapes like triangle, rectangle, ellipse… You can add a regular shape using the Add Shape tool which gives access to particular tools like Add Rectangle, Add Ellipse and Add Triangle. Once you have selected the appropriate tool, you can draw the item following *items creation instructions*. Like other layout items, a regular shape can be manipulated the same way as exposed in *Interakcija sa stavkama rasporeda*.

Белешка

Holding down the `Shift` key while drawing the basic shape with the click and drag method helps you create a perfect square, circle or triangle.

The default shape item can be customized using its *Item Properties* panel. Other than the *items common properties*, this feature has the following functionalities (see Сл.18.66):



Слика18.66: Shape Item Properties Panel

The *Main properties* group shows and allows you to switch the type of the shape item (**Ellipse**, **Rectangle** or **Triangle**) inside the given frame.

You can set the style of the shape using the advanced *symbol* and *color* selector widget…

For the rectangle shape, you can set in different units the value of the *Corner radius* to round of the corners.

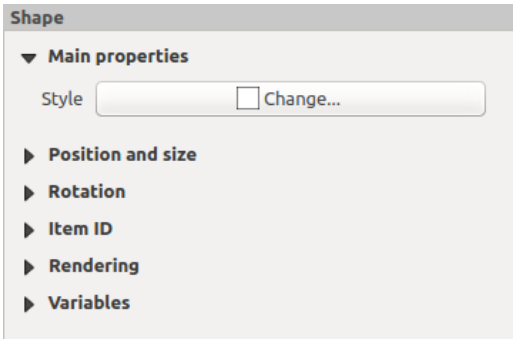
The Node-Based Shape Items

While the *Add Shape* tool provides way to create simple and predefined geometric item, the *Add Node Item* tool helps you create a custom and more advanced geometric item. For polylines or polygons, you can draw as many lines or sides as you want and vertices of the items can be independently and directly manipulated using the *Edit Nodes Item*. The item itself can be manipulated as exposed in *Interakcija sa stavkama rasporeda*.

To add a node-based shape:

1. Click the Add Node Item icon
2. Select either Add Polygon or Add Polyline tool
3. Perform consecutive left clicks to add nodes of your item. If you hold down the `Shift` key while drawing a segment, it is constrained to follow an orientation multiple of 45°.
4. When you're done, right-click to terminate the shape.

You can customize the appearance of the shape in the *Item Properties* panel.



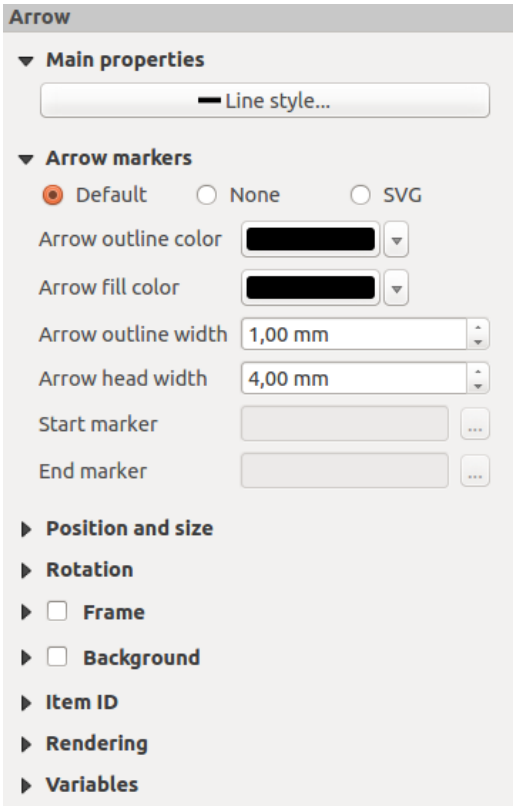
Слика18.67: Polygon Node Shape Item Properties Panel

In the *Main properties*, you can set the style of the shape using the advanced *symbol* and *color* selector widget…

For polyline node items, you can also parameterize the *Line markers* i.e. add:

- start and/or end markers with options:
  - *None*: draws a simple polyline.
  - *Arrow*: adds a regular triangular arrow head that you can customize.
  - *SVG* marker: uses an `SVG` file as arrow head of the item.
- customize the arrow head:
  - *Arrow stroke color*: sets the stroke color of the arrow head.
  - *Arrow fill color*: sets the fill color of the arrow head.
  - *Arrow stroke width*: sets the stroke width of the arrow head.
  - *Arrow head width*: sets the size of the arrow head.

SVG images are automatically rotated with the line. Stroke and fill colors of QGIS predefined SVG images can be changed using the corresponding options. Custom SVG may require some tags following this *instruction*.



Слика18.68: Polyline Node Shape Item Properties Panel

The Arrow Item

The Add Arrow tool is a shortcut to create an arrow-enabled polyline by default and thus has the same properties and behavior as a *polyline node item*.

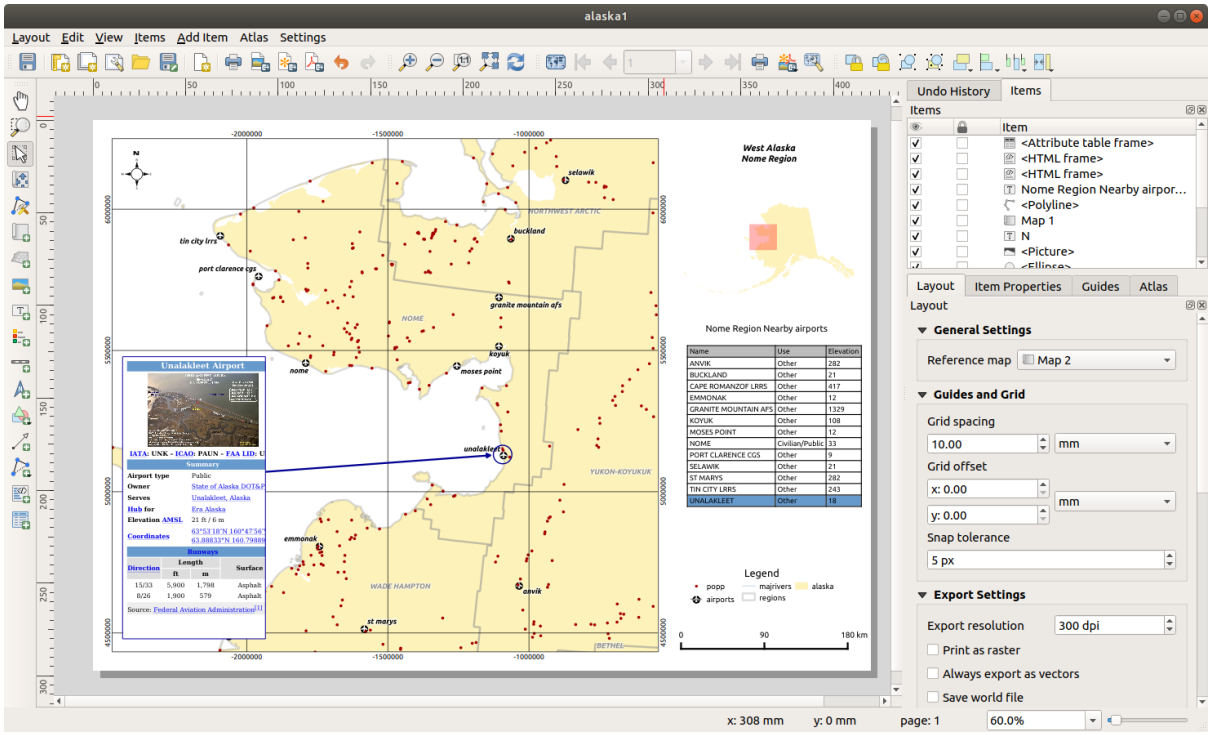
Actually, the arrow item can be used to add a simple arrow, for example, to show the relation between two different print layout items. However, to create a north arrow, the *image item* should be considered first as it gives access to a set of north arrows in `.SVG` format that you can sync with a map item so that it rotates automatically with it.

Editing a node item geometry

A specific tool is provided to edit node-based shapes through Edit Nodes Item. Within this mode, you can select a node by clicking on it (a marker is displayed on the selected node). A selected node can be moved either by dragging it or by using the arrow keys. Moreover, in this mode, you are able to add nodes to an existing shape: double-click on a segment and a node is added at the place you click. Finally, you can remove the currently selected node by hitting the `Del` key.

18.3 Creating an Output

Сл.18.69 shows an example print layout including all the types of layout items described in the previous section.



Слика18.69: Print Layout with map view, legend, image, scale bar, coordinates, text and HTML frame added

From the *Layout* menu or toolbar, you can output the print layout to different file formats, and it is possible to modify the resolution (print quality) and paper size:

- The Print icon allows you to print the layout to a connected printer or a PostScript file, depending on the installed printer drivers.
- The Export as image icon exports the print layout image formats such as `PNG`, `BMP`, `TIF`, `JPG`, and many others…
- The Export as SVG icon saves the print layout as an `SVG` (Scalable Vector Graphic).
- The Export as PDF icon saves the defined print layout directly as a `PDF` (Portable Document Format) file.

18.3.1 Podešavanja izvoza

Whenever you export a print layout, there are a selection of export settings QGIS needs to check in order to produce the most appropriate output. These configurations are:

- The *Export settings* of the *Layout* panel, such as *Export resolution*, *Print as raster* *Always export as vectors* or *Save world file*
- *Exclude page from exports* in the *page item properties* panel
- *Exclude item from exports* in the *item properties* panel

Moreover, a number of predefined checks are automatically applied to the layout. Currently these checks include testing that scalebars are correctly linked to map items, and that map overview items are also correctly linked to a map. If the checks fail, you are shown a nice warning advising you of the issue.

18.3.2 Export as Image

To export a layout as an image:

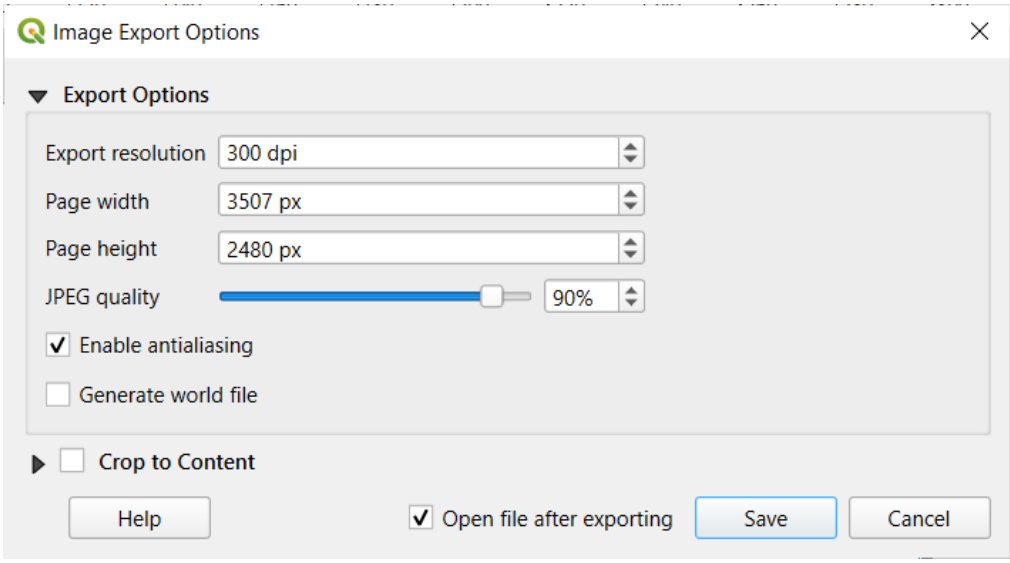
1. Click the  Export as image icon
2. Select the image format, the folder and filename (e.g. `myill.png`) to use. If the layout contains more than one page, each page will be exported to a file with the given filename with the page number appended (e.g. `myill_2.png`).
3. In the next (*Image Export Options*) dialog:
   - You can override the print layout *Export resolution* and the exported page dimensions (as set in *Layout* panel).
   - For JPEG and JPG image formats, you can set the *JPEG Quality* of the output image.
   - Image rendering can also be improved with the *Enable antialiasing* option.
   - If you want to export your layout as a **georeferenced image** (e.g., to share with other projects), check the  *Generate world file* option, and an *ESRI World File* with the same name as the exported image, but a different extension (`.tfw` for TIFF, `.pnw` for PNG, `jgw` for JPEG, …) will be created when exporting. This option can also be checked by default in the *layout panel*.

     Белешка

     For multi-page output, only the page that contains the *reference map* will get a world file (assuming that the *Generate world file* option is checked).

   - By checking  *Crop to content* option, the image output by the layout will include the minimal area enclosing all the items (map, legend, scale bar, shapes, label, image…) of each page of the composition:
     - If the composition includes a single page, then the output is resized to include EVERYTHING on the composition. The page can then be reduced or extended to all items depending on their position (on, above, below, left or right of the page).
     - In case of a multi-page layout, each page will be resized to include items in its area (left and right sides for all pages, plus top for the first page and bottom for the last page). Each resized page is exported to a separate file.

     The *Crop to content* dialog also lets you add margins around the cropped bounds.
   - By checking  *Open file after exporting* the exported file will automatically open in the default image viewer.



Слика18.70: Image Export Options

Савет

Use image formats that support transparency when items extend beyond the paper extent

Layout items may be placed outside the paper extent. When exporting with the *Crop to content* option, the resulting image may therefore extend beyond the paper extent. Since the background outside of the paper extent will be transparent, for image formats that do not support transparency (e.g. `BMP` and `JPG`) the transparent background will be rendered as full black, „corrupting" the image. Use transparency-compatible formats (e.g. `TIFF` and `PNG`) in such cases.

Белешка

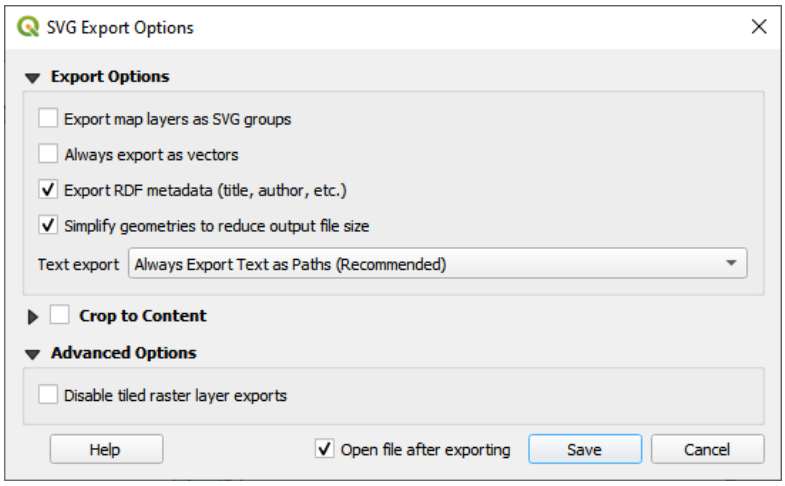
When supported by the format (e.g. `PNG`) and the underlying Qt library, the exported image may include *project metadata* (author, title, date, description…)

18.3.3 Export as SVG

To export a layout as SVG:

1. Click the Export as SVG icon
2. Fill in the path and filename (used as a base name for all the files in case of multi-page composition, as for image export)
3. In the next *SVG Export Options* dialog, you can override the layout default *export settings* or configure new ones:
   - *Export map layers as SVG groups*: exported items are grouped within layers whose name matches the layer names from QGIS, making it much easier to understand the contents of the document.
   - *Always export as vectors*: some rendering options require items to be rasterized for a better rendering. Check this option to keep the objects as vectors with the risk that the appearance of the output file may not match the print layout preview (for more details, see *Podešavanja izvoza*).
   - *Export RDF metadata* of the document such as the title, author, date, description…

- *Simplify geometries to reduce output file size*: this avoids exporting ALL geometry vertices, which can result in a ridiculously complex and large export file size that could fail to load in other applications. Geometries will be simplified while exporting the layout in order to remove any redundant vertices which are not discernably different at the export resolution (e.g. if the export resolution is `300 dpi`, vertices that are less than `1/600 inch` apart will be removed).
- Set the *Text export*: controls whether text labels are always or preferably exported as *text or outline objects*.
- Apply *Crop to content option*
- *Disable tiled raster layer exports*: When exporting files, QGIS uses a built-in raster layer tiled rendering that saves memory. Sometimes, this can cause visible „seams" in the rasters for generated files. Checking this option would fix that, at the cost of a higher memory usage during exports.
- By checking *Open file after exporting* the exported file will automatically open in the default SVG viewer.



Слика18.71: SVG Export Options

Белешка

Currently, the SVG output is very basic. This is not a QGIS problem, but a problem with the underlying Qt library. This will hopefully be sorted out in future versions.

18.3.4 Export as PDF

To export a layout as PDF:

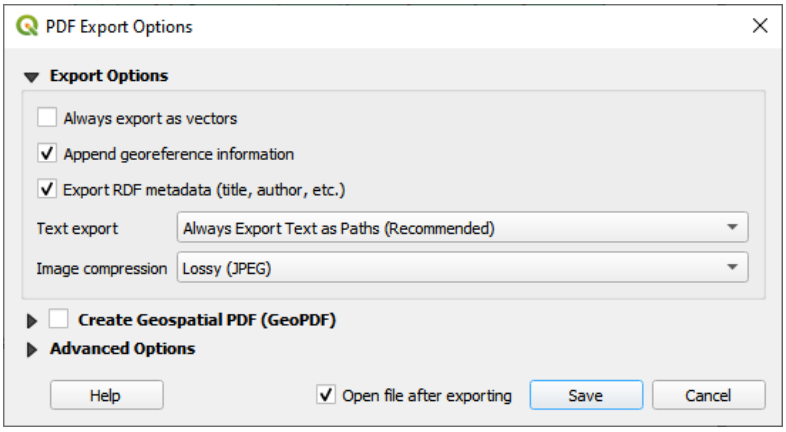
1. Click the Export as PDF icon
2. Fill in the path and filename: unlike for image and SVG export, all the pages in the layout are exported to a single PDF file.
3. In the next *PDF Export Options* dialog, you can override the layout default *export settings* or configure new ones:
   - *Always export as vectors*: some rendering options require items to be rasterized for a better rendering. Check this option to keep the objects as vectors with the risk that the appearance of the output file may not match the print layout preview (for more details, see *Podešavanja izvoza*).

Белешка

Mind opacity settings for vector layers

When you set a layer-wide opacity (or opacity for any vector symbology) to less than 100%, it forces a rasterized rendering of those objects during PDF exports. Depending on the size of the layer, this may considerably increase the PDF file size.

- *Append georeference information*: available only if the *reference map*, from which the information is taken, is on the first page.
- *Export RDF metadata* of the document such as the title, author, date, description…
- Set the *Text export*: controls whether text labels are always or preferably exported as *text or outline objects*.
- Control the PDF *Image compression* using:
  - *Lossy (JPEG)*, which is the default compression mode
  - or *Lossless*, which creates bigger files in most cases, but is much more suitable for printing outputs or for post-production in external applications (requires Qt 5.13 or later).
- *Create Geospatial PDF*: Generate a georeferenced PDF file.
- *Disable tiled raster layer exports*: When exporting files, QGIS uses tiled based rendering that saves memory. Sometimes, this can cause visible „seams" in the rasters for generated files. Checking this option would fix that, at the cost of a higher memory usage during exports.
- *Simplify geometries to reduce output file size*: Geometries will be simplified while exporting the layout by removing vertices that are not discernably different at the export resolution (e.g. if the export resolution is `300 dpi`, vertices that are less than `1/600 inch` apart will be removed). This can reduce the size and complexity of the export file (very large files can fail to load in other applications).
- By checking *Open file after exporting* the exported file will automatically open in the default PDF viewer.



Слика18.72: PDF Export Options

Белешка

Geospatial PDF export is supported, and a number of Geospatial PDF specific options are available:

- *Format* (Geospatial PDF format - there are some variations),

- *Include multiple map themes* (specify map themes to include),
- *Include vector feature information* (choose the layers and group them into logical PDF groups).

Белешка

Exporting a print layout to formats that supports georeferencing (e.g. `PDF` and `TIFF`) creates a georeferenced output by default.

Белешка

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 Generate an Atlas

Atlas functions allow you to create map books in an automated way. Atlas uses the features of a table or vector layer (*Coverage layer*) to create an output for each feature (**atlas feature**) in the table / layer. The most common usage is to zoom a map item to the current atlas feature. Further use cases include:

- a map item showing, for another layer, only features that share the same attribute as the atlas feature or are within its geometry.
- a label or HTML item whose text is replaced as features are iterated over
- a table item showing attributes of associated *parent or children* features of the current atlas feature…

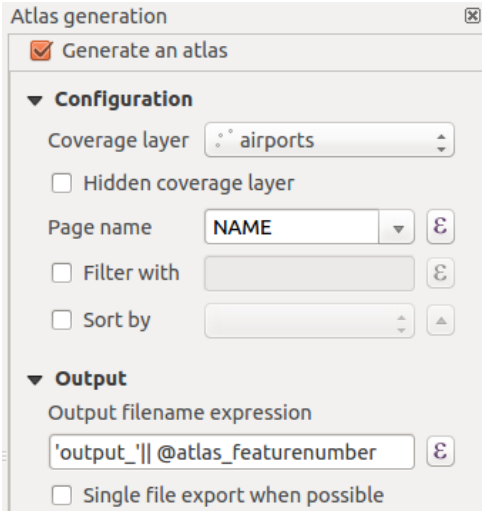
For each feature, the output is processed for all pages and items according to their exports settings.

Савет

Use variables for more flexibility

QGIS provides a large panel of functions and *variables*, including atlas related ones, that you can use to manipulate the layout items, but also the symbology of the layers, according to atlas status. Combining these features gives you a lot of flexibility and helps you easily produce advanced maps.

To enable the generation of an atlas and access atlas parameters, refer to the *Atlas* panel. This panel contains the following (see Сл.18.73):



Слика18.73: Atlas Panel

- *Generate an atlas* enables or disables atlas generation.
- *Configuration*
  - A *Coverage layer* combo box that allows you to choose the table or vector layer containing the features to iterate over.
  - An optional *Hidden coverage layer* that, if checked, will hide the coverage layer (but not the other layers) during the generation.
  - An optional *Page name* combo box to specify the name for the feature page(s). You can select a field of the coverage layer or set an *expression*. If this option is empty, QGIS will use an internal ID, according to the filter and/or the sort order applied to the layer.
  - An optional *Filter with* text area that allows you to specify an expression for filtering features from the coverage layer. If the expression is not empty, only features that evaluate to `True` will be processed.
  - An optional *Sort by* that allows you to sort features of the coverage layer (and the output), using a field of the coverage layer or an expression. The sort order (either ascending or descending) is set by the two-state *Sort direction* button that displays an up or a down arrow.
- *Output* - this is where the output of the atlas can be configured:
  - An *Output filename expression* textbox that is used to generate a filename for each atlas feature. It is based on expressions. is meaningful only for rendering to multiple files.
  - A *Single file export when possible* that allows you to force the generation of a single file if this is possible with the chosen output format (`PDF`, for instance). If this field is checked, the value of the *Output filename expression* field is meaningless.

– An *Image export format* drop-down list to select the output format when using the  Export atlas as Images… button.

Control map by atlas

The most common usage of atlas is with the map item, zooming to the current atlas feature, as iteration goes over the coverage layer. This behavior is set in the *Controlled by atlas* group properties of the map item. See *Controlled by atlas* for different settings you can apply on the map item.

Customize labels with expression

In order to adapt labels to the feature the atlas iterates over, you can include expressions. Make sure that you place the expression part (including functions, fields or variables) between `[%` and `%]` (see *The Label Item* for more details).

For example, for a city layer with fields `CITY_NAME` and `ZIPCODE`, you could insert this:

```
The area of [% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ',
format_number($area/1000000, 2) ) %] km2
```

or, another combination:

```
The area of [% upper(CITY_NAME)%],[%ZIPCODE%] is
[%format_number($area/1000000,2) %] km2
```

The information `[% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ', format_number($area/1000000, 2) ) %]` is an expression used inside the label. Both expressions would result in the following type of label in the generated atlas:

```
The area of PARIS,75001 is 1.94 km2
```

Explore Data-defined override buttons with atlas

There are several places where you can use a  Data defined override button to override the selected setting. This is particularly useful with atlas generation. See *Data defined override setup* for more details on this widget.

For the following examples the `Regions` layer of the QGIS sample dataset is used and selected as *Coverage layer* for the atlas generation. We assume that it is a single page layout containing a map item and a label item.

When the height (north-south) of a region extent is greater than its width (east-west), you should use *Portrait* instead of *Landscape* orientation to optimize the use of paper. With a  Data Defined Override button you can dynamically set the paper orientation.

Right-click on the page and select *Page Properties* to open the panel. We want to set the orientation dynamically, using an expression depending on the region geometry, so press the  button of field *Orientation*, select *Edit…* to open the *Expression string builder* dialog and enter the following expression:

```
CASE WHEN bounds_width(@atlas_geometry) > bounds_height(@atlas_geometry)
THEN 'Landscape' ELSE 'Portrait' END
```

Now if you *preview the atlas*, the paper orients itself automatically, but item placements may not be ideal. For each Region you need to reposition the location of the layout items as well. For the map item you can use the  button of its *Width* property to set it dynamic using the following expression:

```
@layout_pagewidth - 20
```

Likewise, use the  button of the *Height* property to provide the following expression to constrain map item size:

```
@layout_pageheight - 20
```

To ensure the map item is centered in the page, set its *Reference point* to the upper left radio button and enter `10` for its *X* and *Y* positions.

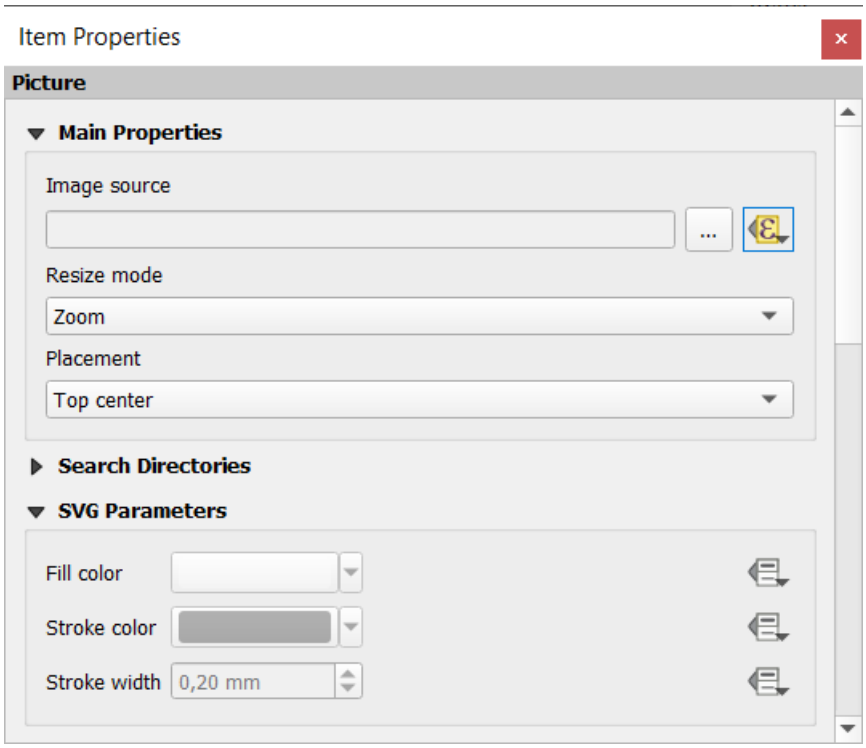
Let's add a title above the map in the center of the page. Select the label item and set the horizontal alignment to *Center*. Next move the label to the right position, choose the middle button for the *Reference point*, and provide the following expression for field *X*:

```
@layout_pagewidth / 2
```

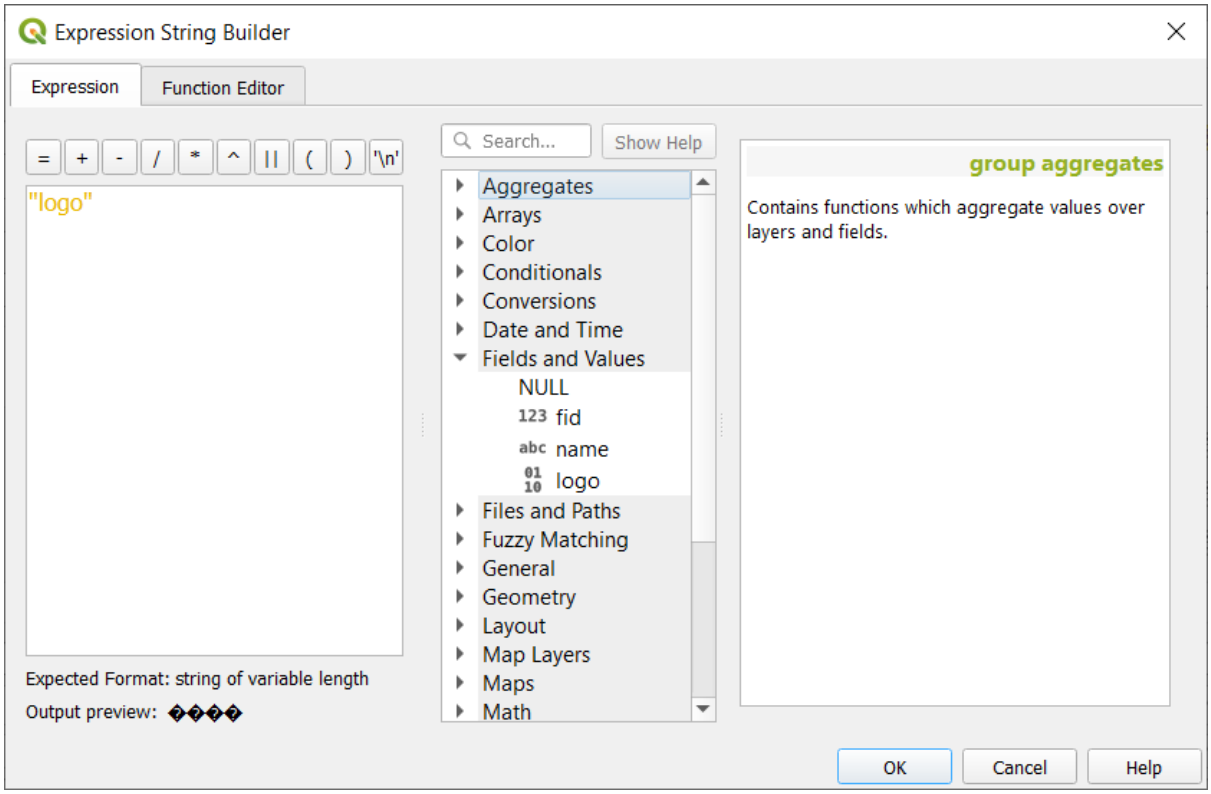
For all other layout items you can set the position in a similar way so they are correctly positioned both for portrait and landscape. You can also do more tweaks such as customizing the title with feature attributes (see *Customize labels with expression* example), changing images, resizing the number of legend columns number according to page orientation, …

The information provided here is an update of the excellent blog (in English and Portuguese) on the Data Defined Override options Multiple_format_map_series_using_QGIS_2.6 .

Another example for using data-defined override buttons is the usage of a dynamic picture. For the following examples we use a geopackage layer containing a BLOB field called `logo` with the field type binary (see *Creating a new GeoPackage layer*). For every feature there is defined a different picture so that the atlas can iterate over as described in *Preview and generate an atlas*. All you need to do is add a picture in the print layout and go to its *Item properties* in the atlas context. There you can find a data-defined override button in the *Image source* section of the *Main Properties*.



In the following window choose *Edit* so that the *Expression String Builder* opens. From the *Fields and values* section you can find the BLOB field that was defined in the geopackage layer. Double-click the field name `logo` and click *OK*.



The atlas iterates over the entries in the BLOB field provided that you choose the geopackage layer as *Coverage layer* (further instructions you can find in *Preview and generate an atlas*).

These are just two examples of how you can use some advanced settings with atlas.

Preview and generate an atlas



Слика18.74: Atlas Preview toolbar

Once the atlas settings have been configured, and layout items (map, table, image…) linked to it, you can create a preview of all the pages by choosing *Atlas* ► *Preview Atlas* or clicking the Preview Atlas icon. You can then use the arrows to navigate through all the features:

- First feature
- Previous feature
- Next feature
- Last feature

You can also use the combo box to select and preview a specific feature. The combo box shows atlas feature names according to the expression set in the atlas *Page name* option.

As for simple compositions, an atlas can be generated in different ways (see *Creating an Output* for more information - just use tools from the *Atlas* menu or toolbar instead of the *Layout* menu.

This means that you can directly print your compositions with *Atlas* ► *Print Atlas*. You can also create a PDF using *Atlas* ► *Export Atlas as PDF…*: You will be asked for a directory to save all the generated PDF files, except if the *Single file export when possible* has been selected. In that case, you'll be prompted to give a filename.

With *Atlas* ► *Export Atlas as Images…* or *Atlas* ► *Export Atlas as SVG…* tool, you're also prompted to select a folder. Each page of each atlas feature composition is exported to the image file format set in *Atlas* panel or to SVG.

Белешка

With multi-page output, an atlas behaves like a layout in that only the page that contains the *Општа подешавања* will get a world file (for each feature output).

Савет

Print a specific atlas feature

If you want to print or export the composition of only one feature of the atlas, simply start the preview, select the desired feature in the drop-down list and click on *Layout* ► *Print* (or *Export…* to any supported file format).

Use project defined relations for atlas creation

For users with HTML and Javascript knowledge it is possible to operate on GeoJSON objects and use project defined relations from the QGIS project. The difference between this approach and using expressions directly inserted into the HTML is that it gives you a full, unstructured GeoJSON feature to work with. This means that you can use existing Javascript libraries and functions that operate on GeoJSON feature representations.

The following code includes all related child features from the defined relation. Using the JavaScript `setFeature` function it allows you to make flexible HTML which represents relations in whatever format you like (lists, tables, etc). In the code sample, we create a dynamic bullet list of the related child features.

```
// 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>
```

During atlas creation there will be an iteration over the coverage layer containing the parent features. On each page, you will see a bullet list of the related child features following the parent's identifier.

18.4 Creating a Report

This section will help you set up a report in QGIS.

18.4.1 What is it?

By definition, a GIS report is a document containing information organized in a narrative way, containing maps, text, graphics, tables, etc. A report can be prepared ad hoc, periodic, recurring, regular, or as required. Reports may refer to specific periods, events, occurrences, subjects or locations.

In QGIS, a *Report* is an extension of a *Layouts*.

Reports allow users to output their GIS projects in a simple, quick and structured way.

A report can be created with *Project* ► *New Report* or inside the *Project* ► *Layout Manager*.

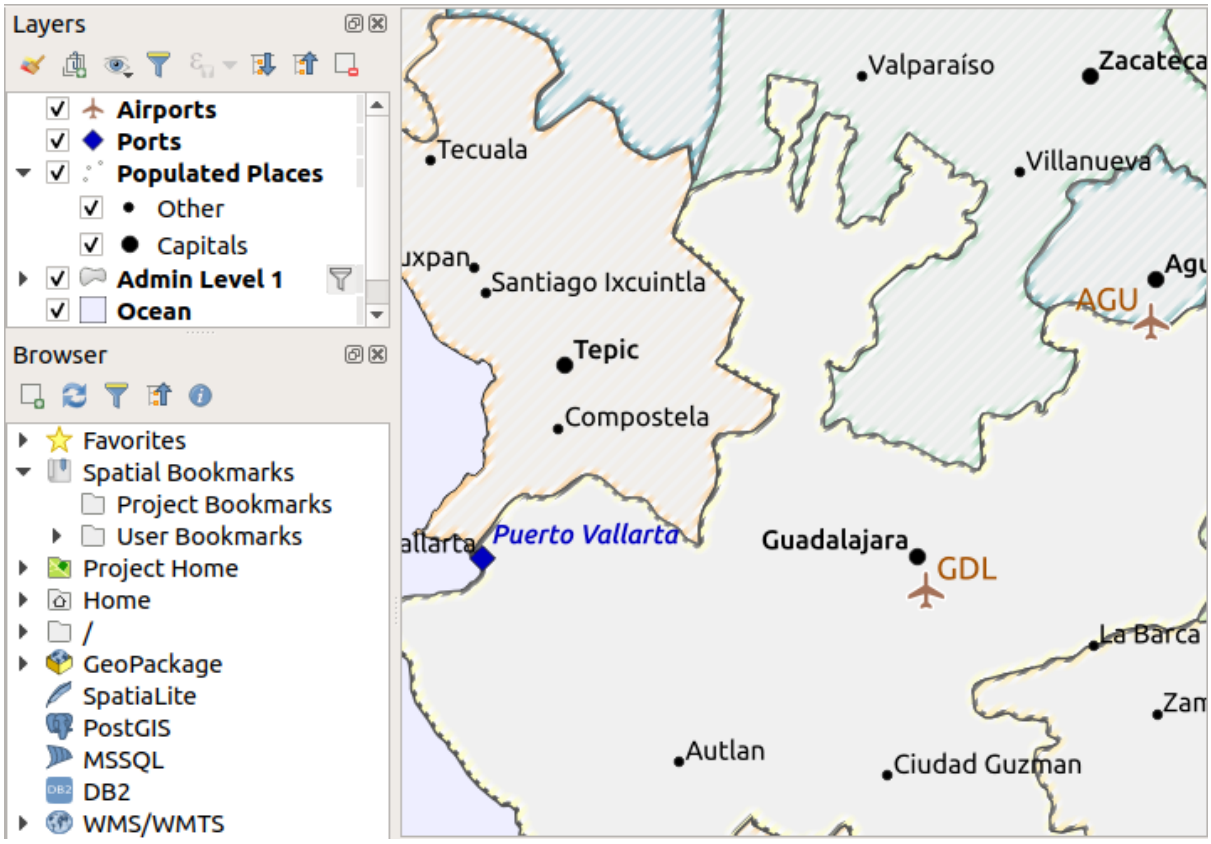
Белешка

The maps in QGIS reports behave in the same way as maps in print layouts and atlases. We will concentrate on the specifics of QGIS reports. For details on map handling, see the sections on *print layouts* and *atlases*.

18.4.2 Get started

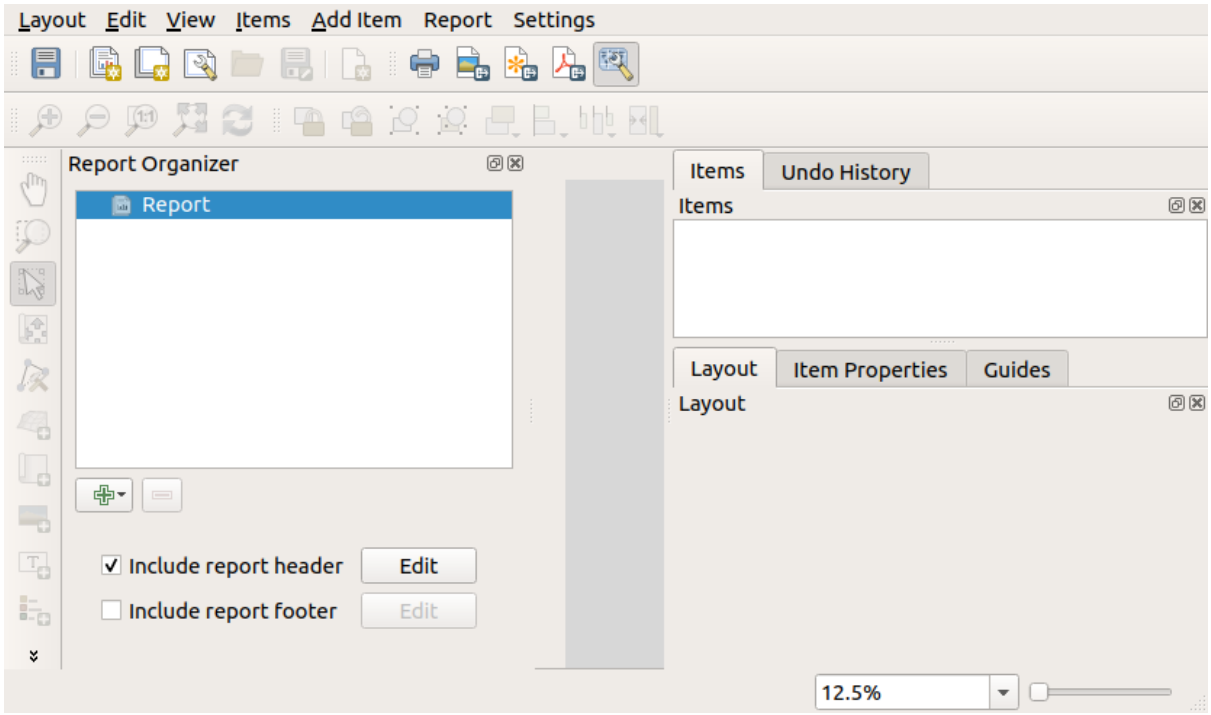
In the *Layout Manager* dialog a report can be created through *New from template* by selecting the dropdown option *Empty Report* and hitting the *Create…* button.

For this example, we use some administrative boundaries, populated places, ports and airports from the Natural Earth dataset (1:10M).



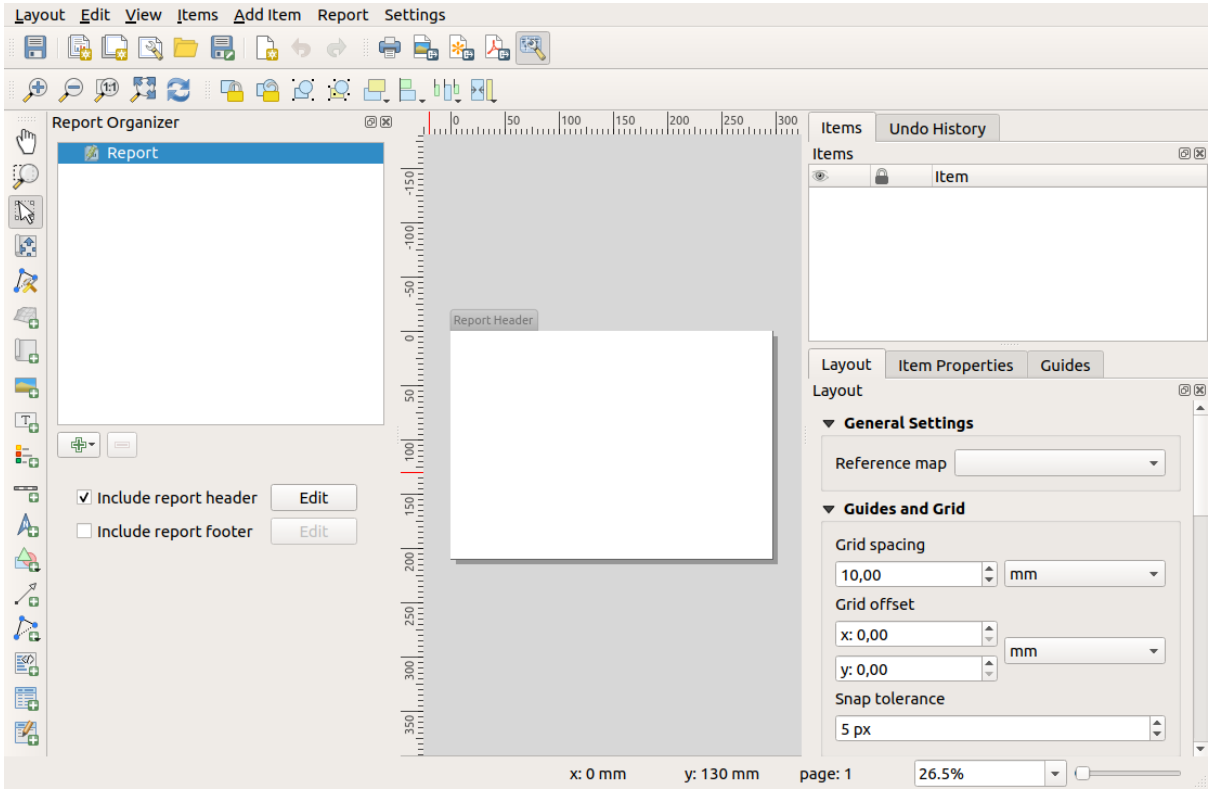
Using the *Project* ► *New Report* command, we create a blank report. Initially, there is not much to look at – the dialog which is displayed looks much like the print layout designer, except for the *Report Organizer* panel to the

left:



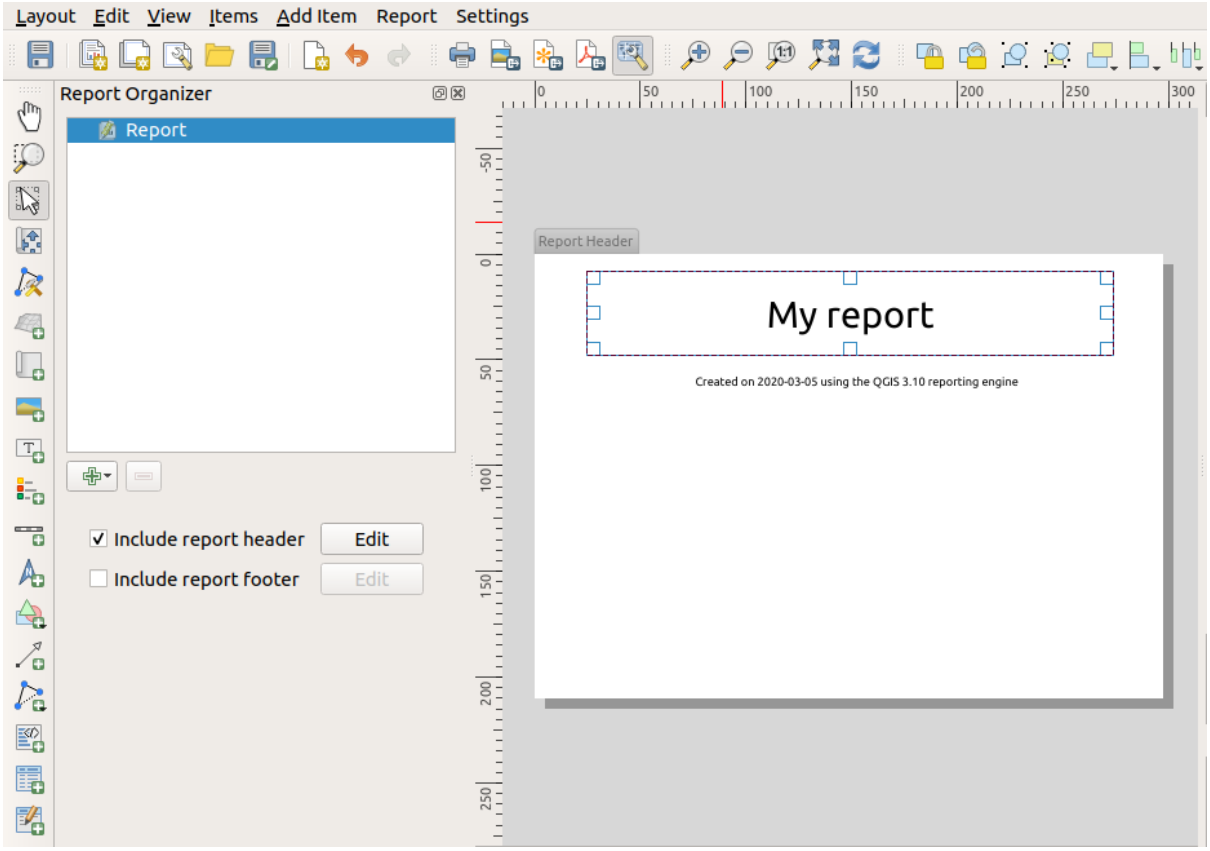
18.4.3 Layout Report Workspace

QGIS reports can consist of multiple, nested sections. In our new blank report we initially only have the main report section. The only options for this report section is *Include report header* and *Include report footer*. If we enable these options, a header will be included as the first page(s) (individual parts of reports can be multi-page if desired) in the report, and a footer will constitute the last page(s). Enable the header (*Include report header*), and hit the *Edit* button next to it:

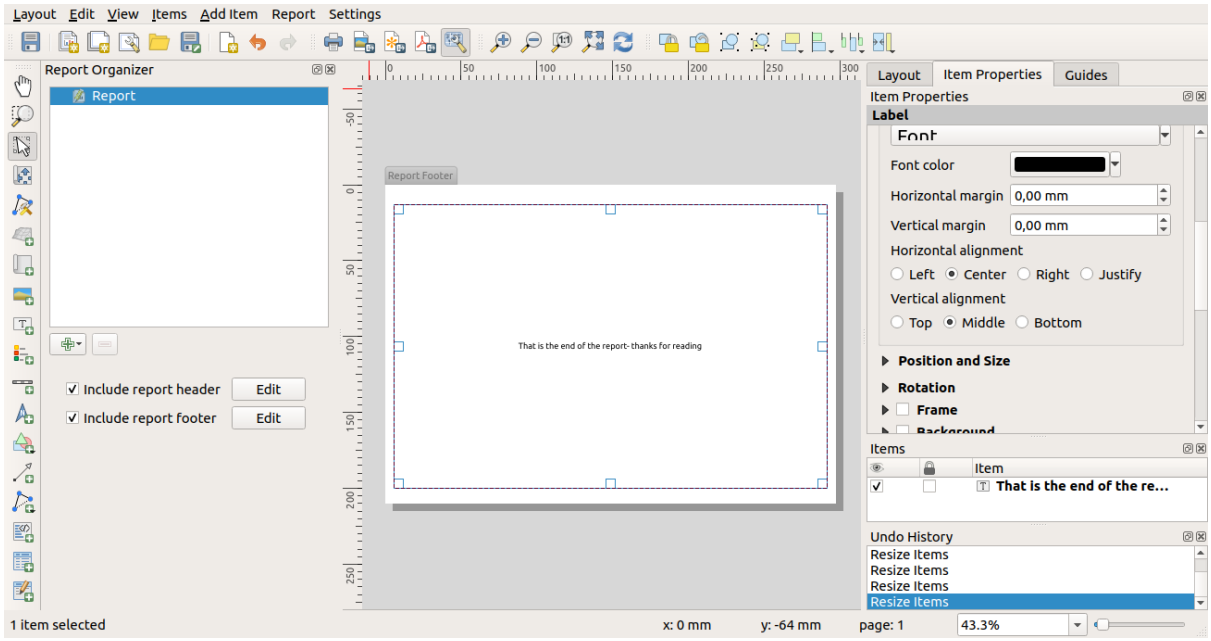


A few things happen as a result. Firstly, an edit pencil is shown next to *Report* in the *Report Organizer*, indicating that the report section is currently being edited in the designer. We also see a new page with a small *Report Header* title. The page has *landscape* orientation by default, but this (and other properties of the page) can be changed by right-clicking on the page and choosing *Page properties*. This will bring up the *Item properties* tab for the page, and page *Size*, *Width*, *Height*, and more can be specified.

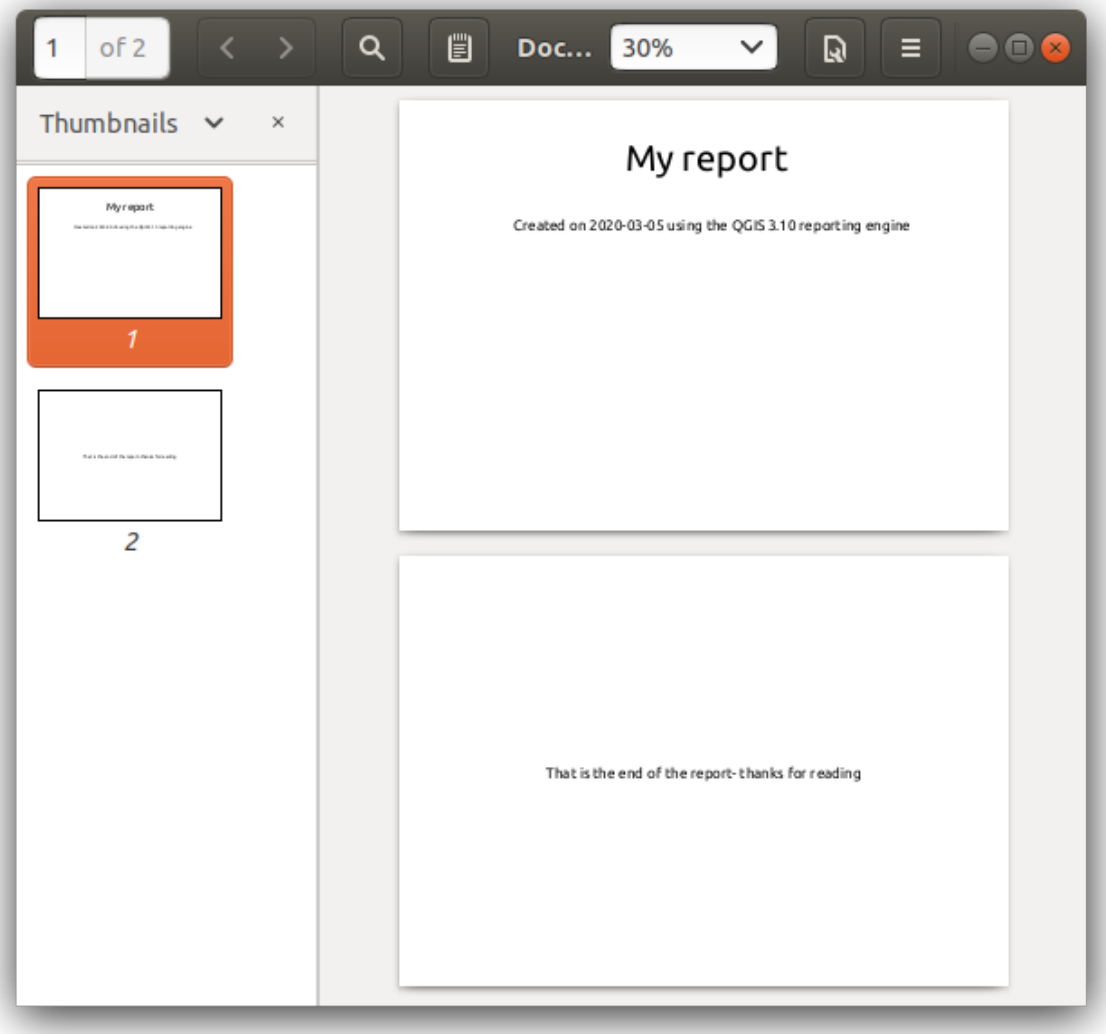
In QGIS reports, every component of the report is made up of individual layouts. They can be created and modified using the same tools as for standard print layouts – so you can use any desired combination of labels, pictures, maps, tables, etc. Let us add some items to our report header to demonstrate:



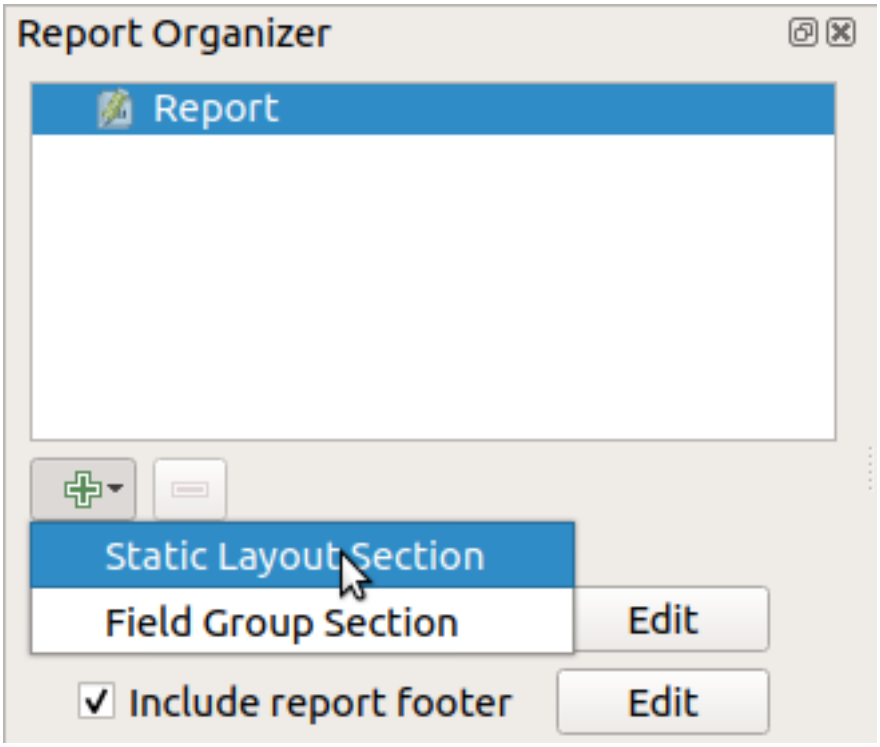
We will also create a simple footer for the report by checking the *Include report footer* option and hitting *Edit*.



Before proceeding further, let us export this report and see what we get. Exporting is done from the *Report* menu – in this case we select *Export Report as PDF…* to render the whole report to a PDF file. Here is the not-very-impressive result – a two page PDF consisting of our header and footer:



Let us make things more interesting. By hitting the Add Section button in the *Report Organizer*, we are given a choice of new sections to add to our report.

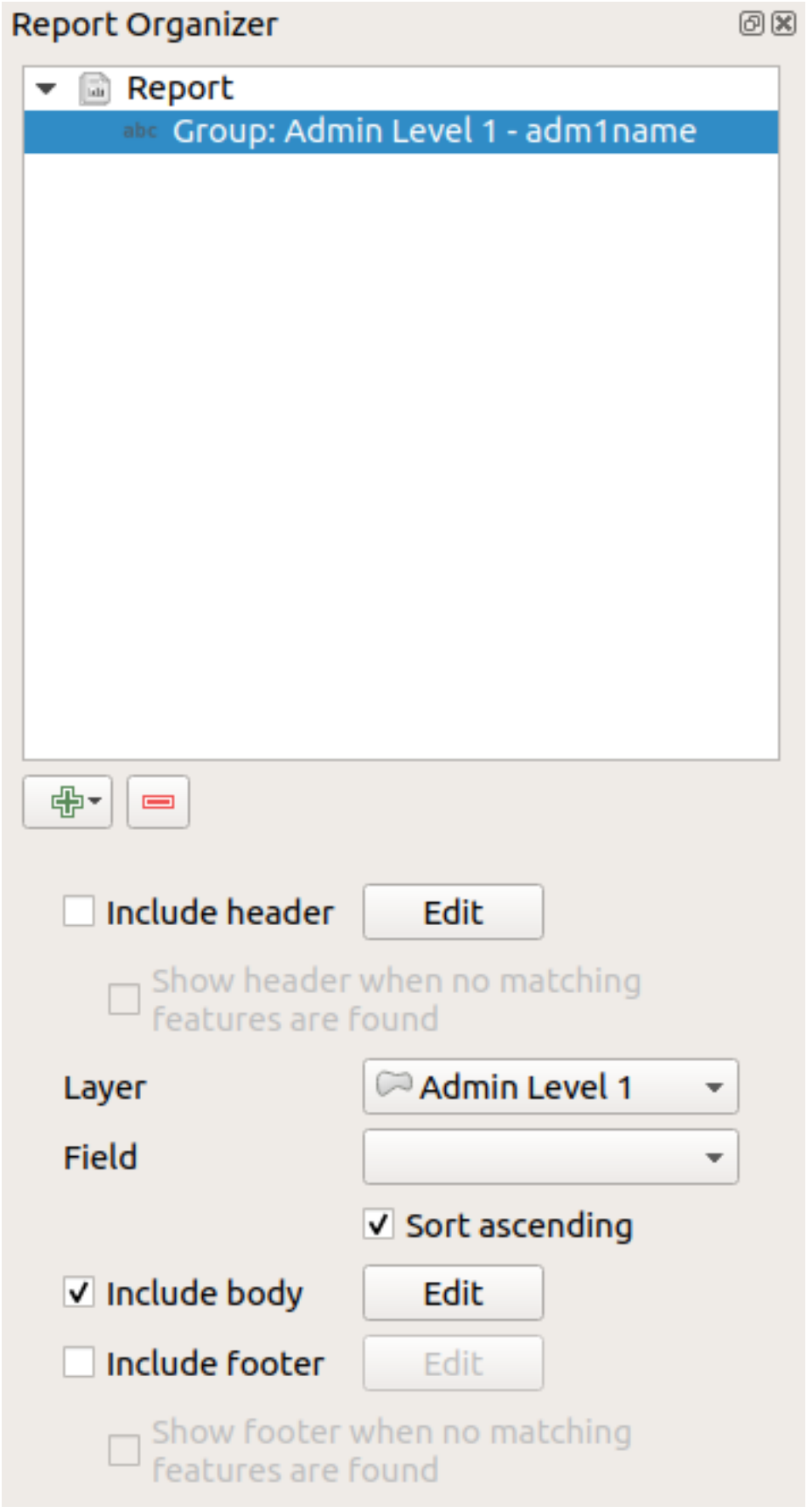


There are two options: *Static Layout Section* and *Field Group Section*.

The *Add Static Layout Section* is a single, static body layout. This can be used to embed static layouts mid-way through a report.

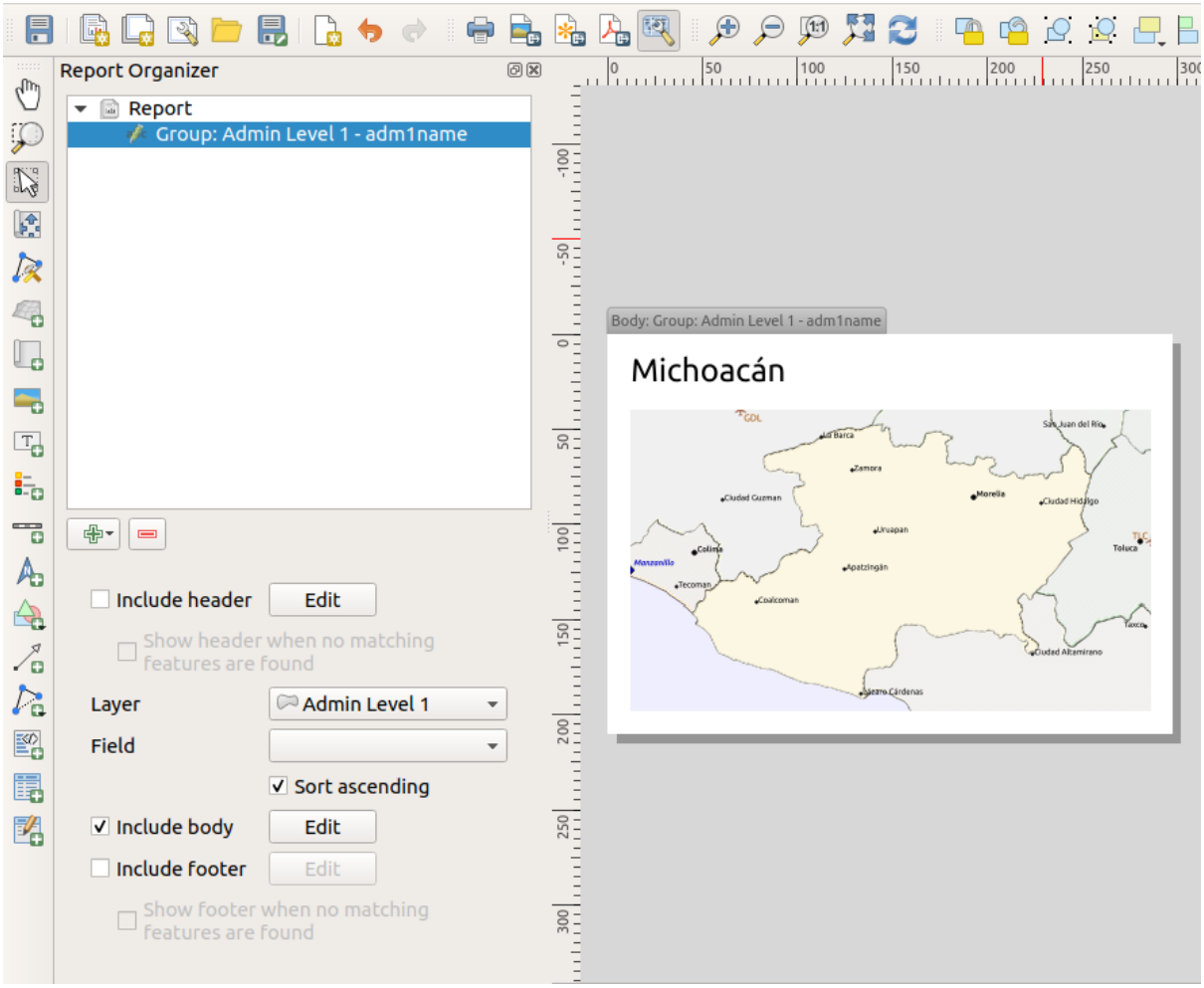
The *Field Group Section* repeats its body layout for every feature of a layer. The features are sorted by the selected grouping feature (with an option for ascending/descending sort). If a field group section has child sections (e.g. another field group section with a different field), then only features with unique values for the group feature are iterated over. This allows nested reports.

For now we will add a *Field Group Section* to our report. At its most basic level, you can think of a *Field Group Section* as the equivalent of a *print atlas*: you select a layer to iterate over, and the report will insert a section for each feature found. Selecting the new *Field Group Section* reveals a number of new related settings:



In this case we've setup our Field Group so that we iterate over all the states from the *Admin Level 1* layer, using the

values from the *adm1name* field. The same options to include header and footer are present, together with a new option to include a *body* for this section. We'll do that, and edit the body:

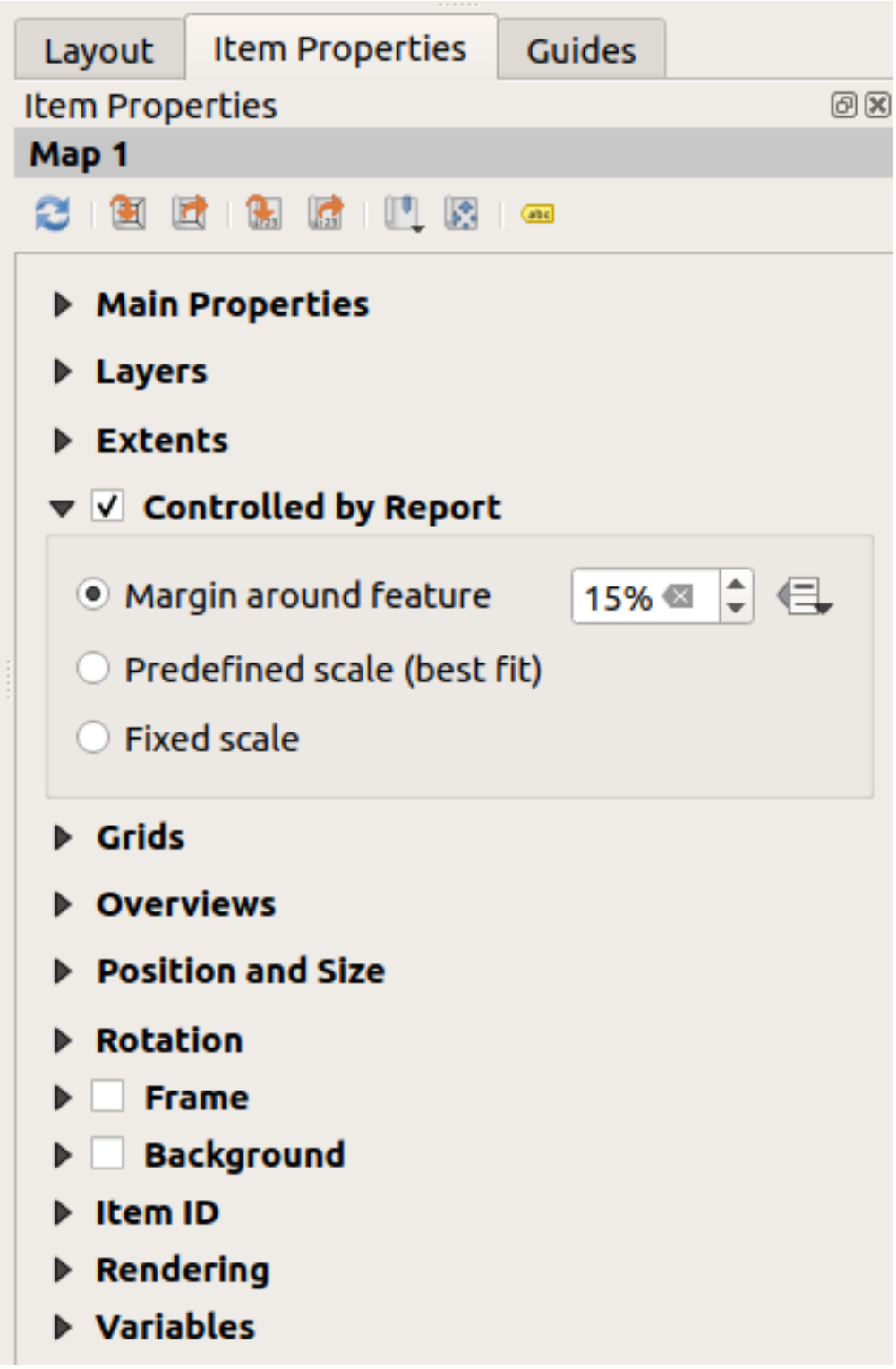


Our body now consists of a map and a label showing the name of the state. To include the name of the state, we selected *Add Item* ► *Add Label* and data defined the text under *Main Properties* with the help of *Insert or Edit an Expression*….

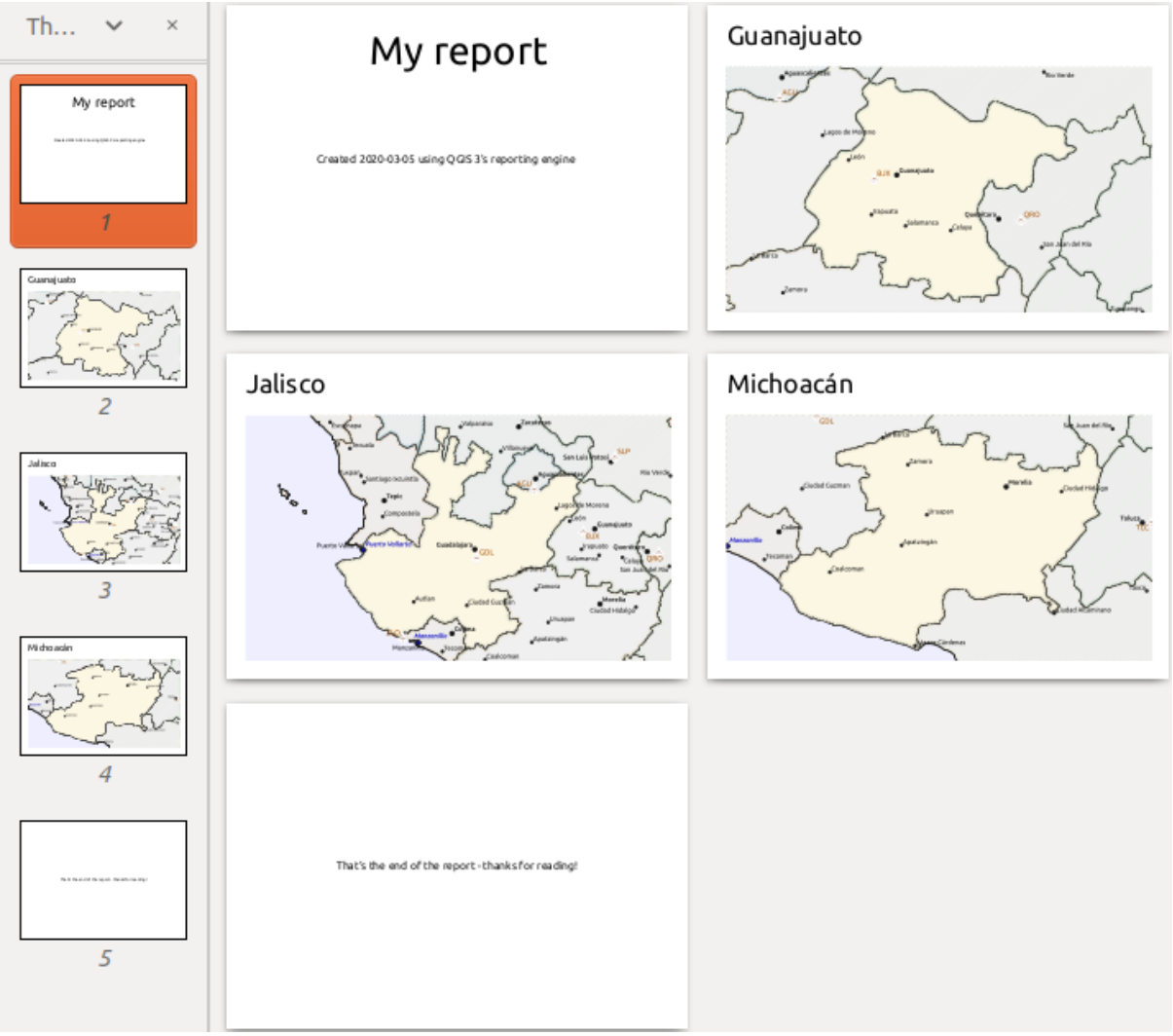
The result was the following expression (*name* is the name of the attribute in the *Admin Level 1* layer that contains the name of the state):

```
[% "name" %]
```

The map is set to follow the current report feature (enabled by checking *Controlled by Report* – just like a map item in an atlas will follow the current atlas feature when *Controlled by Atlas* is checked):



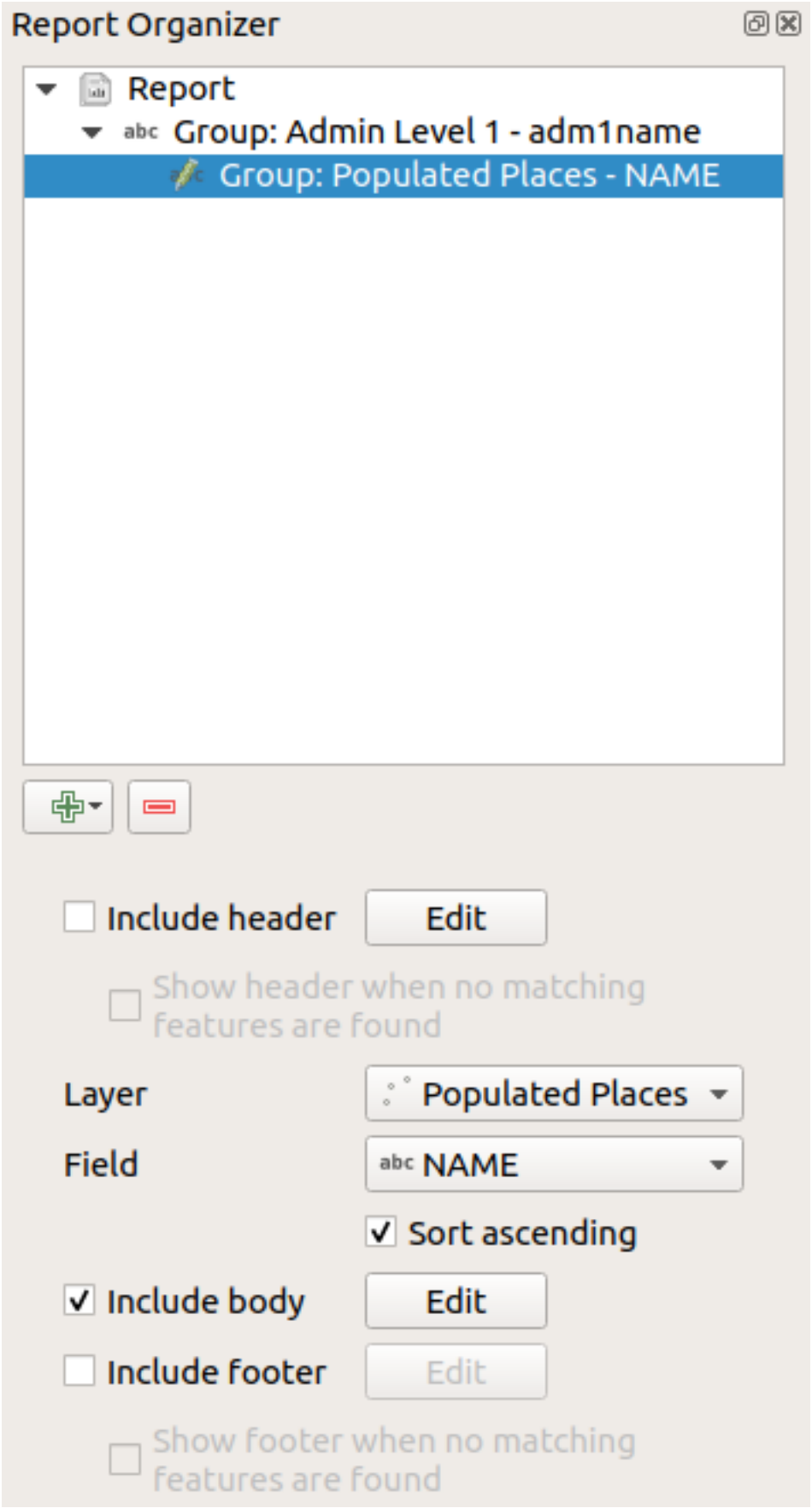
If we went ahead and exported our report now, we'd get something like this:



Слика18.75: The report header, a page for each state, and the report footer.

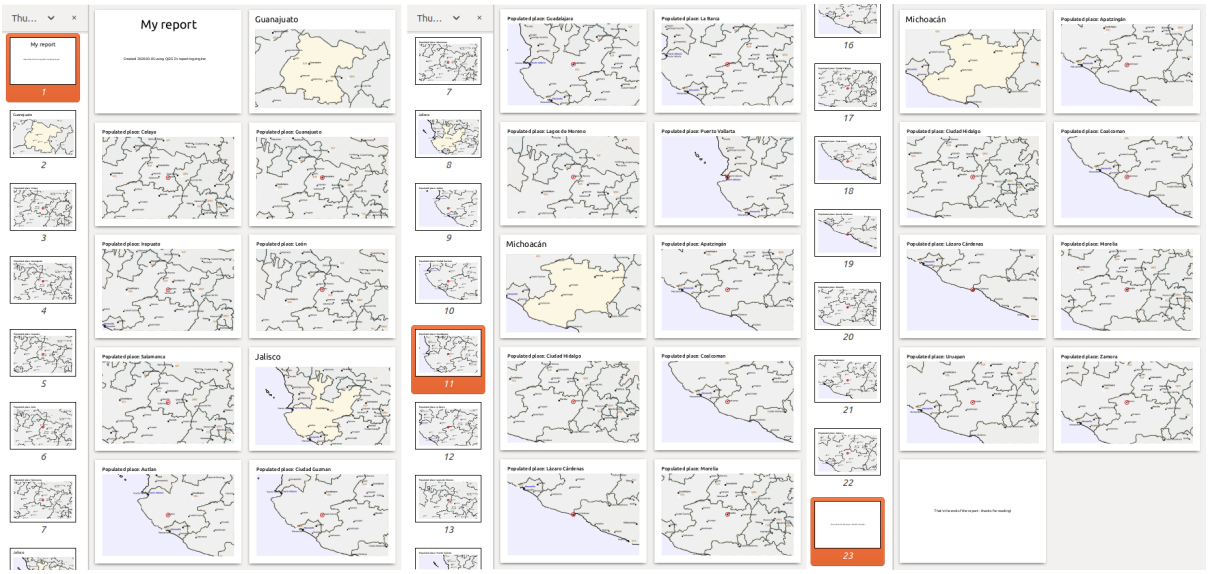
So more or less an atlas, but with a header and footer page.

Let us make things more interesting by adding a subsection to our state group. We do this by first selecting the *Admin Level 1* field group in the organizer, then hitting the Add Field button and adding a new *Field Group Section*:



When iterating over the features of a *Field Group Section*, the features will be filtered to match the defining field

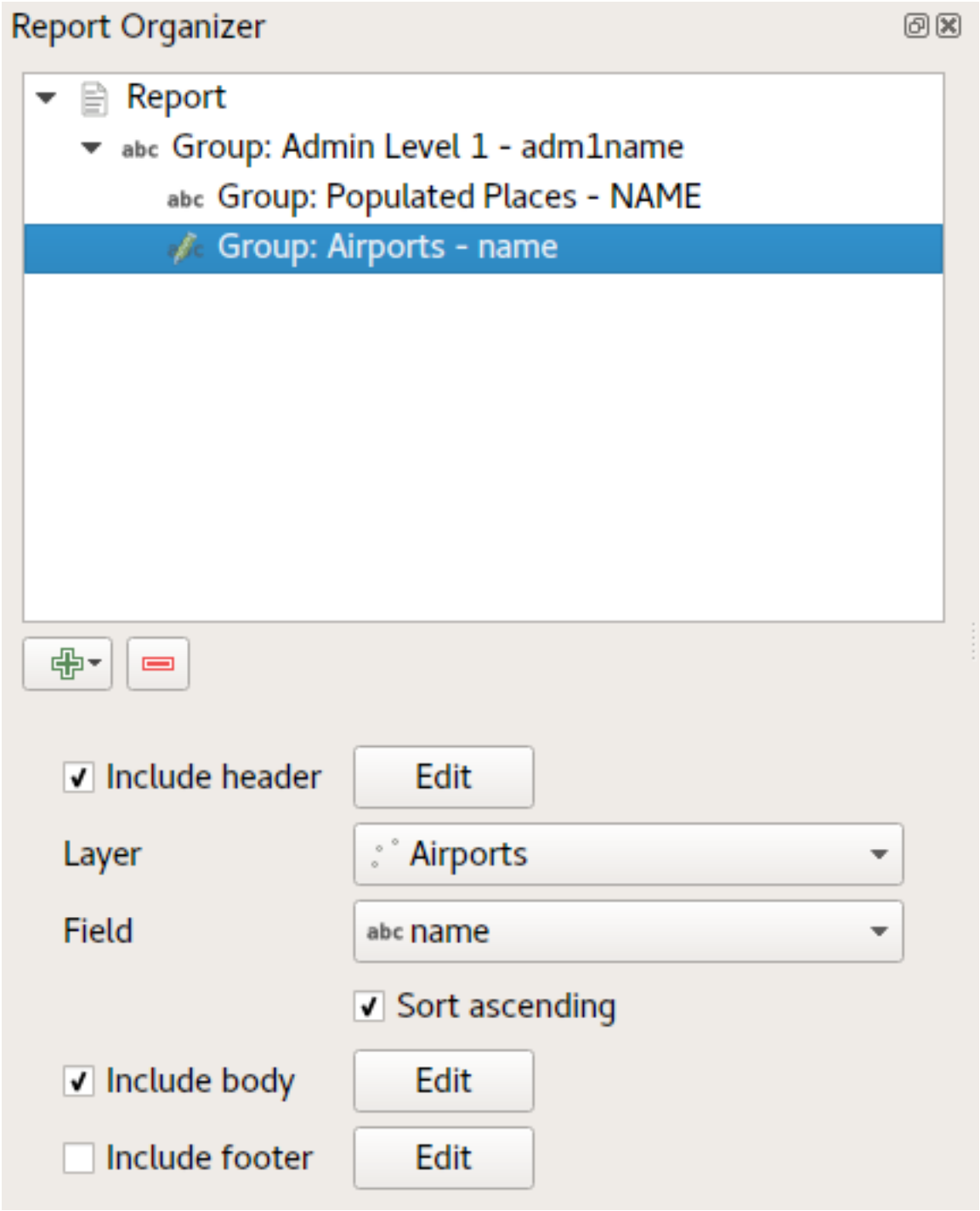
of its parent group (`adm1name` in this case). Here, the subsection we added will iterate over a *Populated Places* layer, including a body section for each place encountered. The magic here is that the *Populated Places* layer has an attribute with the same name as the defining field in the parent layer, *adm1name*, tagging each place with the state it is contained within (if you're lucky your data will already be structured like this – if not, run the *Join Attributes by Location* Processing algorithm and create your own field). When we export this report, QGIS will grab the first state from the *Admin Level 1* layer, and then iterate over all the *Populated Places* with a matching *adm1name* value. Here's what we get:



Here we created a basic body for the Populated Places group, including a map of the place and a table of some place attributes. So our report is now a report header, a page for the first state, followed by a page for every populated place within that state, then the rest of the states with their populated places, and finally the report footer. If we were to add a header for the Populated Places group, it would be included just before listing the populated places for each state, as shown in the illustration below.

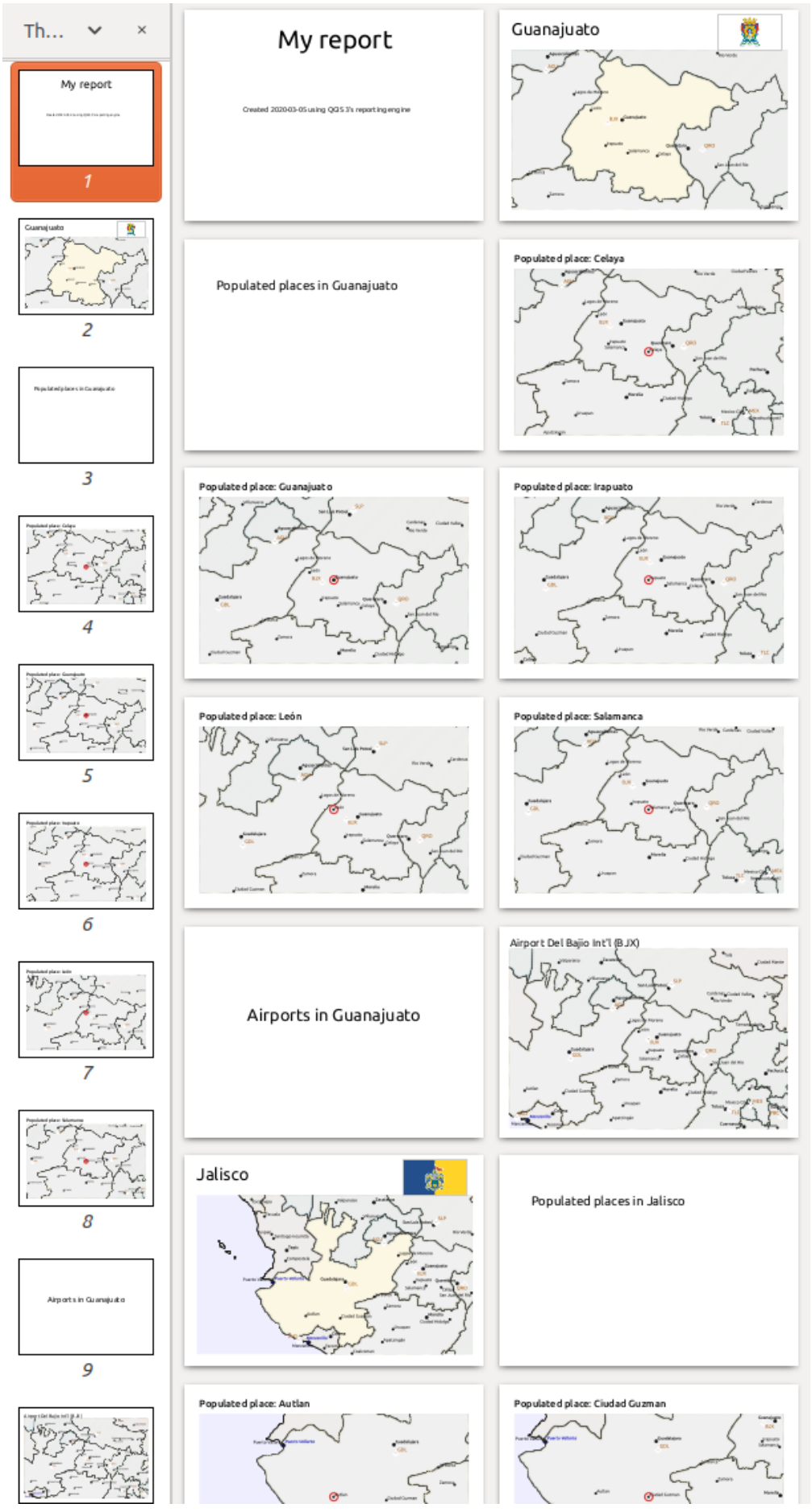
Similarly, a footer for the Populated Places group would be inserted after the final place for each state is included.

In addition to nested subsections, subsections in a report can also be included consecutively. If we add a second subsection to the *Admin Level 1 group* for *Airports*, then (if the *Airports* layer has an attribute *adm1name* that can link it to the parent group) our report will first list ALL the populated places for each state, followed by all the airports within that state, before proceeding to the next state.



The key point here is that our *Airports group* is a subsection of the *Admin Level 1 group* – not the *Populated Places group*.

In this case our report would be structured like this (note that state flags have also been included - the procedure for adding feature specific pictures in this way is described below):


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

Including pictures in a report

Pictures can be quite useful in reports, and QGIS allows pictures in both the static and dynamic parts of a report. Pictures are added in the same way as for standard print layouts, and for the static report parts (and static pictures in dynamic parts) there is not more to it.

But if you want illustrations that are tailored to the report features, your layer must have an attribute that can be used to define the picture to include.

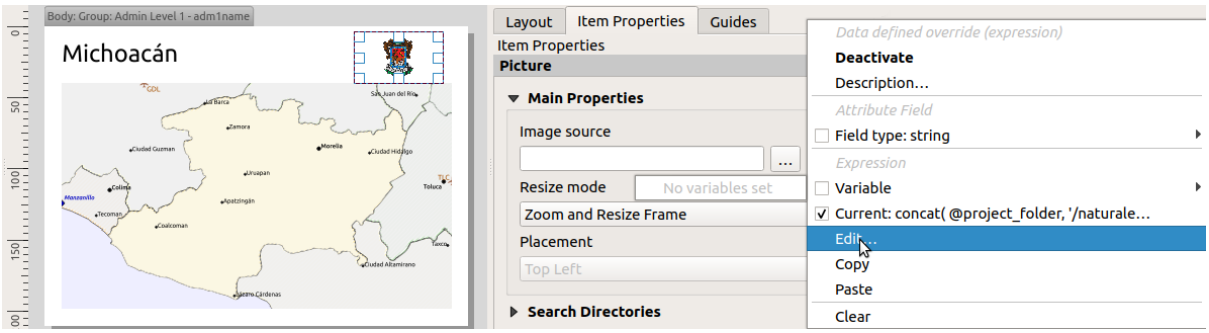
QGIS depends on absolute file names for images in reports.

For dynamic pictures, you first add a picture to the body part of the group, as usual. In the *Item properties* of the picture, you set the *Image Source* using the Data defined override button, and either select an attribute that contains the absolute path of the images or *Edit...* (to enter an expression that generates the absolute image path).

Below is an example expression that uses string concatenation to specify the absolute path to the pictures, using the directory where the project file is located (`@project_path`) and an attribute (`adm1name`) from which the file name is generated (in this case by transforming the string in the `adm1name` attribute to uppercase, and appending '_flag.png'):

```
concat(@project_folder, '/naturalearth/pictures/' ,
       upper("adm1name"), '_flag.png')
```

This means that the pictures are located in the `naturalearth/pictures` subdirectory of the project file directory.



Highlighting the current report feature in a map

In the above report, the report features are emphasized in the maps using highlighting (state) and circles (populated places). To emphasize the report features in the maps (apart from placing them at the centre of the maps), you must data define the style using a comparison between its `@id` and the `@atlas_featureid`, as for atlases.

For instance, if you would like to use a thicker line / border for the report feature than the other features you can data define the line width:

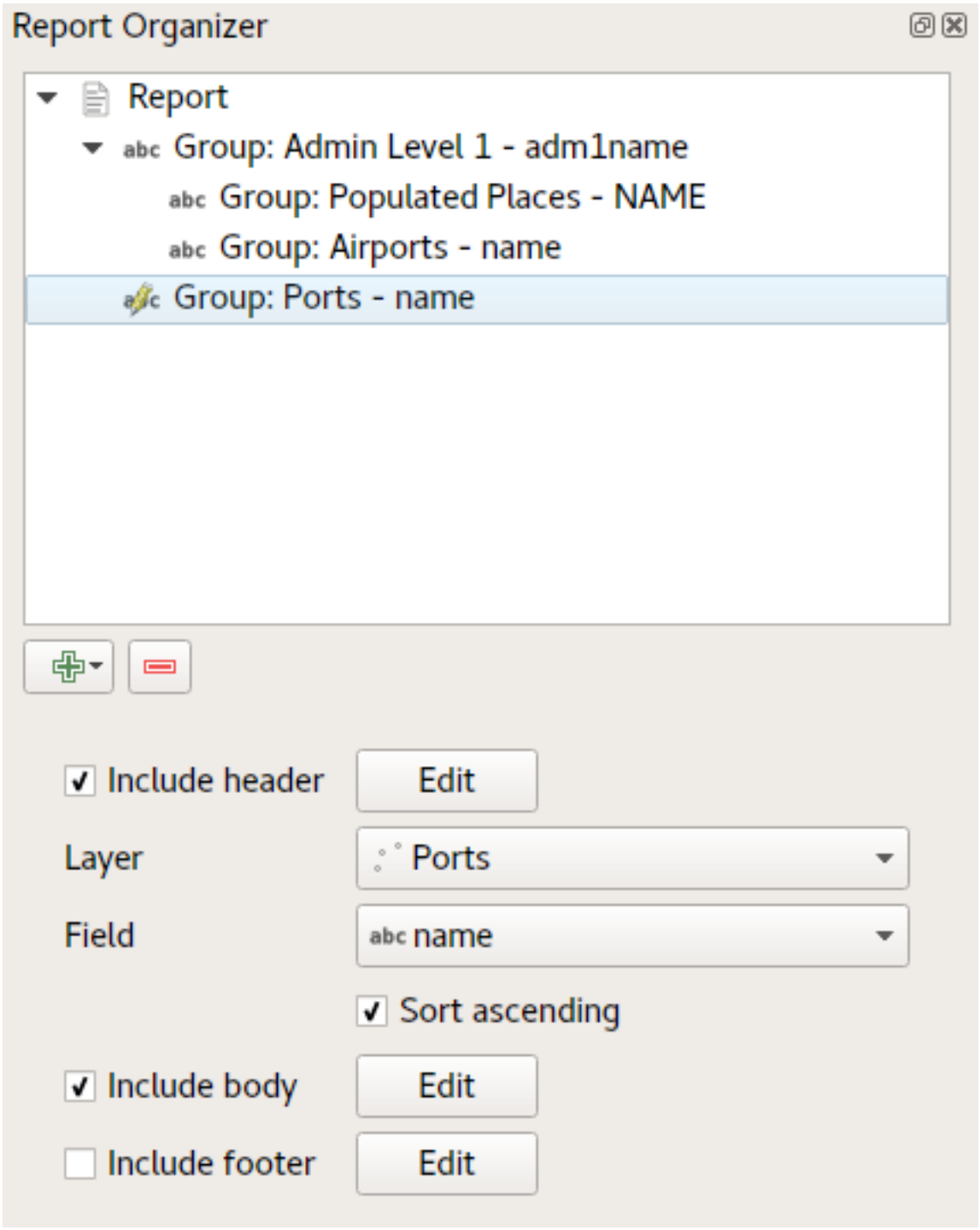
```
if($id=@atlas_featureid, 2.0, 0.1)
```

The report feature will get a 2 units wide polygon outline, while all other features will get a 0.1 units wide line. It is also possible to data define the colour (non-transparent dark magenta for the report feature and semi-transparent light gray for the other features):

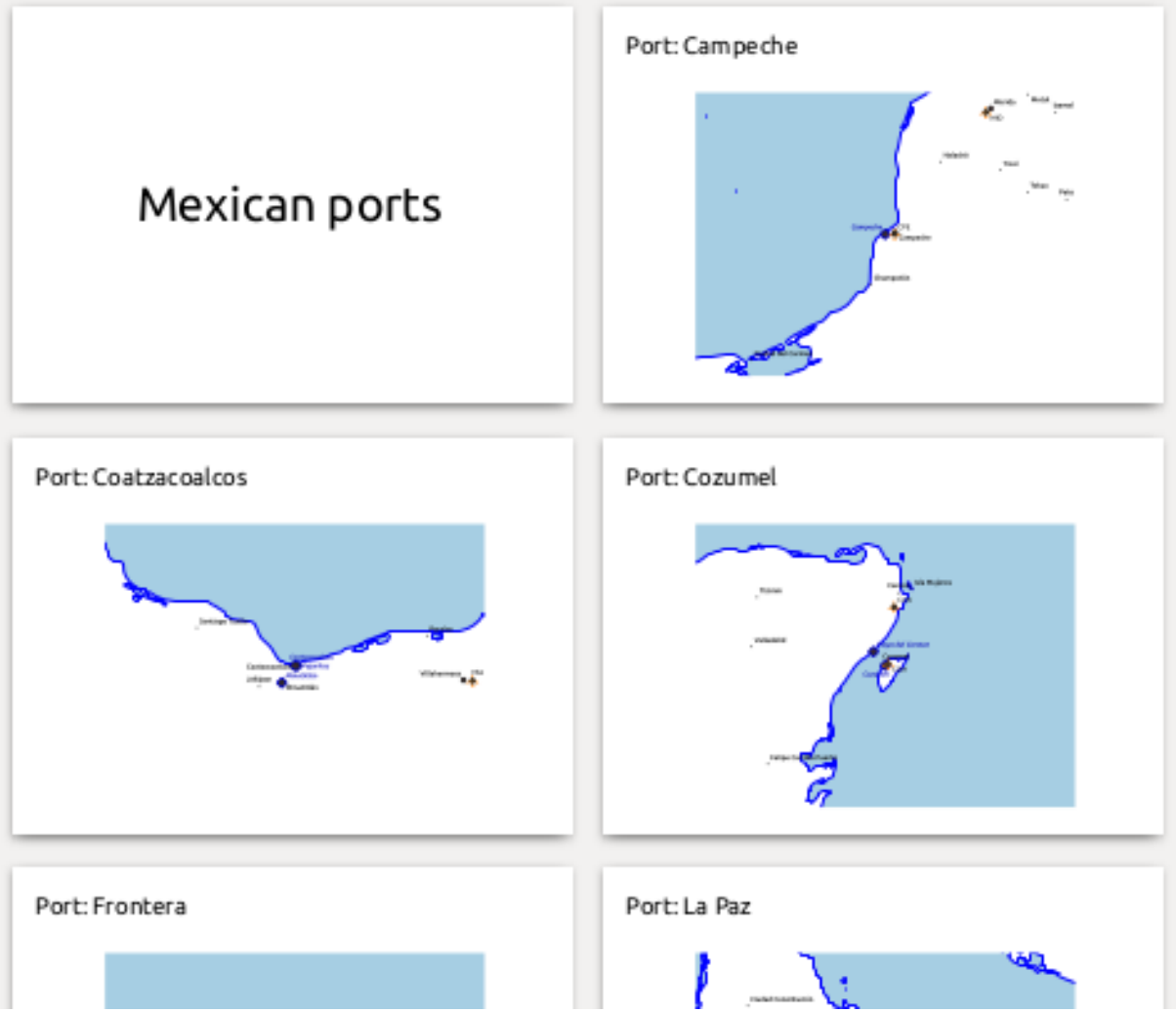
```
if($id=@atlas_featureid, '#FF880088', '#88CCCCCC')
```

More level 1 groups

Combining nested and consecutive sections, together with section headers and footers allows for tons of flexibility. For instance, in the below report we add another field group as a child of the main report for the *Ports* layer. Now, after listing the states together with their populated places and airports, we'll get a summary list of all the ports in the region:



This results in the last part of our report exporting as:



18.4.4 Podešavanja izvoza

When you export a report (*Report* ► *Export Report as Images… / SVG… / PDF…*), you will be asked for a file name, and then you get the opportunity to tune the export settings to get the most appropriate output.

As you see, reports in QGIS are extremely powerful and flexible!

Белешка

The current information was adapted from a North Road blog, Exploring Reports in QGIS 3.0 - the Ultimate Guide!

ГЛАВА 19

Working with OGC / ISO protocols

The Open Geospatial Consortium (OGC) is an international organization with membership of more than 300 commercial, governmental, nonprofit and research organizations worldwide. Its members develop and implement standards for geospatial content and services, GIS data processing and exchange.

Describing a basic data model for geographic features, an increasing number of specifications are developed by OGC to serve specific needs for interoperable location and geospatial technology, including GIS. Further information can be found at https://www.ogc.org/.

Important OGC specifications supported by QGIS are:

- **WMS** — Web Map Service (*WMS/WMTS Client*)
- **WMTS** — Web Map Tile Service (*WMS/WMTS Client*)
- **WFS** — Web Feature Service (*WFS and WFS-T Client*)
- **WFS-T** — Web Feature Service - Transactional (*WFS and WFS-T Client*)
- **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*).

You can also share your maps and data through the WMS, WMTS, WFS, WFS-T and WCS protocols using a webserver with QGIS Server, UMN MapServer or GeoServer installed.

19.1 WMS/WMTS Client

19.1.1 Overview of WMS Support

QGIS currently can act as a WMS client that understands WMS 1.1, 1.1.1 and 1.3 servers. In particular, it has been tested against publicly accessible servers such as DEMIS.

A WMS server acts upon requests by the client (e.g., QGIS) for a raster map with a given extent, set of layers, symbolization style, and transparency. The WMS server then consults its local data sources, rasterizes the map, and sends it back to the client in a raster format. For QGIS, this format would typically be JPEG or PNG.

WMS is generically a REST (Representational State Transfer) service rather than a full-blown Web service. As such, you can actually take the URLs generated by QGIS and use them in a web browser to retrieve the same images that QGIS uses internally. This can be useful for troubleshooting, as there are several brands of WMS server on the market and they all have their own interpretation of the WMS standard.

WMS layers can be added quite simply, as long as you know the URL to access the WMS server, you have a serviceable connection to that server, and the server understands HTTP as the data transport mechanism.

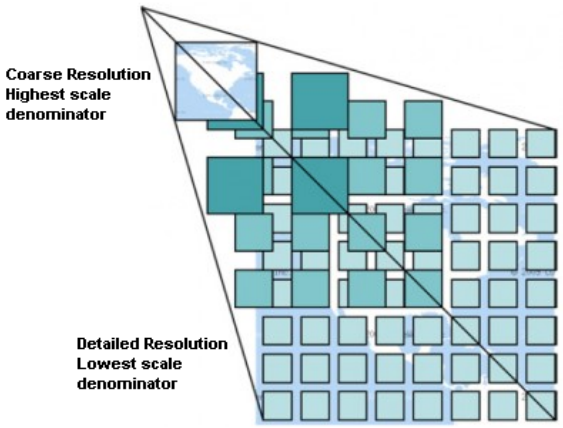
Additionally, QGIS will cache your WMS responses (i.e. images) for 24h as long as the GetCapabilities request is not triggered. The GetCapabilities request is triggered every time the *Connect* button in the *WMS/WMTS* dialog is used to retrieve the WMS server capabilities. This is an automatic feature meant to optimize project loading time. If a project is saved with a WMS layer, the corresponding WMS tiles will be loaded from the cache the next time the project is opened as long as they are not older than 24h.

19.1.2 Overview of WMTS Support

QGIS can also act as a WMTS client. WMTS is an OGC standard for distributing tile sets of geospatial data. This is a faster and more efficient way of distributing data than WMS because with WMTS, the tile sets are pre-generated, and the client only requests the transmission of the tiles, not their production. A WMS request typically involves both the generation and transmission of the data. A well-known example of a non-OGC standard for viewing tiled geospatial data is Google Maps.

In order to display the data at a variety of scales close to what the user might want, the WMTS tile sets are produced at several different scale levels and are made available for the GIS client to request them.

This diagram illustrates the concept of tile sets:



Слика19.1: Concept of WMTS tile sets

The two types of WMTS interfaces that QGIS supports are via Key-Value-Pairs (KVP) and RESTful. These two interfaces are different, and you need to specify them to QGIS differently.

1. In order to access a **WMTS KVP** service, a QGIS user must open the WMS/WMTS interface and add the following string to the URL of the WMTS tile service:

   ```
   "?SERVICE=WMTS&REQUEST=GetCapabilities"
   ```

   An example of this type of address is:

   ```
   https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
   ↪request=GetCapabilities
   ```

For testing the topo2 layer in this WMTS works nicely. Adding this string indicates that a WMTS web service is to be used instead of a WMS service.

2. The **RESTful WMTS** service takes a different form, a straightforward URL. The format recommended by the OGC is:

```
{WMTSBaseURL}/1.0.0/WMTSCapabilities.xml
```

This format helps you to recognize that it is a RESTful address. A RESTful WMTS is accessed in QGIS by simply adding its address in the WMS setup in the URL field of the form. An example of this type of address for the case of an Austrian basemap is:

```
https://maps.wien.gv.at/basemap/1.0.0/WMTSCapabilities.xml
```

Белешка

You can still find some old services called WMS-C. These services are quite similar to WMTS (i.e., same purpose but working a little bit differently). You can manage them the same as you do WMTS services. Just add `?tiled=true` at the end of the url. See https://wiki.osgeo.org/wiki/Tile_Map_Service_Specification for more information about this specification.

When you read WMTS, you can often think WMS-C also.

19.1.3 Selecting WMS/WMTS Servers

The first time you use the WMS/WMTS feature in QGIS, there are no servers defined.

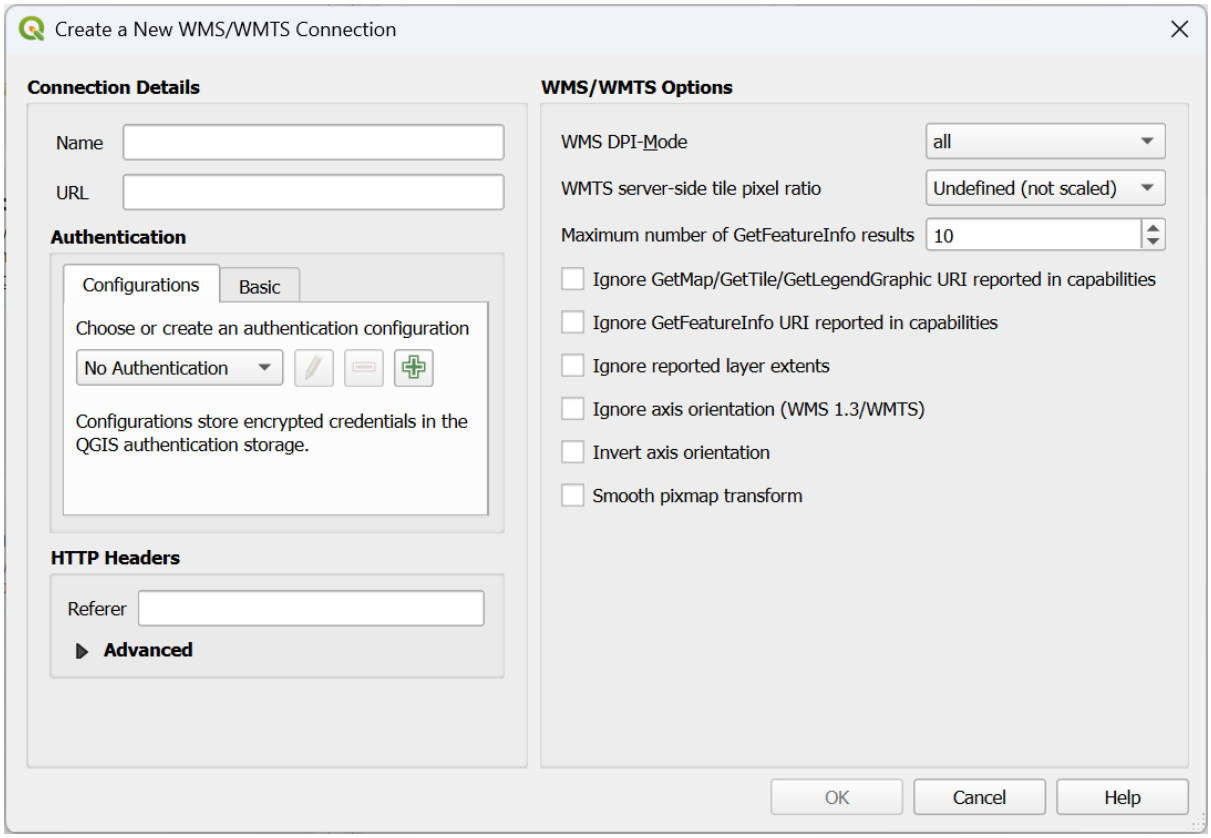
You then need to create connections to the server you are targeting:

1. Go to the *WMS/WMTS* tab of the *Data Source Manager* dialog, either by:
   - clicking the Open Data Source Manager button (or pressing `Ctrl+L`) and enabling the tab
   - clicking the Add WMS layer button on the *Manage Layers* toolbar
   - or selecting *Layer* ► *Add Layer* ► *Add WMS/WMTS Layer…* menu
2. Press *New* from the *Layers* tab. The *Create a New WMS/WMTS Connection…* dialog appears.

Савет

Right-click the *WMS/WMTS* entry from within the *Browser panel* and select *New Connection…* also opens the *Create a New WMS/WMTS Connection…* dialog.

3. Then enter the parameters to connect to your desired WMS server, as listed below:



Слика19.2: Creating a connection to a WMS 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 WMS servers.
- *URL*: URL of the server providing the data. This must be a resolvable host name – the same format as you would use to open a telnet connection or ping a host, i.e. the base URL only. For example, you shouldn't have fragments such as `request=GetCapabilities` or `version=1.0.0` in your URL.
- *Authentication* (optional): using a *stored configuration* or a basic authentication with *Username* and *Password*.

Упозорење

Entering **username** and **password** in the *Authentication* tab will keep unprotected credentials in the connection configuration. Those **credentials will be visible** if, for instance, you shared the project file with someone. Therefore, it's advisable to save your credentials in an *Authentication configuration* instead (*Configurations* tab). See *Sistem za autentifikaciju* for more details.

- HTTP *Referer*
- *WMS DPI-Mode*: Available options are **all**, **off**, **QGIS**, **UMN** and **GeoServer**

- *WMTS server-side tile pixel ratio*: When rendering WMTS layers, allows to scale up or down the tiles based on the device screen DPI. Available options are *Undefined (not scaled)*, *Standard (96 DPI)* and *High (192 DPI)*.
- *Maximum number of GetFeatureInfo results*: specifies a default value for the maximum number of results returned per layer by a GetFeatureInfo request using this connection (see FEATURE_COUNT parameter). Default value is `10`. Set to `0` to use server default value (usually `1`): no FEATURE_COUNT parameter will be added to the request.
- ☐ *Ignore GetMap/GetTile/GetLegendGraphic URI reported in capabilities*: if checked, use given URI from the *URL* field above.
- ☐ *Ignore GetFeatureInfo 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 WMS 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 (WMS 1.3/WMTS)*
- ☐ *Invert axis orientation*
- ☐ *Smooth pixmap transformation*

4. Pritisnite *OK*

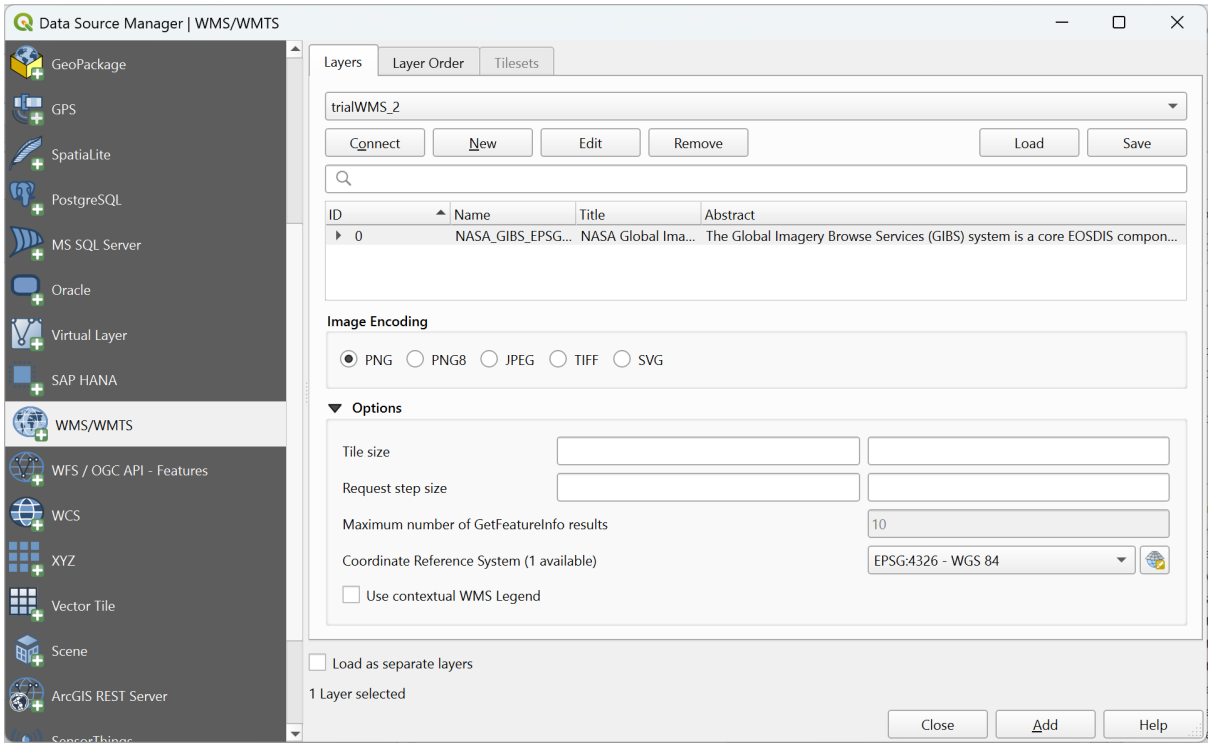
Once the new WMS/WMTS server connection has been created, it will be preserved for future QGIS sessions. Note that it is also possible to *Load* the connection parameters from a `.XML` file or *Save* them to a `.XML` file.

If you need to set up a proxy server to be able to receive WMS services from the internet, you can add your proxy server in the options. Choose *Settings* ► *Options* and click on the *Network* tab. There, you can add your proxy settings and enable them by setting ☑ *Use proxy for web access*. Make sure that you select the correct proxy type from the *Proxy type* … ▾ drop-down menu.

19.1.4 Loading WMS/WMTS Layers

Once you have successfully filled in your parameters, you can use the *Connect* button to retrieve the capabilities of the selected server. This includes the image encoding, layers, layer styles and projections. Since this is a network operation, the speed of the response depends on the quality of your network connection to the WMS server. While downloading data from the WMS server, the download progress is visualized in the lower left corner of the main QGIS dialog.

Your screen should now look a bit like Сл.19.3, which shows the response provided by a WMS server.



Слика19.3: Dialog for adding a WMS server, with filter on available layers

The upper part of the *Layers* tab of the dialog shows a tree structure that can include layer groups embedding layers with their associated image style(s) served by the server. Each item can be identified by:

- an *ID*
- a *Name*
- a *Title*
- and an *Abstract*.

The list can be filtered using the widget in the top right corner.

Image Encoding

The *Image encoding* section lists the formats that are supported by both the client and server. Choose one depending on your image accuracy requirements.

Савет

Image Encoding

You will typically find that a WMS server offers you the choice of JPEG or PNG image encoding. JPEG is a lossy compression format, whereas PNG faithfully reproduces the raw raster data.

Use JPEG if you expect the WMS data to be photographic in nature and/or you don't mind some loss in picture quality. This trade-off typically reduces by five times the data transfer requirement compared with PNG.

Use PNG if you want precise representations of the original data and you don't mind the increased data transfer requirements.

Options

The Options area of the dialog provides means to configure the WMS requests. You can define:

- *Tile size* if you want to set tile sizes (e.g., 256x256) to split up the WMS request into multiple requests.

- *Request step size*: if you want to reduce the effect of cut labels at tile borders, increasing the step size creates larger requests, fewer tiles and fewer borders. The default value is 2000.
- The *Maximum number of GetFeatureInfo results*: specifies the maximum number of results returned by a GetFeatureInfo request, for the layer(s) being loaded. Allows to override the *connection's default value* for specific layers.
- Each WMS layer can be presented in multiple CRSs, depending on the capability of the WMS server. If you select a WMS from the list, a field with the default projection provided by the web server appears. Press the  Select CRS widget to replace the default projection of the WMS with another CRS supported by the WMS server.

  A dialog similar to the one shown in Сл.6.4 will appear. The main difference with the WMS version of the dialog is that only those CRSs supported by the WMS server will be shown.
- Finally you can activate  *Use contextual WMS Legend* if the WMS Server supports this feature. Then only the relevant legend for your current map view extent will be shown and thus will not include legend items for items you can't see in the current map.

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.

Layer Order

The *Layer Order* tab lists the selected layers available from the current connected WMS server.

WMS layers rendered by a server are overlaid in the order listed in the *Layers* tab, from top to bottom of the list. If you want to change the overlay order, you can use the *Up* and *Down* buttons of the *Layer Order* tab.

Transparency

The *Global transparency* setting from the *Layer Properties* is hard coded to be always on, where available.

19.1.5 Tilesets

When using WMTS (Cached WMS) services you are able to browse through the *Tilesets* tab given by the server. Additional information like tile size, formats and supported CRS are listed in this table.

```
# example of WMTS service

https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
↪request=GetCapabilities
```

Selecting a layer to load, it is also possible to apply an *Interpretation method*, converting it into a raster layer of single band float type, ready for styling using QGIS usual *raster renderers*.

In combination with this feature, you can use the tile scale slider by selecting *View* ► *Panels* (or  *Settings* ► *Panels*), then choosing *Tile Scale Panel*. This gives you the available scales from the tile server with a nice slider docked in.

19.1.6 Using the Identify Tool

Once you have added a WMS server, and if any layer from a WMS server is queryable, you can then use the  Identify tool to select a pixel on the map canvas. A query is made to the WMS server for each selection made. The results of the query are returned in plain text. The formatting of this text is dependent on the particular WMS server used.

Format selection

If multiple output formats are supported by the server, a combo box with supported formats is automatically added to the identify results dialog and the selected format may be stored in the project for the layer.

GML format support

The  Identify tool supports WMS server response (GetFeatureInfo) in GML format (it is called Feature in the QGIS GUI in this context). If „Feature" format is supported by the server and selected, results of the Identify tool are vector features, as from a regular vector layer. When a single feature is selected in the tree, it is highlighted in the map and it can be copied to the clipboard and pasted to another vector layer. See the example setup of the UMN Mapserver below to support GetFeatureInfo in GML format.

```
# 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 Viewing Properties

Once you have added a WMS server, you can view its properties by right-clicking on it in the legend and selecting *Properties*. The WMS/WMTS layer properties is much like the raster layer properties so you will find detailed description at *Raster Properties Dialog*. However, there are some differences, which will be explained below.

Information properties

Metadata Tab

The tab *Metadata* displays a wealth of information about the WMS server, generally collected from the capabilities statement returned from that server. Many definitions can be gleaned by reading the WMS standards (see OPEN--GEOSPATIAL-CONSORTIUM in *Literatura i veb reference*), but here are a few handy definitions:

- **Server Properties**
  - **WMS Version** — The WMS version supported by the server.
  - **Image Formats** — The list of MIME-types the server can respond with when drawing the map. QGIS supports whatever formats the underlying Qt libraries were built with, which is typically at least `image/png` and `image/jpeg`.
  - **Identity Formats** — The list of MIME-types the server can respond with when you use the Identify tool. Currently, QGIS supports the `text-plain` type.
- **Layer Properties**
  - **Selected** — Whether or not this layer was selected when its server was added to this project.

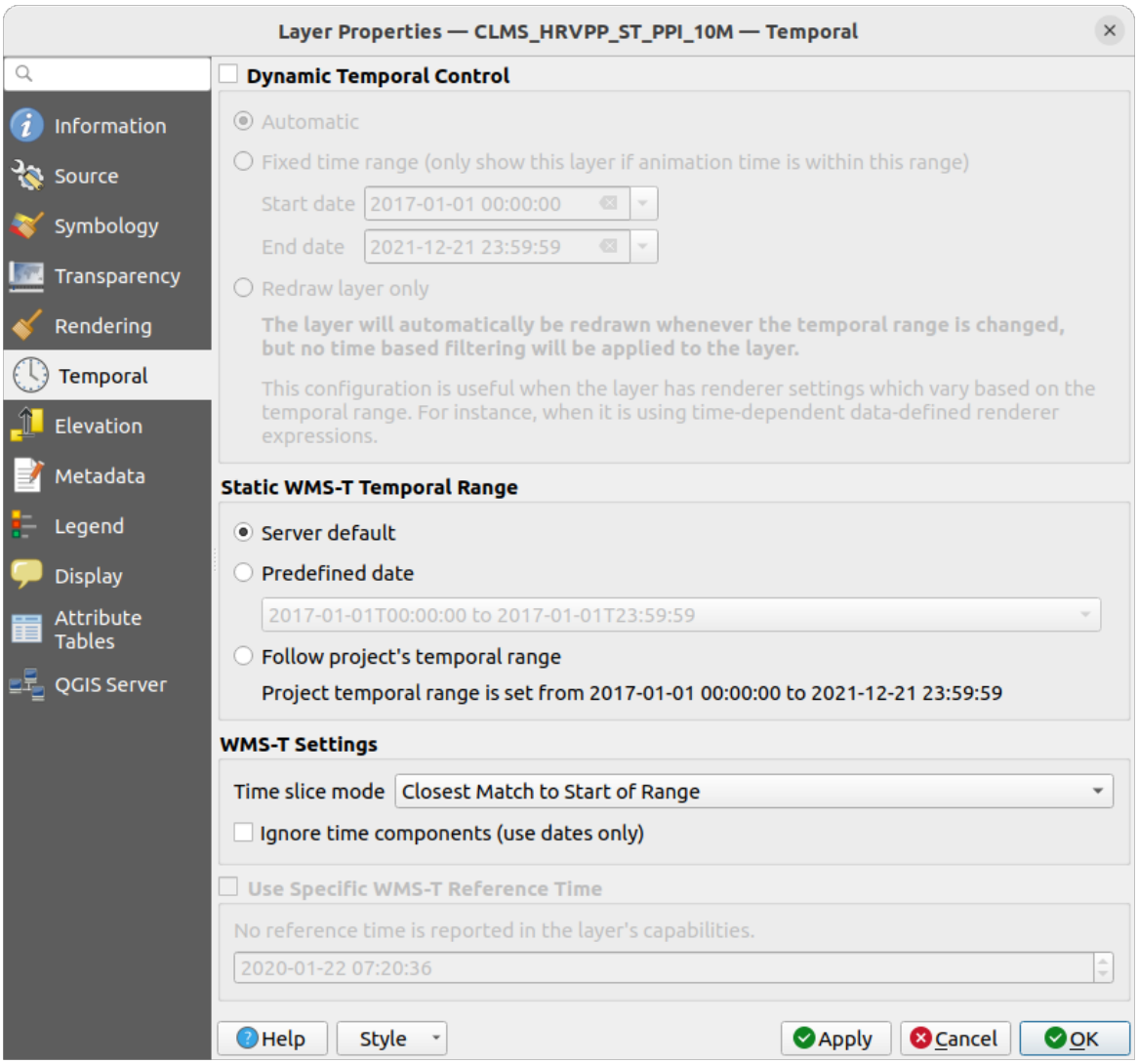
- **Visible** — Whether or not this layer is selected as visible in the legend (not yet used in this version of QGIS).
- **Can Identify** — Whether or not this layer will return any results when the Identify tool is used on it.
- **Can be Transparent** — Whether or not this layer can be rendered with transparency. This version of QGIS will always use transparency if this is `Yes` and the image encoding supports transparency.
- **Can Zoom In** — Whether or not this layer can be zoomed in by the server. This version of QGIS assumes all WMS layers have this set to `Yes`. Deficient layers may be rendered strangely.
- **Cascade Count** — WMS servers can act as a proxy to other WMS servers to get the raster data for a layer. This entry shows how many times the request for this layer is forwarded to peer WMS servers for a result.
- **Fixed Width, Fixed Height** — Whether or not this layer has fixed source pixel dimensions. This version of QGIS assumes all WMS layers have this set to nothing. Deficient layers may be rendered strangely.
- **WGS 84 Bounding Box** — The bounding box of the layer, in WGS 84 coordinates. Some WMS servers do not set this correctly (e.g., UTM coordinates are used instead). If this is the case, then the initial view of this layer may be rendered with a very 'zoomed-out' appearance by QGIS. The WMS webmaster should be informed of this error, which they may know as the WMS XML elements `LatLonBoundingBox`, `EX_GeographicBoundingBox` or the CRS:84 `BoundingBox`.
- **Available in CRS** — The projections that this layer can be rendered in by the WMS server. These are listed in the WMS-native format.
- **Available in style** — The image styles that this layer can be rendered in by the WMS server.

Temporal properties

Raster *temporal properties* (namely *Dynamic Temporal Control*) can be set for WMS and WMTS layers. By default, when a time-dimension enabled WMS or WMTS layer is added to the project, it is indicated in the *Layers* panel with the Temporal Layer icon next to it. Its *Temporal* properties default to the *Automatic* temporal mode, meaning that the layer will follow the temporal controller's current time range by default.

You can then opt to show a specific static time value for the layer by unchecking *Dynamic Temporal Control* and picking an option under *Static WMS-T Temporal Range*:

- *Server default*
- *Predefined date* with a server exposing data for non-contiguous temporal ranges or *Predefined range* with a server exposing a range of available dates. A *Start date* and *End date* are necessary in the latter case. Their expected formatting can be deduced from the reference time option (see below). depending on whether the provider has data for contiguous period or not
- *Follow project's temporal range* as defined in the project's properties dialog



Слика19.4: Temporal properties of a WMTS layer

Whatever temporal data control is in use, there are some *WMS-T Settings* to help display the correct temporal data:

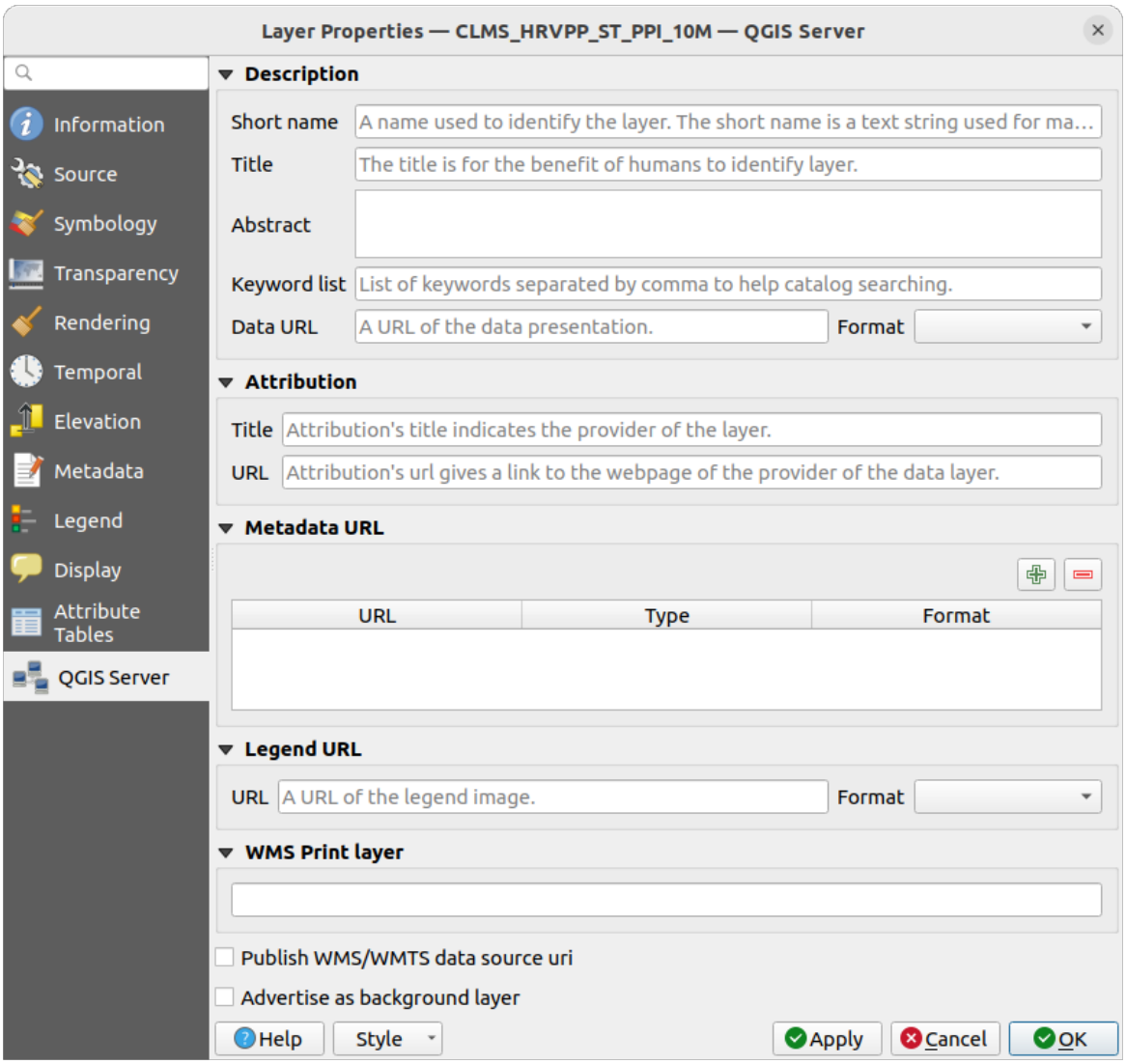
- *Time slice mode* which can be:
  - *Use whole temporal range*
  - *Match to start of range*
  - *Match to end of range*
  - *Closest match to start of range*
  - *Closest match to end of range*
- *Ignore time components (use dates only)*: If checked, the time component of temporal queries will be discarded and only the date component will be used in server requests.

You can also *Use Specific WMS-T Reference Time* picked from times reported in the layer's capabilities. Convenient for servers which expose a non-contiguous set of date time instances (instead of a range of dates).

QGIS Server properties

In addition to the raster layer *properties*, publishing a WMS/WMTS layer with QGIS Server will display following options:

- *WMS Print layer*: Allows to set alternative WMS layer that should be used for printing (in the GetProjectSettings reply). Convenient for WMTS layers that are generally not suitable for printing.
- ☑ *Publish WMS/WMTS data source uri*: Allows a web client to fetch the WMS/WMTS data directly
- ☑ *Advertise as background layer*



Слика19.5: QGIS Server properties of a WMS/WMTS layer

19.1.8 Show WMS legend graphic in table of contents and layout

The QGIS WMS data provider is able to display a legend graphic in the table of contents' layer list and in the print layout. The WMS legend will be shown only if the WMS server has GetLegendGraphic capability and the layer has getCapability url specified, so you additionally have to select a styling for the layer.

If a legendGraphic is available, it is shown below the layer. It is little and you have to click on it to open it in real dimension (due to QgsLegendInterface architectural limitation). Clicking on the layer's legend will open a frame with the legend at full resolution.

In the print layout, the legend will be integrated at it's original (downloaded) dimension. Resolution of the legend graphic can be set in the item properties under *Legend* ► *WMS LegendGraphic* to match your printing requirements.

The legend will display contextual information based on your current scale. The WMS legend will be shown only if the WMS server has GetLegendGraphic capability and the layer has getCapability url specified, so you have to select a styling.

19.2 WCS Client

A Web Coverage Service (WCS) provides access to raster data in forms that are useful for client-side rendering, as input into scientific models, and for other clients. The WCS may be compared to the WFS and the WMS. As WMS and WFS service instances, a WCS allows clients to choose portions of a server's information holdings based on spatial constraints and other query criteria.

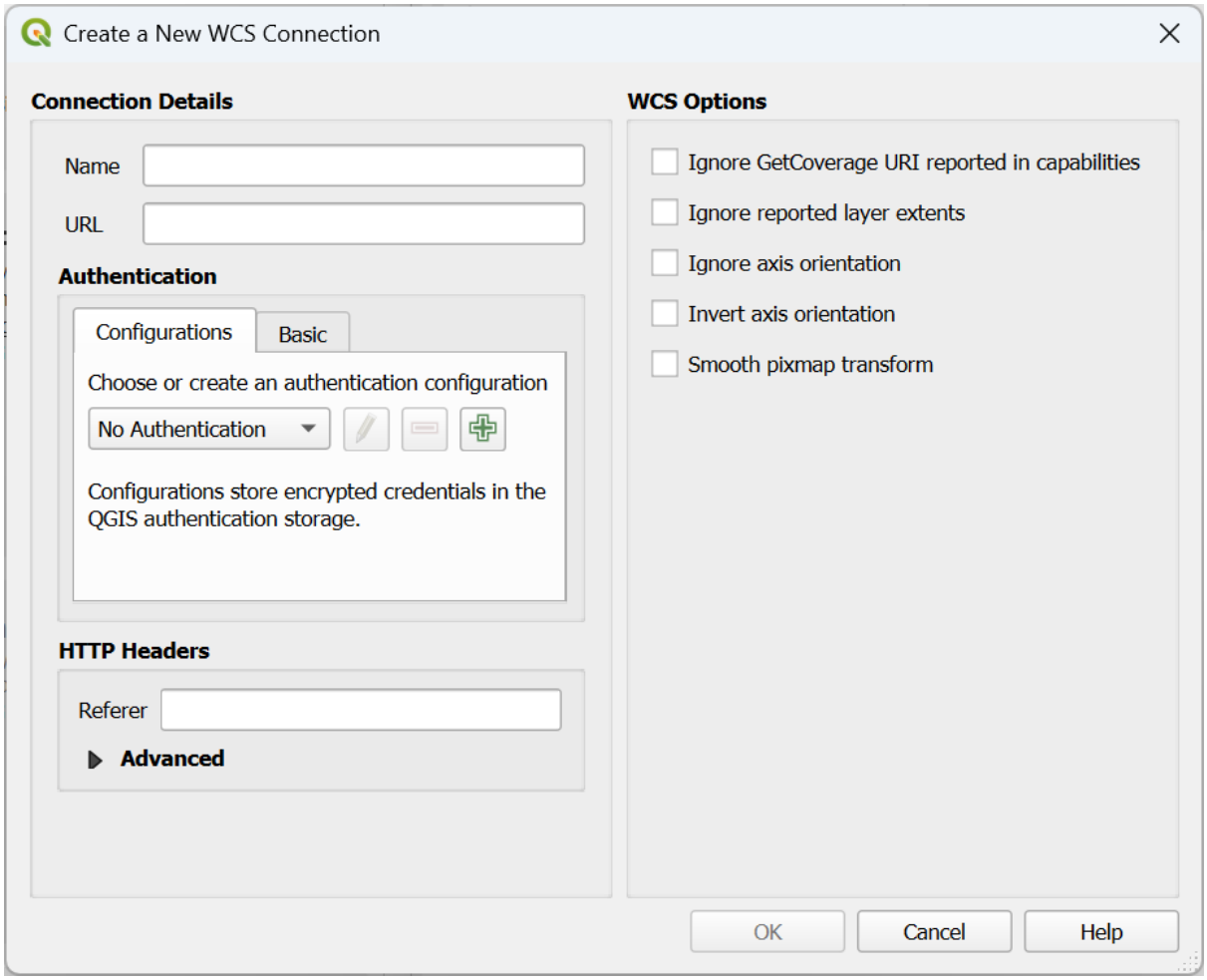
QGIS has a native WCS provider and supports both version 1.0 and 1.1 (which are significantly different), but currently it prefers 1.0, because 1.1 has many issues (i.e., each server implements it in a different way with various particularities).

The native WCS provider handles all network requests and uses all standard QGIS network settings (especially proxy). It is also possible to select cache mode ('always cache', 'prefer cache', 'prefer network', 'always network'), and the provider also supports selection of time position, if temporal domain is offered by the server.

Loading a WCS Layer

To be able to load a WCS Layer, first create a connection to the WCS server:

1. Open the *Data Source Manager* dialog by pressing the Open Data Source Manager button
2. Enable the *WCS* tab
3. Click on *New…* to open the *Create a New WCS Connection* dialog



Слика19.6: Creating a connection to a WCS server

- *Name*: A name for the connection. This name will be used in the Server Connections drop-down box so that you can distinguish it from other WCS servers.
- *URL*: URL of the server providing the data. This must be a resolvable host name – the same format as you would use to open a telnet connection or ping a host, i.e. the base URL only. For example, you shouldn't have fragments such as `request=GetCapabilities` or `version=1.0.0` in your URL.
- *Authentication* (optional): using a *stored configuration* or a basic authentication with *Username* and *Password*.

> **Упозорење**
>
> Entering **username** and **password** in the *Authentication* tab will keep unprotected credentials in the connection configuration. Those **credentials will be visible** if, for instance, you shared the project file with someone. Therefore, it's advisable to save your credentials in an *Authentication configuration* instead (*Configurations* tab). See *Sistem za autentifikaciju* for more 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*
- ☐ *Invert axis orientation*
- ☐ *Smooth pixmap transformation*

4. Press *OK* to create the connection.

Note that any proxy settings you may have set in your preferences are also recognized. Also note that it is possible to *Load* the connection parameters from a `.XML` file or *Save* them to a `.XML` file.

Now we are ready to load WCS layers from the above connection.

19.3 WFS and WFS-T Client

In QGIS, a WFS layer behaves pretty much like any other vector layer. You can identify and select features, and view the attribute table. QGIS supports WFS 1.0.0, 1.1.0, 2.0 and OGC API - Features (OAPIF), including editing (through WFS-T). QGIS also supports background download and progressive rendering, on-disk caching of downloaded features and version autodetection.

Layers of servers implementing OGC API - Features - Part 4: Create, Replace, Update and Delete can be turned into editing mode to allow creating, modifying and deleting features. Note that each created/modified/deleted feature requires a dedicated network request, so performance might suffer in case of simultaneous modification of hundreds or more features at a time.

In general, adding a WFS layer is very similar to the procedure used with WMS. There are no default servers defined, so you have to add your own. You can find WFS servers by using the *MetaSearch plugin* or your favourite web search engine. There are a number of lists with public URLs, some of them maintained and some not.

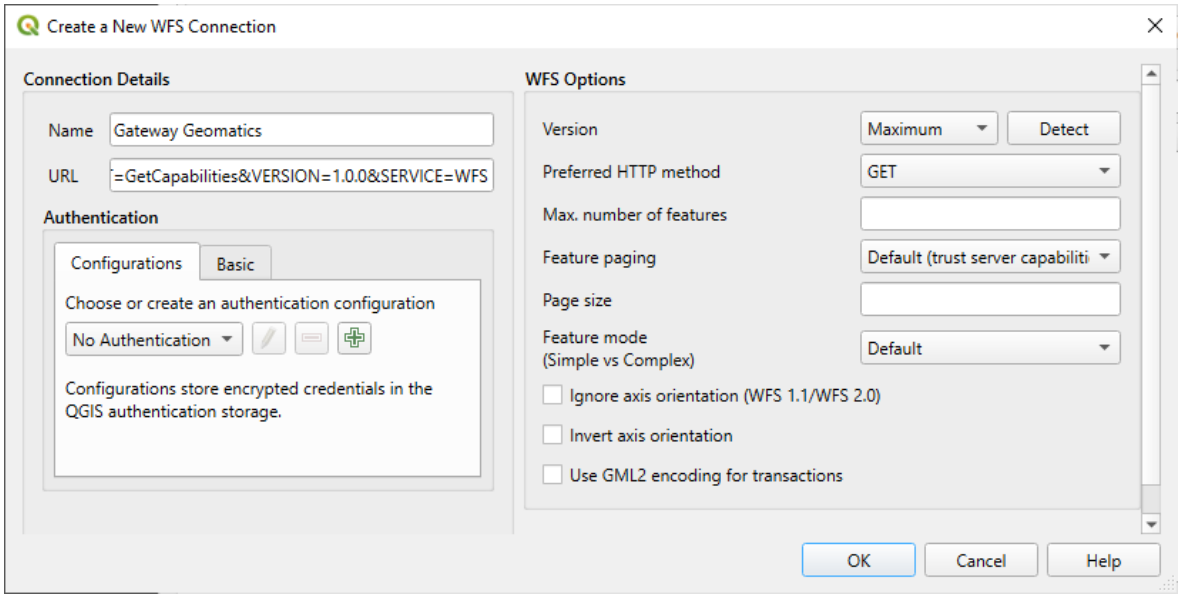
Loading a WFS Layer

As an example, we use the Gateway Geomatics WFS server and display a layer.

```
https://demo.gatewaygeomatics.com/cgi-bin/wfs_gateway?REQUEST=GetCapabilities&
↪VERSION=1.0.0&SERVICE=WFS
```

To be able to load a WFS Layer, first create a connection to the WFS server:

1. Open the *Data Source Manager* dialog by pressing the Open Data Source Manager button
2. Enable the *WFS / OGC API - Features* tab
3. Click on *New…* to open the *Create a New WFS Connection* dialog
4. In the *Connection Details* group, enter `Gateway Geomatics` as name
5. Enter the URL (see above)
6. Optionally, enter the *Authentication* settings



Слика19.7: Creating a connection to a WFS server

Белешка

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:
   - Indicate the WFS version of the server. If unknown, press the *Detect* button to automatically retrieve it.
   - 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.
     - Force to *Ignore axis orientation (WFS 1.1/WFS 2.0)*
     - *Invert axis orientation*.
     - *Use GML2 encoding for transactions*.

Упозорење

Entering **username** and **password** in the *Authentication* tab will keep unprotected credentials in the connection configuration. Those **credentials will be visible** if, for instance, you shared the project file with someone. Therefore, it's advisable to save your credentials in an *Authentication configuration* instead (*Configurations* tab). See *Sistem za autentifikaciju* for more details.

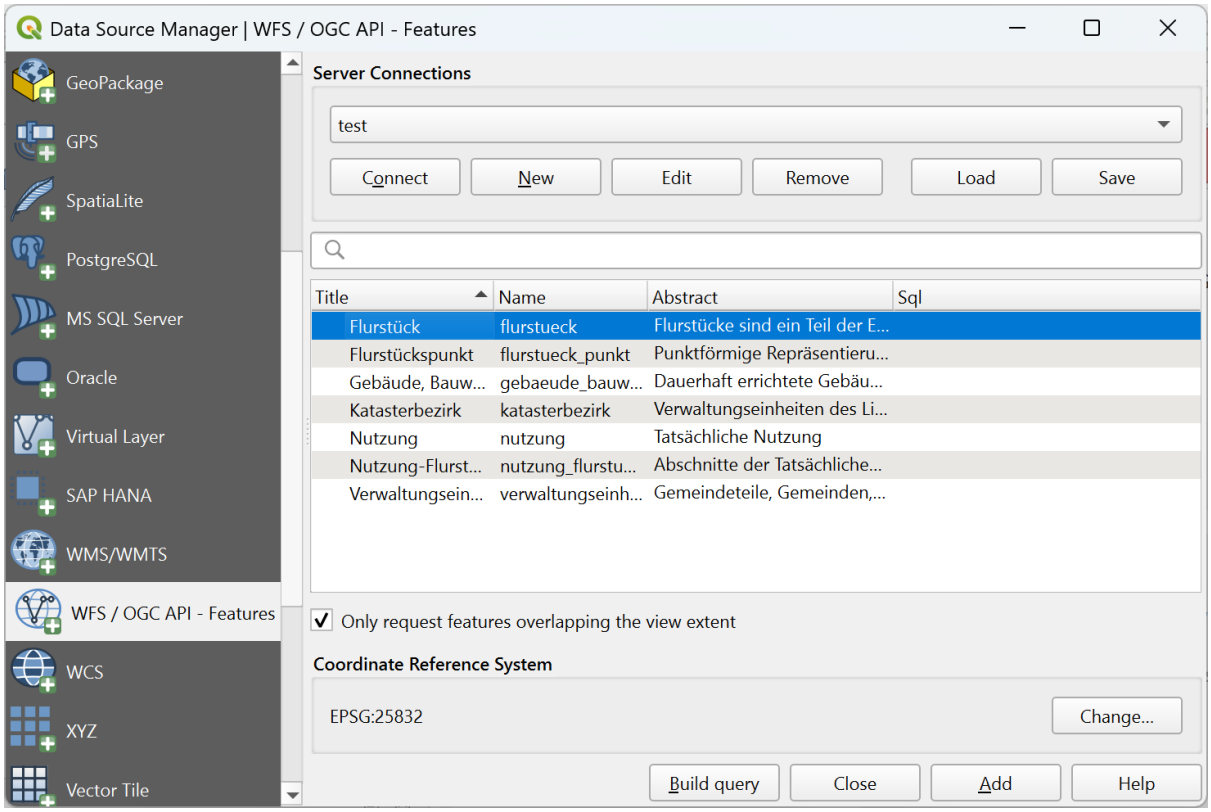
8. Press *OK* to create the connection.

Note that any proxy settings you may have set in your preferences are also recognized. Also note that it is possible to *Load* the connection parameters from a `.XML` file or *Save* them to a `.XML` file.

Now we are ready to load WFS layers from the above connection.

1. Choose 'Gateway Geomatics' from the *Server Connections* drop-down list.
2. Click *Connect*
3. Select the *Parks* layer in the list
4. You can also choose whether to:
   - *Only request features overlapping the view extent*
   - *Change…* the layer's CRS to any other supported by the service
   - or build a query to specify particular features to retrieve from the service: double-click on the row of the layer to open the *SQL Query Composer* dialog. That dialog provides widgets to write an advanced SQL query relying on available tables and columns of the service, with sorting and filtering and a bunch of SQL functions, spatial predicates and operators.

     The query you build will appear after validation in the *SQL* column within the *WFS / OGC API - Features* table, and the filtered layer will display the icon next to it in the *Layers* panel. It is thus possible to adjust the query at any moment.
5. Click *Add* to add the layer to the map, using its default name from the queried service.



Слика19.8: Adding a WFS layer

You'll notice the download progress is visualized in the lower left of the QGIS main window. Once the layer is loaded, you can identify and select a couple of features and view the attribute table.

19.4 SensorThings

QGIS supports connections to OGC SensorThings API, a standard providing an open and unified framework to interconnect IoT sensing devices, data, and applications over the Web. It is an open standard addressing the syntactic and semantic interoperability of the Internet of Things. It is based on the Observations and Measurements data model, a standardized model for observations, and for features involved in sampling when making observations.

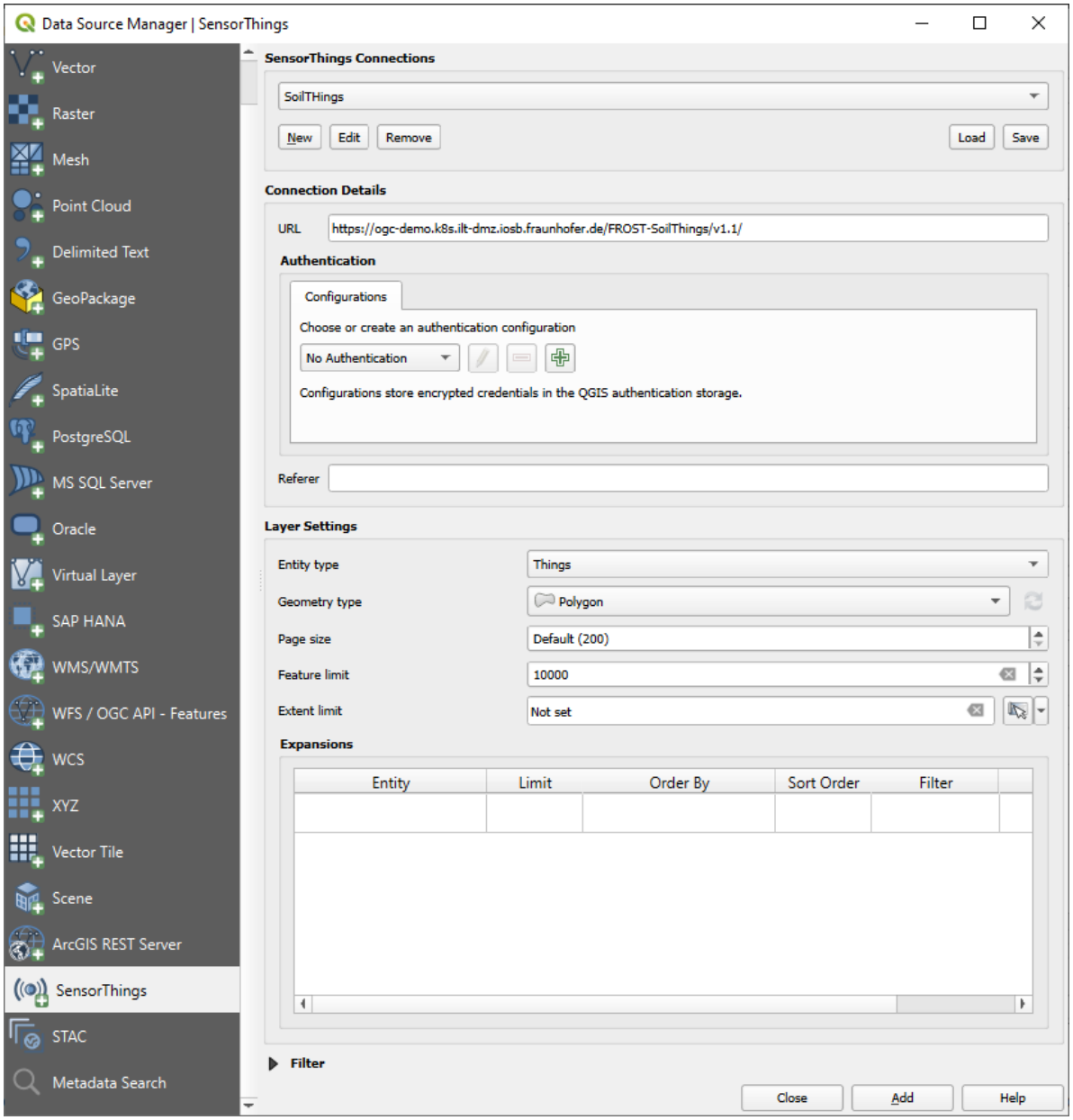
19.4.1 Setting connection

To add SensorThings data to QGIS use the *SensorThings* tab in the *Data Source Manager* dialog.

To establish a new connection, press *New* (or *New SensorThings Connection* from the Browser panel) and provide *Name* and *URL*. Advanced options, such as *authentication* and a *Referer*, can also be configured.

Press *OK* to establish the connection. Then you will be able to:

- *Edit* the SensorThings connection settings
- *Remove* the SensorThings connection

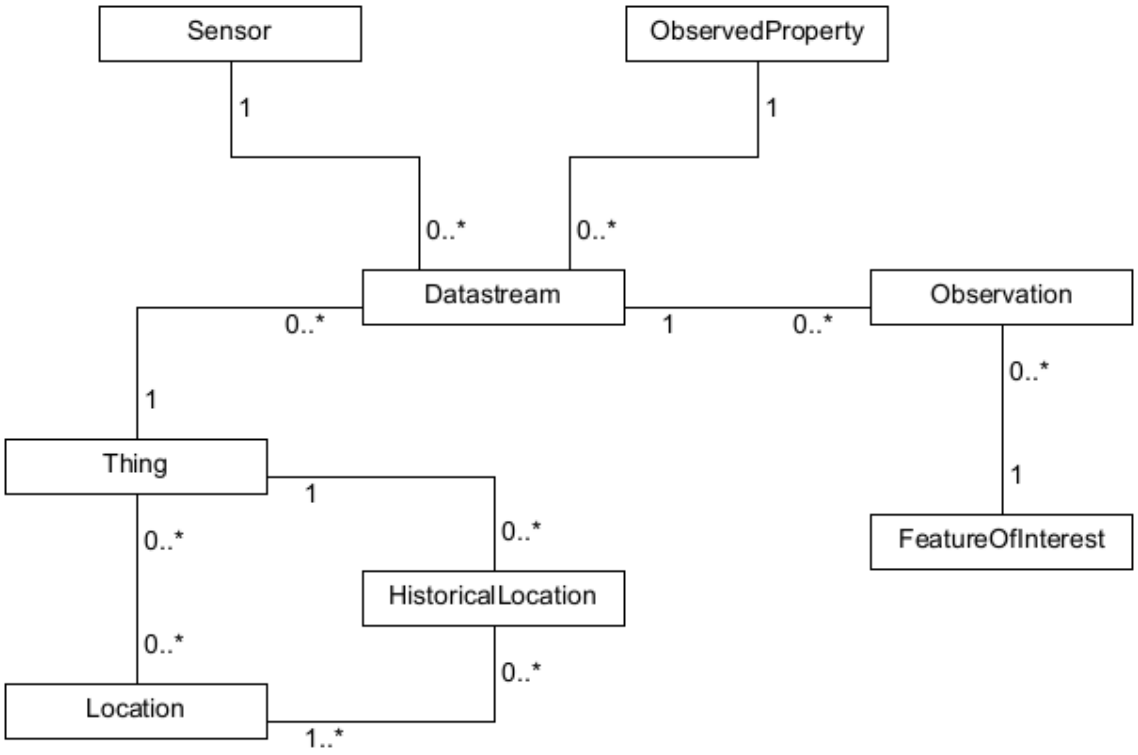


Слика19.9: SensorThings Connection dialog

Configurations can be saved to an `.XML` file (*Save*) through the *SensorThings* entry in *Data Source Manager* dialog or its contextual menu in the *Browser* panel (*Save Connections*). Likewise, configurations can be added from a file (*Load*).

19.4.2 Loading SensorThings data

Relations between layers (so-called entities) stored in a SensorThings dataset are expressed in the diagram below.



Слика19.10: Data model Observations and Measurements (Source: Sensor Web Tutorial by SIST network)

Any type of entity can be loaded in QGIS, but not all are spatial data. To load an entity, there are *Layer Settings* that can be configured:

- *Entity Type*: the entity to load from the data model as layer in QGIS
- *Geometry Type*: the geometry type of the selected entity to load. Press Check available types to limit the list to the actually supported geometry types.
- *Page Size*
- *Feature Limit* sets a maximum number of features to request from the service
- *Extent Limit* sets a maximum extent limit for the layer, so that only features within the extent are requested
- *Expansions*: The data model of SensorThings provides a mechanism of expansion of the results to related entities, similar to how tables are joined together in a relational database. Using this approach, you can expand the selected layer to include data from other items. This will flatten the relationship, creating as many parent features as children, and additional properties are added as columns in the attribute table.

Use *Filter* to build a query to filter the data, using SensorThings filter syntax.

Белешка

The above settings and filtering options are also available for update in the layer properties dialog, *Source* tab, once loaded in QGIS.

Press *Add* to load the selected entity type as layer in QGIS.

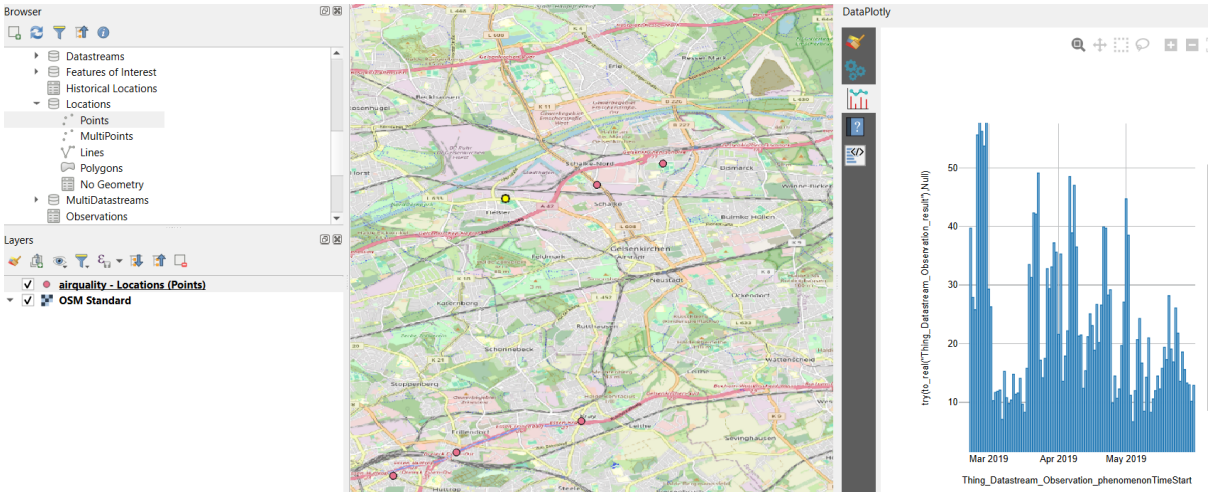
19.4.3 Working with a vector layer from SensorThings

A SensorThings layer is loaded in QGIS as a vector layer. As such, it displays the same tabs in the *layer properties* and allows same feature interactions using the selection or identify tools. There are however some specificities you should consider while working with SensorThings data.

Because of the data model, the result property of a SensorThings Observation is a string field. In case you want to use its numerical representation in for example a graduated style, use an expression to convert the value to real and try() in case this fails (e.g., `try( to_real("Observation_result"), Null)`).

In case you want to create a chart of the observations at one or more locations, you can install the QGIS plugin *Data Plotly*.

1. Now select the observations at a point location in the map view.
2. Open the plotly panel and activate the *Use only selected features* checkbox.
3. Select on the x-column a date-time property and on the y-column the *Observation_result*. This will plot the observations at that location over time.
4. Verify to filter by a single Observed Property.
5. Notice that the chart changes as soon as you select other locations on the map.



Слика19.11: Use Data plotly to plot the air quality observations at a location

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 Setting connection

STAC connections can be added in QGIS using either the *Browser panel* or the *Data Source Manager*:

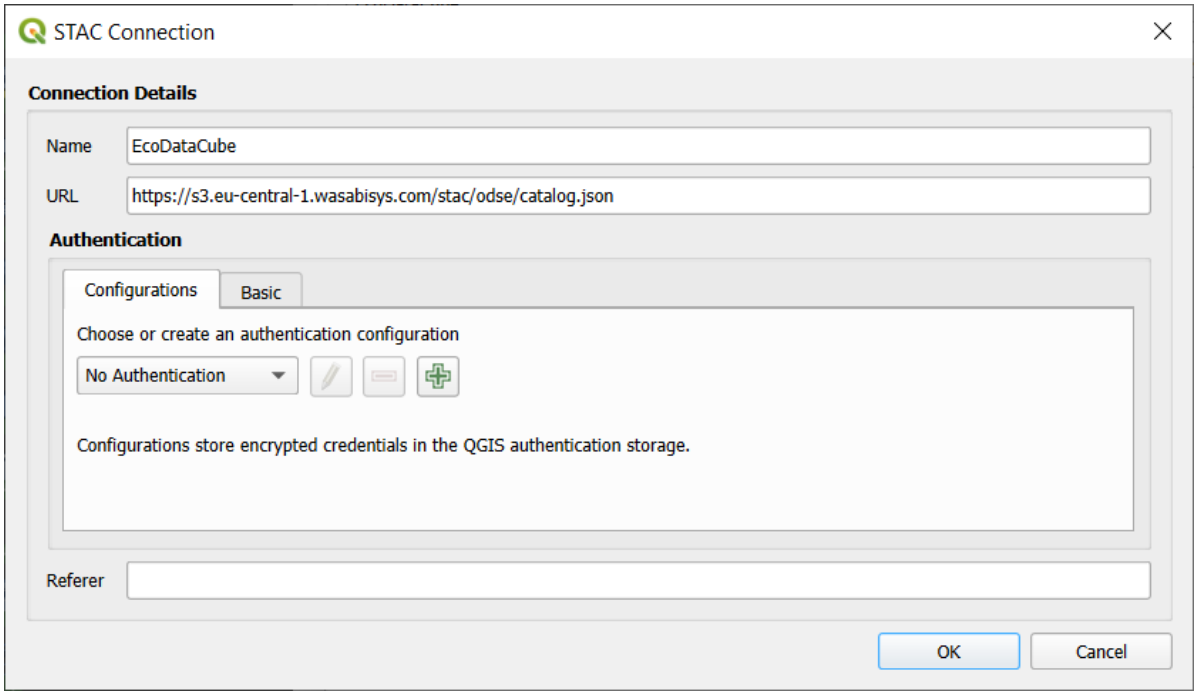
- **Browser Panel:** In the *Browser panel*, right-click on the *STAC* entry and select *New STAC Connection…*. In the dialog that appears, enter a *Name* for the connection, the *URL* of the STAC catalog and optionally fill in *Authentication* credentials and a *Referer*. Then click *OK*.

  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:

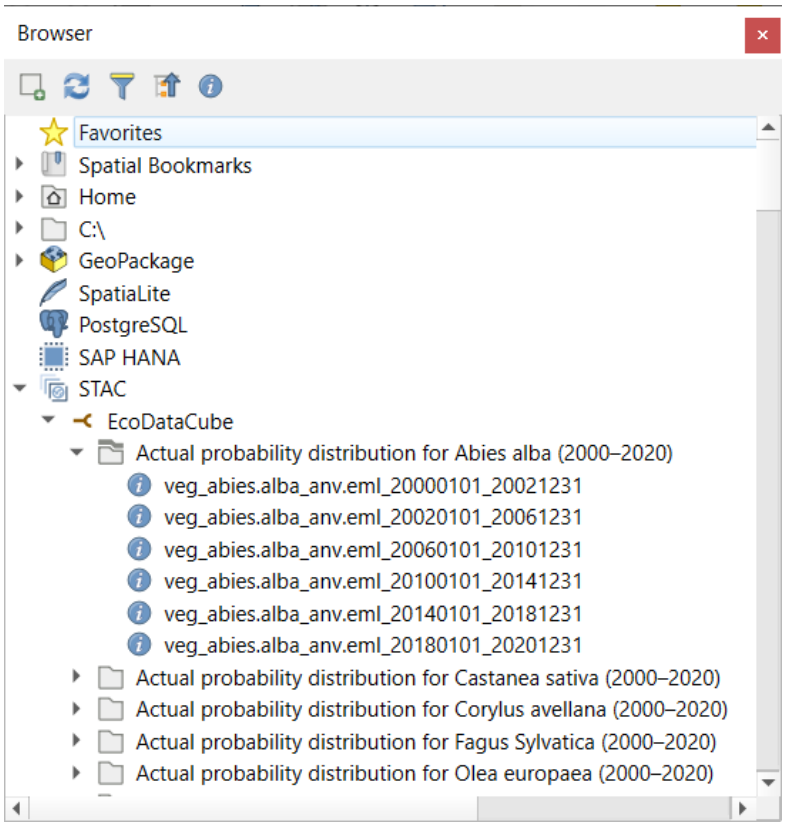
  - *Edit* the STAC connection settings
  - *Remove* the STAC connection



Слика19.12: 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.



Слика19.13: STAC connection expanded in the Browser, showing Collections and Items

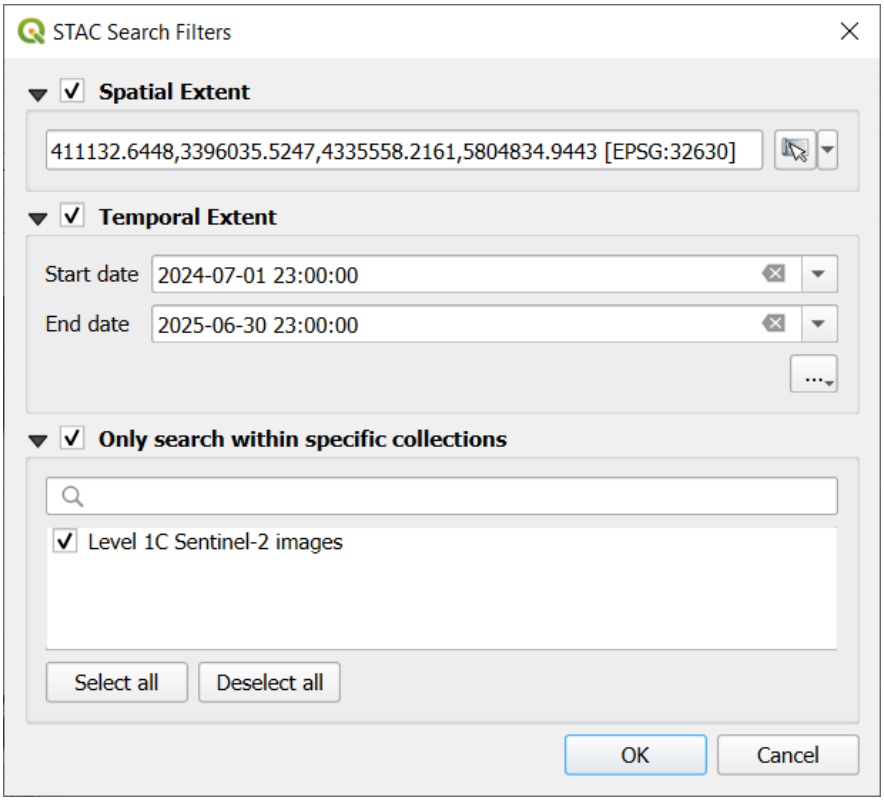
Right-click any STAC *Item* and choose *Details…* to view its metadata. The details panel shows the item's JSON content and a map of its coverage. If an item's asset is a cloud-optimized format (e.g. a `.COG`), you can add it directly to the map canvas. Otherwise, *Download Assets…* to save it locally 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…*



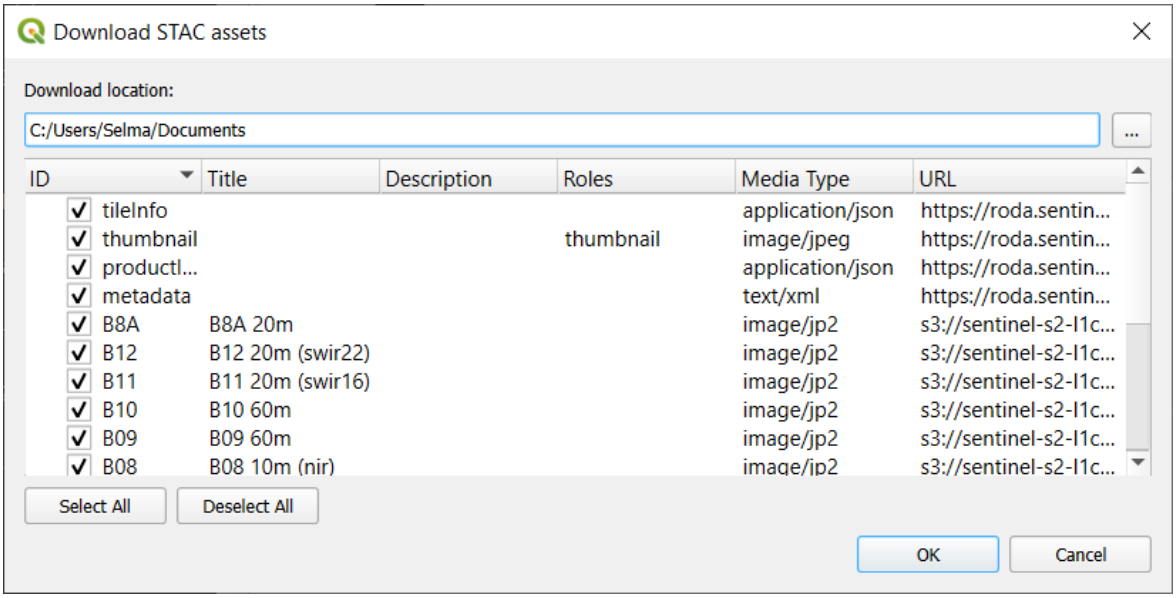
Слика19.14: Define spatial and temporal filters for a STAC catalog search

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.



Слика19.15: 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 *Using ArcGIS REST Servers* for instructions to connect to an ArcGIS REST server.

When loading layers from 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.

ГЛАВА 20

Rad sa GPS podacima

20.1 Introducing GNSS/GPS Data

20.1.1 What is GPS?

GPS, the **Global Positioning System**, is a satellite-based system that allows anyone with a GPS receiver to find their exact position anywhere in the world. GPS is used as an aid in navigation, for example in airplanes, in boats and by hikers. The GPS receiver uses the signals from the satellites to calculate its latitude, longitude and (sometimes) elevation. Most receivers also have the capability to store:

- locations (known as **waypoints**)
- sequences of locations that make up a planned **route**
- and a **track** log of the receiver's movement over time.

Waypoints, routes and tracks are the three basic feature types in GPS data. QGIS displays waypoints in point layers, while routes and tracks are displayed in linestring layers.

Белешка

QGIS supports also GNSS receivers. But we keep using the term GPS in this documentation.

20.1.2 Transferring or loading GPS data

Loading a GPX file

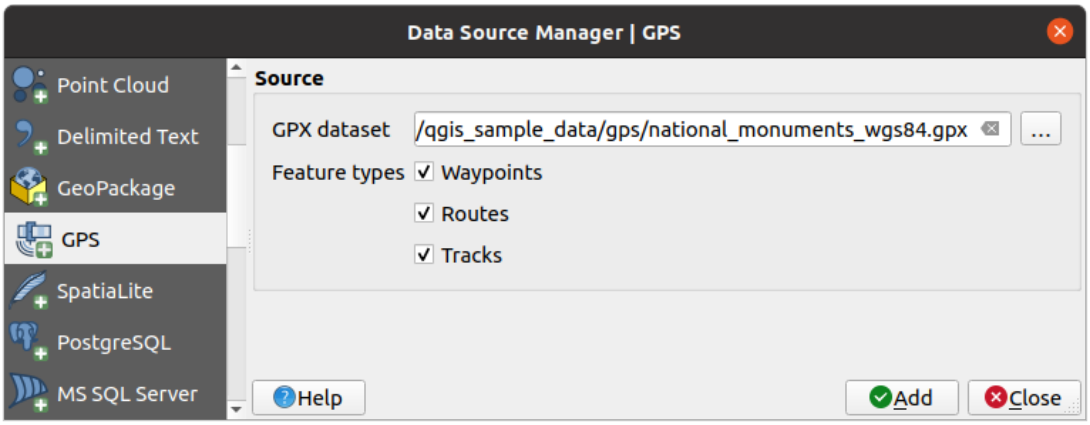
There are dozens of different file formats for storing GPS data. The format that QGIS uses is called GPX (GPS eXchange format), which is a standard interchange format that can contain any number of waypoints, routes and tracks in the same file.

To load a `GPX` file:

1. Open the *GPS* tab in the *Data Source Manager* dialog, i.e.:
   - Click the  Open Data Source Manager button on the toolbar (or press `Ctrl+L`) and enable the target tab

- Or select *Layer* ► *Add Layer* ► *Add GPX Layer…*

2. Use the … Browse button next to the *GPX dataset* option to select the GPX file
3. Use the check boxes to select the *Feature types* you want to load from the file. Each feature type (*Waypoints*, *Tracks* or *Routes*) will be loaded in a separate layer.

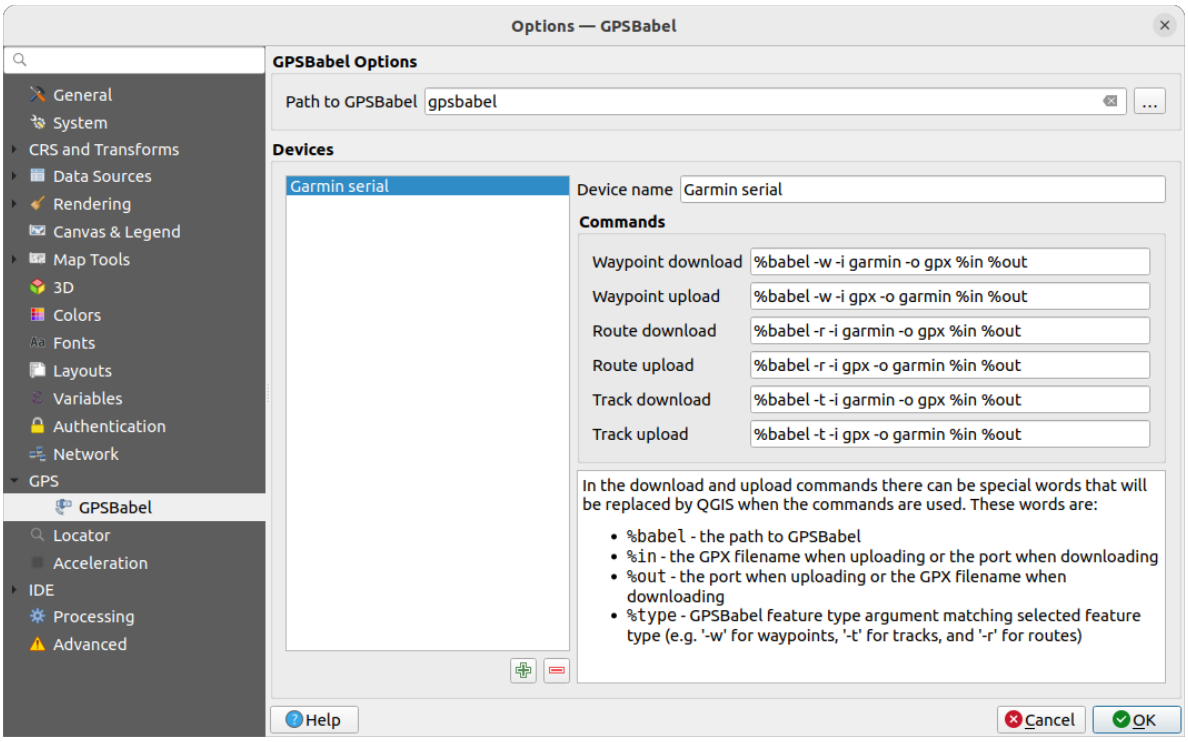


Слика20.1: Loading GPS Data dialog

Loading to or from a device

There are lots of different types of GPS devices and formats. Since QGIS uses GPX files, you need a way to convert other GPS file formats to GPX. QGIS can do that using the free program GPSBabel. GPSBabel can help you convert waypoints, tracks, and routes between popular GPS receivers such as Garmin or Magellan and mapping programs like Google Earth or Basecamp. Literally hundreds of GPS receivers and programs are supported. It can also transfer GPS data between your computer and a GPS device.

Under *Settings* ► *Options* ► *GPS* ► *GPSBabel*, QGIS allows you to define your own device type and set parameters of conversion that could later be used by the *Processing GPS algorithms*.



Слика20.2: GPS Babel settings

1. First you have to define the *Path to GPSBabel* binaries.
2. Then you may want to add your device. You can update devices list using Add new device or Remove device button.
3. For each device:
   - you provide a *Device name*
   - you configure different *Commands* QGIS will use while interacting with it, such as:
     - *Waypoint download* from the device
     - *Waypoint upload* to the device
     - *Route download* from the device
     - *Route upload* to the device
     - *Track download* from the device
     - *Track upload* to the device

   While the commands are usually GPSBabel commands, you can also use any other command line program that can create a GPX file. QGIS will replace the keywords `%type`, `%in`, and `%out` when it runs the command.

   As an example, if you create a device type with the download command `gpsbabel %type -i garmin -o gpx %in %out` and then use it to download waypoints from port `/dev/ttyS0` to the file `output.gpx`, QGIS will replace the keywords and run the command `gpsbabel -w -i garmin -o gpx /dev/ttyS0 output.gpx`.

   Read the GPSBabel manual for the command line options that may be specific to your use case.

Once you have created a new device type, it will appear in the device lists for the GPS download and upload algorithms.

Белешка

GPS units allow you to store data in different coordinate systems. When downloading a GPX file (from your GPS unit or a web site) and then loading it in QGIS, be sure that the data stored in the GPX file uses WGS 84 (latitude/longitude). QGIS expects this, and it is the official GPX specification. See GPX 1.1 Schema Documentation.

20.2 Live GPS tracking

QGIS can help you do field mapping with a GPS receiver. Such a live tracking operation is done using the *GPS Toolbar*. Some *device configuration* may be needed before connecting QGIS and the GPS receiver.

20.2.1 GPS toolbar

The *GPS Toolbar* provides the main tools to control a live tracking session. It can be activated from *View* ► *Toolbars* ► *GPS*. It follows the state of the project, GPS, and current GPS track, and enables actions only when they make sense. Once a device is detected by QGIS, you will be able to interact with it:

-  Connect to GPS
-  Disconnect from GPS
-  Recenter map on GPS location: the map will immediately recenter on the current GPS position regardless of the *automatic recentering parameter*.
-  Set destination layer for GPS digitized features: By default, QGIS will *Follow active layer* for feature digitizing, meaning that GPS digitizing tools will adapt to the layer selected in *Layers* panel and created features will be stored in it accordingly. While this may be useful in some contexts, it also implies careful interaction with other layers to avoid storing inadvertently features in undesired layers. This option allows you to explicitly indicate a layer for the data storage during the live tracking session, and you could switch depending on your needs. Exposed tools will adapt as well to the selected layer type.

  The GPS destination layer is made automatically editable when a feature is being created, and the user is notified accordingly.

- actions for feature creation:
  -  Add vertex to GPS track using GPS current location
  -  Create a new point feature at current GPS location
  -  Create a new line feature using the current GPS track
  -  Create a new polygon feature using the current GPS track
  -  Reset track
-  Show GPS information: opens the *GPS Information* panel
- a display box for quick access to some GPS information components you may need:
  -  Show location
  -  Show altitude (Geoid)
  -  Show altitude (WGS 84 ellipsoid)
  -  Show ground speed

  - Show bearing
  - Show total track length
  - Show distance from start of track

- The Settings button opens a drop-down menu with common settings which are expected to be modified mid-session:
  - *Show location marker*
  - *Show bearing line*
  - *Rotate map to match GPS direction*
  - Options to control map recentering:
    * *Always recenter map* when the GPS position is offset from the map canvas center by a certain distance (as a ratio of the map canvas extent)
    * *Recenter map when leaving extent*
    * *Never recenter*
  - *Automatically add track vertices* whenever new locations are received from the GPS device.
  - *Automatically save added features*: features created from GPS locations are immediately committed to their target layers (skipping the usual layer edit buffer)
  - *Time stamp destination* to adjust the field for time fix storage
  - *Log to GeoPackage/SpatiaLite…*: When activated, the user will be prompted to select an existing GeoPackage/SpatiaLite file or enter a new file name. A `gps_points` and `gps_tracks` table will be created in the file with a predefined structure.

    All incoming GPS messages will be logged to the `gps_points` layer, along with speed, bearing, altitude and accuracy information from the GPS.

    When the GPS is disconnected (or QGIS closed), the entire recorded GPS track will be added to the `gps_tracks` table (along with some calculated information like track length, start and end times).
  - *Log NMEA sentences…*: enables logging all raw NMEA strings from the device to a text file
  - *GPS settings…* to access the GPS *global options* dialog

Савет

Live status bar information on position

When a GPS device is connected and the user moves the cursor over the map canvas, a live status bar message displays the distance and bearing from the cursor to the GPS position. Project distance and bearing settings are respected in this display. On a touch screen device, use a tap-and-hold event to trigger the live status bar message.

20.2.2 GPS Information Panel

For full monitoring of a live GPS tracking in QGIS, you may need to enable the *GPS Information Panel* (*View* ► *Panels* ► or press `Ctrl+0`).

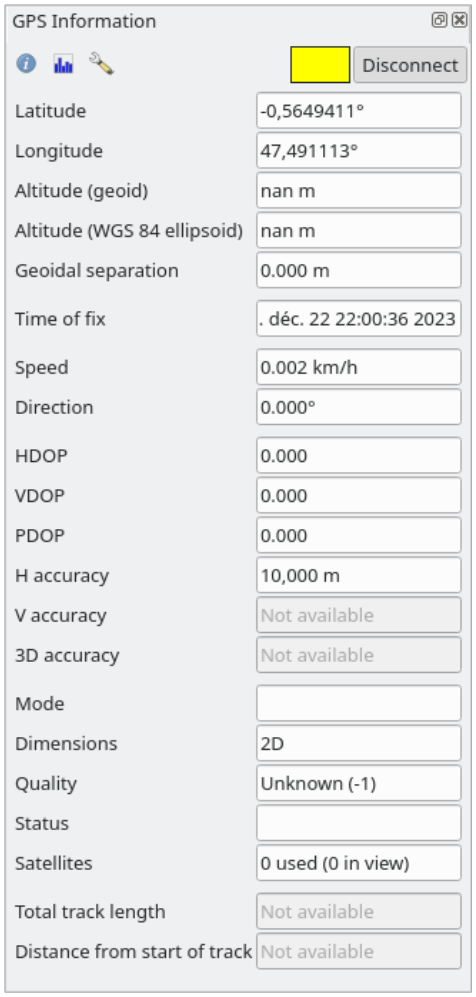
In the top right corner of the *GPS Information Panel*, you press *Connect* to initiate connection between QGIS and a plugged-in GPS receiver, or *Disconnect* them.

In the top left of the panel, following buttons are accessible:

- Position: live details on the GPS position and sensors
- Signal: signal strength of satellite connections
- Settings: drop-down menu to *live tracking options* you may need to modify during a session

Position and additional attributes

In the Position tab, if the GPS is receiving signals from satellites, you will see your position in latitude, longitude and altitude together with additional attributes.



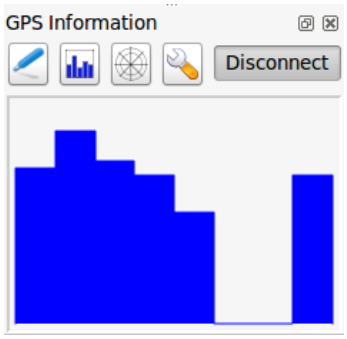
Слика20.3: GPS tracking position and additional attributes

- *Latitude*
- *Longitude*
- *Altitude (Geoid)*: Altitude/elevation above or below the mean sea level
- *Altitude (WGS 84 ellipsoid)*: Altitude/elevation above or below the WGS-84 Earth ellipsoid
- *Geoidal separation*: Difference between the WGS-84 Earth ellipsoid and mean sea level (geoid), – means mean sea level is below ellipsoid
- *Time of fix*
- *Speed*: Ground speed
- *Direction*: Bearing measured in degrees clockwise from true north to the direction of travel

- *HDOP*: Horizontal dilution of precision
- *VDOP*: Vertical dilution of precision
- *PDOP*: Dilution of precision
- *H accuracy*: Horizontal accuracy in meters
- *V accuracy*: Vertical accuracy in meters
- *3D accuracy*: 3D Root Mean Square (RMS) in meters
- *Mode*: GPS receiver configuration 2D/3D mode, can be `automatic` or `manual`
- *Dimensions*: Position fix dimension, can be `2D`, `3D` or `No fix`
- *Quality*: Positioning quality indicator
- *Status*: Position fix status, can be `Valid` or `Invalid`
- *Satellites*: Count of satellites used in obtaining the fix
- *Total track length*: Total distance of current GPS track
- *Distance from start of track*: Direct distance from first vertex in current GPS track to latest vertex

Signal

With  Signal tab, you can see the signal strength of the satellites you are receiving signals from.



Слика20.4: GPS tracking signal strength

20.2.3 Connect to a Bluetooth GPS for live tracking

With QGIS you can connect a Bluetooth GPS for field data collection. To perform this task you need a GPS Bluetooth device and a Bluetooth receiver on your computer.

At first you must let your GPS device be recognized and paired to the computer. Turn on the GPS, go to the Bluetooth icon on your notification area and search for a New Device.

On the right side of the Device selection mask make sure that all devices are selected so your GPS unit will probably appear among those available. In the next step a serial connection service should be available, select it and click on *Configure* button.

Remember the number of the COM port assigned to the GPS connection as resulting by the Bluetooth properties.

After the GPS has been recognized, make the pairing for the connection. Usually the authorization code is `0000`.

Now open *GPS information* panel and switch to GPS options screen. Select the COM port assigned to the GPS connection and click the *Connect*. After a while a cursor indicating your position should appear.

If QGIS can't receive GPS data, then you should restart your GPS device, wait 5-10 seconds then try to connect again. Usually this solution work. If you receive again a connection error make sure you don't have another Bluetooth receiver near you, paired with the same GPS unit.

20.2.4 Examples of GPS devices connection in QGIS

Using GPSMAP 60cs

MS Windows

Easiest way to make it work is to use a middleware (freeware, not open) called GPSGate.

Launch the program, make it scan for GPS devices (works for both USB and BT ones) and then in QGIS:

1. In *Settings* ► *Options* ► *GPS*, detect the plugged-in device. You can use the  *Autodetect* mode.
2. Press *Connect* in the *GPS Information* panel

Ubuntu/Mint GNU/Linux

As for Windows the easiest way is to use a server in the middle, in this case GPSD, so

1. Install the program

   ```
   sudo apt install gpsd
   ```

2. Then load the `garmin_gps` kernel module

   ```
   sudo modprobe garmin_gps
   ```

3. And then connect the unit and check with `dmesg` the actual device being used by the unit, for example `/dev/ttyUSB0`.
4. Now you can launch gpsd

   ```
   gpsd /dev/ttyUSB0
   ```

5. And finally connect with the QGIS live tracking tool.

Using BTGP-38KM datalogger (only Bluetooth)

Using GPSD (under Linux) or GPSGate (under Windows) is effortless.

Using BlueMax GPS-4044 datalogger (both BT and USB)

MS Windows

The live tracking works for both USB and BT modes, by using GPSGate or even without it, just use the  *Autodetect* mode, or point the tool to the right port.

Ubuntu/Mint GNU/Linux

For USB

The live tracking works both with GPSD

```
gpsd /dev/ttyACM3
```

or without it, by connecting the QGIS live tracking tool directly to the device (for example `/dev/ttyACM3`).

For Bluetooth

The live tracking works both with GPSD

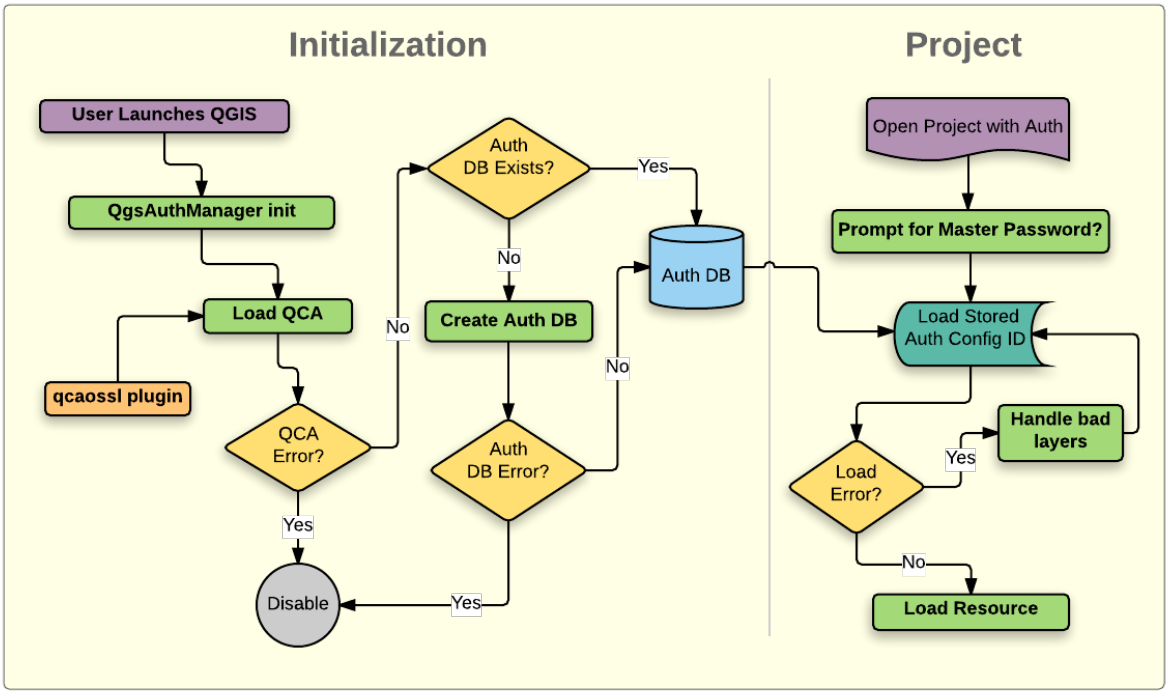
```
gpsd /dev/rfcomm0
```

or without it, by connecting the QGIS live tracking tool directly to the device (for example `/dev/rfcomm0`).

ГЛАВА 21

Sistem za autentifikaciju

21.1 Authentication System Overview



Слика21.1: Anatomy of authentication system

21.1.1 Authentication database

The authentication system stores authentication configurations in an SQLite database file located, by default, at `<profile directory>/qgis-auth.db`.

This authentication database can be moved between QGIS installations without affecting other current QGIS user preferences, as it is completely separate from normal QGIS settings. A configuration ID (a random 7-character alphanumeric string) is generated when initially storing a configuration to the database. This represents the configuration, thereby allowing the ID to be stored in plain text application components, (such as project, plugin, or settings files) without disclosure of its associated credentials.

Белешка

The parent directory of the *qgis-auth.db* can be set using the following environment variable, `QGIS_AUTH_DB_DIR_PATH`, or set on the command line during launch with the `--authdbdirectory` option.

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 supported 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

Белешка

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.

Упозорење

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.

Упозорење

Any database not based on SQLite is considered to be read-only (this can be changed by Python plugins if necessary).

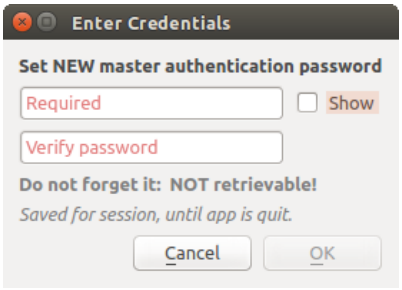
This is an advanced feature, designed to allow one or more custom authentication databases or even custom Python implementations of credentials storages to be used by QGIS.

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 Master password

To store or access sensitive information within the database, a user must define a *master password*. A new master password is requested and verified when initially storing any encrypted data to the database. When sensitive information is accessed, the user is prompted for the master password. The password is then cached for the remainder of the session (until application is quit), unless the user manually chooses an action to clear its cached value. Some instances of using the authentication system do not require input of the master password, such as when selecting an existing authentication configuration, or applying a configuration to a server configuration (such as when adding a WMS layer).

You can choose to save the password in the `Wallet/Keyring` of your computer.



Слика21.2: Input new master password

Белешка

A path to a file containing the master password can be set using the following environment variable, `QGIS_AUTH_PASSWORD_FILE`.

Managing the master password

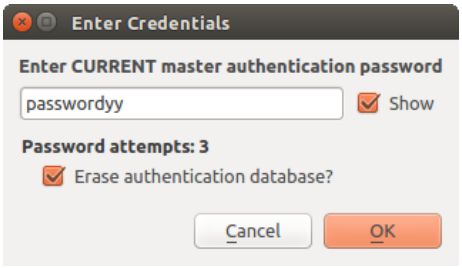
Once set, the master password can be reset; the current master password will be needed prior to resetting. During this process, there is an option to generate a complete backup of the current database.



Слика21.3: Resetting master password

If the user forgets the master password, there is no way to retrieve or override it. There is also no means of retrieving encrypted information without knowing the master password.

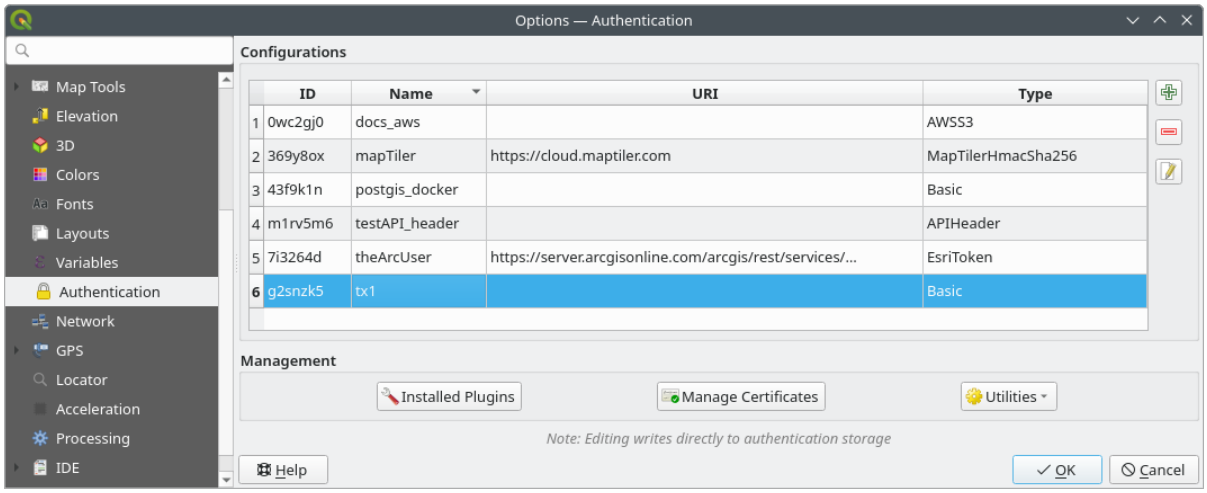
If a user inputs their existing password incorrectly three times, the dialog will offer to erase the database.



Слика21.4: Password prompt after three invalid attempts

21.1.4 Authentication Configurations

You can manage authentication configurations from *Configurations* in the *Authentication* tab of the QGIS Options dialog (*Settings* ► *Options*).

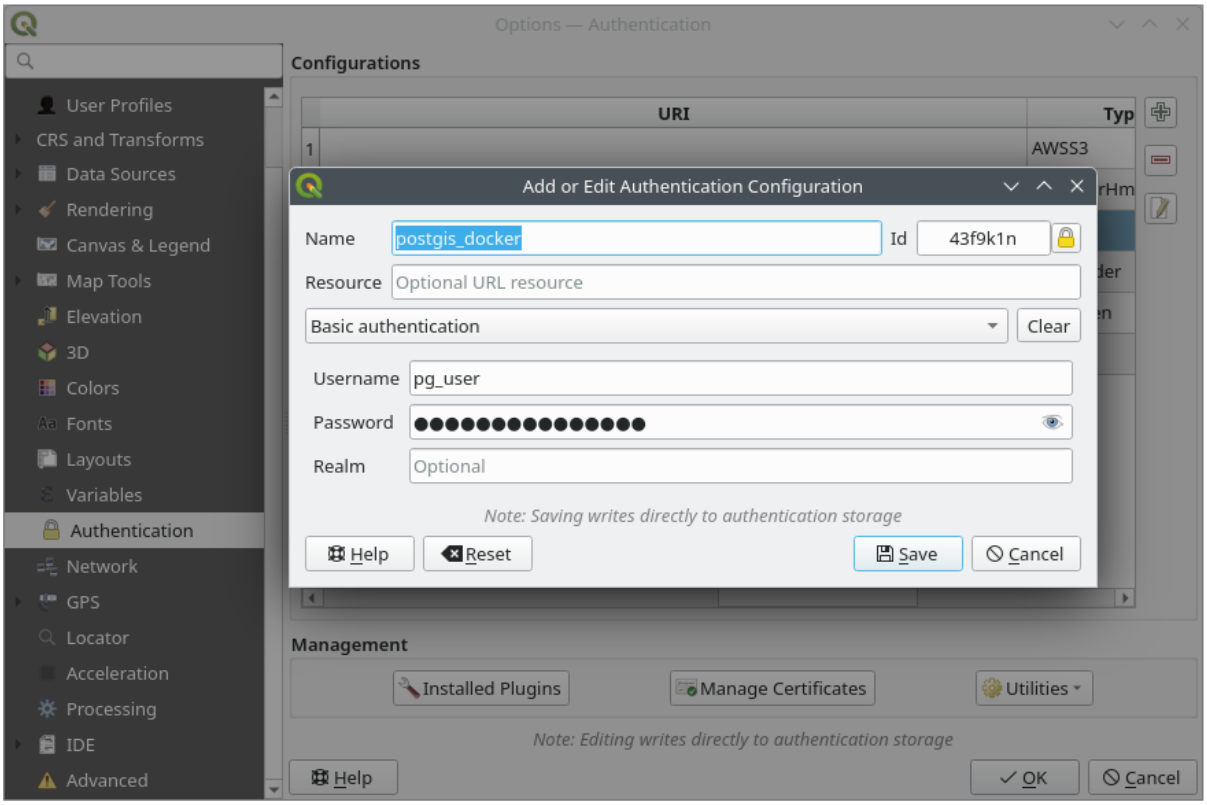


Слика21.5: Configurations editor

To create a new authentication configuration:

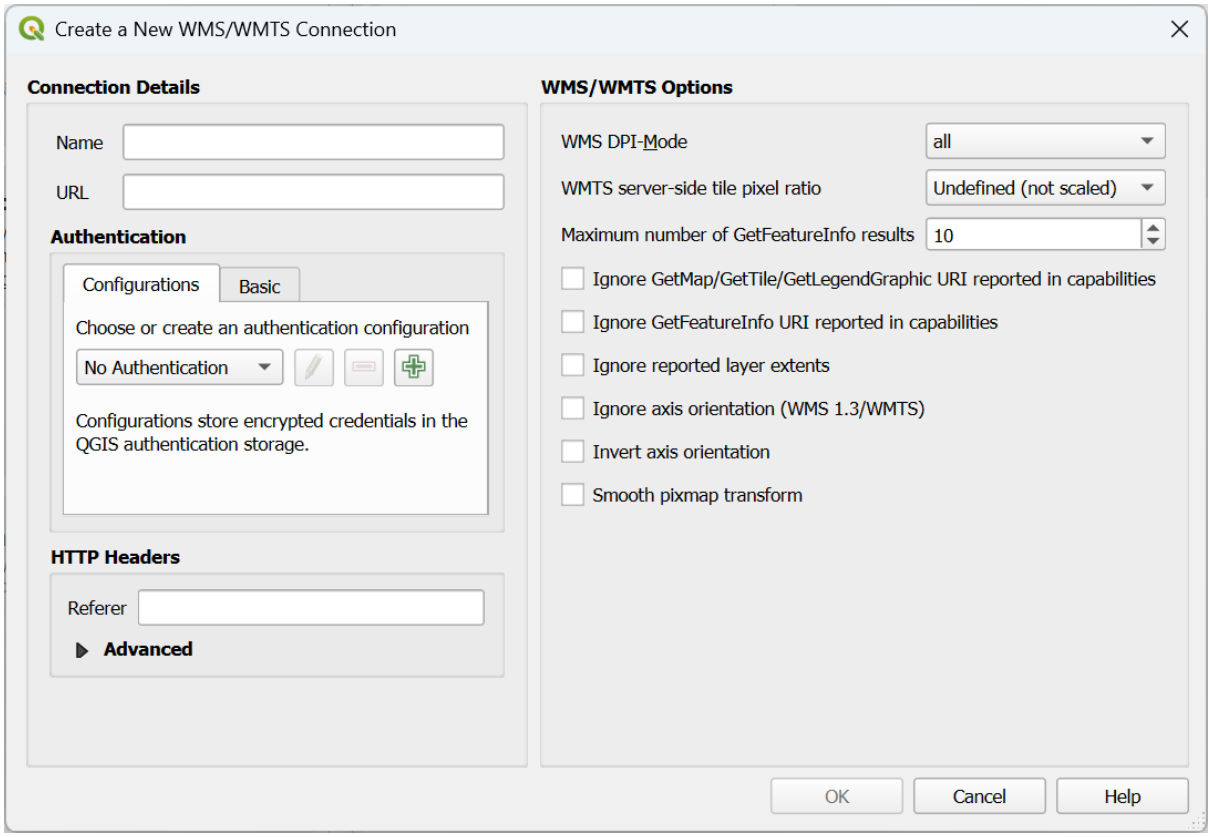
1. Press the Add a new authentication configuration
2. Provide a *Name* and optional *Resource* URL
3. A seven-characters *ID* will be assigned to the configuration for identification purpose. It can be customized after you pressed the Unlock to edit ID button. Be aware that updating that ID may break 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.



Слика21.6: Adding config from within Configuration editor

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.

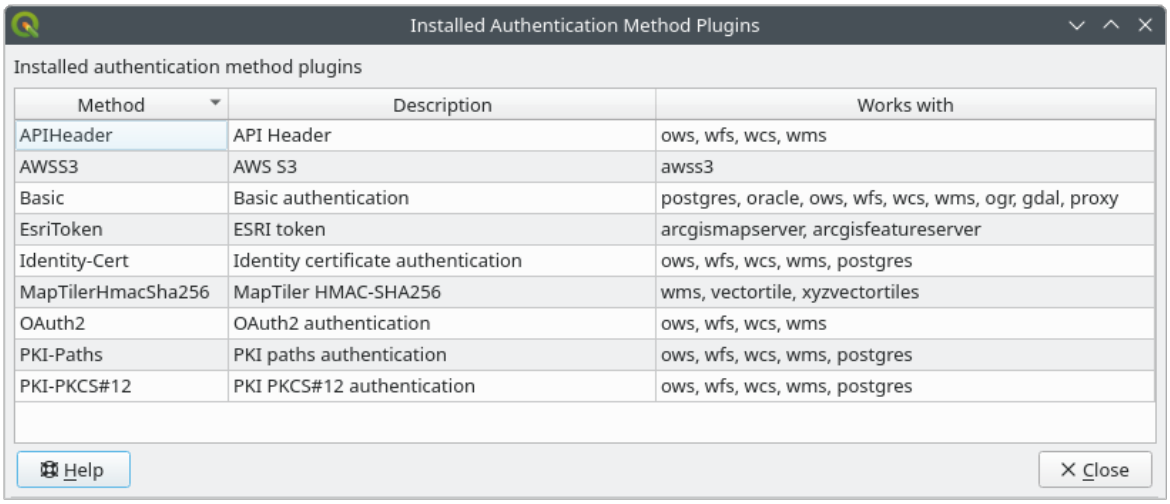


Слика21.7: WMS connection dialog showing authentication configuration widget

When creating or editing an authentication configuration, the info required is a name, an authentication method and any other info that the authentication method requires (see more about the available authentication types in *Authentication Methods*).

21.1.5 Authentication Methods

Available authentications are provided by C++ plugins much in the same way data provider plugins are supported by QGIS. The method of authentication that can be selected is relative to the access needed for the resource/provider, e.g. HTTP(S) or database, and whether there is support in both QGIS code and a plugin. As such, some authentication method plugins may not be applicable everywhere an authentication configuration selector is shown. A list of available authentication method plugins and their compatible resource/providers can be accessed going to *Settings* ► *Options* and, in the *Authentication* tab, click the *Installed Plugins* button.



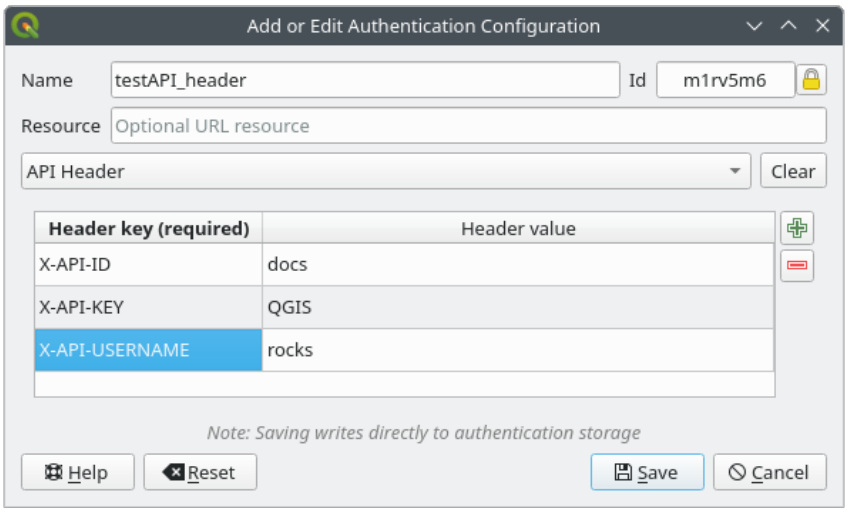
Слика21.8: Available method plugins list

Plugins can be created for new authentication methods that do not require QGIS to be recompiled. Since the support for plugins is currently C++-only, QGIS will need to be restarted for the new dropped-in plugin to become available to the user. Ensure your plugin is compiled against the same target version of QGIS if you intend to add it to an existing target install.

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.

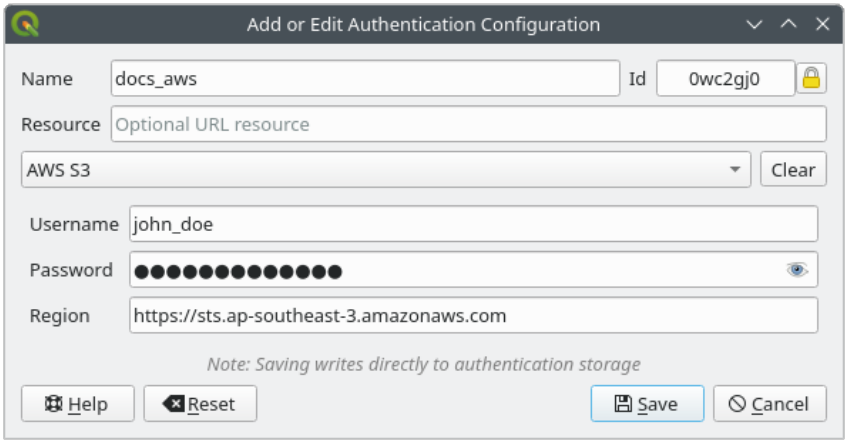
You can add one or more headers with the button; provide a *Header Name* and *Header Value* for each entry. Use to delete the selected header or the *Clear* button to remove all entries.



Слика21.9: API Header authentication configs

AWS S3 Authentication

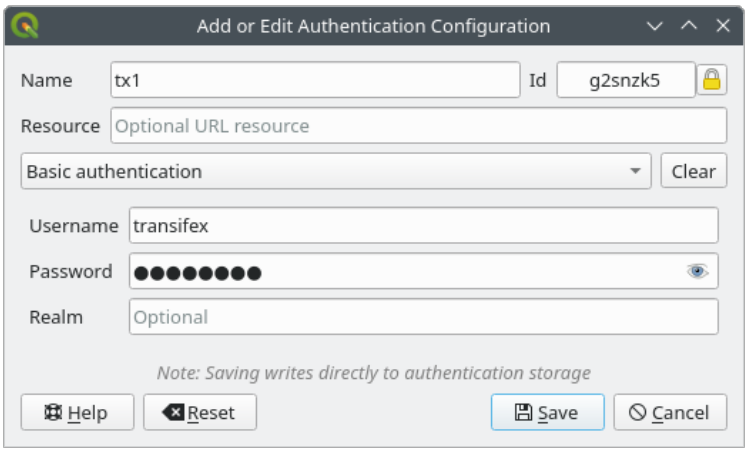
The AWS S3 authentication method allows you to connect to Amazon S3 resources by providing a *Name*, *Resource*, *Username*, *Password*, and *Region*.



Слика21.10: AWS S3 authentication configs

Basic Authentication

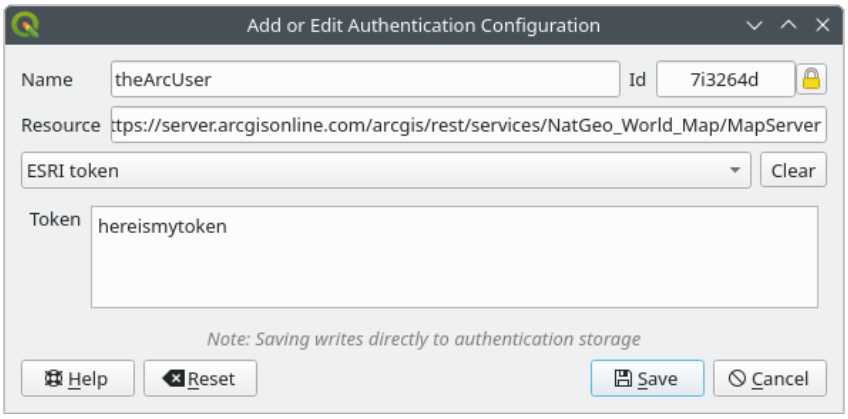
The Basic authentication method is used for services that require standard HTTP authentication with a *Username*, *Password* and *Realm*.



Слика21.11: Basic HTTP authentication configs

ESRI Token Authentication

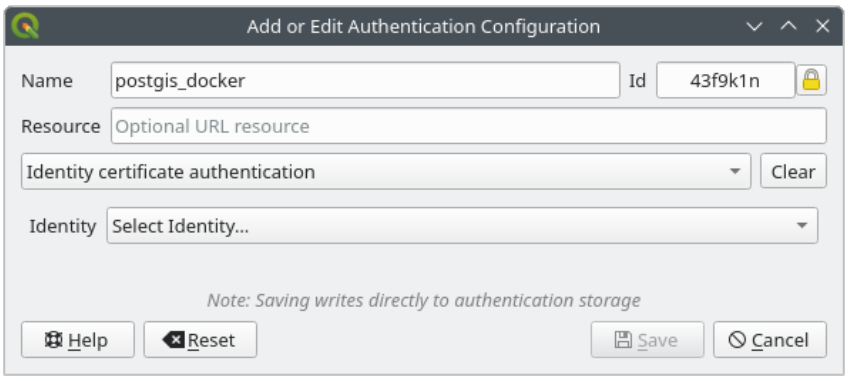
The ESRI token authentication method is used for *ArcGIS REST Servers* that require token-based authentication.



Слика21.12: ESRI Token authentication configs

Identity certificate authentication

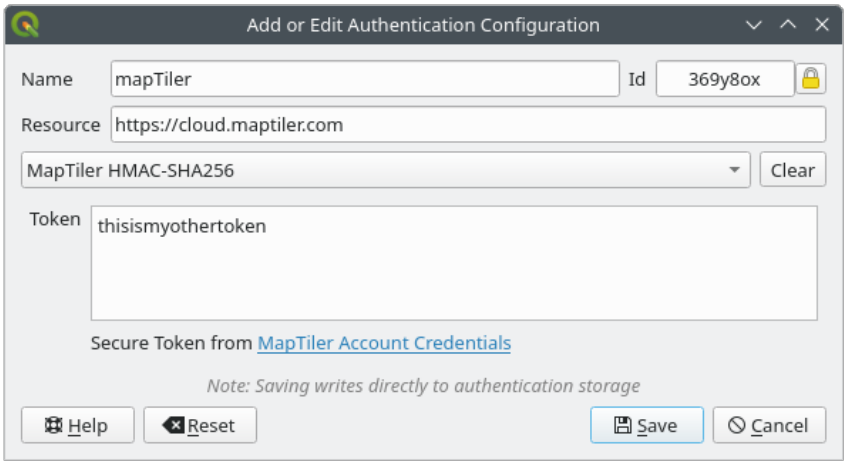
The Identity certificate authentication method allows you to connect using a client *identity certificate*.



Слика21.13: 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.



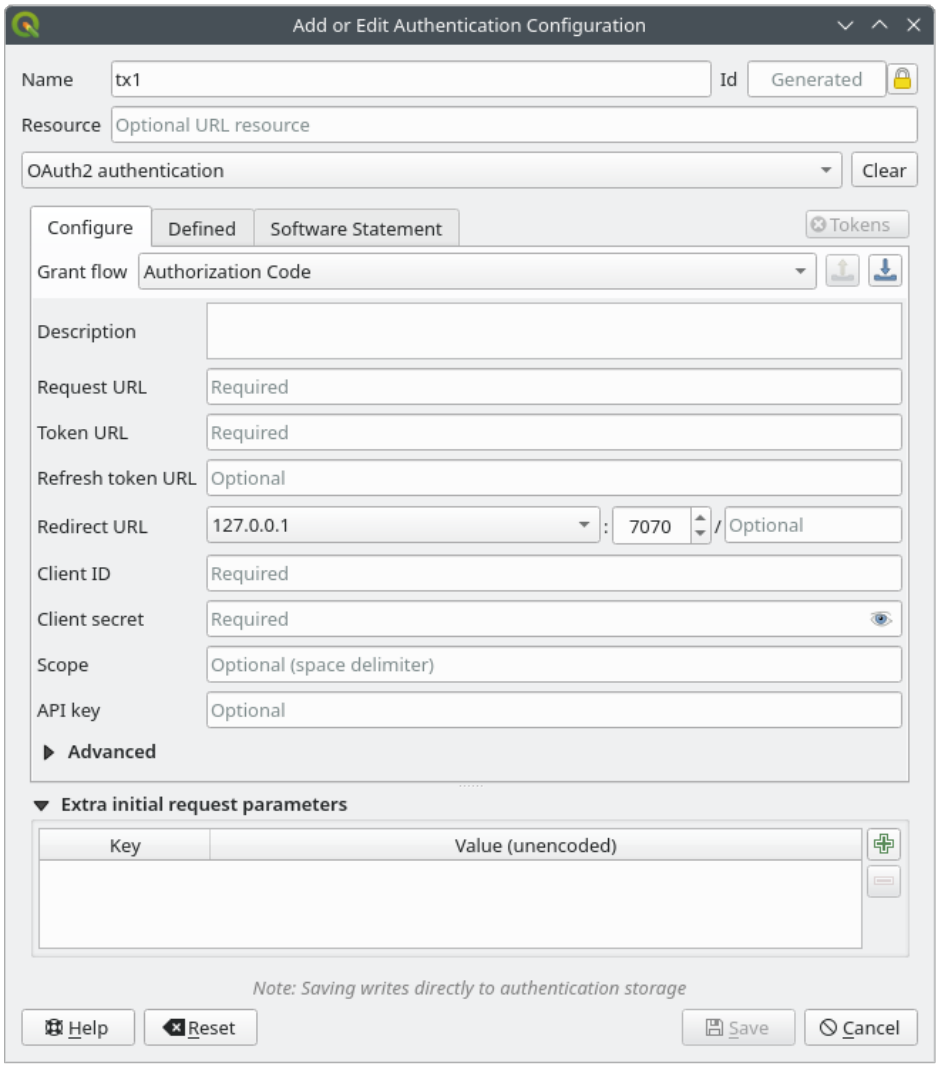
Слика21.14: MapTiler HmacSha256 authentication configs

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).

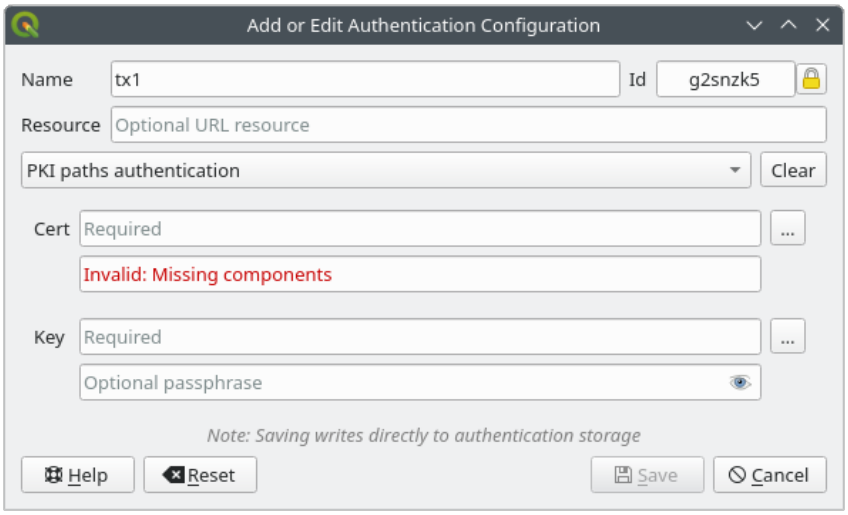
An advanced OAuth2 configuration option, *Extra token header(s)*, allows attaching additional tokens returned by the OAuth2 token endpoint as HTTP(S) request headers.



Слика21.15: OAuth2 authentication configs

PKI-Paths Authentication

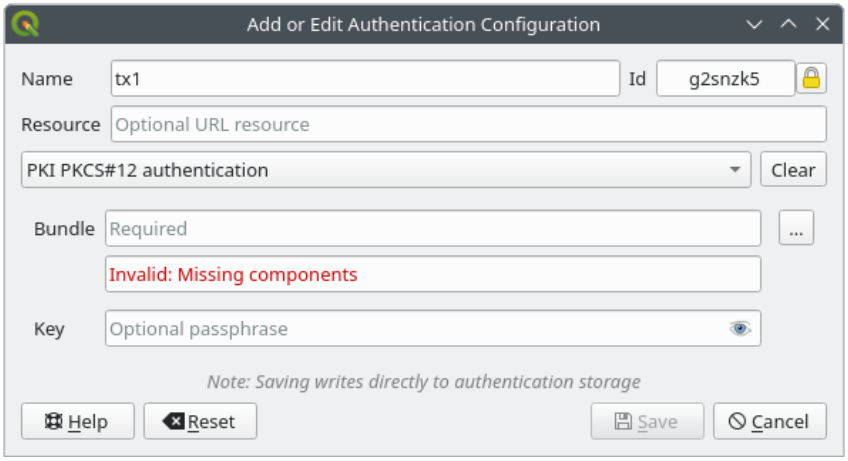
The PKI-Paths authentication method allows you to connect using separate certificate and key files stored on your system.



Слика21.16: PKI paths authentication configs

PKI-PKCS#12 Authentication

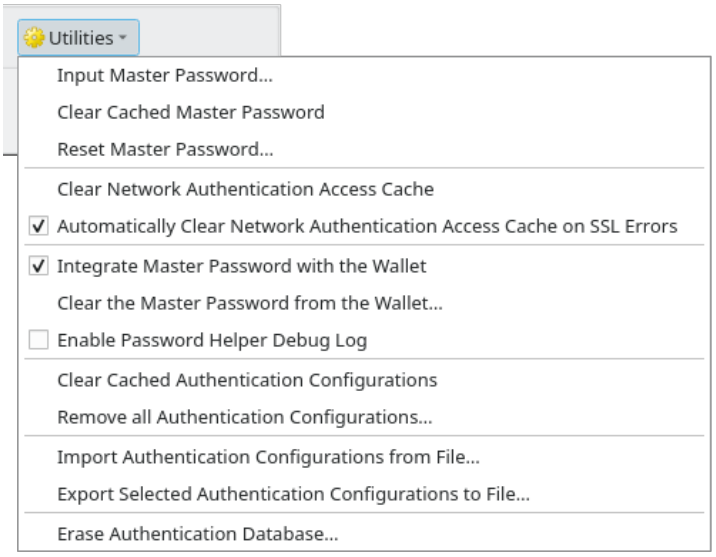
The PKI-PKCS#12 authentication method allows you to connect using a single bundle file containing both the certificate and private key, with an optional passphrase for the key.



Слика21.17: PKI PKCS#12 file paths authentication configs

21.1.6 Master Password and Auth Config Utilities

Under the Options menu (*Settings* ► *Options*) in the *Authentication* tab, there are several utility actions to manage the authentication database and configurations:



Слика21.18: Utilities menu

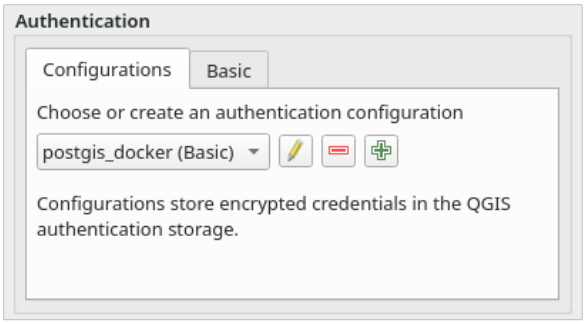
- *Input master password…*: opens the master password input dialog, independent of performing any authentication database command
- *Clear cached master password*: unsets the master password if it has been set
- *Reset master password…*: opens a dialog to change the master password (the current password 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 Using authentication configurations

Typically, an authentication configuration is selected in a configuration dialog for a network services (such as WMS). However, the selector widget can be embedded anywhere authentication is needed or in non-core functionality, like in third-party PyQGIS or 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 

to create a configuration you could use. More details at *Authentication*.



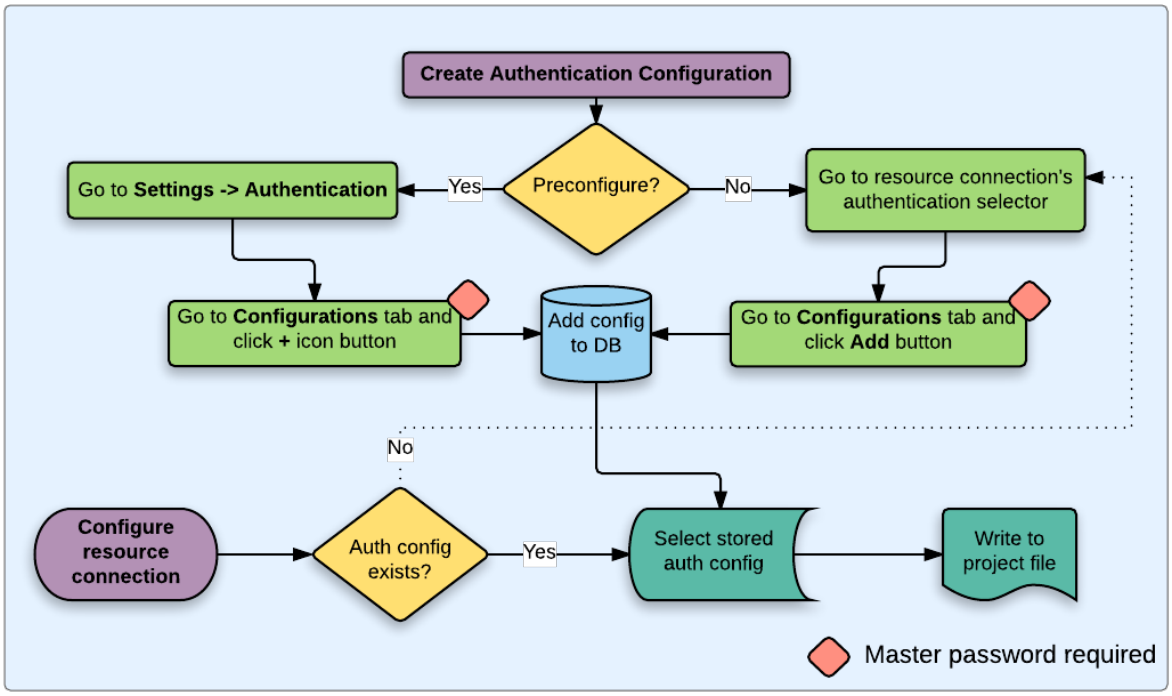
Слика21.19: Authentication configuration selector with selected config

21.1.8 Python bindings

All classes and public functions have sip bindings, except QgsAuthCrypto, since management of the master password hashing and auth database encryption should be handled by the main app, and not via Python.

Once the master password is entered, the API is open to access authentication configs in the authentication database, similar to how Firefox works. However, no wall against PyQGIS access has been defined. This may lead to issues where a user downloads/installs a malicious PyQGIS plugin or standalone app that gains access to authentication credentials.

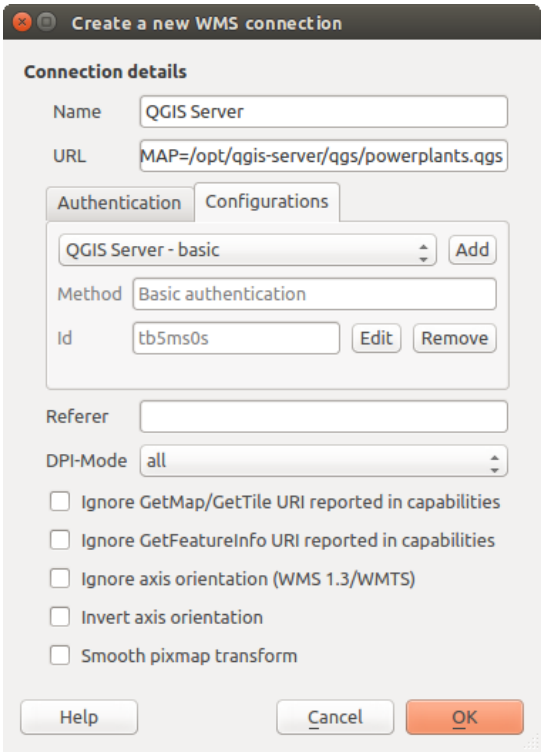
21.2 User Authentication Workflows



Слика21.20: Generic user workflow

21.2.1 HTTP(S) authentication

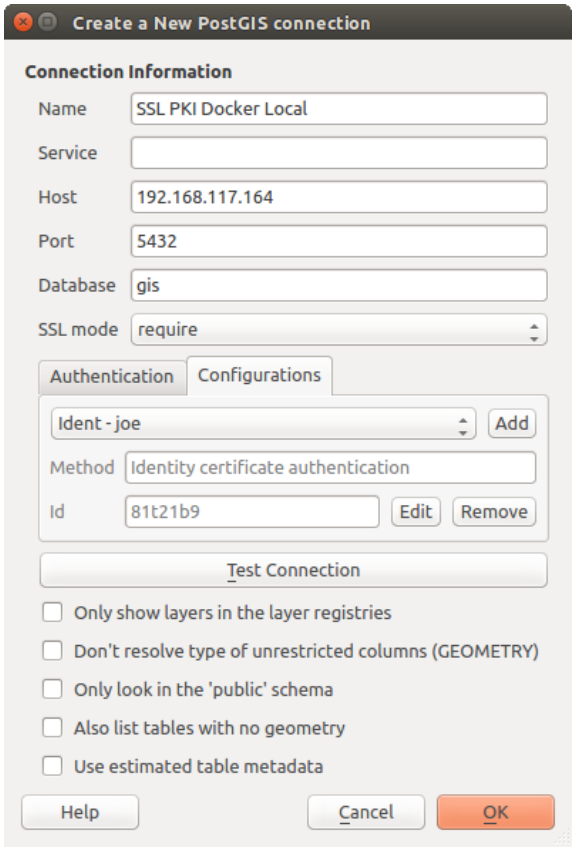
One of the most common resource connections is via HTTP(S), e.g. web mapping servers, and authentication method plugins often work for these types of connections. Method plugins have access to the HTTP request object and can manipulate both the request as well as its headers. This allows for many forms of internet-based authentication. When connecting via HTTP(S) using the standard username/password authentication method will attempt HTTP BASIC authentication upon connection.



Слика21.21: Configuring a WMS connection for HTTP BASIC

21.2.2 Database authentication

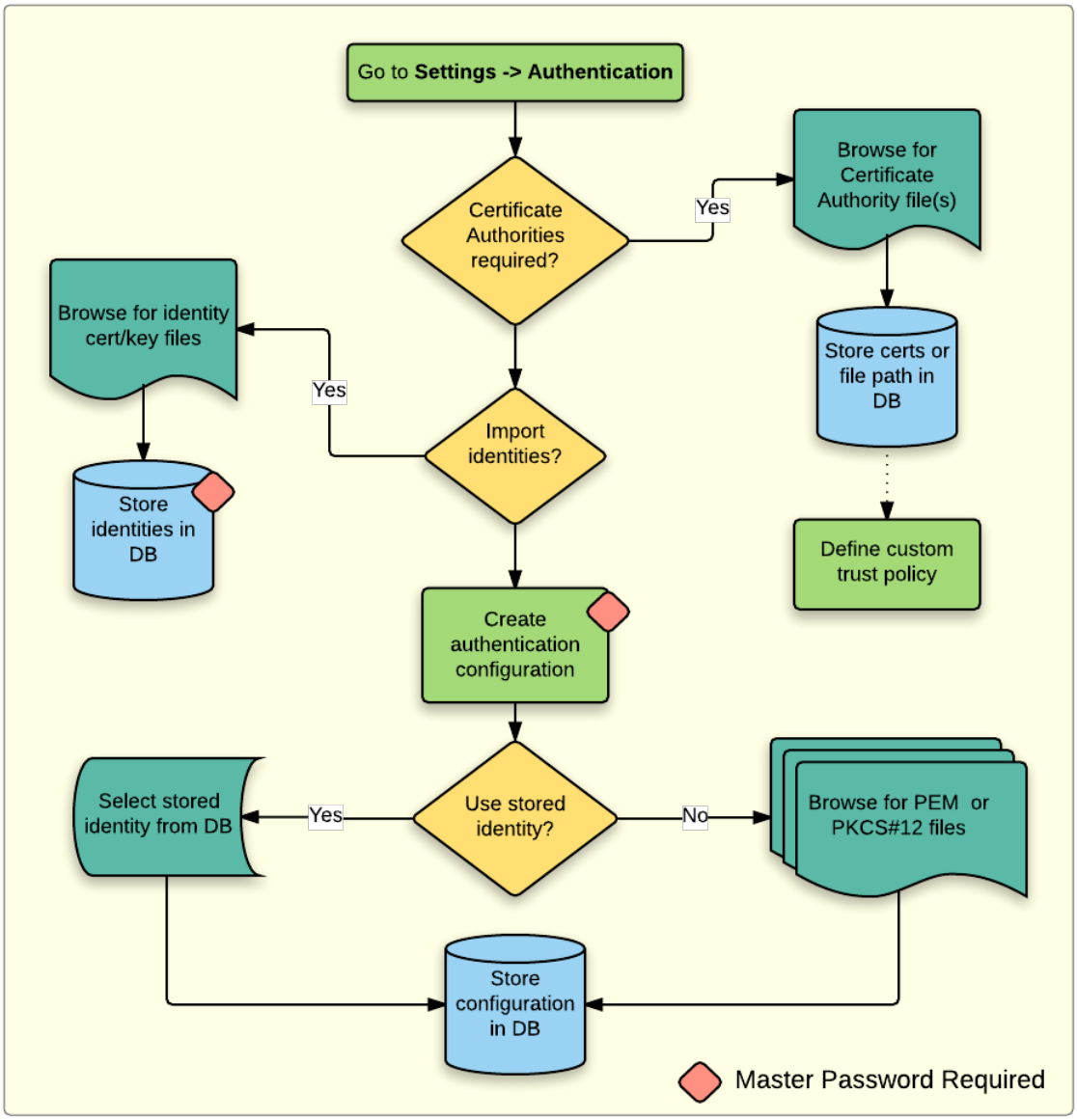
Connections to database resources are generally stored as `key=value` pairs, which will expose usernames and (optionally) passwords, if *not* using an authentication configuration. When configuring with the auth system, the `key=value` will be an abstracted representation of the credentials, e.g. `authfg=81t21b9`.



Слика21.22: Configuring a Postgres SSL-with-PKI connection

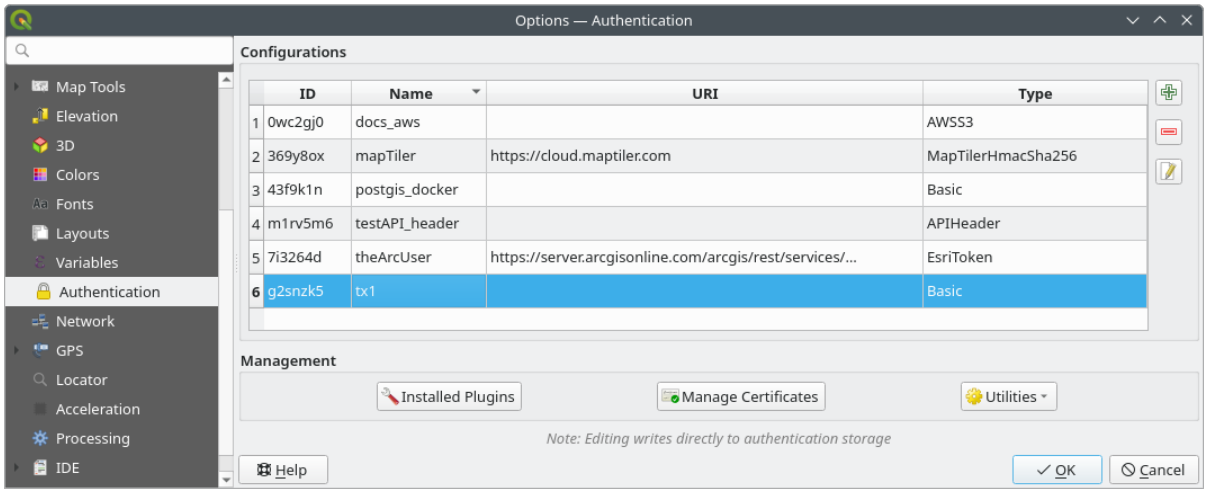
21.2.3 PKI authentication

When configuring PKI components within the authentication system, you have the option of importing components into the database or referencing component files stored on your filesystem. The latter may be useful if such components change frequently, or where the components will be replaced by a system administrator. In either instance you will need to store any passphrase needed to access private keys within the database.



Слика21.23: PKI configuration workflow

All PKI components can be managed in separate editors within the **Certificate Manager**, which can be accessed in the *Authentication* tab in QGIS *Options* dialog (*Settings* ► *Options*) by clicking the *Manage Certificates* button.



Слика21.24: Opening the Certificate Manager

In the *Certificate Manager*, there are editors for **Identities**, **Servers** and **Authorities**. Each of these are contained in their own tabs, and are described below in the order they are encountered in the workflow chart above. The tab order is relative to frequently accessed editors once you are accustomed to the workflow.

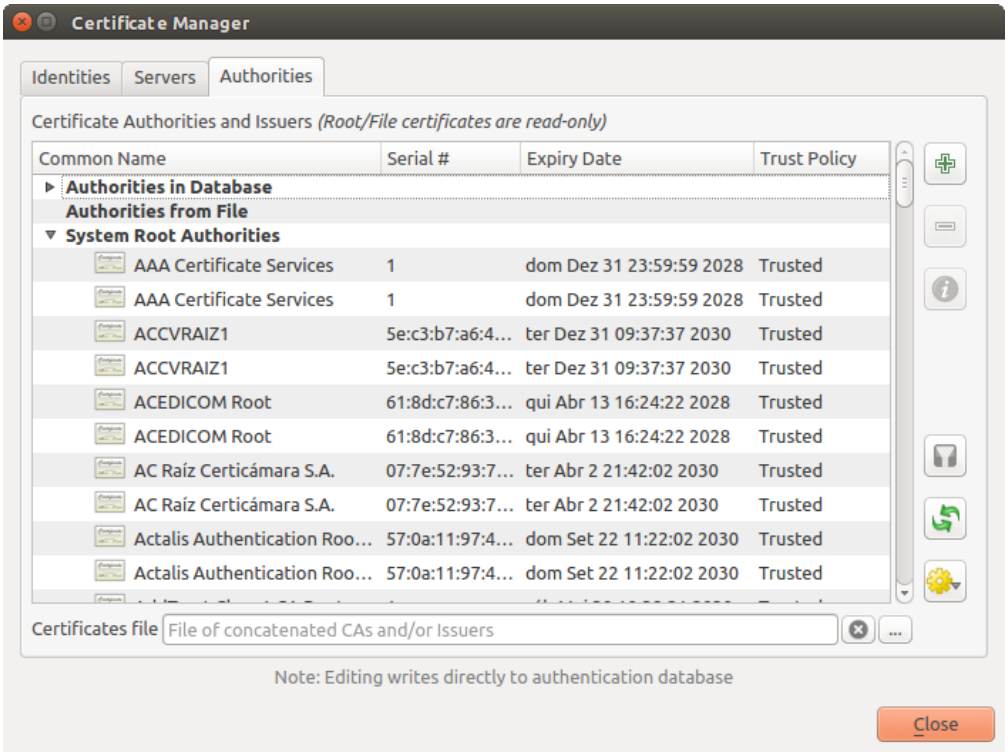
Белешка

Because all authentication system edits write immediately to the authentication database, there is no need to click the *Options* dialog *OK* button for any changes to be saved. This is unlike other settings in the Options dialog.

Authorities

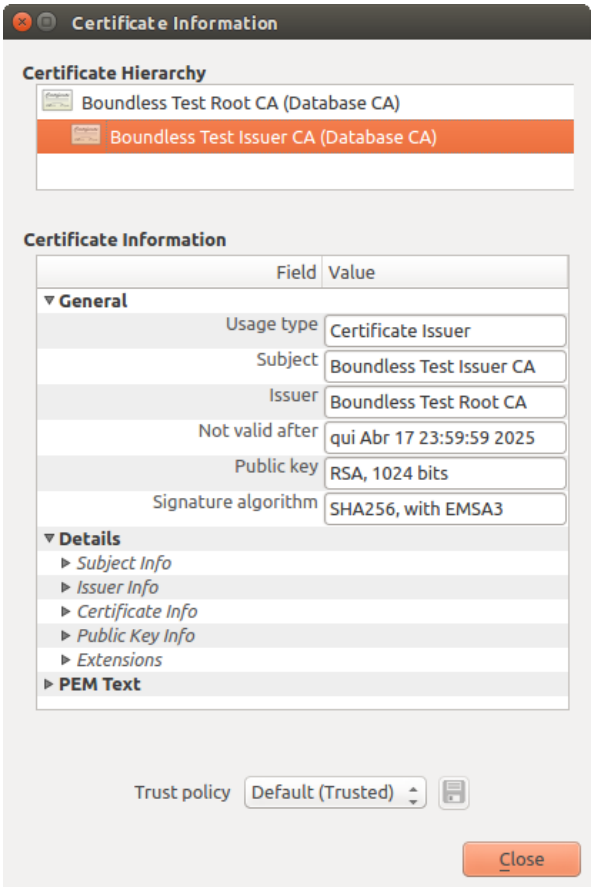
You can manage available Certificate Authorities (CAs) from the **Authorities** tab in the **Certificate manager** from the **Authentication** tab of the QGIS **Options** dialog.

As referenced in the workflow chart above, the first step is to import or reference a file of CAs. This step is optional, and may be unnecessary if your PKI trust chain originates from root CAs already installed in your operating system (OS), such as a certificate from a commercial certificate vendor. If your authenticating root CA is not in the OS's trusted root CAs, it will need to be imported or have its file system path referenced. (Contact your system administrator if unsure.)



Слика21.25: Authorities editor

By default, the root CAs from your OS are available; however, their trust settings are not inherited. You should review the certificate trust policy settings, especially if your OS root CAs have had their policies adjusted. Any certificate that is expired will be set to untrusted and will not be used in secure server connections, unless you specifically override its trust policy. To see the QGIS-discoverable trust chain for any certificate, select it and click the Show information for certificate.



Слика21.26: Certificate info dialog

You can edit the *Trust policy*  for any selected certificate within the chain. Any change in trust policy to a selected certificate will not be saved to the database unless the  Save certificate trust policy change to database button is clicked *per* selected certification. Closing the dialog will **not** apply the policy changes.

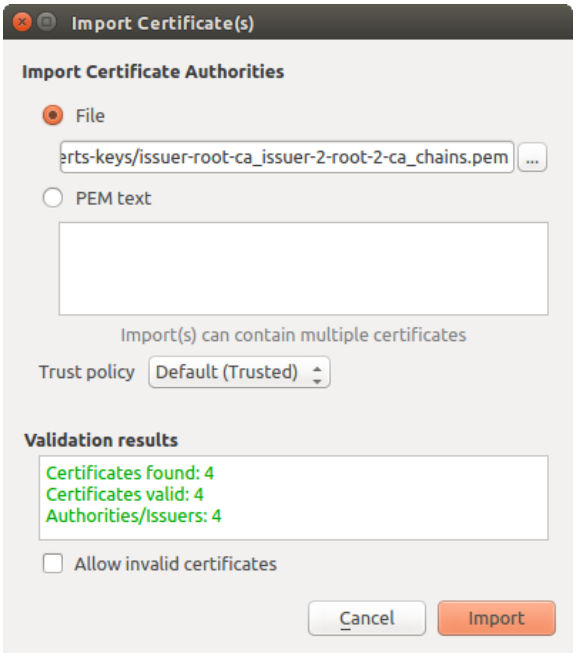
You can review the filtered CAs, both intermediate and root certificates, that will be trusted for secure connections or change the default trust policy by clicking the  **Options** button.

Упозорење

Changing the default trust policy may result in problems with secure connections.

You can import CAs or save a file system path from a file that contains multiple CAs, or import individual CAs. The standard PEM format for files that contain multiple CA chain certifications has the root cert at the bottom of the file and all subsequently signed child certificates above, towards the beginning of the file.

The CA certificate import dialog will find all CA certificates within the file, regardless of order, and also offers the option to import certificates that are considered invalid (in case you want to override their trust policy). You can override the trust policy upon import, or do so later within the **Authorities** editor.



Слика21.27: Import certificates dialog

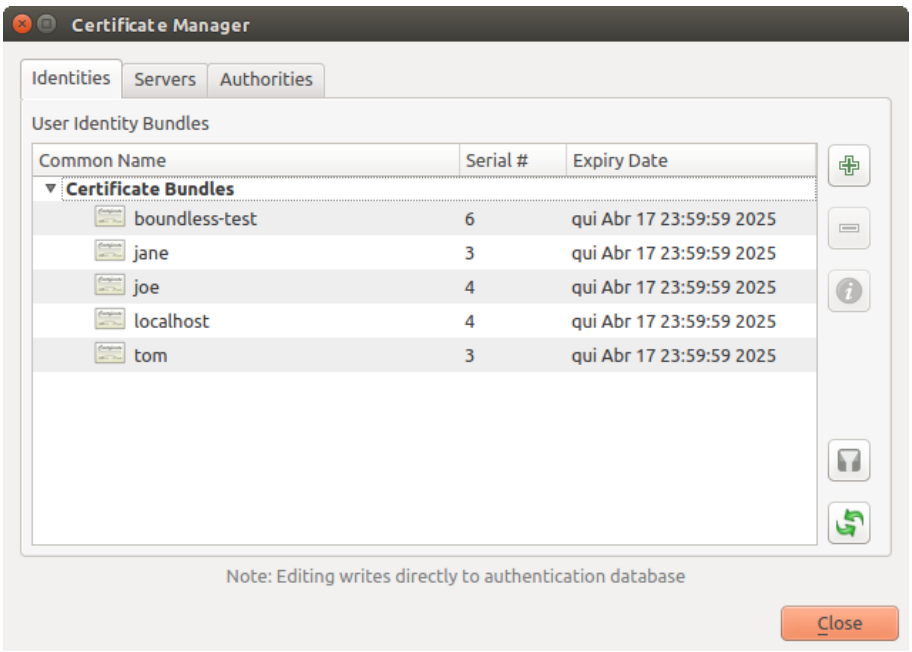
Белешка

If you are pasting certificate information into the *PEM text* field, note that encrypted certificates are not supported.

Identities

You can manage available client identity bundles from the *Identities* tab in the *Certificate manager* from the **Authentication** tab of the QGIS **Options** dialog. An identity is what authenticates you against a PKI-enabled service and usually consists of a client certificate and private key, either as separate files or combined into a single „bundled" file. The bundle or private key is often passphrase-protected.

Once you have any Certificate Authorities (CAs) imported you can optionally import any identity bundles into the authentication database. If you do not wish to store the identities, you can reference their component file system paths within an individual authentication configuration.

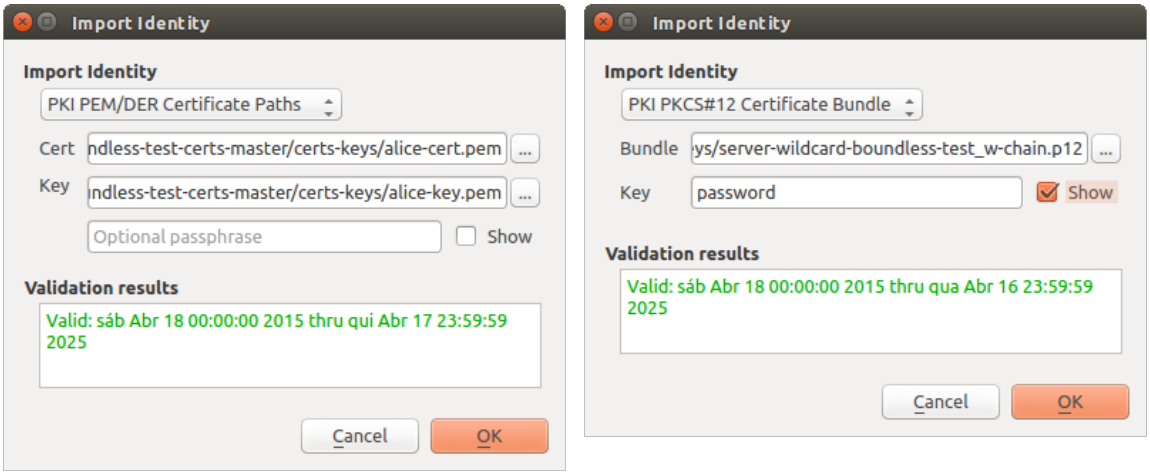


Слика21.28: Identities editor

When importing an identity bundle, it can be passphrase-protected or unprotected, and can contain CA certificates forming a trust chain. Trust chain certifications will not be imported here; they can be added separately under the *Authorities* tab.

Upon import the bundle's certificate and private key will be stored in the database, with the key's storage encrypted using the QGIS master password. Subsequent usage of the stored bundle from the database will only require input of the master password.

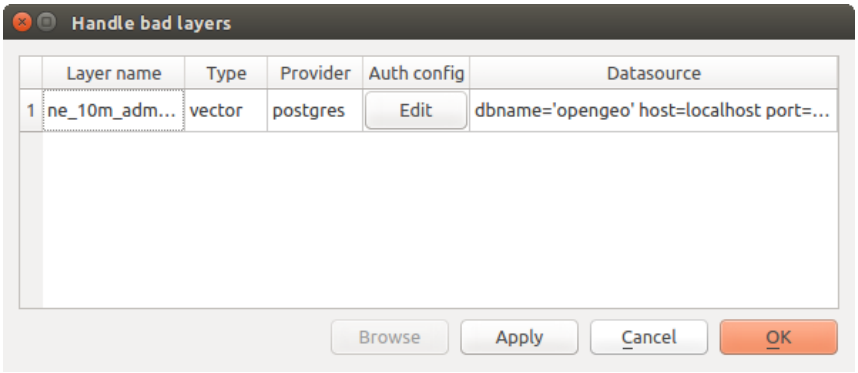
Personal identity bundles consisting of PEM/DER (.pem/.der) and PKCS#12 (.p12/.pfx) components are supported. If a key or bundle is passphrase-protected, the password will be required to validate the component prior to import. Likewise, if the client certificate in the bundle is invalid (for example, its effective date has not yet started or has elapsed) the bundle can not be imported.



Слика21.29: PEM/DER and PKCS#12 identities import

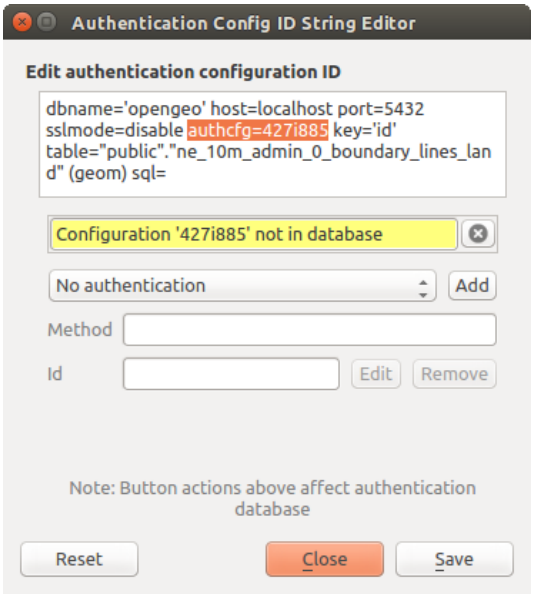
21.2.4 Handling bad layers

Occasionally, the authentication configuration ID that is saved with a project file is no longer valid, possibly because the current authentication database is different than when the project was last saved, or due to a credentials mismatch. In such cases the *Handle bad layers* dialog will be presented upon QGIS launch.



Слика21.30: Handle bad layers with authentication

If a data source is found to have an authentication configuration ID associated with it, you will be able to edit it. Doing so will automatically edit the data source string, much in the same way as opening the project file in a text editor and editing the string.

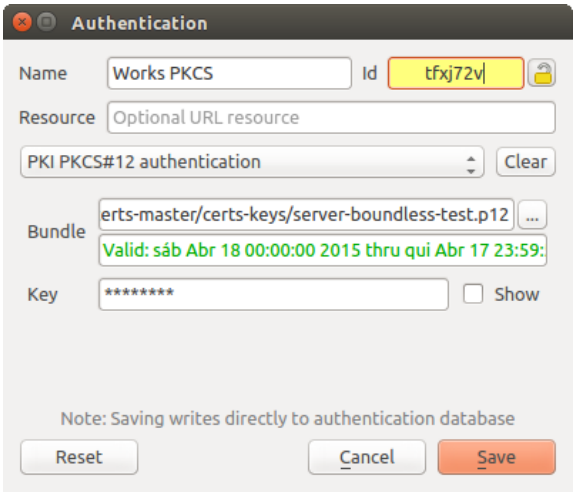


Слика21.31: Edit bad layer's authentication config ID

21.2.5 Changing authentication config ID

Occasionally, you will need to change the authentication configuration ID that is associated with accessing a resource. There are instances where this is useful:

- **Resource auth config ID is no longer valid**: This can occur when you have switched auth databases add need to *align* a new configuration to the ID already associated with a resource.
- **Shared project files**: If you intended to share projects between users, e.g. via a shared file server, you can *predefine* a 7-character (containing **a-z** and/or **0-9**) that is associated with the resource. Then, individual users change the ID of an authentication configuration that is specific to their credentials of the resource. When the project is opened, the ID is found in the authentication database, but the credentials are different per user.



Слика21.32: Changing a layer's authentication config ID (unlocked yellow text field)

Упозорење

Changing the auth config ID is considered an advanced operation and should only be done with full knowledge as to why it is necessary. This is why there is a lock button that needs clicked, to unlock the ID's text field prior

to editing the ID.

21.2.6 QGIS Server support

When using a project file, with layers that have authentication configurations, as a basis for a map in QGIS Server, there are a couple of additional setup steps necessary for QGIS to load the resources:

- Authentication database needs to be available
- Authentication database's master password needs to be available

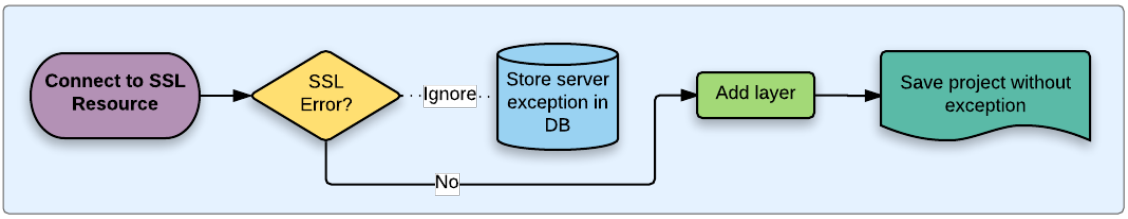
When instantiating the authentication system, Server will create or use `qgis-auth.db` file in the active *user profile*, or the directory defined by the `QGIS_AUTH_DB_DIR_PATH` environment variable. It may be that the Server's user has no HOME directory, in which case, use the environment variable to define a directory that the Server's user has read/write permissions and is not located within the web-accessible directories.

To pass the master password to Server, write it to the first line of file at a path on the file system readable by the Server processes user and defined using the `QGIS_AUTH_PASSWORD_FILE` environment variable. Ensure to limit the file as only readable by the Server's process user and to not store the file within web-accessible directories.

Белешка

`QGIS_AUTH_PASSWORD_FILE` variable will be removed from the Server environment immediately after accessing.

21.2.7 SSL server exceptions



Слика21.33: SSL server exception

You can manage SSL server configurations and exceptions from the **Servers** tab in the **Authentication** section of the QGIS **Options** dialog.

Sometimes, when connecting to an SSL server, there are errors with the SSL „handshake" or the server's certificate. You can ignore those errors or create an SSL server configuration as an exception. This is similar to how web browsers allow you to override SSL errors, but with more granular control.

Упозорење

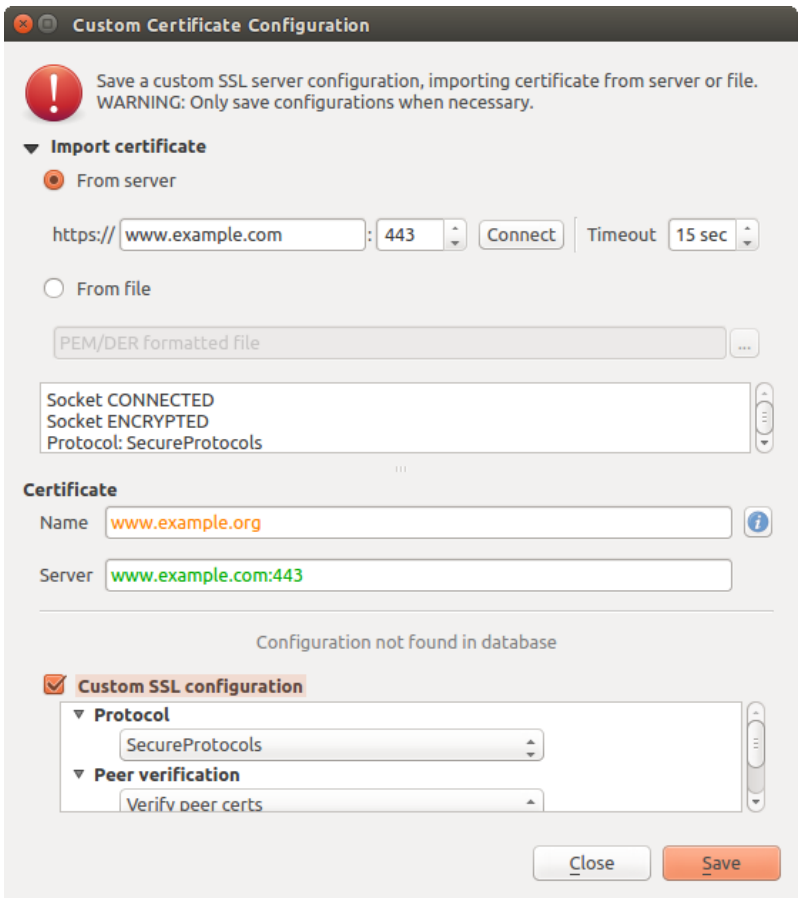
You should not create an SSL server configuration unless you have complete knowledge of the entire SSL setup between the server and client. Instead, report the issue to the server administrator.

Белешка

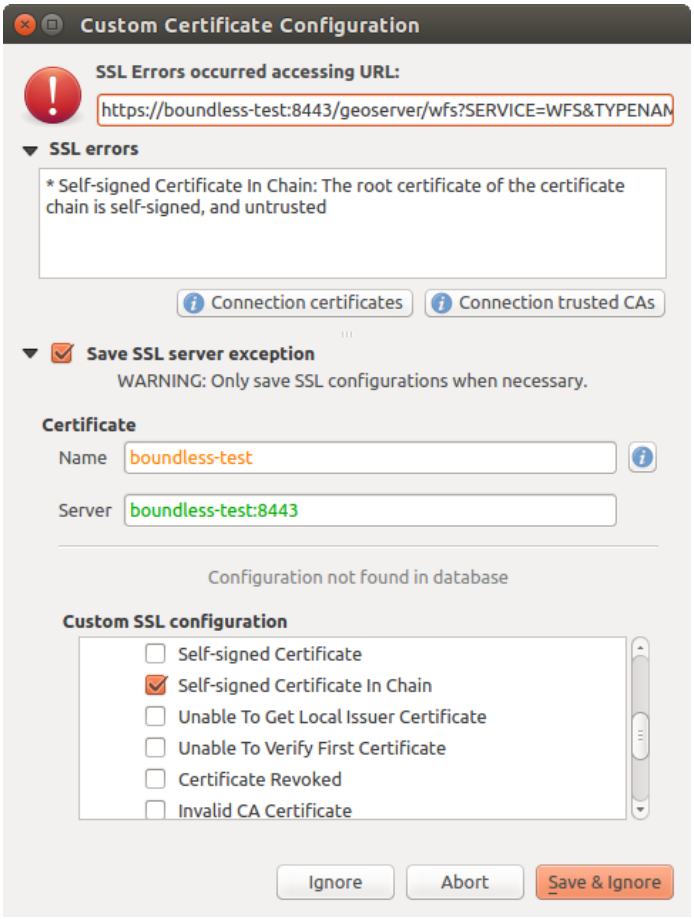
Some PKI setups use a completely different CA trust chain to validate client identities than the chain used to validate the SSL server certificate. In such circumstances, any configuration created for the connecting server

will not necessarily fix an issue with the validation of your client identity, and only your client identity's issuer or server administrator can fix the issue.

You can pre-configure an SSL server configuration by clicking the button. Alternatively, you can add a configuration when an SSL error occurs during a connection and you are presented with an **SSL Error** dialog (where the error can be ignored temporarily or saved to the database and ignored):

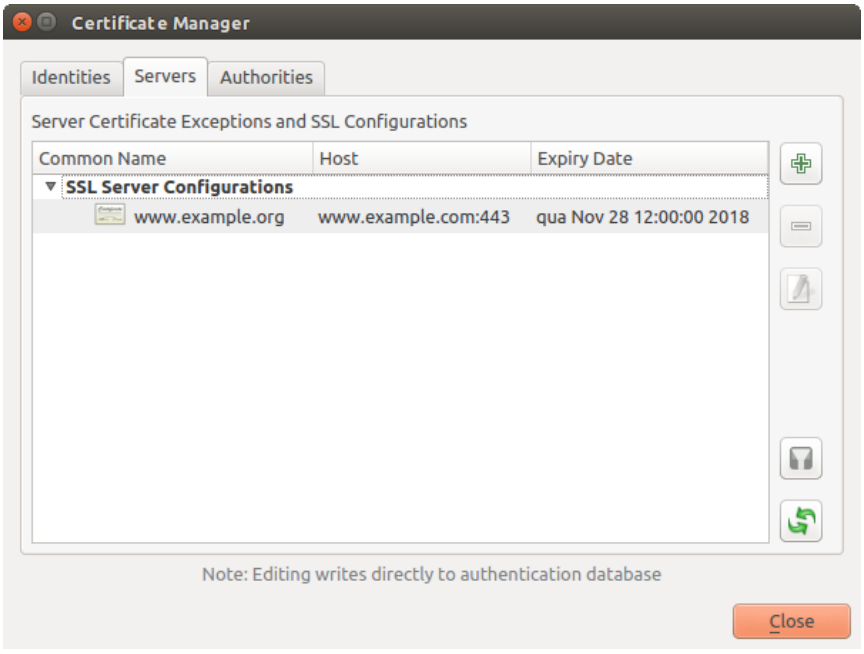


Слика21.34: Manually adding configuration

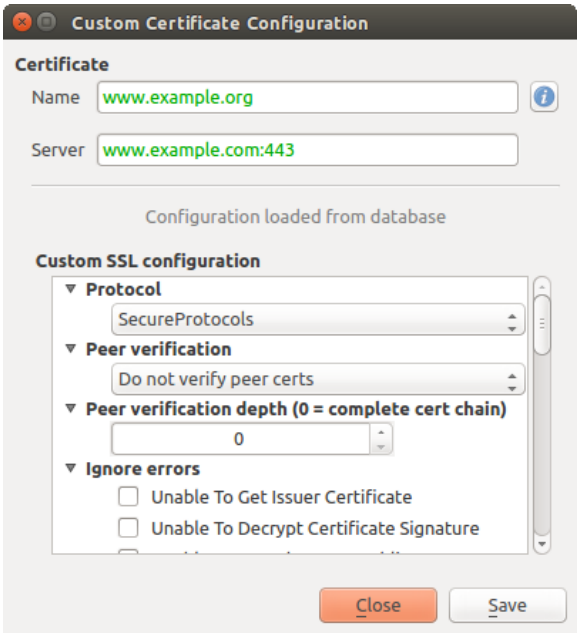


Слика21.35: Adding configuration during SSL error

Once an SSL configuration is saved to the database, it can be edited or deleted.



Слика21.36: Existing SSL configuration



Слика21.37: Editing an existing SSL configuration

If you want to pre-configure an SSL configuration and the import dialog is not working for your server's connection, you can manually trigger a connection via the **Python Console** by running the following code (replace `https://bugreports.qt-project.org` with the URL of your server):

```
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)
```

This will open an SSL error dialog if any errors occur, where you can choose to save the configuration to the database.

ГЛАВА 22

GRASS GIS Integration

GRASS integration provides access to GRASS GIS databases and functionalities (see GRASS-PROJECT in *Literatura i veb reference*). The integration consists of two parts: provider and plugin. The provider allows to browse, manage and visualize GRASS raster and vector layers. The plugin can be used to create new GRASS locations and mapsets, change GRASS region, create and edit vector layers and analyze GRASS 2-D and 3-D data with more than 400 GRASS modules. In this section, we'll introduce the provider and plugin functionalities and give some examples of managing and working with GRASS data.

The provider supports GRASS version 6 and 7, the plugin supports GRASS 6 and 7 (starting from QGIS 2.12). QGIS distribution may contain provider/plugin for either GRASS 6 or GRASS 7 or for both versions at the same time (binaries have different file names). Only one version of the provider/plugin may be loaded on runtime however.

22.1 Demo dataset

As an example, we will use the QGIS Alaska dataset (see section *Preuzimanje primernih podataka*). It includes a small sample GRASS `LOCATION` with three vector layers and one raster elevation map. Create a new folder called `grassdata`, download the QGIS 'Alaska' dataset `qgis_sample_data.zip` from https://qgis.org/downloads-list/#data and unzip the file into `grassdata`.

More sample GRASS `LOCATIONs` are available at the GRASS website.

22.2 Loading GRASS raster and vector layers

If the provider is loaded in QGIS, the location item with GRASS  icon is added in the browser tree under each folder item which contains GRASS location. Go to the folder `grassdata` and expand location `alaska` and mapset `demo`.

You can load GRASS raster and vector layers like any other layer from the browser either by double click on layer item or by dragging and dropping to map canvas or legend.

Савет

GRASS Data Loading

If you don't see GRASS location item, verify in *Help* ► *About* ► *Providers* if GRASS vector provider is loaded.

22.3 Importing data into a GRASS LOCATION via drag and drop

This section gives an example of how to import raster and vector data into a GRASS mapset.

1. In QGIS browser navigate to the mapset you want to import data into.
2. In QGIS browser find a layer you want to import to GRASS, note that you can open another instance of the browser (*Browser Panel (2)*) if source data are too far from the mapset in the tree.
3. Drag a layer and drop it on the target mapset. The import may take some time for larger layers, you will see animated icon in front of new layer item until the import finishes.

When raster data are in different CRS, they can be reprojected using an *Approximate* (fast) or *Exact* (precise) transformation. If a link to the source raster is created (using `r.external`), the source data are in the same CRS and the format is known to GDAL, the source data CRS will be used. You can set these options in the *Browser* tab in *GRASS Options*.

If a source raster has more bands, a new GRASS map is created for each layer with **.<band number>** suffix and group of all maps with icon is created. External rasters have a different icon .

22.4 Managing GRASS data in QGIS Browser

- Copying maps: GRASS maps may be copied between mapsets within the same location using drag and drop.
- Deleting maps: Right click on a GRASS map and select *Delete* from context menu.
- Renaming maps: Right click on a GRASS map and select *Rename* from context menu.

22.5 GRASS Options

GRASS options may be set in *GRASS Options* dialog, which can be opened by right clicking on the location or mapset item in the browser and then choosing *GRASS Options*.

22.6 Starting the GRASS plugin

To use GRASS functionalities in QGIS, you must select and load the GRASS plugin using the Plugin Manager. To do this, go to the menu *Plugins* ► *Manage and Install Plugins…*, select *GRASS* and click *OK*.

The following main features are provided with the GRASS menu (*Plugins* ► *GRASS*) when you start the GRASS plugin:

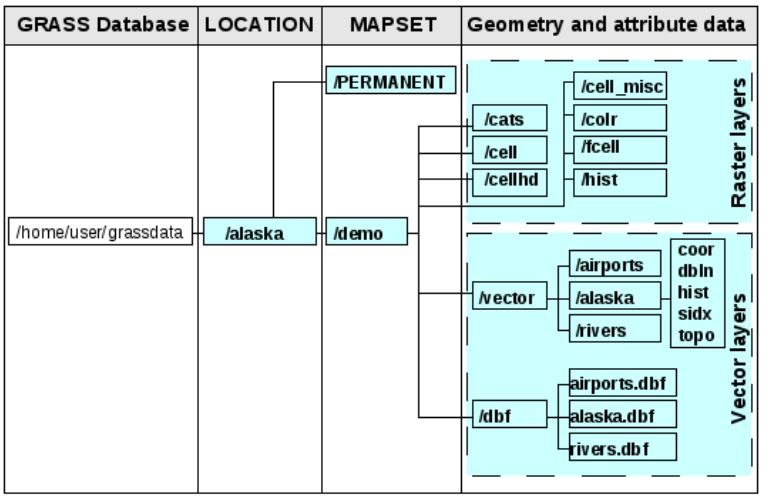
- Open Mapset
- New Mapset
- Close Mapset
- Open GRASS Tools
- Display Current GRASS Region
- GRASS Options

22.7 Opening GRASS mapset

A GRASS mapset must be opened to get access to GRASS Tools in the plugin (the tools are disabled if no mapset is open). You can open a mapset from the browser: right click on mapset item and then choose *Open mapset* from context menu.

22.8 GRASS LOCATION and MAPSET

GRASS data are stored in a directory referred to as GISDBASE. This directory, often called `grassdata`, must be created before you start working with the GRASS plugin in QGIS. Within this directory, the GRASS GIS data are organized by projects stored in subdirectories called `LOCATIONs`. Each `LOCATION` is defined by its coordinate system, map projection and geographical boundaries. Each `LOCATION` can have several `MAPSETs` (subdirectories of the `LOCATION`) that are used to subdivide the project into different topics or sub-regions, or as workspaces for individual team members (see Neteler & Mitasova 2008 in *Literatura i veb reference*). In order to analyze vector and raster layers with GRASS modules, you generally have to import them into a GRASS `LOCATION`. (This is not strictly true – with the GRASS modules `r.external` and `v.external` you can create read-only links to external GDAL-supported datasets without importing them. This is not the usual way for beginners to work with GRASS, therefore this functionality will not be described here.)



Слика22.1: GRASS data in the alaska LOCATION

22.9 Importing data into a GRASS LOCATION

See section *Importing data into a GRASS LOCATION via drag and drop* to find how data can be easily imported by dragging and dropping in the browser.

This section gives an example of how to import raster and vector data into the 'alaska' GRASS `LOCATION` provided by the QGIS 'Alaska' dataset in traditional way, using standard GRASS modules. Therefore, we use the landcover raster map `landcover.img` and the vector GML file `lakes.gml` from the QGIS 'Alaska' dataset (see *Preuzimanje primernih podataka*).

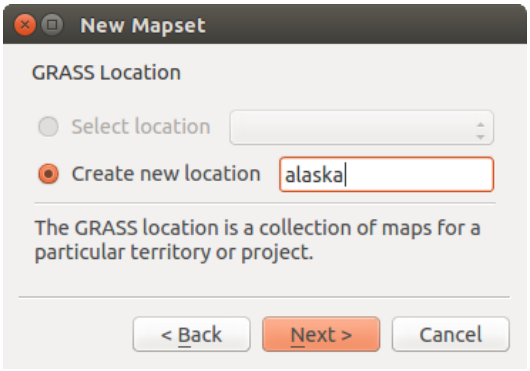
1. Start QGIS and make sure the GRASS plugin is loaded.
2. In the GRASS toolbar, click the Open MAPSET icon to bring up the *MAPSET* wizard.
3. Select as GRASS database the folder `grassdata` in the QGIS Alaska dataset, as `LOCATION` 'alaska', as `MAPSET` 'demo' and click *OK*.
4. Now click the Open GRASS tools icon. The GRASS Toolbox (see section *The GRASS Toolbox*) dialog appears.
5. To import the raster map `landcover.img`, click the module `r.in.gdal` in the *Modules Tree* tab. This GRASS module allows you to import GDAL-supported raster files into a GRASS `LOCATION`. The module dialog for `r.in.gdal` appears.
6. Browse to the folder `raster` in the QGIS 'Alaska' dataset and select the file `landcover.img`.
7. As raster output name, define `landcover_grass` and click *Run*. In the *Output* tab, you see the currently running GRASS command `r.in.gdal -o input=/path/to/landcover.img output=landcover_grass`.

8. When it says **Successfully finished**, click *View Output*. The `landcover_grass` raster layer is now imported into GRASS and will be visualized in the QGIS canvas.
9. To import the vector GML file `lakes.gml`, click the module `v.in.ogr` in the *Modules Tree* tab. This GRASS module allows you to import OGR-supported vector files into a GRASS `LOCATION`. The module dialog for `v.in.ogr` appears.
10. Browse to the folder `gml` in the QGIS 'Alaska' dataset and select the file `lakes.gml` as OGR file.
11. As vector output name, define `lakes_grass` and click *Run*. You don't have to care about the other options in this example. In the *Output* tab you see the currently running GRASS command `v.in.ogr -o dsn=/path/to/lakes.gml output=lakes\_grass`.
12. When it says **Successfully finished**, click *View Output*. The `lakes_grass` vector layer is now imported into GRASS and will be visualized in the QGIS canvas.

22.9.1 Creating a new GRASS LOCATION

As an example, here is the sample GRASS `LOCATION alaska`, which is projected in the Albers Equal Area projection using feet as units. This sample GRASS `LOCATION alaska` will be used for all examples and exercises in the following GRASS-related sections. It is useful to download and install the dataset on your computer (see *Preuzimanje primernih podataka*).

1. Start QGIS and make sure the GRASS plugin is loaded.
2. Visualize the `alaska.shp` shapefile (see section *Loading a layer from a file*) from the QGIS Alaska dataset (see *Preuzimanje primernih podataka*).
3. In the GRASS toolbar, click on the  New mapset icon to bring up the *MAPSET* wizard.
4. Select an existing GRASS database (GISDBASE) folder `grassdata`, or create one for the new `LOCATION` using a file manager on your computer. Then click *Next*.
5. We can use this wizard to create a new `MAPSET` within an existing `LOCATION` (see section *Adding a new MAPSET*) or to create a new `LOCATION` altogether. Select  *Create new location* (see Сл.22.2).
6. Enter a name for the `LOCATION` – we used 'alaska' – and click *Next*.
7. Define the projection by clicking on the radio button  *Projection* to enable the projection list.
8. We are using Albers Equal Area Alaska (feet) projection. Since we happen to know that it is represented by the EPSG ID 2964, we enter it in the search box. (Note: If you want to repeat this process for another `LOCATION` and projection and haven't memorized the EPSG ID, click on the  CRS Status icon in the lower right-hand corner of the status bar (see section *Working with Projections*)).
9. In *Filter*, insert 2964 to select the projection.
10. Click *Next*.
11. To define the default region, we have to enter the `LOCATION` bounds in the north, south, east, and west directions. Here, we simply click on the button *Set Current QGIS Extent*, to apply the extent of the loaded layer `alaska.shp` as the GRASS default region extent.
12. Click *Next*.
13. We also need to define a `MAPSET` within our new `LOCATION` (this is necessary when creating a new `LOCATION`). You can name it whatever you like - we used 'demo'. GRASS automatically creates a special `MAPSET` called `PERMANENT`, designed to store the core data for the project, its default spatial extent and coordinate system definitions (see Neteler & Mitasova 2008 in *Literatura i veb reference*).
14. Check out the summary to make sure it's correct and click *Finish*.
15. The new `LOCATION`, 'alaska', and two `MAPSETs`, 'demo' and 'PERMANENT', are created. The currently opened working set is 'demo', as you defined.
16. Notice that some of the tools in the GRASS toolbar that were disabled are now enabled.



Слика22.2: Creating a new GRASS LOCATION or a new MAPSET in QGIS

If that seemed like a lot of steps, it's really not all that bad and a very quick way to create a LOCATION. The LOCATION 'alaska' is now ready for data import (see section *Importing data into a GRASS LOCATION*). You can also use the already-existing vector and raster data in the sample GRASS LOCATION 'alaska', included in the QGIS 'Alaska' dataset *Preuzimanje primernih podataka*, and move on to section *The GRASS vector data model*.

22.9.2 Adding a new MAPSET

A user has write access only to a GRASS MAPSET which he or she created. This means that besides access to your own MAPSET, you can read maps in other users' MAPSETs (and they can read yours), but you can modify or remove only the maps in your own MAPSET.

All MAPSETs include a WIND file that stores the current boundary coordinate values and the currently selected raster resolution (see Neteler & Mitasova 2008 in *Literatura i veb reference*, and section *The GRASS region tool*).

1. Start QGIS and make sure the GRASS plugin is loaded.
2. In the GRASS toolbar, click on the  New mapset icon to bring up the *MAPSET* wizard.
3. Select the GRASS database (GISDBASE) folder grassdata with the LOCATION 'alaska', where we want to add a further MAPSET called 'test'.
4. Click *Next*.
5. We can use this wizard to create a new MAPSET within an existing LOCATION or to create a new LOCATION altogether. Click on the radio button  *Select location* (see Сл.22.2) and click *Next*.
6. Enter the name test for the new MAPSET. Below in the wizard, you see a list of existing MAPSETs and corresponding owners.
7. Click *Next*, check out the summary to make sure it's all correct and click *Finish*.

22.10 The GRASS vector data model

It is important to understand the GRASS vector data model prior to digitizing. In general, GRASS uses a topological vector model. This means that areas are not represented as closed polygons, but by one or more boundaries. A boundary between two adjacent areas is digitized only once, and it is shared by both areas. Boundaries must be connected and closed without gaps. An area is identified (and labelled) by the **centroid** of the area.

Besides boundaries and centroids, a vector map can also contain points and lines. All these geometry elements can be mixed in one vector and will be represented in different so-called 'layers' inside one GRASS vector map. So in GRASS, a layer is not a vector or raster map but a level inside a vector layer. This is important to distinguish carefully. (Although it is possible to mix geometry elements, it is unusual and, even in GRASS, only used in special cases such as vector network analysis. Normally, you should prefer to store different geometry elements in different layers.)

It is possible to store several 'layers' in one vector dataset. For example, fields, forests and lakes can be stored in one vector. An adjacent forest and lake can share the same boundary, but they have separate attribute tables. It is also possible to attach attributes to boundaries. An example might be the case where the boundary between a lake and a forest is a road, so it can have a different attribute table.

The 'layer' of the feature is defined by the 'layer' inside GRASS. 'Layer' is the number which defines if there is more than one layer inside the dataset (e.g., if the geometry is forest or lake). For now, it can be only a number. In the future, GRASS will also support names as fields in the user interface.

Attributes can be stored inside the GRASS `LOCATION` as dBase, SQLite3 or in external database tables, for example, PostgreSQL, MySQL, Oracle, etc.

Attributes in database tables are linked to geometry elements using a 'category' value.

'Category' (key, ID) is an integer attached to geometry primitives, and it is used as the link to one key column in the database table.

Савет

Learning the GRASS Vector Model

The best way to learn the GRASS vector model and its capabilities is to download one of the many GRASS tutorials where the vector model is described more deeply. See https://grass.osgeo.org/learn/manuals/ for more information, books and tutorials in several languages.

22.11 Creating a new GRASS vector layer

To create a new GRASS vector layer, select one of following items from mapset context menu in the browser:

- New Point Layer
- New Line Layer
- New Polygon Layer

and enter a name in the dialog. A new vector map will be created and layer will be added to canvas and editing started. Selecting type of the layer does not restrict geometry types which can be digitized in the vector map. In GRASS, it is possible to organize all sorts of geometry types (point, line and polygon) in one vector map. The type is only used to add the layer to the canvas, because QGIS requires a layer to have a specific type.

It is also possible to add layers to existing vector maps selecting one of the items described above from context menu of existing vector map.

In GRASS, it is possible to organize all sorts of geometry types (point, line and area) in one layer, because GRASS uses a topological vector model, so you don't need to select the geometry type when creating a new GRASS vector. This is different from shapefile creation with QGIS, because shapefiles use the Simple Feature vector model (see section *Creating new vector layers*).

22.12 Digitizing and editing a GRASS vector layer

GRASS vector layers can be digitized using the standard QGIS digitizing tools. There are however some particularities, which you should know about, due to

- GRASS topological model versus QGIS simple feature
- complexity of GRASS model
  - multiple layers in single maps
  - multiple geometry types in single map
  - geometry sharing by multiple features from multiple layers

The particularities are discussed in the following sections.

Save, discard changes, undo, redo

Упозорење

All the changes done during editing are immediately written to vector map and related attribute tables.

Changes are written after each operation, it is however, possible to do undo/redo or discard all changes when closing editing. If undo or discard changes is used, original state is rewritten in vector map and attribute tables.

There are two main reasons for this behaviour:

- It is the nature of GRASS vectors coming from conviction that user wants to do what he is doing and it is better to have data saved when the work is suddenly interrupted (for example, blackout)
- Necessity for effective editing of topological data is visualized information about topological correctness, such information can only be acquired from GRASS vector map if changes are written to the map.

Toolbar

The 'Digitizing Toolbar' has some specific tools when a GRASS layer is edited:

Табела22.1: GRASS Digitizing Tools

Ikona	Alatka	Svrha
	New Point	Digitize new point
	New Line	Digitize new line
	New Boundary	Digitize new boundary
	New Centroid	Digitize new centroid (label existing area)
	New Closed Boundary	Digitize new closed boundary

Савет

Digitizing polygons in GRASS

If you want to create a polygon in GRASS, you first digitize the boundary of the polygon. Then you add a centroid (label point) into the closed boundary. The reason for this is that a topological vector model links the attribute information of a polygon always to the centroid and not to the boundary.

Category

Category, often called cat, is sort of ID. The name comes from times when GRASS vectors had only singly attribute „category". Category is used as a link between geometry and attributes. A single geometry may have multiple categories and thus represent multiple features in different layers. Currently it is possible to assign only one category per layer using QGIS editing tools. New features have automatically assigned new unique category, except boundaries. Boundaries usually only form areas and do not represent linear features, it is however possible to define attributes for a boundary later, for example in different layer.

New categories are always created only in currently being edited layer.

It is not possible to assign more categories to geometry using QGIS editing, such data are properly represented as multiple features, and individual features, even from different layers, may be deleted.

Attributes

Attributes of currently edited layer can only be modified. If the vector map contains more layers, features of other layers will have all attributes set to '<not editable (layer #)>' to warn you that such attribute is not editable. The

reason is, that other layers may have and usually have different set of fields while QGIS only supports one fixed set of fields per layer.

If a geometry primitive does not have a category assigned, a new unique category is automatically assigned and new record in attribute table is created when an attribute of that geometry is changed.

Савет

If you want to do bulk update of attributes in table, for example using 'Field Calculator' (*Korišćenje kalkulatora polja*), and there are features without category which you don't want to update (typically boundaries), you can filter them out by setting 'Advanced Filter' to `cat is not null`.

Editing style

The topological symbology is essential for effective editing of topological data. When editing starts, a specialized 'GRASS Edit' renderer is set on the layer automatically and original renderer is restored when editing is closed. The style may be customized in layer properties 'Style' tab. The style can also be stored in project file or in separate file as any other style. If you customize the style, do not change its name, because it is used to reset the style when editing is started again.

Савет

Do not save project file when the layer is edited, the layer would be stored with 'Edit Style' which has no meaning if layer is not edited.

The style is based on topological information which is temporarily added to attribute table as field 'topo_symbol'. The field is automatically removed when editing is closed.

Савет

Do not remove 'topo_symbol' field from attribute table, that would make features invisible because the renderer is based on that column.

Snapping

To form an area, vertices of connected boundaries must have **exactly** the same coordinates. This can be achieved using snapping tool only if canvas and vector map have the same CRS. Otherwise, due conversion from map coordinates to canvas and back, the coordinate may become slightly different due to representation error and CRS transformations.

Савет

Use layer's CRS also for canvas when editing.

Limitations

Simultaneous editing of multiple layers within the same vector at the same time is not supported. This is mainly due to the impossibility of handling multiple undo stacks for a single data source.

On Linux and macOS only one GRASS layer can be edited at time. This is due to a bug in GRASS which does not allow to close database drivers in random order. This is being solved with GRASS developers.

Савет

GRASS Edit Permissions

You must be the owner of the GRASS `MAPSET` you want to edit. It is impossible to edit data layers in a `MAPSET` that is not yours, even if you have write permission.

22.13 The GRASS region tool

The region definition (setting a spatial working window) in GRASS is important for working with raster layers. Vector analysis is by default not limited to any defined region definitions. But all newly created rasters will have the spatial extension and resolution of the currently defined GRASS region, regardless of their original extension and resolution. The current GRASS region is stored in the `$LOCATION/$MAPSET/WIND` file, and it defines north, south, east and west bounds, number of columns and rows, horizontal and vertical spatial resolution.

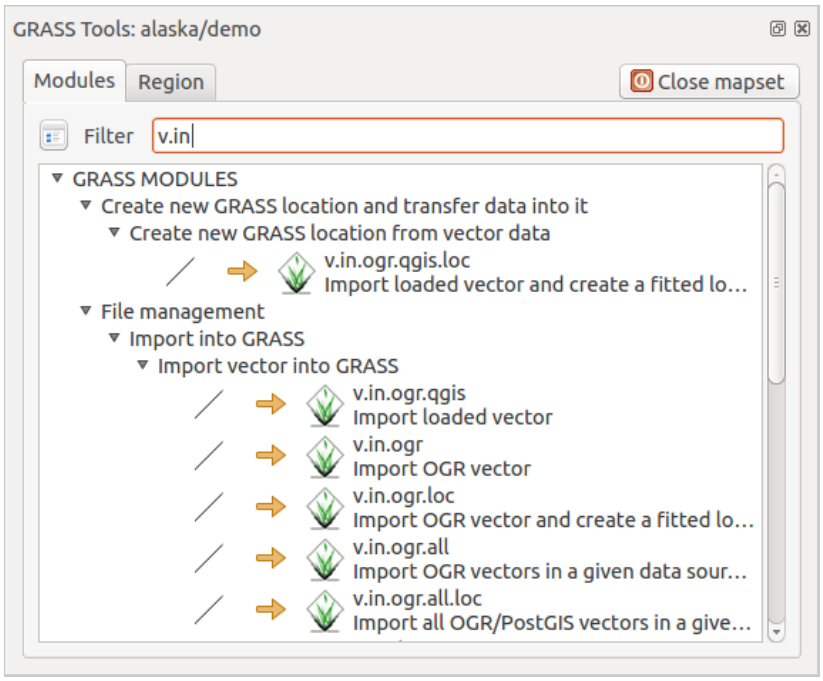
It is possible to switch on and off the visualization of the GRASS region in the QGIS canvas using the  Display current GRASS region button.

The region can be modified in 'Region' tab in 'GRASS Tolls' dock widget. Type in the new region bounds and resolution, and click *Apply*. If you click on *Select the extent by dragging on canvas* you can select a new region interactively with your mouse on the QGIS canvas dragging a rectangle.

The GRASS module `g.region` provides a lot more parameters to define an appropriate region extent and resolution for your raster analysis. You can use these parameters with the GRASS Toolbox, described in section *The GRASS Toolbox*.

22.14 The GRASS Toolbox

The  Open GRASS Tools box provides GRASS module functionalities to work with data inside a selected GRASS `LOCATION` and `MAPSET`. To use the GRASS Toolbox you need to open a `LOCATION` and `MAPSET` that you have write permission for (usually granted, if you created the `MAPSET`). This is necessary, because new raster or vector layers created during analysis need to be written to the currently selected `LOCATION` and `MAPSET`.



Слика22.3: GRASS Toolbox and Module Tree

22.14.1 Working with GRASS modules

The GRASS plugin provides a user-friendly toolbox with about 200 of the available GRASS modules and functionalities.

A complete list of GRASS modules available in the graphical Toolbox in QGIS is available in the GRASS wiki at https://grasswiki.osgeo.org/wiki/GRASS-QGIS_relevant_module_list.

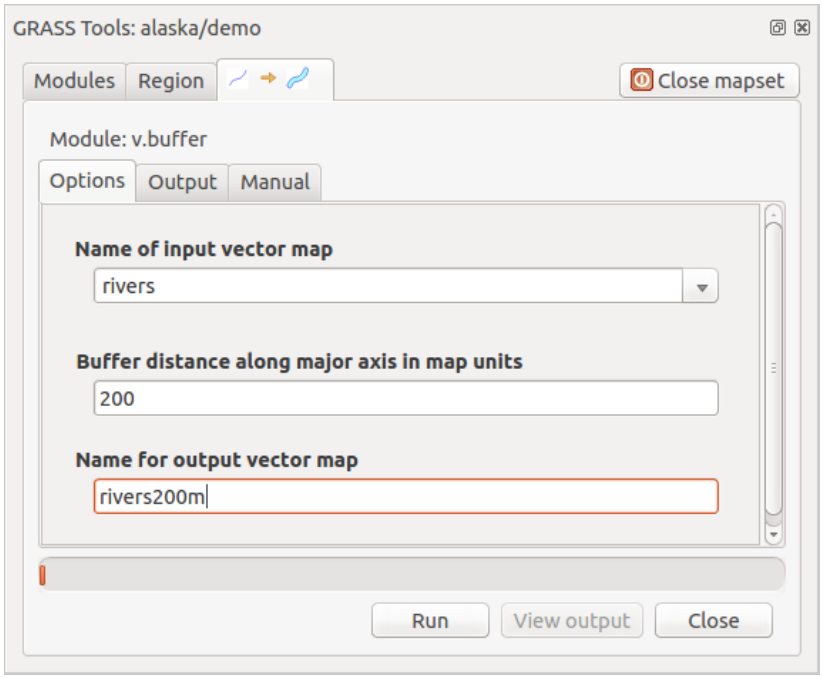
It is also possible to customize the GRASS Toolbox content. This procedure is described in section *Customizing the GRASS Toolbox*.

As shown in Сл.22.3, you can look for the appropriate GRASS module using the thematically grouped *Modules Tree* or the searchable *Modules List* tab.

By clicking on a graphical module icon, a new tab will be added to the Toolbox dialog, providing three new sub-tabs: *Options*, *Output* and *Manual*.

Options

The *Options* tab provides a simplified module dialog where you can usually select a raster or vector layer visualized in the QGIS canvas and enter further module-specific parameters to run the module.

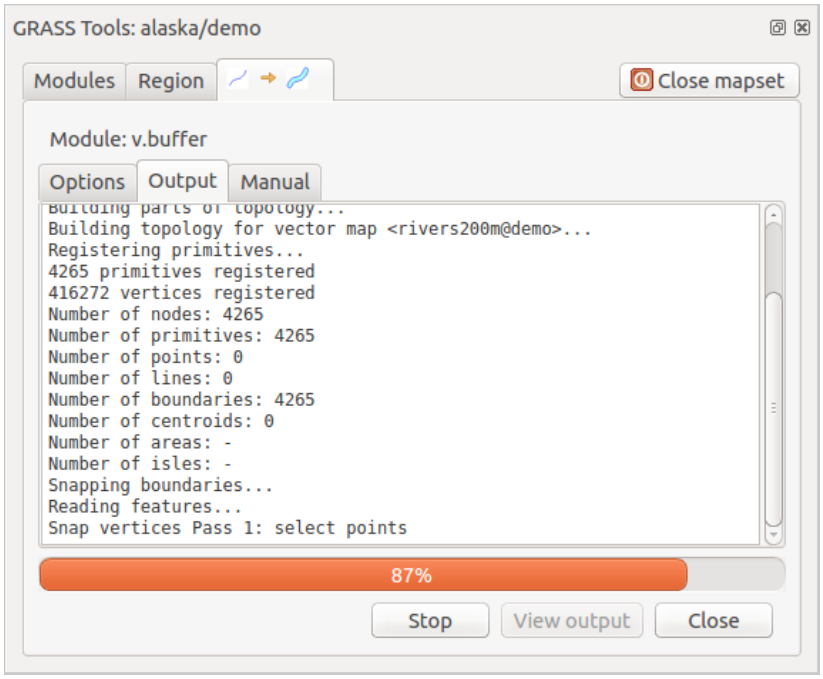


Слика22.4: GRASS Toolbox Module Options

The provided module parameters are often not complete to keep the dialog simple. If you want to use further module parameters and flags, you need to start the GRASS shell and run the module in the command line.

A new feature since QGIS 1.8 is the support for a *Show Advanced Options* button below the simplified module dialog in the *Options* tab. At the moment, it is only added to the module `v.in.ascii` as an example of use, but it will probably be part of more or all modules in the GRASS Toolbox in future versions of QGIS. This allows you to use the complete GRASS module options without the need to switch to the GRASS shell.

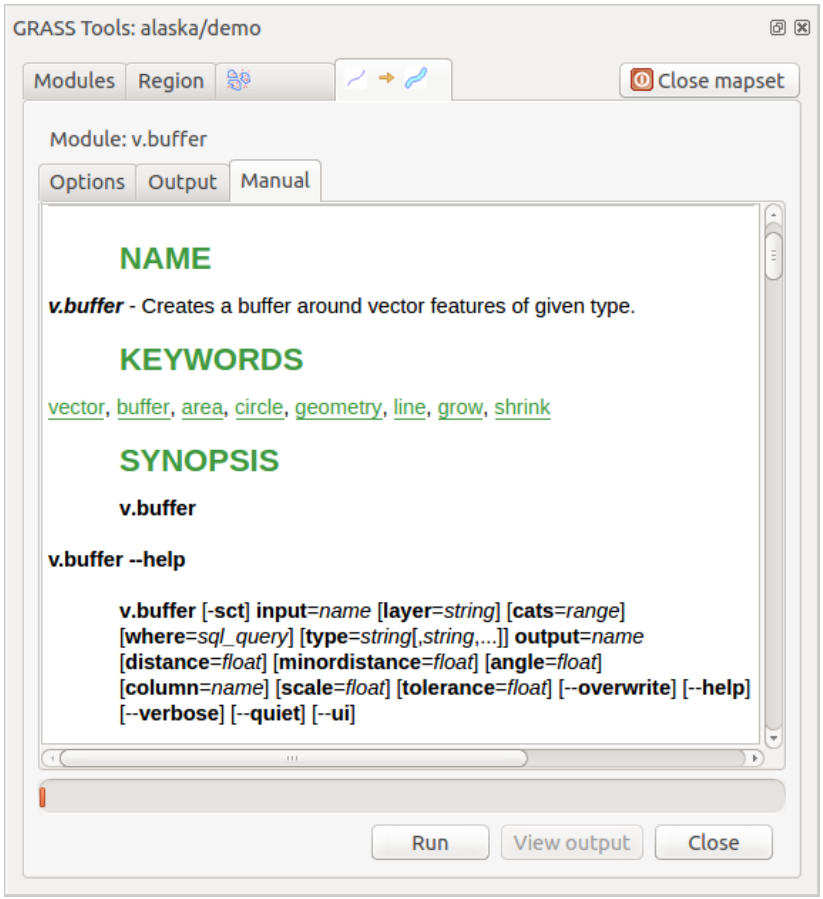
Izlaz



Слика22.5: GRASS Toolbox Module Output

The *Output* tab provides information about the output status of the module. When you click the *Run* button, the module switches to the *Output* tab and you see information about the analysis process. If all works well, you will finally see a `Successfully finished` message.

Manual



Слика22.6: GRASS Toolbox Module Manual

The *Manual* tab shows the HTML help page of the GRASS module. You can use it to check further module parameters and flags or to get a deeper knowledge about the purpose of the module. At the end of each module manual page, you see further links to the `Main Help index`, the `Thematic index` and the `Full index`. These links provide the same information as the module `g.manual`.

Савет

Display results immediately

If you want to display your calculation results immediately in your map canvas, you can use the 'View Output' button at the bottom of the module tab.

22.14.2 GRASS module examples

The following examples will demonstrate the power of some of the GRASS modules.

Creating contour lines

The first example creates a vector contour map from an elevation raster (DEM). Here, it is assumed that you have the Alaska `LOCATION` set up as explained in section *Importing data into a GRASS LOCATION*.

- First, open the location by clicking the Open mapset button and choosing the Alaska location.
- Now open the Toolbox with the Open GRASS tools button.
- In the list of tool categories, double-click *Raster* ► *Surface Management* ► *Generate vector contour lines*.

- Now a single click on the tool **r.contour** will open the tool dialog as explained above (see *Working with GRASS modules*).
- In the *Name of input raster map* enter `gtopo30`.
- Type into the *Increment between Contour levels*  the value 100. (This will create contour lines at intervals of 100 meters.)
- Type into the *Name for output vector map* the name `ctour_100`.
- Click *Run* to start the process. Wait for several moments until the message `Successfully finished` appears in the output window. Then click *View Output* and *Close*.

Since this is a large region, it will take a while to display. After it finishes rendering, you can open the layer properties window to change the line color so that the contours appear clearly over the elevation raster, as in *The Vector Properties Dialog*.

Next, zoom in to a small, mountainous area in the center of Alaska. Zooming in close, you will notice that the contours have sharp corners. GRASS offers the **v.generalize** tool to slightly alter vector maps while keeping their overall shape. The tool uses several different algorithms with different purposes. Some of the algorithms (i.e., Douglas Peuker and Vertex Reduction) simplify the line by removing some of the vertices. The resulting vector will load faster. This process is useful when you have a highly detailed vector, but you are creating a very small-scale map, so the detail is unnecessary.

Савет

The simplify tool

Note that QGIS has a *Vector* ► *Geometry Tools* ► *Simplify geometries* tool that works just like the GRASS **v.generalize** Douglas-Peuker algorithm.

However, the purpose of this example is different. The contour lines created by `r.contour` have sharp angles that should be smoothed. Among the **v.generalize** algorithms, there is Chaiken's, which does just that (also Hermite splines). Be aware that these algorithms can **add** additional vertices to the vector, causing it to load even more slowly.

- Open the GRASS Toolbox and double-click the categories *Vector* ► *Develop map* ► *Generalization*, then click on the **v.generalize** module to open its options window.
- Check that the 'ctour_100' vector appears as the *Name of input vector*.
- From the list of algorithms, choose Chaiken's. Leave all other options at their default, and scroll down to the last row to enter in the field *Name for output vector map* 'ctour_100_smooth', and click *Run*.
- The process takes several moments. Once `Successfully finished` appears in the output windows, click *View Output* and then *Close*.
- You may change the color of the vector to display it clearly on the raster background and to contrast with the original contour lines. You will notice that the new contour lines have smoother corners than the original while staying faithful to the original overall shape.



Слика22.7: GRASS module v.generalize to smooth a vector map

Савет

Other uses for r.contour

The procedure described above can be used in other equivalent situations. If you have a raster map of precipitation data, for example, then the same method will be used to create a vector map of isohyetal (constant rainfall) lines.

Creating a Hillshade 3-D effect

Several methods are used to display elevation layers and give a 3-D effect to maps. The use of contour lines, as shown above, is one popular method often chosen to produce topographic maps. Another way to display a 3-D effect is by hillshading. The hillshade effect is created from a DEM (elevation) raster by first calculating the slope and aspect of each cell, then simulating the sun's position in the sky and giving a reflectance value to each cell. Thus, you get sun-facing slopes lighted; the slopes facing away from the sun (in shadow) are darkened.

- Begin this example by loading the `gtopo30` elevation raster. Start the GRASS Toolbox, and under the Raster category, double-click to open *Spatial analysis* ► *Terrain analysis*.
- Then click **r.shaded.relief** to open the module.
- Change the *azimuth angle* 1,00 270 to 315.
- Enter `gtopo30_shade` for the new hillshade raster, and click *Run*.
- When the process completes, add the hillshade raster to the map. You should see it displayed in grayscale.
- To view both the hillshading and the colors of the `gtopo30` together, move the hillshade map below the `gtopo30` map in the table of contents, then open the *Properties* window of `gtopo30`, switch to the *Transparency* tab and set its transparency level to about 25%.

You should now have the `gtopo30` elevation with its colormap and transparency setting displayed **above** the grayscale hillshade map. In order to see the visual effects of the hillshading, turn off the `gtopo30_shade` map, then turn it back on.

Raster statistics in a vector map

The next example shows how a GRASS module can aggregate raster data and add columns of statistics for each polygon in a vector map.

- Again using the Alaska data, refer to *Importing data into a GRASS LOCATION* to import the `shapefiles/trees.shp` file into GRASS.

- Now an intermediate step is required: centroids must be added to the imported trees map to make it a complete GRASS area vector (including both boundaries and centroids).
- From the Toolbox, choose *Vector* ► *Manage features*, and open the module **v.centroids**.
- Enter as the *output vector map* 'forest_areas' and run the module.
- Now load the `forest_areas` vector and display the types of forests - deciduous, evergreen, mixed - in different colors: In the layer *Properties* window, *Symbology* tab, choose from *Legend type*  'Unique value' and set the *Classification field* to 'VEGDESC'. (Refer to the explanation of the symbology tab in *Symbology Properties* of the vector section.)
- Next, reopen the GRASS Toolbox and open *Vector* ► *Vector update* by other maps.
- Click on the **v.rast.stats** module. Enter `gtopo30` and `forest_areas`.
- Only one additional parameter is needed: Enter *column prefix* `elev`, and click *Run*. This is a computationally heavy operation, which will run for a long time (probably up to two hours).
- Finally, open the `forest_areas` attribute table, and verify that several new columns have been added, including `elev_min`, `elev_max`, `elev_mean`, etc., for each forest polygon.

22.14.3 Customizing the GRASS Toolbox

Nearly all GRASS modules can be added to the GRASS Toolbox. An XML interface is provided to parse the pretty simple XML files that configure the modules' appearance and parameters inside the Toolbox.

A sample XML file for generating the module `v.buffer` (v.buffer.qgm) looks like this:

```
<?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>
```

The parser reads this definition and creates a new tab inside the Toolbox when you select the module.

ГЛАВА 23

QGIS okvir za obradu

23.1 Introduction

This chapter introduces the QGIS processing framework, a geoprocessing environment that can be used to call native and third-party algorithms from QGIS, making your spatial analysis tasks more productive and easy to accomplish.

As a *Core plugin*, Processing is installed by default but you need to activate it:

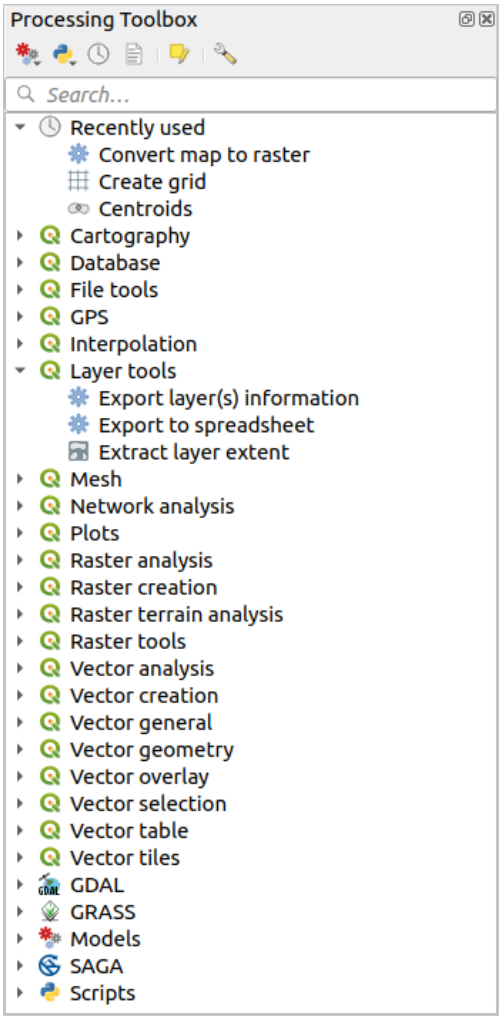
1. Go to *Plugins* ► *Manage and install plugins…*
2. Click on the *Installed* tab at the left
3. Check the box next to the  *Processing* entry
4. Close the dialog.

   A *Processing* menu is now available in the top menu bar. From there you can reach the main components of this framework.

In the following sections, we will review how to use the graphical elements of this framework and make the most out of each one of them.

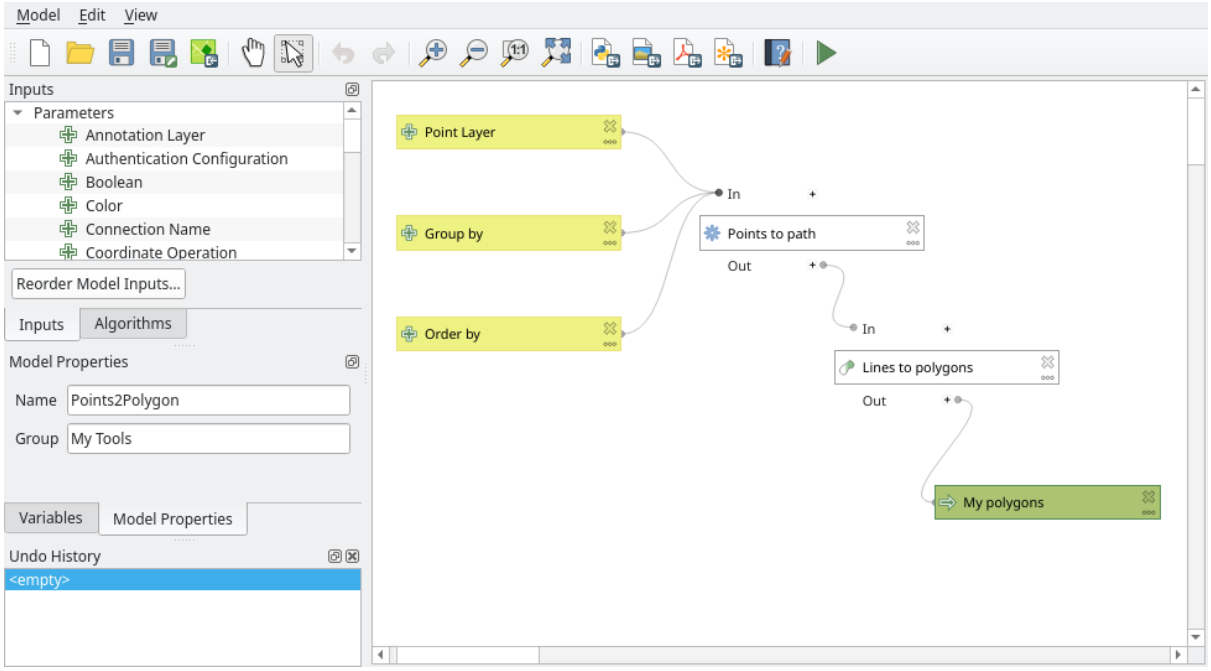
There are four basic elements in the framework GUI, which are used to run algorithms for different purposes. Choosing one tool or another will depend on the kind of analysis that is to be performed and the particular characteristics of each user and project. All of them (except for the batch processing interface, which is called from the toolbox or the algorithm execution dialog, as we will see) can be accessed from the *Processing* menu item (you will see more entries; the remaining ones are not used to execute algorithms and will be explained later in this chapter).

- The *Toolbox*: The main element of the GUI, it is used to execute a single algorithm or run a batch process based on that algorithm.



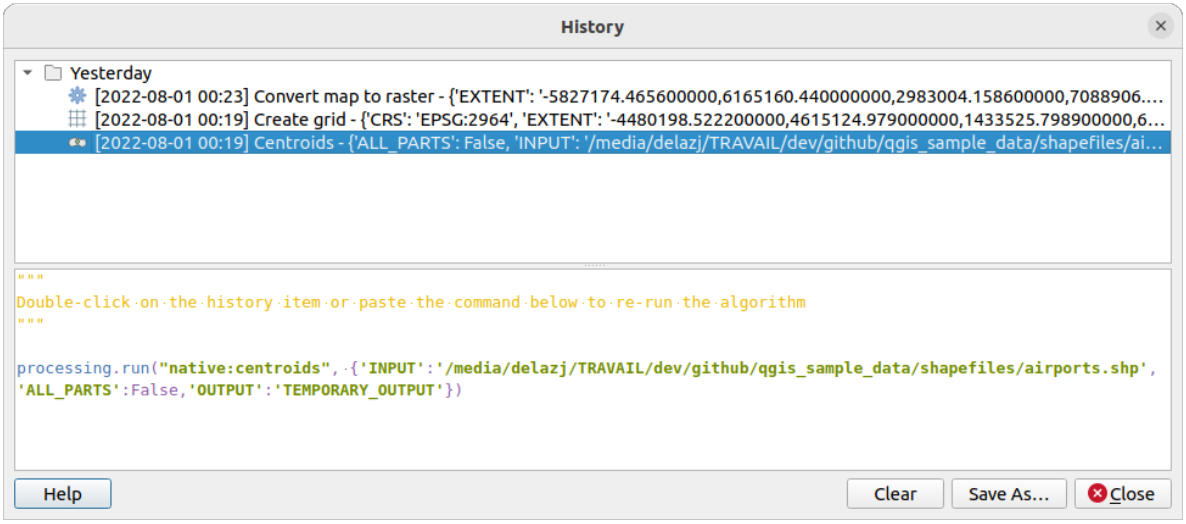
Слика23.1: Processing Toolbox

- The *Model Designer*: Several algorithms can be combined graphically using the modeler to define a workflow, creating a single process that involves several subprocesses.



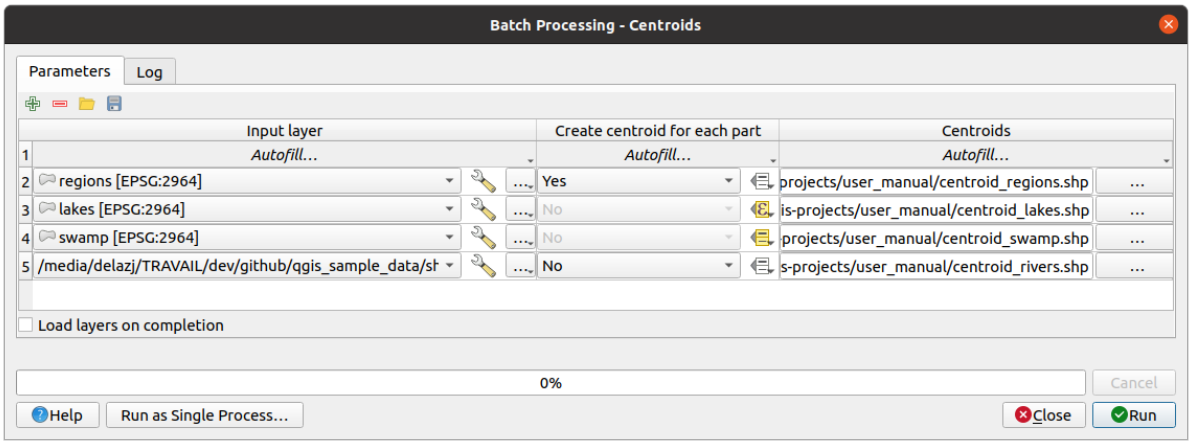
Слика23.2: Processing Modeler

- The *History* manager: All actions performed using any of the aforementioned elements are stored in a history file and can be later easily reproduced using the history manager.



Слика23.3: Processing History

- The *Batch Processing* interface: This interface allows you to execute batch processes and automate the execution of a single algorithm on multiple datasets.



Слика23.4: Batch Processing interface

In the following sections, we will review each one of these elements in detail.

23.2 Configuring the Processing Framework

The Processing Options menu (*Settings*► *Options* ► *Processing* tab) allows you to configure how algorithms work. Configuration parameters are structured in separate blocks that you can select on the left-hand side of the dialog.

23.2.1 General

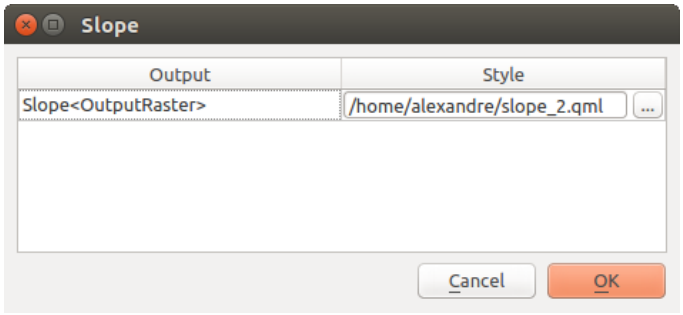
The *General* block contains the default settings to control how the algorithm dialog as well as input or output parameters should behave. Some of the settings can however be overridden at the algorithm level, per *algorithm run* or for *individual parameters*.

- *Default output raster layer extension* is by default `tif`
- *Default output vector layer extension* is by default `gpkg`
- *Invalid features filtering* when executing algorithm:
  - *Do not filter (better performance)*: all the features (with valid and invalid geometries) are processed, but the result may be erroneous depending on how the geometry invalidity affects the operations
  - *Skip (ignore) features with invalid geometries*, meaning that only a subset of your dataset (the valid geometry features) will be processed
  - *Stop algorithm execution when a geometry is invalid*: you'll need to track and fix the invalid geometries if you want the algorithm to process the whole layer. Algorithms like *Proveri validnost* or *Popravi geometrije* can help you achieve this.

  The *Invalid features filtering* setting can be overridden on a per-input basis, at algorithm runtime.
- *Keep dialog open after running algorithm*. Once an algorithm has finished execution and its output layers are loaded into the QGIS project, the algorithm dialog is closed. If you want to keep it open (to run the algorithm again with different parameters, or to better check the output that is written to the log tab), check this option.
- *Max Threads*
- *Output folder* for non temporary outputs: If no folder path is provided for the Processing execution outputs, this is the folder in which they will be saved. Default is `processing` under the user home directory (e.g., `C:\Users\<YOURUSERNAME>\processing\` on Windows).

- *Override temporary output folder path*: Temporary outputs are saved by default in the `tmp` folder on the machine. This option helps you set a different place for storage.
- *Pre-execution script* and *Post-execution script*. These parameters point to files that contain scripts written using the processing scripting functionality, explained in the section covering scripting and the console.
- *Prefer output filename for layer names*. The name of each resulting layer created by an algorithm is defined by the algorithm itself. In some cases, a fixed name might be used, meaning that the same output name will be used, no matter which input layer is used. In other cases, the name might depend on the name of the input layer or some of the parameters used to run the algorithm. If this checkbox is checked, the name will be taken from the output filename instead. Notice that, if the output is saved to a temporary file, the filename of this temporary file is usually a long and meaningless one intended to avoid collision with other already existing filenames.
- *Results group name*. If you want to obtain all processing result layers in a group in the *Layers* panel, set a group name for this parameter. The group may exist already or not. QGIS will add all output layers to such a group. By default, this parameter is empty, so all output layers are added to different places in the *Layers* panel, depending on the item that is active when running an algorithm. Note that output layers will be loaded to the *Layers* panel only if *Open output file after running algorithm* is checked in the algorithm dialog.
- *Show algorithms with known issues*: By default, QGIS avoids display of broken algorithms (generally from third-party providers). If checked, they will be available in the Processing toolbox, with a warning icon and a tooltip explaining they have issues. Use at your own risks.
- *Show feature count for output vector layers*. Since calculating the feature count might take some time in certain data formats, this option is off by default.
- *Show layer CRS definition in selection boxes*
- *Show tooltip when there are disabled providers*
- *Style for line layers*, *Style for point layers*, *Style for polygons layers* and *Style for raster layers* are used for setting the default rendering style for output layers (that is, layers generated by processing algorithms). Just create the style you want using QGIS, save it to a file, and then enter the path to that file in the settings so the algorithms can use it. Whenever a layer is loaded by Processing and added to the QGIS canvas, it will be rendered with that style.

  Rendering styles can be configured individually for each algorithm and each one of its outputs. Just right-click on the name of the algorithm in the toolbox and select *Edit rendering styles for outputs*. You will see a dialog like the one shown next.



Слика23.5: Rendering Styles

  Select the style file (`.qml`) that you want for each output and press *OK*.
- *Warn before executing if parameter CRS's do not match*: By default, QGIS native algorithms (i.e. the ones listed under the *Menus* ► *QGIS (native C++)* group) transparently reproject the input layers to the first one's CRS before execution. Check this option to get a notification from the other tools that do not support reprojection, when the inputs CRS are not identical. Third-party providers are not concerned.

23.2.2 Menus

The  *Menus* block controls whether an algorithm, script or model (built-in or provided by plugins) should be made available through a dedicated menu or toolbar (along with the Processing Toolbox). For each item of each provider, you can:

- *Add button in toolbar*, making it available in the *Processing Algorithms* toolbar
- assign an *Icon* to the algorithm
- set a *Menu path*: the algorithm will then be available through an existing or a custom menu, e.g. `Vect&or/MyTopAlgorithms` will add a shortcut in a *MyTopAlgorithms* sub-menu of the default *Vector* menu.

Restart QGIS to apply the settings. At any time, your changes can be *Reset to defaults*.

23.2.3 Models and Scripts

In the  *Models* and  *Scripts* blocks, a default path, set under the active *User profile* folder, is provided for storing models and scripts respectively. As for the other options, you can modify the path, e.g., to lead to a remote or shared folder for all your users.

23.2.4 Providers

You will also find a block for algorithm  *Providers*. This is the place where installed providers expose their settings.

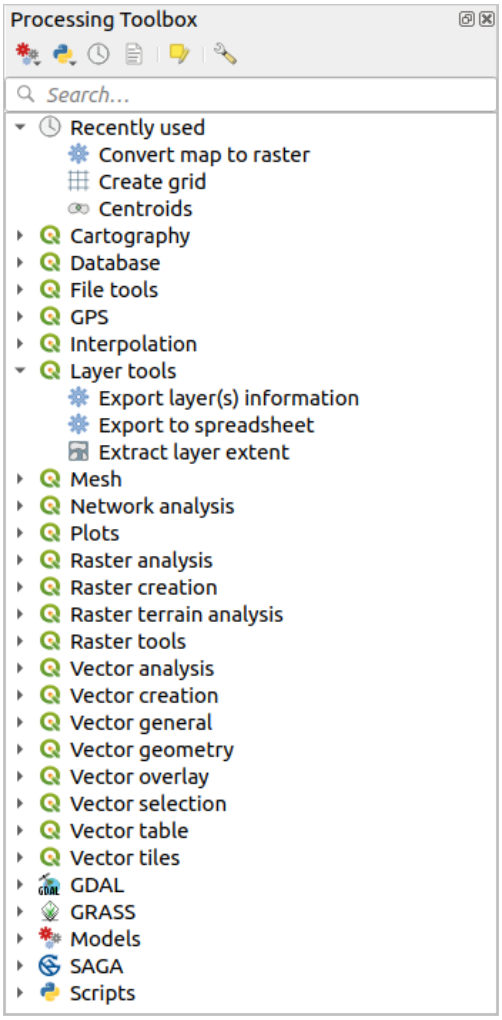
By default, QGIS is installed with two algorithm providers:

- *GDAL* whose algorithms you may *Activate* (or not) in this dialog. Read more about the *GDAL algorithms*.
- *GRASS* whose algorithms require the installation of the *GRASS GIS Processing Provider* from the *Plugins Manager*. You could then set some dedicated settings such as:
  -  *For raster layers, use r.external (faster) instead of r.in.gdal*
  -  *For vector layers, use v.external (faster) instead of v.in.ogr*
  - *Location of GRASS docs*: by default, and depending on your OS, QGIS utilizes a hardcoded list of paths for finding the local GRASS documentation and opening it whenever you hit the *Help* button of a GRASS algorithm dialog. Filling this option will override the default path, allowing you to use a custom location, or remote documentation.
  -  *Log console output*
  -  *Log execution commands*

Installed plugins that provide Processing algorithms may also have their provider listed in this group, with custom settings.

23.3 The Toolbox

The *Processing Toolbox* is the main element of the processing GUI, and the one that you are more likely to use in your daily work. It shows the list of all available **algorithms** grouped in different blocks called *Providers*, and custom **models** and **scripts** you can add to extend the set of tools. Hence the toolbox is the access point to run them, whether as a single process or as a batch process involving several executions of the same algorithm on different sets of inputs.



Слика23.6: Processing Toolbox

Providers can be (de)activated in the *Processing settings dialog*. By default, only providers that do not rely on third-party applications (that is, those that only require QGIS elements to be run) are active. Algorithms requiring external applications might need additional configuration. Configuring providers is explained in a *later chapter* in this manual.

In the upper part of the toolbox dialog, you will find a set of tools to:

- work with Models: *Create New Model…*, *Open Existing Model…* and *Add Model to Toolbox…*;
- work with Scripts: *Create New Script…*, *Create New Script from Template…*, *Open Existing Script…* and *Add Script to Toolbox…*;
- open the History panel;
- open the Results Viewer panel;
- toggle the toolbox to the *in-place modification mode* using the Edit Features In-Place button: only the algorithms that are suitable to be executed on the active layer without outputting a new layer are displayed;
- open the Options dialog.

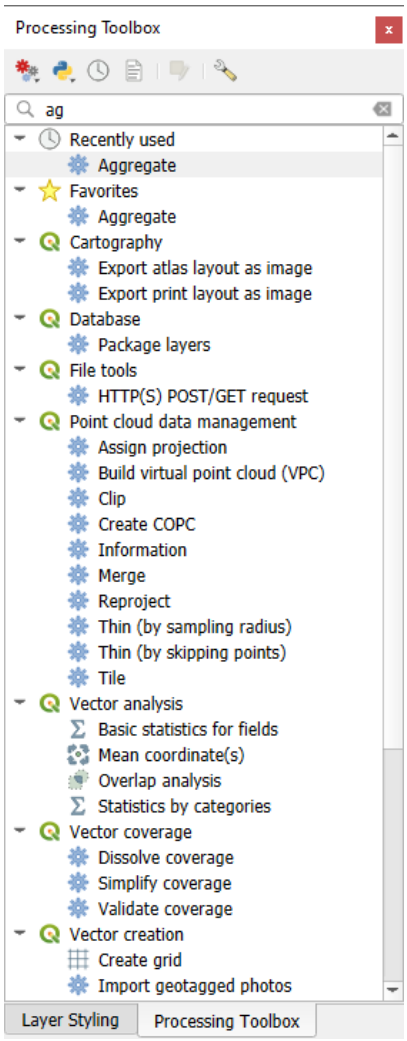
Below this toolbar is a *Search…* box to help you easily find the tools you need. You can enter any word or phrase on the text box. Notice that, as you type, the number of algorithms, models or scripts in the toolbox is reduced to just those that contain the text you have entered in their names or keywords.

The top of the algorithm list shows the most Recently Used tools and the Favorites tools, making it convenient to reexecute them. The Recently Used list is there by default, showing the algorithms you have recently executed. The Favorites list is created when you manually add an algorithm to it.

You can right-click on any algorithm in the toolbox and choose from the contextual menu to:

- *Execute…* to run the algorithm.
- *Execute as Batch Process…* to run the algorithm in batch mode.
- *Edit Rendering Styles for Outputs…* to customize the rendering styles for the algorithm's outputs.
- *Add to Favorites* to add the algorithm to the Favorites list.
- *Remove from Favorites* to remove the algorithm from the Favorites list.

Unlike the Recently Used list, the Favorites list remains static and is not affected by which algorithms were executed.

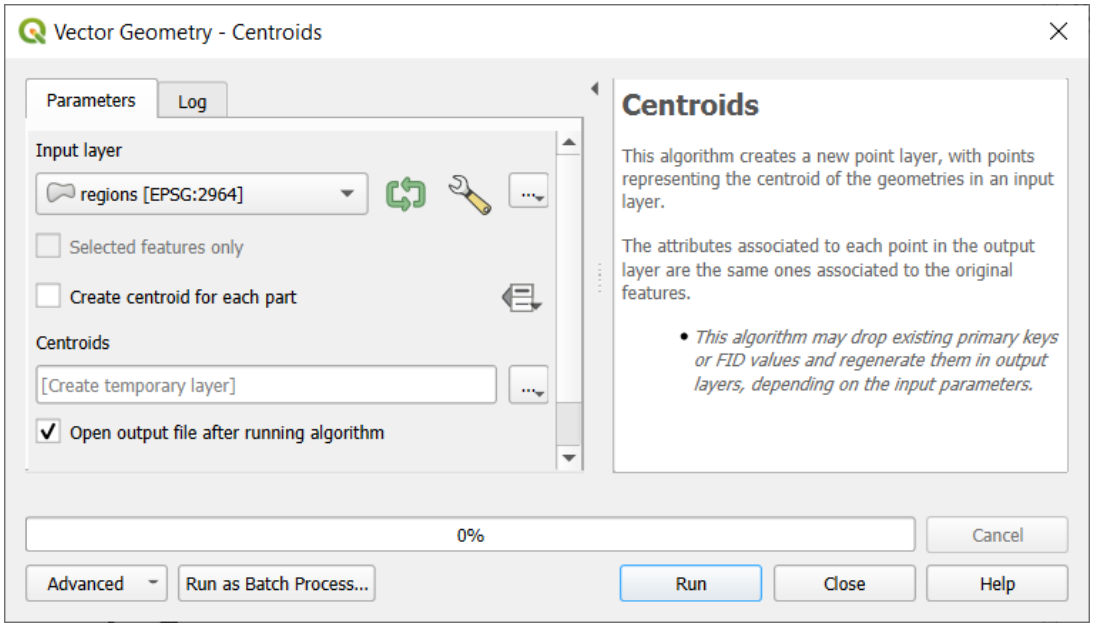


Слика23.7: Processing Toolbox showing search results

To execute a tool, just double-click on its name in the toolbox.

23.3.1 The algorithm dialog

Once you double-click on the name of the algorithm that you want to execute, a dialog similar to that in the Сл.23.8 below is shown (in this case, the dialog corresponds to the `Centroids` algorithm). Dialog title will include the group name from which the algorithm originates.



Слика23.8: Algorithm Dialog - Parameters

The dialog shows two tabs (*Parameters* and *Log*) on the left part, the algorithm description on the right, and a set of buttons at the bottom.

Parameter types

The *Parameters* tab is used to set the input values that the algorithm needs to be executed. It shows a list of input values and configuration parameters to be set. It of course has a different content, depending on the requirements of the algorithm to be executed, and is created automatically based on those requirements.

Савет

Setting your own default values for algorithm parameters

Algorithm dialogs open with some parameters prefilled with values from QGIS installation. It is however possible to set *your own default values* for specific algorithm parameters so that they are used at algorithm startup.

Although the number and type of parameters depend on the characteristics of the algorithm, the structure is similar for all of them. The parameters found in the table can be of one of the following types.

- A **vector layer**, to select from a list of all vector layers available (currently opened) in QGIS. You can also use unloaded layers: press the … button on the widget right-hand side, and select:
  - *Select file…*: selects file on disk using the Operating System file explorer
  - *Browse for layer…*: opens the *Browser panel*, allowing to take the layers directly from database sources (PostgreSQL, SQL Server, Oracle, …), web services (WFS, AFS, …) or files on disk.



Слика23.9: Vector input widget

Савет

Drag-and-drop as convenient alternative for layer selection

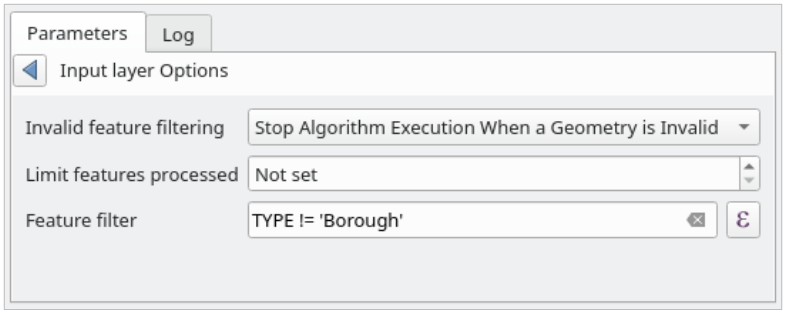
Besides selecting from the drop-down menu or browsing sources, you can also drag and drop a layer from the *Layers* or *Browser* panel, or from the Operating System file explorer directly into the combobox.

Белешка

By default, the layer widget shows the CRS of the layer along with its name. If you do not want to see this additional information, you can disable this functionality in the Processing Settings dialog, unchecking the *General* ► *Show layer CRS definition in selection boxes* option.

The vector input widget also has following features:

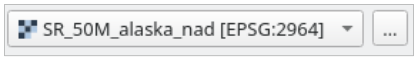
- an iterator button: If toggled, the algorithm will be executed iteratively on each one of its features, instead of just once for the whole layer, producing as many outputs as times the algorithm is executed. This allows for automating the process when all features in a layer have to be processed separately. If the algorithm contains several input vectors you can iterate over, the iteration will be processed only on the first toggled parameter, in the order parameters are declared in the algorithm.
- Advanced options button to adjust settings to use for that specific parameter. These settings concern:
  - *Invalid feature filtering*: allows the *default method* for handling features with invalid geometries to be overridden
  - *Limit features processed*: optional limit on number of features processed from the source
  - *Feature filter*: allows to enter an expression to subset the layer dynamically when running the tool, avoiding the need for separate steps to set layer filters or create layer subsets.



Слика23.10: Advanced options for vector input widget

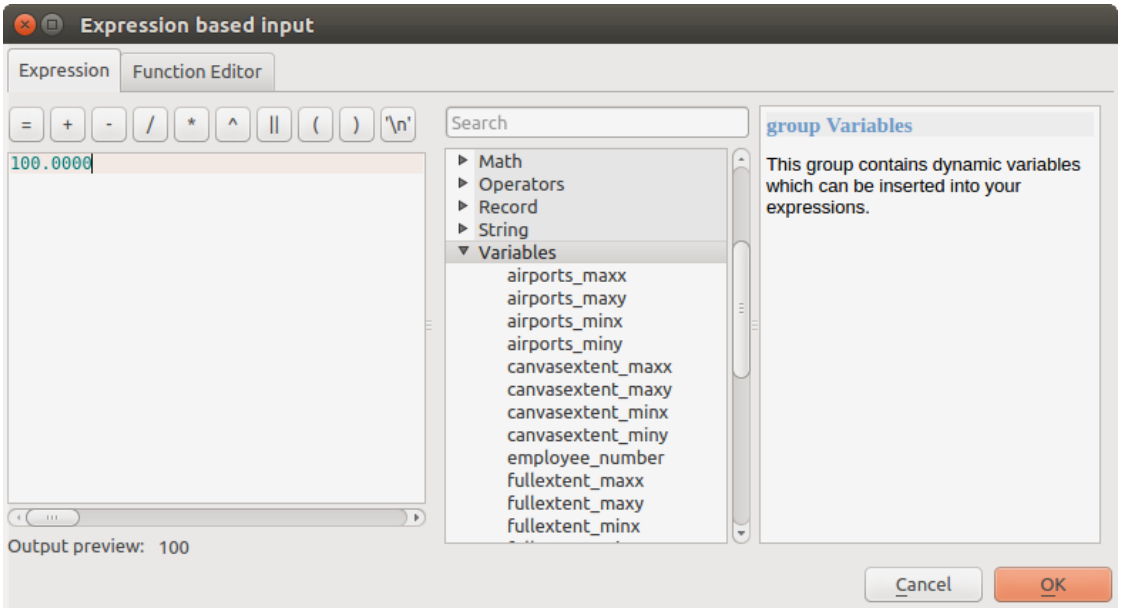
- It is also possible to limit the algorithm execution on the vector layer to its *Selected features only*.

- A **table**, to select from a list of all available in QGIS. Non-spatial tables are loaded into QGIS like vector layers, and use the *same widget*.
- A **raster layer**, to select from a list of all raster layers available in QGIS. The selector contains as well a … button on its right-hand side, to let you select filenames that represent layers currently not loaded in QGIS.



Слика23.11: Raster input widget

- An **option**, to choose from a selection list of possible options.
- A **numerical value**, to be introduced in a spin box. In some contexts (when the parameter applies at the feature level and not at the layer's), you will find a Data-defined override button by its side, allowing you to open the *expression builder* and enter a mathematical expression to generate variable values for the parameter. Some useful variables related to data loaded into QGIS can be added to your expression, so you can select a value derived from any of these variables, such as the cell size of a layer or the northernmost coordinate of another one.



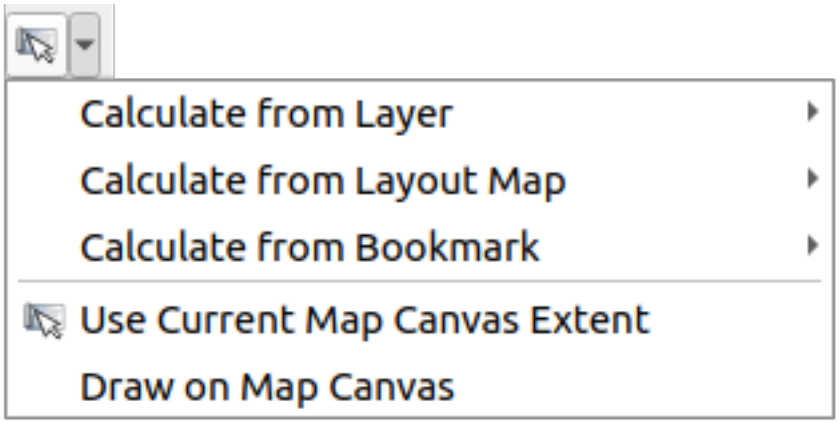
Слика23.12: Expression based input

- A **range**, with min and max values to be introduced in two text boxes.
- A **text string**, to be introduced in a text box.
- A **field**, to choose from the attributes table of a vector layer or a single table selected in another parameter.
- A **coordinate reference system**. You can select it among the recently used ones from the drop-down list or from the *CRS selection* dialog that appears when you click on the button on the right-hand side.

> **Савет**
>
> **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.

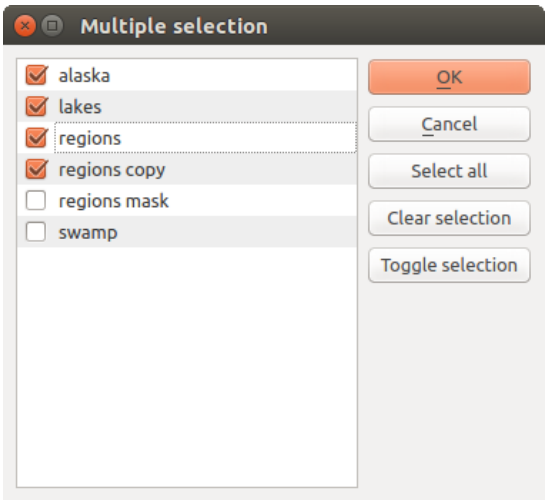
- An **extent**, a text box defining a rectangle through its corners coordinate in the format `xmin, xmax, ymin, ymax`. Press the Set to current map canvas extent button to use the map canvas extent. Clicking the arrow on the right-hand side of the value selector, a pop-up menu will appear, giving you options to:
  - *Calculate from layer* ►: fills the text box with the coordinates of the bounding box of a layer to select among the loaded ones

- *Calculate from layout map* ►: fills the text box with the coordinates of a map item selected from a layout in the current project
- *Calculate from bookmark* ►: fills the text box with the coordinates of a saved bookmark
- *Use current map canvas extent*
- *Draw on canvas*: the parameters window will hide itself, so you can click and drag onto the canvas. Once you have defined the extent rectangle, the dialog will reappear, containing the values in the extent text box.



Слика23.13: Extent selector

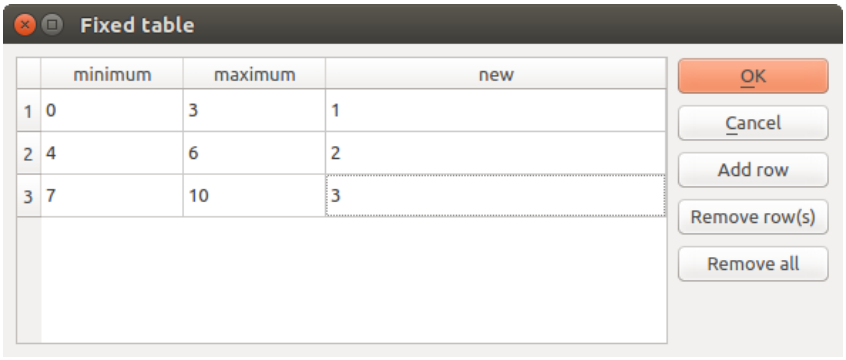
- A **list of elements** (whether raster or vector layers, tables, fields) to select from. Click on the … button at the left of the option to see a dialog like the following one. Multiple selection is allowed and when the dialog is closed, number of selected items is displayed in the parameter text box widget.



Слика23.14: Multiple Selection

- A **small table** to be edited by the user. These are used to define parameters like lookup tables or convolution kernels, among others.

  Click on the button on the right side to see the table and edit its values.



Слика23.15: Fixed Table

Depending on the algorithm, the number of rows can be modified or not by using the buttons on the right side of the window.

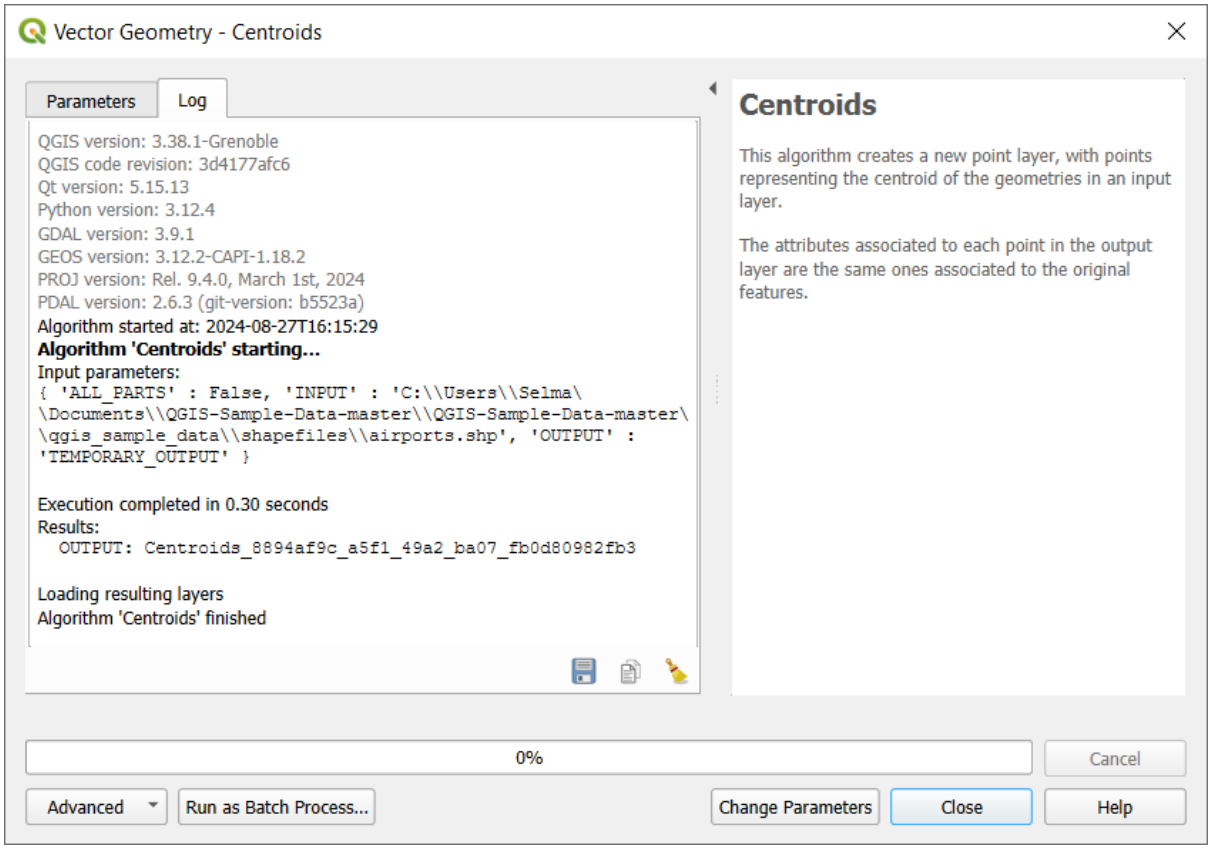
Белешка

Some algorithms require many parameters to run, e.g. in the *Rasterski kalkulator* you have to specify manually the cell size, the extent and the CRS. You can avoid to choose all the parameters manually when the algorithm has the `Reference layers` parameter. With this parameter you can choose the reference layer and all its properties (cell size, extent, CRS) will be used.

Logging the execution

Along with the *Parameters* tab, there is another tab named *Log* (see Сл.23.16 below). Information provided by the algorithm during its execution is written in this tab, allowing you to track the execution as well as being aware and having more details about the algorithm as it runs. You can directly click on the names of output files, folders, or HTML files listed in the log. Doing so will open the folder containing the generated file and automatically select it. Information on algorithm execution is also output in the *View* ► *Panels* ► *Log Messages Panel*.

Notice that not all algorithms write information to the *Log* tab, and many of them might run silently without producing any output other than the final files. Check the *Log Messages Panel* in that case.



Слика23.16: Algorithm Dialog - Log

At the bottom of the *Log* tab you will find buttons to *Save Log to File*, *Copy Log to Clipboard* and *Clear Log*. These are particularly handy when you have checked the *Keep dialog open after running algorithm* in the *General* part of the Processing options.

Other tools

On the right hand side of the dialog you will find a short description of the algorithm, which will help you understand its purpose and its basic ideas. If such a description is not available, the description panel will not be shown.

For a more detailed help file, which might include description of every parameter it uses, or examples, you will find a *Help* button at the bottom of the dialog bringing you to the *Processing algorithms documentation* or to the provider documentation (for some third-party providers).

The *Advanced* ► menu provides functions to reuse the configuration defined in the dialog without running the algorithm:

- *Algorithm Settings...*: allows to override processing settings for the current algorithm execution. More details at *Override algorithm settings*.
- *Copy as Python Command*: allows for easy copying of the equivalent *PyQGIS command* to run the tool using the parameters defined in the dialog
- *Copy as qgis_process Command*: allows for easy generation of *qgis_process command*, including its environment settings like the distance units, area units, ellipsoid, and any tricky parameter values like GeoPackage outputs with specific layers
- *Copy as JSON*: all the settings of the command are copied in a `JSON` format, ready to be consumed by qgis_process. This is a convenient way to see the expected format of the commands, even for complex

parameters (like TIN interpolation parameters). You can store these easily and then restore them later by pasting the values.

- *Paste Settings* in a `JSON` format

The *Run as Batch Process…* button triggers the *batch processing mode* allowing to configure and run multiple instances of the algorithm with a variety of parameters. A *Run as Single Process…* button helps you switch back from the batch mode.

When an algorithm execution finishes (either successfully or not), a new button *Change Parameters* is shown as long as the *Log* tab is active.

Override algorithm settings

Triggered from within the *Advanced* drop-down menu at the bottom of an algorithm dialog, the *Algorithm Settings…* shows a panel allowing users to control general processing settings which apply to that algorithm execution only. It is intended to be a place where a user can override their *global processing settings* on an ad-hoc basis without having to change their usual default settings.

Settings that can be overridden are:

- *Invalid feature filtering*: unlike the existing per-parameter setting override for this, setting the handling method here will apply to **ALL inputs** for the algorithm
- *Calculation settings*, such as *Distance units* and *Area units* to use for distance/area measurements
- *Environment settings*, such as *Temporary folder* and *Number of threads to use*

A note on projections

Processing algorithm execution are always performed in the input layer coordinate reference system (CRS). Due to QGIS's on-the-fly reprojecting capabilities, although two layers might seem to overlap and match, that might not be true if their original coordinates are used without reprojecting them onto a common coordinate system. Whenever you use more than one layer as input to a *QGIS native algorithm*, whether vector or raster, the layers will all be reprojected to match the coordinate reference system of the first input layer.

This is however less true for most of the external applications whose algorithms are exposed through the processing framework as they assume that all of the layers are already in a common coordinate system and ready to be analyzed.

By default, the parameters dialog will show a description of the CRS of each layer along with its name, making it easy to select layers that share the same CRS to be used as input layers. If you do not want to see this additional information, you can disable this functionality in the Processing settings dialog, unchecking the *Show layer CRS definition in selection boxes* option.

If you try to execute an algorithm using as input two or more layers with unmatching CRSs, a warning dialog will be shown. This occurs thanks to the *Warn before executing if layer CRS's do not match* option.

You still can execute the algorithm, but be aware that in most cases that will produce wrong results, such as empty layers due to input layers not overlapping.

Савет

Use Processing algorithms to do intermediate reprojection

When an algorithm can not successfully perform on multiple input layers due to unmatching CRSs, use QGIS internal algorithm such as *Reprojektovanje sloja* to perform layers' reprojection to the same CRS before executing the algorithm using these outputs.

23.3.2 Data objects generated by algorithms

Data objects generated by an algorithm can be of different types. They can be a layer (vector with or without geometry, raster, mesh, point cloud, …), stored as a plain file on disk or in a database, text-based files such as `qml` style file or graphics `.html` file, a folder, …

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.

The **output** parameter for setting the result of the algorithm execution provides a text box for filling the path to the output result, and a drop-down … button for additional options (the availability depends on the algorithm you run):

- *Skip Output*: if you are not interested in a given output of the algorithm
- *Create Temporary Layer* (`TEMPORARY_OUTPUT`): the output is stored in a vector *temporary scratch layer*.
- *Save to File…*: you will be prompted with a save file dialog, where you can select the desired file path. Supported file extensions are shown in the file format selector of the dialog, depending on the kind of output and the algorithm.

  The format of the output is defined by the filename extension. The supported formats depend on what is supported by the algorithm itself. To select a format, just select the corresponding file extension (or add it, if you are directly typing the file path instead). If the extension of the file path you entered does not match any of the supported formats, a default extension will be appended to the file path, and the file format corresponding to that extension will be used to save the layer or table. Default extensions are `.dbf` for tables, `.tif` for raster layers and `.gpkg` for vector layers. These can be modified in the setting dialog, selecting any other of the formats supported by QGIS.

  You can set a default folder for output data objects. Go to the settings dialog (you can open it from the *Settings* ► *Options* ► *Processing* menu), and in the *General* group, you will find a parameter named *Output folder*. This output folder is used as the default path in case you type just a filename with no path (i.e., `myfile.shp`) when executing an algorithm.
- *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*.
- *Save to Geopackage…*: the output layer will be saved as a named table of a new or existing GeoPackage database
- *Save to Database Table…*: the output layer will be saved as a new table in a database such as GeoPackage, SpatiaLite, PostgreSQL or MS SQL Server. A connection should already exist. You'll need to provide the name of the new table and depending on the type of target database, select a schema,
- *Append to Layer…*: adds features output by the algorithm to an existing layer. Upon selecting the target layer, a field mapping panel opens, allowing to manually set how fields from the output layer are mapped to the target layer's fields.

- *Save to Directory*: Use a new or existing directory for storing the algorithm outputs
- *Save to a Temporary Directory*: creates a directory within the set `TEMP` folder for storing the algorithm outputs. The files will be deleted once you exit QGIS.

When running an algorithm that uses a vector layer in iterative mode, the entered file path is used as the base path for all generated files, which are named using the base name and appending a number representing the index of the iteration. The file extension (and format) is used for all such generated files.

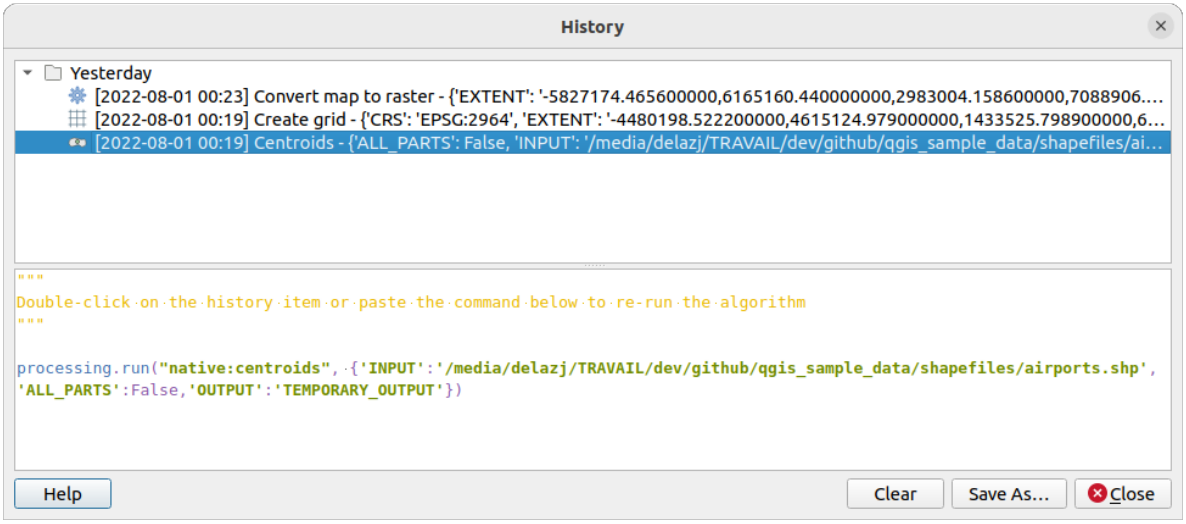
Algorithms also generate graphics and text as HTML files. These results are shown at the end of the algorithm execution in the *Results Viewer* dialog. This dialog will keep the results produced by any algorithm during the current session, and can be shown at any time by selecting *Processing* ► *Results Viewer* from the QGIS main menu.

For layer-based outputs, it is possible to *Open output file after running algorithm* directly in QGIS, as a new entry in the *Layers* panel. By default, the files are opened.

23.4 The history manager

23.4.1 The processing history

Every time you execute an algorithm, information about the process is stored in the history manager. The date and time of the execution are saved, along with the parameters used, making it is easy to track and control all the work that has been developed using the Processing framework, and to reproduce it.



Слика23.17: History

Process information is kept as a command-line expression, even if the algorithm was launched from the toolbox. This makes it useful for those learning how to use the command-line interface, since they can call an algorithm using the toolbox and then check the history manager to see how it could be called from the command line.

Right-click on a row and you should be able to:

- *Copy as Python Command*: allows for easy copying of the equivalent *PyQGIS command* run from the dialog. Same as the code displayed below the commands list.
- *Copy as qgis_process Command*: allows for easy generation of *qgis_process command*, including its environment settings like the distance units, area units, ellipsoid, and any tricky parameter values like GeoPackage outputs with specific layers
- *Copy as JSON*: all the settings of the command are copied in a `JSON` format, ready to be consumed by qgis_process. This is a convenient way to see the expected format of the commands, even for complex parameters (like TIN interpolation parameters). You can store these easily and then restore them later by pasting the values over an algorithm dialog.
- *Create Test...* using the concerned algorithm and parameters, following instructions at Processing README file.

Apart from browsing the entries in the registry, you can also re-execute processes by simply double-clicking on the entry. The algorithm dialog then opens with parameters already set, and you can change any of them to fit your needs and re-run the algorithm.

The *History* dialog also provides a convenient way to contribute to the consolidation of the testing infrastructure of QGIS Processing algorithms and scripts.

23.4.2 The processing log

The history dialog only contains the execution calls, but not the information produced by the algorithm when executed. That information is written to the QGIS log (*View* ► *Panels* ► *Log Messages Panel*).

Third-party algorithms are usually executed by using their command-line interfaces, which communicate with the user via the console. Although that console is not shown, usually a full dump of it is written to the log each time you run one of those algorithms. To avoid cluttering the log with that information, you can disable it for each provider in the settings dialog.

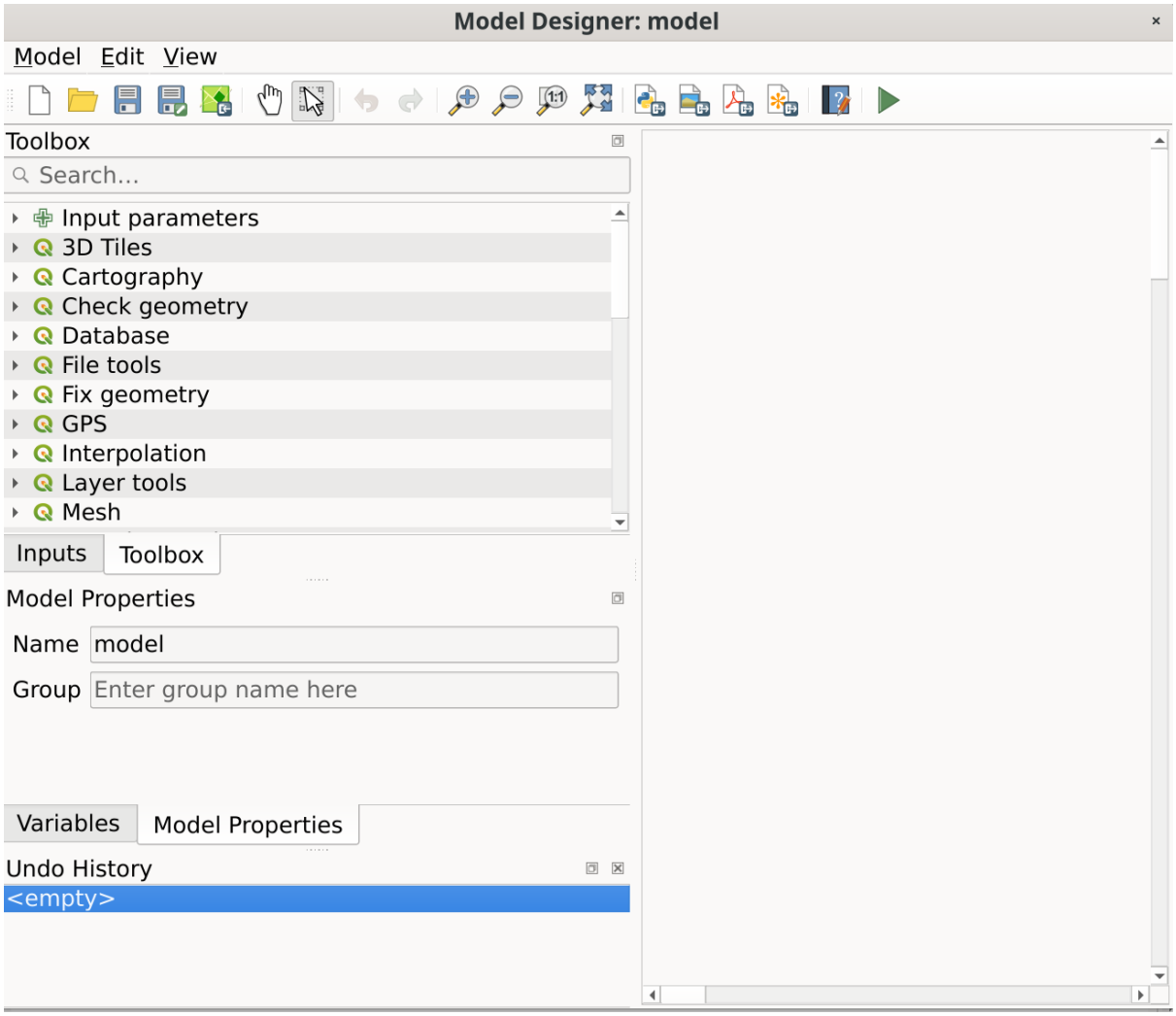
Some algorithms, even if they can produce a result with the given input data, output comments or additional information to log when they detect potential problems with the data, in order to warn you. Make sure you check those messages in the log if you get unexpected results.

23.5 Dizajner modela

Dizajner modela omogućava vam da pravite složene modele pomoću jednostavnog i lakog interfejsa. Pri radu sa GIS-om, većina operacija analize nije izolovana, već deo lanca operacija. Pomoću dizajnera modela, taj lanac operacija može se umotati u jedan proces, čineći zgodnim da se kasnije izvrši sa drugim skupom ulaza. Bez obzira na to koliko koraka i različitih algoritama uključuje, model se izvršava kao jedan algoritam, štedeći vreme i trud.

Dizajner modela može se otvoriti iz menija Obrada (*Obrada* ► *Dizajner modela*).

23.5.1 Interfejs dizajnera modela



Слика23.18: Dizajner modela

U svom glavnom delu, modeler ima radno platno na kojem se može konstruisati struktura modela i tok posla koji predstavlja.

Na vrhu dijaloga, različiti meniji i paleta alatki *Navigacija* daju pristup raznim alatkama.

Meni Model

Oznaka	Prečica	Paleta alatki Navigacija	Opis
Proveri model			Proverava da li algoritmi i ulazi korišćeni u modelu postoje. Zgodno pre objavljivanja modela.
Pokreni model…	`F5`	☑	Izvršava model
Pokreni izabrane korake…	`Shift+F5`		Pokreće samo izabrane korake u modelu, omogućavajući pokretanje podskupa modela
Preuredi ulaze modela…			Postavlja redosled kojim se ulazi predstavljaju korisniku u dijalogu algoritma.
Preuredi izlazne slojeve…			Postavlja određeni redosled koji izlazi modela moraju koristiti pri učitavanju rezultata u projekat.
Otvori model…	`Ctrl+O`	☑	Otvara `.model3` datoteku radi uređivanja ili izvršavanja
Sačuvaj model	`Ctrl+S`	☑	Čuva model na disk kao `.model3` datoteku
Sačuvaj model kao…	`Ctrl+Shif`	☑	Čuva model na disk kao novu `.model3` datoteku
Sačuvaj model u projektu		☑	Ugrađuje datoteku modela u datoteku projekta, čineći je dostupnom pri deljenju datoteke projekta.
Uredi pomoć modela…		☑	Interfejs za dokumentovanje modela, algoritama, parametara i izlaza, kao i autora i verzionisanja
Izvezi ►			
► *Izvezi kao sliku…*		☑	Čuva grafički dizajn modela u format datoteke slike (u ilustrativne svrhe)
► *Izvezi kao PDF…*			Čuva grafički dizajn modela u `PDF` format datoteke (u ilustrativne svrhe)
► *Izvezi kao SVG…*			Čuva grafički dizajn modela u `SVG` format datoteke (u ilustrativne svrhe)
► *Izvezi kao algoritam skriptu…*		☑	Generiše datoteku python skripte koja uključuje instrukcije modela

Meni Uređivanje

Oznaka	Prečica	Paleta alatki Navigacija	Opis
Izaberi sve	`Ctrl+A`		Bira sve komponente modela u dizajneru
Zalepi izabrane komponente za mrežu			lepi i poravnava elemente u mrežu
Ponovi	`Ctrl+Y`	☑	Poništava poslednju otkazanu radnju. Pogledajte i panel *Opozovi/ponovi*.
Poništi	`Ctrl+Z`	☑	Otkazuje prethodnu izmenu. Pogledajte i panel *Opozovi/ponovi*.
Iseci	`Ctrl+X`		Iseca izbor komponenata iz modela.
Kopiraj	`Ctrl+C`		Kopira izbor komponenata iz modela.
Nalepi	`Ctrl+V`		Nalepi isečen ili kopiran izbor komponenata iz jednog modela u drugi ili unutar istog modela. Izabrane komponente zadržavaju svoja originalna svojstva i komentare.
Obriši izabrane komponente	`Del`		Uklanja komponentu iz modela.
Dodaj okvir grupe			Dodaje okvir u pozadini povezanih komponenata radi njihovog vizuelnog grupisanja. Naročito korisno u velikim modelima za održavanje toka posla čistim.

Meni Prikaz

Oznaka	Prečica	Paleta alatki Navigacija	Opis
Uveličaj na ►			Uveličava na obuhvat izabranog okvira grupe
Uvećaj	`Ctrl++`	☑	
Umanji	`Ctrl+-`	☑	
Uveličaj na 100%	`Ctrl+1`	☑	
Uvećaj na pun obuhvat	`Ctrl+0`	☑	Prikazuje sve komponente na trenutnom platnu dizajnera
Prikaži komentare			Prikazuje komentare povezane sa svakim algoritmom ili ulazom u dizajneru modela
Omogući lepljenje			
Uključi/isključi vidljivost panela	`Ctrl+Tab`		Uključuje ili isključuje *panele* u dizajneru

Paneli

Levi deo prozora je odeljak sa pet panela koji se mogu koristiti za dodavanje novih elemenata u model:

1. *Ulazi*: svi *ulazni parametri* koji mogu oblikovati vaš model
2. *Kutija sa alatkama*: ovde možete pronaći dostupne *algoritme obrade*. Uključuje i ulazne parametre. Koristite polje *Pretraga…* za pronalaženje algoritama ili ulaza.

3. *Svojstva modela*: zadajte naziv (obavezno) modela i grupu u kojoj će biti prikazan u *kutiji sa alatkama za obradu*
4. *Promenljive*: Modeli mogu sadržati posebne *promenljive* koje su jedinstvene i dostupne samo njima. Ovim promenljivama može pristupiti bilo koji izraz korišćen unutar modela. Korisne su za kontrolu algoritama unutar modela i kontrolu više aspekata modela promenom jedne promenljive. Promenljive se mogu pregledati i menjati u panelu *Promenljive*.
5. *Istorija opoziva*: ovaj panel će registrovati sve što se dešava u modeleru, olakšavajući otkazivanje stvari koje ste pogrešno uradili.

O dostupnim algoritmima

Neki algoritmi koji se mogu izvršiti iz *Obrada* ► *Kutija sa alatkama* ne pojavljuju se u listi dostupnih algoritama kada dizajnirate model. Da bi bio uključen u model, algoritam mora imati ispravnu semantiku. Ako algoritam nema takvu dobro definisanu semantiku (na primer, ako se broj izlaznih slojeva ne može znati unapred), onda ga nije moguće koristiti unutar modela, i neće se pojaviti u listi algoritama koju možete pronaći u dijalogu modelera. S druge strane, neki algoritmi su specifični za modeler. Ti algoritmi nalaze se unutar grupe „Alati za modelovanje“.

23.5.2 Pravljenje modela

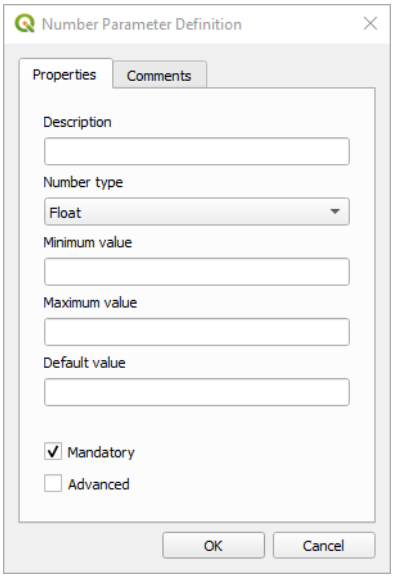
Pravljenje modela podrazumeva dva osnovna koraka:

1. *Definisanje potrebnih ulaza*. Ovi ulazi biće dodati u prozor parametara, tako da korisnik može postaviti njihove vrednosti pri izvršavanju modela. Sam model je algoritam, tako da se prozor parametara generiše automatski kao za sve algoritme dostupne u okviru za obradu.
2. *Definisanje toka posla*. Korišćenjem ulaznih podataka modela, tok posla se definiše dodavanjem algoritama i izborom kako oni koriste definisane ulaze ili izlaze generisane drugim algoritmima u modelu.

Definisanje ulaza

Prvi korak je definisanje ulaza za model. Oni se nalaze u panelu *Ulazi* sa leve strane prozora modelera. Zadržavanje pokazivača miša iznad ulaza prikazaće iskačuću pomoć sa dodatnim informacijama. Za potpunu listu dostupnih parametara u modeleru i njihovu odgovarajuću upotrebu za skriptovanje, pročitajte *Ulazni i izlazni tipovi za algoritme Obrade*.

Pri dvokliku na element, prikazuje se dijalog koji vam omogućava da definišete njegove karakteristike. U zavisnosti od parametra, dijalog će sadržati bar jedan element (opis, što je ono što će korisnik videti pri izvršavanju modela). Na primer, pri dodavanju numeričke vrednosti, kao što se vidi na sledećoj slici, pored opisa parametra morate postaviti podrazumevanu vrednost i opseg ispravnih vrednosti.



Слика23.19: Definisanje parametara modela

Svoj ulaz možete definisati kao obavezan za svoj model označavanjem opcije `Mandatory`, a označavanjem polja `Advanced` možete postaviti da ulaz bude unutar odeljka `Advanced`. Ovo je naročito korisno kada model ima mnogo parametara i neki od njih nisu trivijalni, ali ipak želite da ih birate.

Za svaki dodati ulaz, novi element se dodaje na platno modelera.



Слика23.20: Parametri modela

Ulaze možete dodati i prevlačenjem tipa ulaza iz liste i ispuštanjem na položaj gde ga želite na platnu modelera. Ako želite da promenite parametar postojećeg ulaza, samo dvokliknite na njega i pojaviće se isti dijalog.

Pri korišćenju modela unutar drugog modela, potrebni ulazi i izlazi biće prikazani na platnu.

Definisanje toka posla

U sledećem primeru dodaćemo dva ulaza i dva algoritma. Cilj modela je da kopira vrednosti visina sa DEM rastersko sloja na linijski sloj pomoću algoritma `Drape`, a zatim izračuna ukupan uspon linijskog sloja pomoću algoritma `Climb Along Line`.

U kartici *Ulazi* izaberite dva ulaza kao `Vector Layer` za liniju i `Raster Layer` za DEM. Sada smo spremni da dodamo algoritme u tok posla.

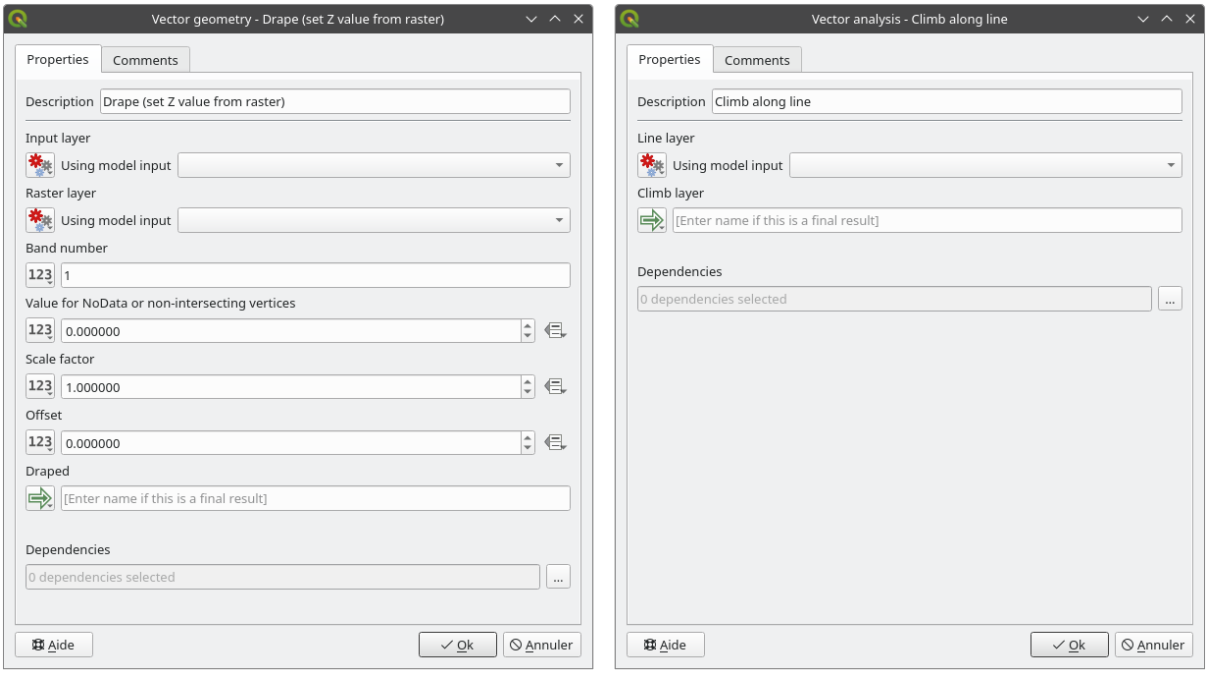
Algoritmi se mogu pronaći u panelu *Kutija sa alatkama*, grupisani u velikoj meri na isti način kao u kutiji sa alatkama za obradu.



Слика23.21: Ulazi modela

Da biste dodali algoritam u model, dvokliknite na njegov naziv ili ga prevucite i ispustite, baš kao za ulaze. Kao i za ulaze, možete promeniti opis algoritma i dodati komentar. Pri dodavanju algoritma, pojaviće se dijalog izvršavanja,

sa sadržajem sličnim onom u panelu izvršavanja koji se prikazuje pri izvršavanju algoritma iz kutije sa alatkama. Sledeća slika prikazuje dijaloge oba algoritma, `Drape (set Z value from raster)` i `Climb along line`.

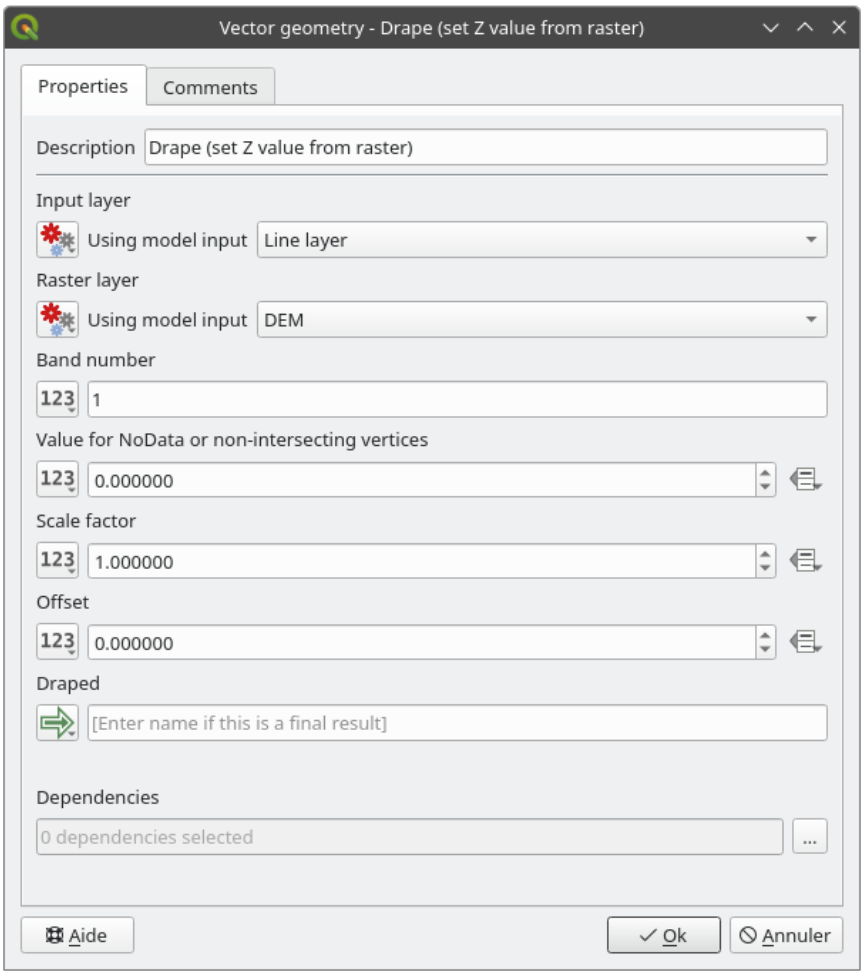


Слика23.22: Parametri algoritma modela

Kao što vidite, ipak postoje neke razlike. Svaki parametar ima padajući meni pored sebe koji omogućava kontrolu kako će biti opslužen tokom toka posla:

- Vrednost: omogućava vam da parametru dodelite statičku vrednost. U zavisnosti od tipa parametra, vidžet će vam omogućiti da unesete broj (`5.0`), nisku (`mytext`), izaberete sloj(eve) učitane u QGIS projektu ili iz foldera, izaberete stavke iz liste, …
- Unapred izračunata vrednost: otvara dijalog *Graditelj izraza* i omogućava vam da definišete izraz za popunjavanje parametra. Ulazi modela zajedno sa nekim drugim statistikama slojeva dostupni su kao **promenljive** i navedeni su na vrhu dijaloga Pretraga Graditelja izraza. Izraz se izračunava jednom pre izvršavanja podređenog algoritma i koristi tokom izvršavanja tog algoritma.
- Ulaz modela: omogućava korišćenje ulaza dodatog u model kao parametra. Kada se klikne, ova opcija će navesti sve pogodne ulaze za parametar.
- Izlaz algoritma: omogućava korišćenje izlaza drugog algoritma kao ulaza trenutnog algoritma. Kao i za ulaze modela, ova opcija će navesti sve pogodne ulaze za parametar.
- **Izlazni parametar** takođe ima gore navedene opcije u svom padajućem meniju:
  - dodavanje statičkih izlaza za podređene algoritme, npr. uvek čuvanje izlaza podređenog algoritma u unapred definisan geopackage ili postgres sloj
  - korišćenje izlaznih vrednosti zasnovanih na izrazu za podređene algoritme, npr. generisanje automatskog naziva datoteke na osnovu današnjeg datuma i čuvanje izlaza u tu datoteku
  - korišćenje ulaza modela, npr. ulaza modela *Datoteka/folder* za zadavanje izlazne datoteke ili foldera
  - korišćenje izlaza drugog algoritma, npr. izlaza algoritma *Napravi direktorijum* (iz *Alati za modelovanje*)
  - dodatna opcija Izlaz modela čini izlaz algoritma dostupnim u modelu. Ako se sloj generisan algoritmom koristi samo kao ulaz za drugi algoritam, nemojte uređivati to tekstualno polje.

Na sledećoj slici možete videti dva ulazna parametra definisana kao `Model Input` i privremeni izlazni sloj:



Слика23.23: Ulazni i izlazni parametri algoritma

Naći ćete i dodatni parametar pod nazivom *Zavisnosti* koji nije dostupan pri pozivanju algoritma iz kutije sa alatkama. Ovaj parametar omogućava vam da definišete redosled kojim se algoritmi izvršavaju, eksplicitnim definisanjem jednog algoritma kao *roditelja* trenutnog. Time se prisiljava *roditeljski* algoritam da se izvrši pre trenutnog.

Kada koristite izlaz prethodnog algoritma kao ulaz svog algoritma, to implicitno postavlja prethodni algoritam kao roditelja trenutnog (i postavlja odgovarajuću strelicu na platnu modelera). Međutim, u nekim slučajevima algoritam može zavisiti od drugog čak i ako ne koristi nijedan izlazni objekat iz njega (na primer, algoritam koji izvršava SQL naredbu nad PostgreSQL bazom podataka i drugi koji uvozi sloj u tu istu bazu podataka). U tom slučaju, samo izaberite prethodni algoritam u parametru *Zavisnosti* i oni će se izvršiti ispravnim redosledom.

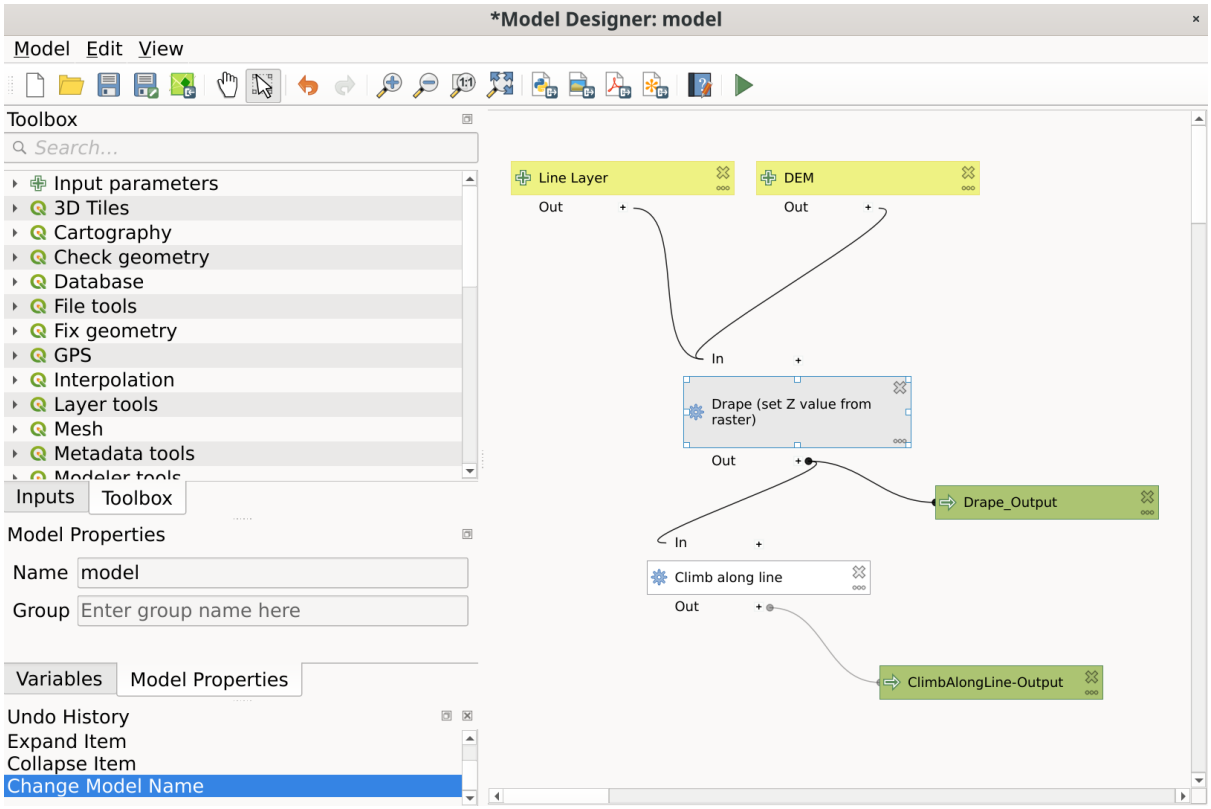
Kada su svim parametrima dodeljene ispravne vrednosti, kliknite na *OK* i algoritam će biti dodat na platno. Biće povezan sa elementima na platnu (algoritmima ili ulazima) koji pružaju objekte koji se koriste kao ulazi za algoritam.

Elementi se mogu prevući na drugi položaj na platnu pomoću alatke  Izaberi/pomeri/poveži stavku. Ovo je korisno da bi struktura modela bila jasnija i intuitivnija. Elementima možete promeniti veličinu, hvatajući njihovu ivicu. Ovo je naročito korisno ako je opis ulaza ili algoritma dug. Sa označenom opcijom *Prikaz* ► *Omogući lepljenje*, promena veličine ili pomeranje stavki može se vezati za virtuelnu mrežu, radi vizuelno strukturiranijeg dizajna algoritma.

Veze između elemenata se automatski ažuriraju i možete videti dugme + na vrhu i na dnu svakog algoritma. Klik na dugme će navesti sve ulaze i izlaze algoritma tako da možete imati brz pregled.

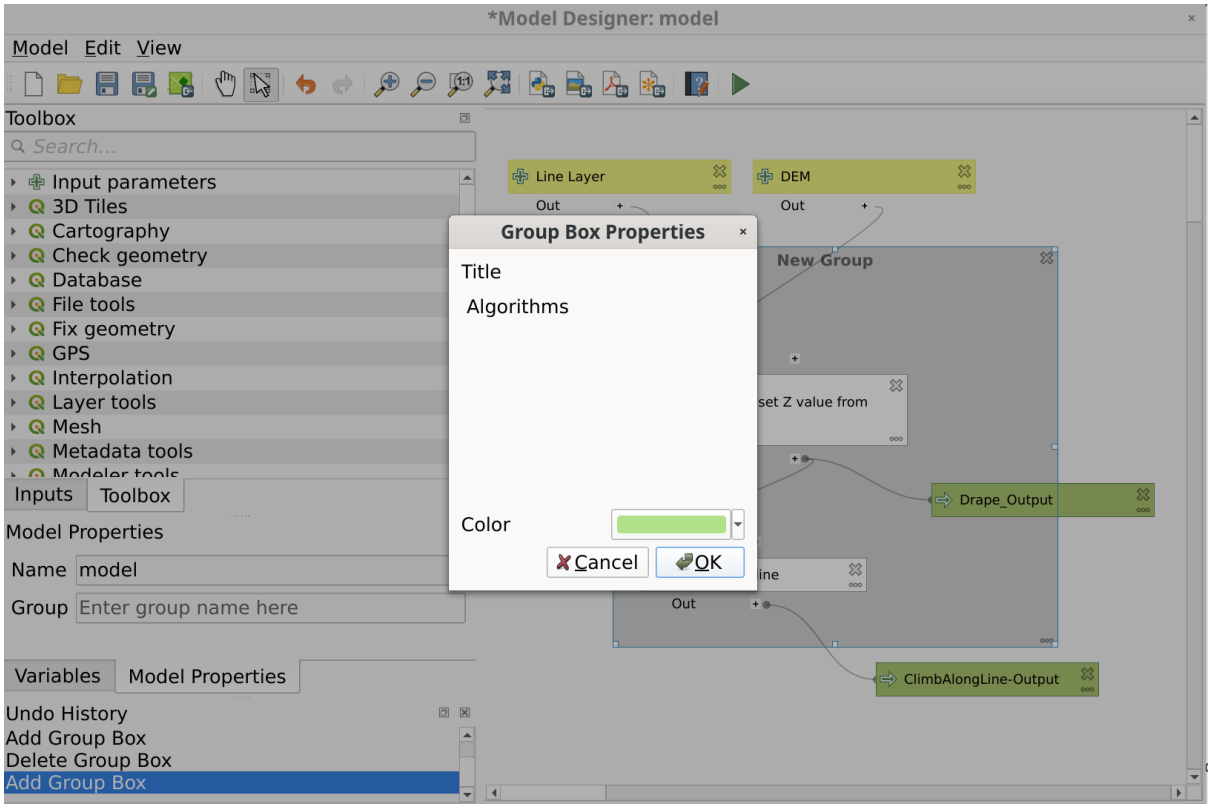
Veze između komponenata modela možete praviti ili uklanjati pomoću interaktivnog interfejsa prevlačenja. Svaka stavka parametra i algoritma na platnu modela uključuje male kružne priključke, izlazne priključke sa desne strane

i ulazne priključke sa leve strane. Da biste povezali komponente, kliknite i prevucite od izlaznog priključka (bilo od ulaza modela ili algoritma) do ulaznog priključka na drugom algoritmu. Vizuelna ivica prati pokazivač i završava vezu pri otpuštanju. Da biste uklonili vezu, odvucite ivicu od ulaznog priključka. Dijalozi algoritama se automatski ažuriraju da odraze izmene napravljene pomoću interfejsa prevlačenja.



Слика23.24: Kompletan model

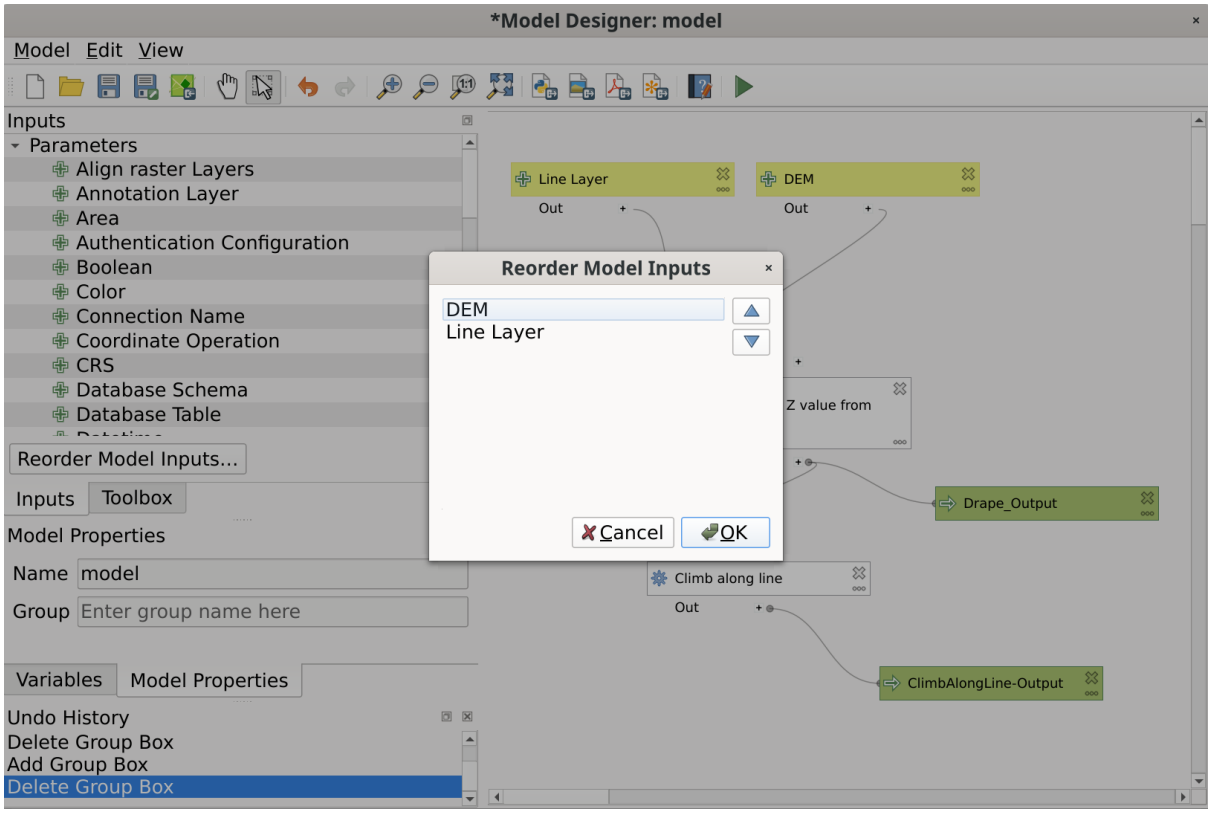
Alatkom *Uređivanje* ► *Dodaj okvir grupe* možete dodati *okvir* koji se može prevlačiti na platno. Ova mogućnost je veoma korisna u velikim modelima za grupisanje povezanih elemenata na platnu modelera i održavanje toka posla čistim. Na primer, mogli bismo grupisati sve algoritme iz primera:



Слика23.25: Okvir grupe modela

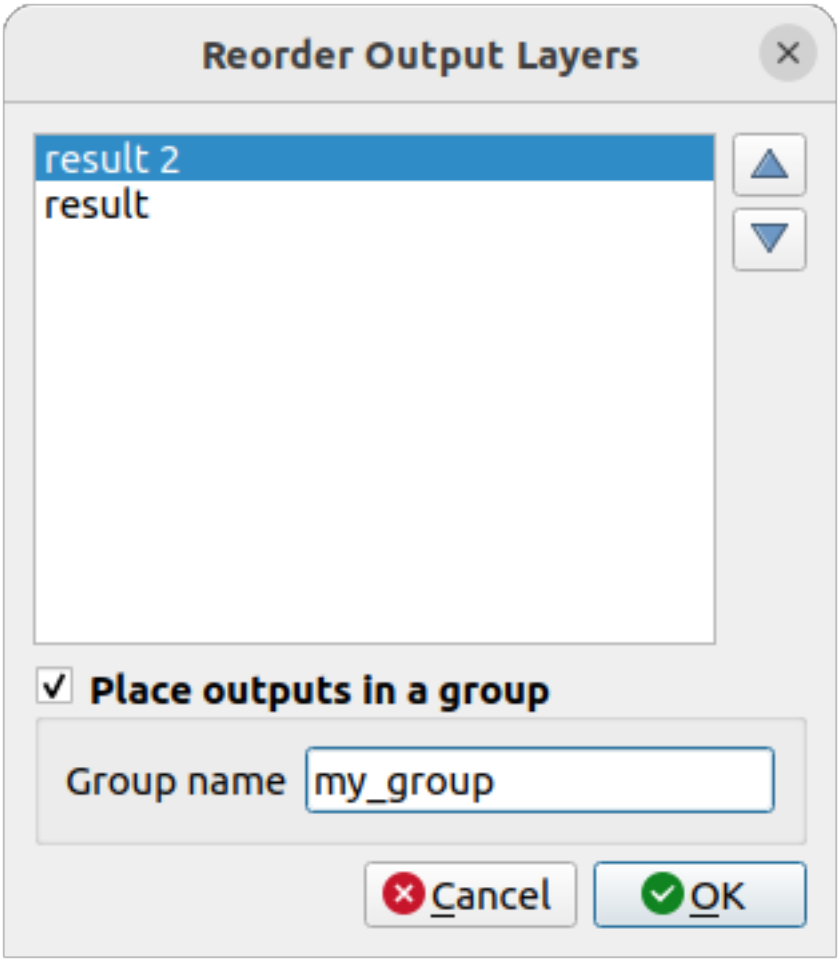
Možete promeniti naziv i boju okvira. Okviri grupe su veoma korisni kada se koriste zajedno sa alatkom *Prikaz ► Uveličaj na ►*, omogućavajući vam da uveličate na određeni deo modela. Takođe možete uveličavati i umanjivati pomoću točkića miša.

Možda ćete želeti da promenite redosled ulaza i način na koji su navedeni u glavnom dijalogu modela. Na dnu panela `Ulaz` naći ćete dugme `Preuredi ulaze modela...` i klikom na njega pojaviće se novi dijalog koji vam omogućava da promenite redosled ulaza:



Слика23.26: Preuredi ulaze modela

Postoji i mogućnost postavljanja određenog redosleda koji izlazi modela moraju koristiti pri učitavanju rezultata u projekat. Time se tvorcu modela daje način da obezbedi da su slojevi logično poređani na platnu pri pokretanju modela, kao što je postavljanje izlaza vektorskog sloja iznad izlaza rasterskog sloja, ili tačkastog sloja iznad poligonskog sloja. Tvorac modela može takođe postaviti opcioni „Naziv grupe“ za izlaze radi automatskog grupisanja izlaza unutar stabla slojeva pomoću novog naziva grupe ili njihovim dodavanjem u postojeću grupu. U meniju `Model` naći ćete unos `Preuredi izlazne slojeve...` i klikom na njega pojaviće se novi dijalog koji vam omogućava da promenite redosled izlaznih slojeva:



Слика23.27: Preuredi izlazne slojeve

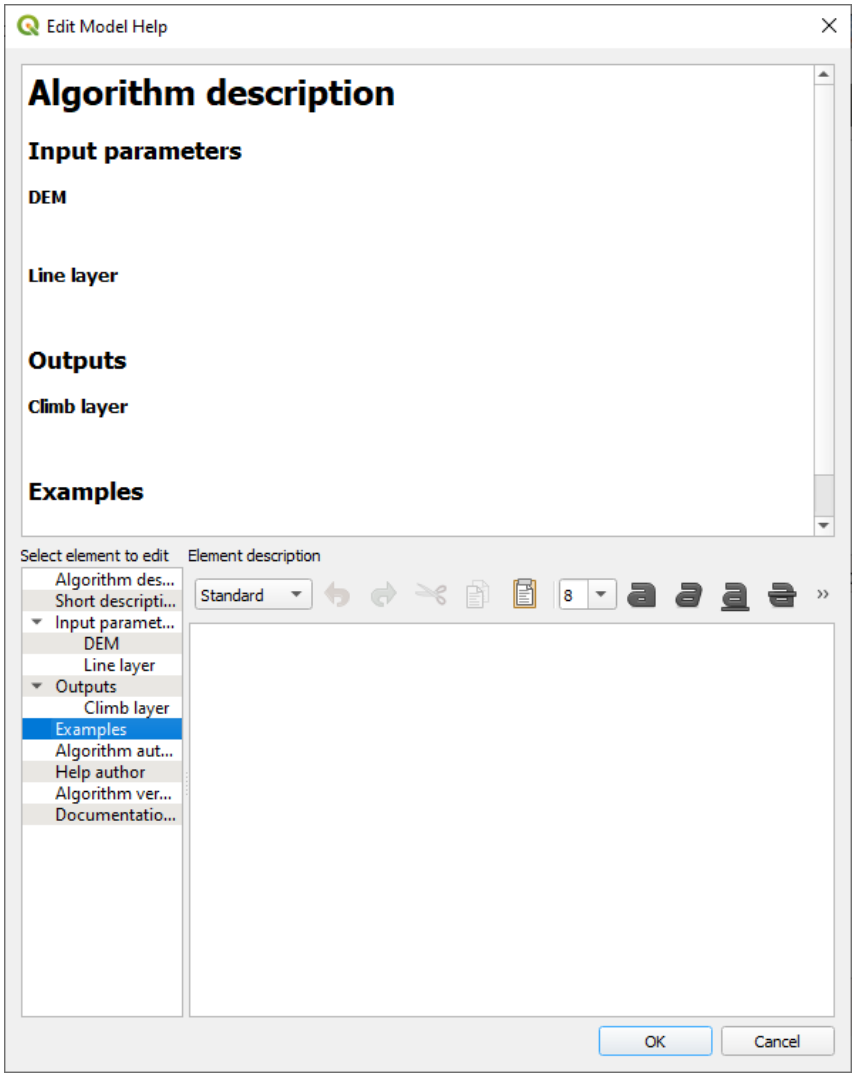
Komentari se takođe mogu dodati ulazima ili algoritmima prisutnim u modeleru. To se može uraditi odlaskom u karticu *Komentar* stavke ili desnim klikom. U istoj kartici može se ručno postaviti boja za pojedinačne komentare modela. Komentari su vidljivi samo na platnu modelera, a ne u konačnom dijalogu algoritma; mogu se sakriti deaktiviranjem *Prikaz* ► *Prikaži komentare*.

Vaš model se može pokrenuti na razne načine:

- Ceo model možete pokrenuti klikom na Pokreni model… iz palete alatki, desnim klikom na model u *panelu Pregledač* ili *Model* ► *Pokreni model…*: pri korišćenju uređivača za izvršavanje modela, sve vrednosti različite od podrazumevanih biće sačuvane u ulazima. To znači da će izvršavanje modela kasnije iz uređivača imati dijalog unapred popunjen tim vrednostima pri svakom narednom pokretanju.
- Možete izabrati elemente modela i pokrenuti samo taj podskup modela: pritisnite opciju *Pokreni izabrane korake…* iz menija *Modeli* ► ili iz kontekstnog menija izabranog algoritma. Početno stanje biće uzeto iz bilo kojih prethodnih izvršavanja modela kroz uređivač, tako da su rezultati prethodnih koraka u modelu dostupni za izabrane korake. Time se omogućava ispravljanje delova velikog modela, bez potrebe za stalnim pokretanjem celog modela radi testiranja. Naročito korisno kada su raniji koraci u modelu vremenski zahtevni.
- Možete pokrenuti podskup modela, počev od određenog algoritma: kliknite desnim tasterom na algoritam i izaberite *Pokreni odavde…*. Isto tako, vrednosti se uzimaju iz prethodnih izvršavanja.
- Da biste koristili algoritam iz kutije sa alatkama za obradu, mora biti sačuvan i dijalog modelera zatvoren, da bi kutija sa alatkama mogla da osveži svoj sadržaj.

Dokumentovanje modela

Potrebno je da dokumentujete svoj model, a to se može uraditi iz samog modelera. Kliknite na dugme Uredi pomoć modela i pojaviće se dijalog kao onaj prikazan u nastavku.



Слика23.28: Uređivanje pomoći

Sa desne strane videćete jednostavnu HTML stranicu, napravljenu korišćenjem opisa ulaznih parametara i izlaza algoritma, zajedno sa nekim dodatnim stavkama kao što je opšti opis modela ili njegov autor. Takođe, postoji odeljak Primer gde možete uneti sopstvene prilagođene primere radi objašnjenja upotrebe modela. Prvi put kada otvorite uređivač pomoći, svi ovi opisi su prazni, ali ih možete uređivati pomoću elemenata sa leve strane dijaloga. Izaberite element u gornjem delu, a zatim napišite njegov opis u tekstualnom polju ispod.

Pomoć modela se čuva kao deo samog modela.

23.5.3 Čuvanje i učitavanje modela

Čuvanje modela

Koristite dugme Sačuvaj model da biste sačuvali trenutni model i dugme Otvori model da biste otvorili prethodno sačuvan model. Modeli se čuvaju sa ekstenzijom `.model3`. Ako je model već sačuvan iz prozora modelera, neće vam biti zatražen naziv datoteke. Pošto već postoji datoteka povezana sa modelom, ta datoteka će se koristiti za naredna čuvanja.

Pre čuvanja modela, morate uneti naziv i grupu za njega u tekstualnim poljima u gornjem delu prozora.

Modeli sačuvani u folderu `models` (podrazumevani folder kada vam se zatraži naziv datoteke za čuvanje modela) pojaviće se u kutiji sa alatkama u odgovarajućoj grani. Kada se kutija sa alatkama pozove, ona pretražuje folder `models` za datoteke sa ekstenzijom `.model3` i učitava modele koje sadrže. Pošto je model sam algoritam, može se dodati u kutiju sa alatkama baš kao bilo koji drugi algoritam.

Modeli se takođe mogu sačuvati unutar datoteke projekta pomoću dugmeta Sačuvaj model u projektu. Modeli sačuvani ovim metodom neće biti upisani kao `.model3` datoteke na disku, već će biti ugrađeni u datoteku projekta.

Modeli projekta dostupni su u meniju *Modeli projekta* kutije sa alatkama i u stavci menija *Projekat* ► *Modeli*.

Folder modela može se postaviti iz dijaloga konfiguracije Obrade, pod grupom *Modeler*.

Modeli učitani iz foldera `models` pojavljuju se ne samo u kutiji sa alatkama, već i u stablu algoritama u kartici *Algoritmi* prozora modelera. To znači da model možete ugraditi kao deo većeg modela, baš kao druge algoritme.

Modeli će se pojaviti u panelu *Pregledač* i mogu se pokrenuti odatle.

Izvoz modela kao Python skripte

Kao što ćemo videti u kasnijem poglavlju, algoritmi obrade mogu se pozvati iz QGIS Python konzole, a novi algoritmi obrade mogu se praviti pomoću Pythona. Brz način za pravljenje takve Python skripte jeste pravljenje modela, a zatim njegov izvoz kao Python datoteke.

Da biste to uradili, kliknite na Izvezi kao algoritam skriptu… na platnu modelera ili kliknite desnim tasterom na naziv modela u kutiji sa alatkama za obradu i izaberite Izvezi model kao Python algoritam….

Izvoz modela kao slike, PDF-a ili SVG-a

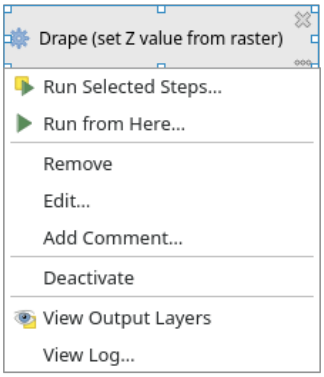
Model se takođe može izvesti kao slika, SVG ili PDF (u ilustrativne svrhe) klikom na Izvezi kao sliku, Izvezi kao PDF ili Izvezi kao SVG.

23.5.4 Uređivanje modela

Možete uređivati model koji trenutno pravite, redefinišući tok posla i odnose između algoritama i ulaza koji definišu model.

Postojeći model takođe možete uređivati iz *panela Pregledač* izborom opcije *Uredi model…*.

Ako kliknete desnim tasterom na algoritam na platnu, videćete kontekstni meni kao onaj prikazan u nastavku:



Слика23.29: Desni klik u modeleru

- Opcije za *pokretanje dela modela* izložene su ranije.
- Izbor opcije *Ukloni* uzrokovaće uklanjanje izabranog algoritma. Algoritam se može ukloniti samo ako nema drugih algoritama koji zavise od njega. To jest, ako se nijedan izlaz iz algoritma ne koristi kao ulaz u drugom.

- Izbor opcije *Uredi…* prikazaće dijalog parametara tako da možete promeniti ulaze i vrednosti parametara. Neće se svi ulazni elementi dostupni u modelu pojaviti kao dostupni ulazi. Slojevi ili vrednosti generisani u naprednijem koraku toka posla definisanog modelom neće biti dostupni ako uzrokuju kružne zavisnosti.

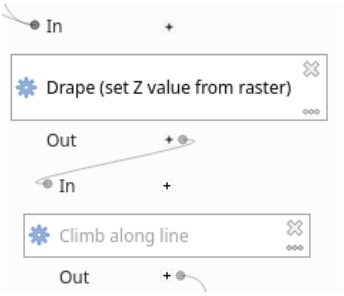
  Izaberite nove vrednosti i kliknite na dugme *OK* kao i obično. Veze između elemenata modela promeniće se na platnu modelera u skladu s tim.

- *Dodaj komentar…* omogućava vam da dodate komentar algoritmu radi boljeg opisa ponašanja.

Белешка

Desni klik na ulazni parametar takođe će vam omogućiti *Ukloni*, *Uredi…* i *Dodaj komentar…*

- Model se može pokrenuti delimično deaktiviranjem nekih njegovih algoritama. Da biste to uradili, izaberite opciju *Deaktiviraj* u kontekstnom meniju koji se pojavljuje pri desnom kliku na element algoritma. Izabrani algoritam i svi oni u modelu koji zavise od njega biće prikazani sivo i neće se izvršavati kao deo modela.



Слика23.30: Model sa deaktiviranim algoritmom

  Pri desnom kliku na algoritam koji nije aktivan, videćete opciju menija *Aktiviraj* koju možete koristiti da biste ga ponovo aktivirali.

- Pri uređivanju modela kroz dizajner (i nakon što ste pokrenuli taj model), desni klik na bilo koji podređeni korak u modelu omogućava izbor  *Prikaži izlazne slojeve*. Time se izlazni slojevi iz tog koraka dodaju kao novi slojevi u trenutni QGIS projekat.

  Ova radnja je dostupna za sve podređene algoritme u modelu, čak i ako model nije konfigurisan da koristi izlaze tih podređenih kao izlaze modela. Ovo je osmišljeno kao korisna radnja za otklanjanje grešaka. Ako model ne uspe (ili daje neočekivane rezultate), možete zatim pratiti model i pregledati izlaze za sumnjive problematične korake. Time se izbegava potreba za dodavanjem privremenih izlaza u model i ponovnim pokretanjem radi testiranja.

  Osim toga, ova radnja je uvek dostupna nakon pokretanja modela, čak i ako sam model nije uspeo (npr. zbog loše konfigurisanog koraka kasnije u modelu).

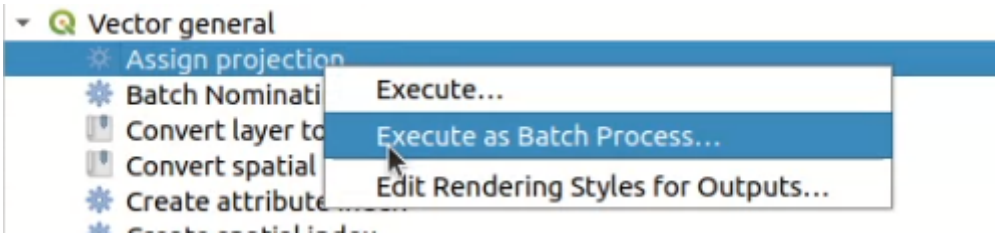
- Radnja *Prikaži dnevnik* pomaže vam da vidite dnevnik za svaki podređeni algoritam nakon što ste zatvorili dijalog algoritma.

23.6 The batch processing interface

23.6.1 Introduction

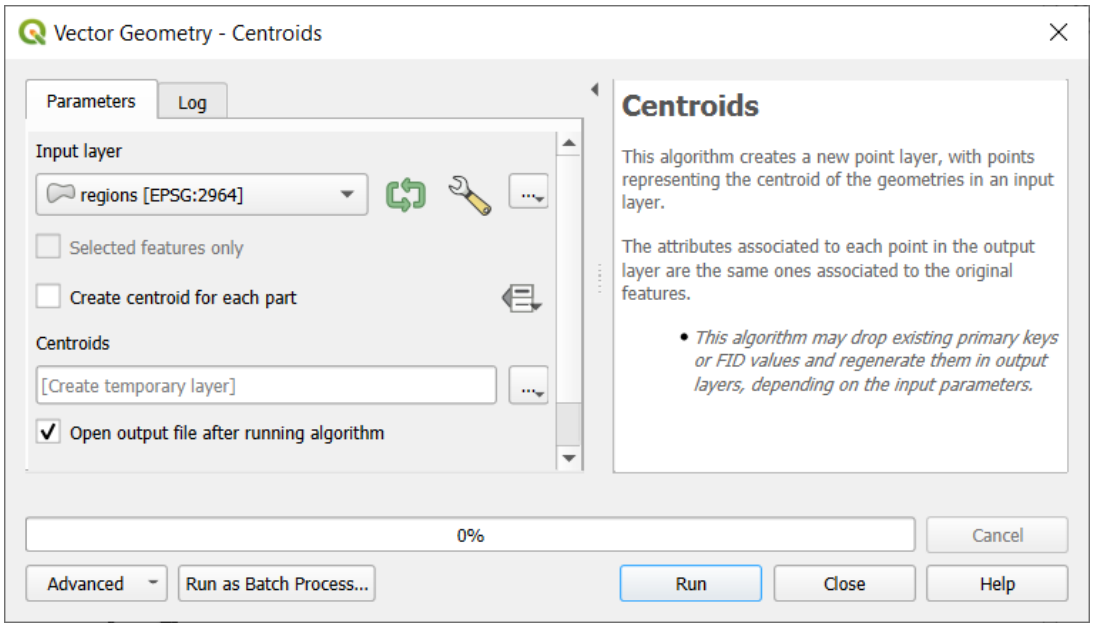
All algorithms (including models) can be executed as a batch process. That is, they can be executed using not just a single set of inputs, but several of them, executing the algorithm as many times as needed. This is useful when processing large amounts of data, since it is not necessary to launch the algorithm many times from the toolbox.

To execute an algorithm as a batch process, right-click on its name in the toolbox and select the *Execute as batch process* option in the pop-up menu that will appear.



Слика23.31: Batch Processing from right-click

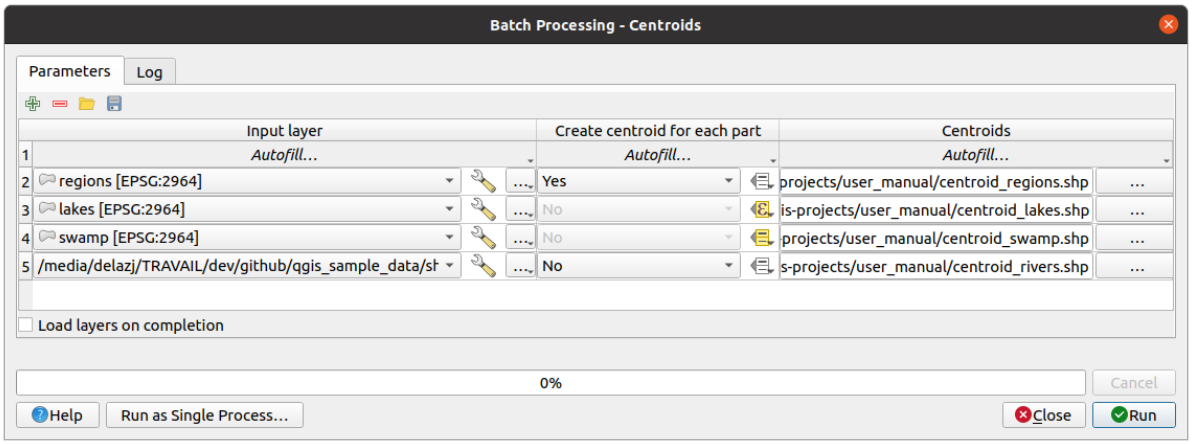
If you have the execution dialog of the algorithm open, you can also start the batch processing interface from there, clicking on the *Run as batch process...* button.



Слика23.32: Batch Processing From Algorithm Dialog

23.6.2 The parameters table

Executing a batch process is similar to performing a single execution of an algorithm. Parameter values have to be defined, but in this case we need not just a single value for each parameter, but a set of them instead, one for each time the algorithm has to be executed. Values are introduced using a table like the one shown next, where each row is an iteration and columns are the parameters of the algorithm.



Слика23.33: Batch Processing

From the top toolbar, you can:

- Toggle advanced mode: Available only when the algorithm has parameters that are marked as advanced, this button allows to show or hide such parameters in the batch dialog.
- Add row: adds a new processing entry for configuration
- Remove row(s): remove selected rows from the table. Row selection is done by clicking the number at the left and allows *keyboard combination* for multi selection.
- Open a batch processing configuration file
- Save the batch processing configuration to a `.JSON` file that can be run afterwards

By default, the table contains just two rows:

- The first row displays in each cell an *Autofill...* ► drop-down menu with *options* to quickly fill the cells below. Available options depend on the parameter type.
- The second row (as well as each subsequent one) represents a single execution of the algorithm, and each cell contains the value of one of the parameters. It is similar to the parameters dialog that you see when executing an algorithm from the toolbox, but with a different arrangement.

At the bottom of the table, you can set whether to *Load layers on completion*.

Once the size of the table has been set, it has to be filled with the desired values.

23.6.3 Filling the parameters table

For most parameters, setting the value is trivial. The appropriate widget, same as in the *single process dialog*, is provided, allowing to just type the value, or select it from a list of possible values, depending on the parameter type. This also includes data-define widget, when compatible.

To automate the batch process definition and avoid filling the table cell by cell, you may want to press down the *Autofill...* menu of a parameter and select any of the following options to replace values in the column:

- *Fill Down* will take the input for the first process and enter it for all other processes.
- *Calculate by Expression...* will allow you to create a new QGIS expression to use to update all existing values within that column. Existing parameter values (including those from other columns) are available for use inside the expression via *variables*. E.g. setting the number of segments based on the buffer distance of each layer:

```
CASE WHEN @DISTANCE > 20 THEN 12 ELSE 8 END
```

- *Add Values by Expression...* will add new rows using the values from an expression which returns an array (as opposed to *Calculate by Expression...*, which works only on existing rows). The intended use case is to allow populating the batch dialog using complex numeric series. For example adding rows for a batch buffer using the expression `generate_series(100, 1000, 50)` for distance parameter results in new rows with values 100, 150, 200, …. 1000.
- When setting a file or layer parameter, more options are provided:
  - *Add Files by Pattern...*: adds new rows to the table for files matching a *File pattern* in a folder to *Look in*. E.g. `*.shp` will add to the list all the `SHP` files in the folder. Check *Search recursively* to also browse sub-folders.
  - *Select Files...* individually on disk
  - *Add All Files from a Directory...*
  - *Select from Open Layers...* in the active project

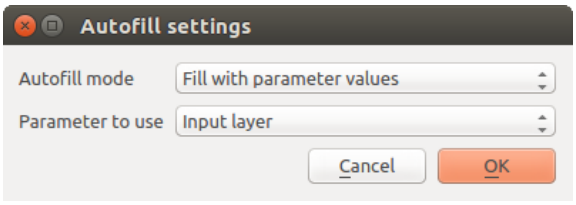
Output data parameter exposes the same capabilities as when executing the algorithm as a single process. Depending on the algorithm, the output can be:

- skipped, if the cell is left empty
- 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).



Слика23.34: Batch Processing Save

If the default value (*Do not autofill*) is selected, it will just put the selected filename in the selected cell from the parameters table. If any of the other options is selected, all the cells **below** the selected one will be automatically filled based on a defined criteria:

  - *Fill with numbers*: incrementally appends a number to the file name
  - *Fill with parameter values*: you can select a parameter whose value in the same row is appended to the file name. This is particularly useful for naming output data objects according to input ones.

- saved as a layer within a database container:

```
# 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 Executing the batch process

To execute the batch process once you have introduced all the necessary values, just click on *Run*. The *Log* panel is activated and displays details and steps of the execution process. Progress of the global batch task will be shown in the progress bar in the lower part of the dialog.

23.7 Using processing algorithms from the console

The console allows advanced users to increase their productivity and perform complex operations that cannot be performed using any of the other GUI elements of the processing framework. Models involving several algorithms can be defined using the command-line interface, and additional operations such as loops and conditional sentences can be added to create more flexible and powerful workflows.

There is not a processing console in QGIS, but all processing commands are available instead from the QGIS built-in *Python console*. That means that you can incorporate those commands into your console work and connect processing algorithms to all the other features (including methods from the QGIS API) available from there.

The code that you can execute from the Python console, even if it does not call any specific processing method, can be converted into a new algorithm that you can later call from the toolbox, the graphical modeler or any other component, just like you do with any other algorithm. In fact, some algorithms that you can find in the toolbox are simple scripts.

In this section, we will see how to use processing algorithms from the QGIS Python console, and also how to write algorithms using Python.

23.7.1 Calling algorithms from the Python console

The first thing you have to do is to import the processing functions with the following line:

```
>>> from qgis import processing
```

Now, there is basically just one (interesting) thing you can do with that from the console: ute an algorithm. That is done using the `run()` method, which takes the id of the algorithm to ute as its first parameter, and then a variable number of additional parameters depending on the requirements of the algorithm. So the first thing you need to know is the name of the algorithm to ute. That is not the name you see in the toolbox, but rather a unique command–line name. To find the right name for your algorithm, you can use the `processingRegistry`. Type the following line in your console:

```
>>> for alg in QgsApplication.processingRegistry().algorithms():
        print(alg.id(), "->", alg.displayName())
```

You will see something like this (with some extra dashes added to improve readability).

```
3d:tessellate --------------> Tessellate
gdal:aspect ----------------> Aspect
gdal:assignprojection ------> Assign projection
gdal:buffervectors ---------> Buffer vectors
gdal:buildvirtualraster ----> Build Virtual Raster
```

(наставак на следећој страни)

(настављено са претходне стране)

```
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
...
```

That's a list of all the available algorithm IDs, sorted by provider name and algorithm name, along with their corresponding names.

Once you know the command-line name of the algorithm, the next thing to do is to determine the right syntax to ute it. That means knowing which parameters are needed when calling the `run()` method.

There is a method to describe an algorithm in detail, which can be used to get a list of the parameters that an algorithm requires and the outputs that it will generate. To get this information, you can use the `algorithmHelp()` method. Pass as parameter the ID of the algorithm, not the full descriptive name.

Calling the method with `native:buffer` as parameter (`qgis:buffer` is an alias for `native:buffer` and will also work), you get the following description:

```
>>> 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
```

(наставак на следећој страни)

(настављено са претходне стране)

```
      - QgsVectorLayer

DISTANCE: Distance

   Parameter type: QgsProcessingParameterDistance

   Accepted data types:
      - int
      - float
      - QgsProperty

SEGMENTS: Segments

   Parameter type: QgsProcessingParameterNumber

   Accepted data types:
      - int
      - float
      - QgsProperty

END_CAP_STYLE: End cap style

   Parameter type: QgsProcessingParameterEnum

   Available values:
      - 0: Round
      - 1: Flat
      - 2: Square

   Accepted data types:
      - 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
```

(наставак на следећој страни)

(настављено са претходне стране)

```
DISSOLVE: Dissolve result

   Parameter type: QgsProcessingParameterBoolean

   Accepted data types:
      - bool
      - int
      - str
      - QgsProperty

OUTPUT: Buffered

   Parameter type: QgsProcessingParameterFeatureSink

   Accepted data types:
      - str: destination vector file, e.g. 'd:/test.shp'
      - str: 'memory:' to store result in temporary memory layer
      - str: using vector provider ID prefix and destination URI,
             e.g. 'postgres:...' to store result in PostgreSQL table
      - QgsProcessingOutputLayerDefinition
      - QgsProperty

----------------
Outputs
----------------

OUTPUT:  <QgsProcessingOutputVectorLayer>
   Buffered
```

Now you have everything you need to run any algorithm. As we have already mentioned, algorithms can be run using: `run()`. Its syntax is as follows:

```
>>> processing.run(name_of_the_algorithm, parameters)
```

Where parameters is a dictionary of parameters that depend on the algorithm you want to run, and is exactly the list that the `algorithmHelp()` method gives you.

```
>>> 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'})
```

If a parameter is optional and you do not want to use it, then don't include it in the dictionary.

If a parameter is not specified, the default value will be used.

Depending on the type of parameter, values are introduced differently. The next list gives a quick review of how to introduce values for each type of input parameter:

- Raster Layer, Vector Layer or Table. Simply use a string with the name that identifies the data object to use (the name it has in the QGIS Table of Contents) or a filename (if the corresponding layer is not opened, it will be opened but not added to the map canvas). If you have an instance of a QGIS object representing the layer, you can also pass it as parameter.

- Enumeration. If an algorithm has an enumeration parameter, the value of that parameter should be entered using an integer value. To know the available options, you can use the `algorithmHelp()` command, as above. For instance, the `native:buffer` algorithm has an enumeration called 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 this case, the parameter has three options. Notice that ordering is zero-based.

- Boolean. Use `True` or `False`.
- Multiple input. The value is a string with input descriptors separated by semicolons (`;`). As in the case of single layers or tables, each input descriptor can be the data object name, or its file path.
- Table Field from XXX. Use a string with the name of the field to use. This parameter is case-sensitive.
- Fixed Table. Type the list of all table values separated by commas (`,`) and enclosed between quotes (`"`). Values start on the upper row and go from left to right. You can also use a 2-D array of values representing the table.
- CRS. Enter the EPSG code number of the desired CRS.
- Extent. You must use a string with `xmin`, `xmax`, `ymin` and `ymax` values separated by commas (`,`).

Boolean, file, string and numerical parameters do not need any additional explanations.

Input parameters such as strings, booleans, or numerical values have default values. The default value is used if the corresponding parameter entry is missing.

For output data objects, type the file path to be used to save it, just as it is done from the toolbox. If the output object is not specified, the result is saved to a temporary file (or skipped if it is an optional output). The extension of the file determines the file format. If you enter a file extension not supported by the algorithm, the default file format for that output type will be used, and its corresponding extension appended to the given file path.

Unlike when an algorithm is uted from the toolbox, outputs are not added to the map canvas if you ute that same algorithm from the Python console using the `run()` method. That method returns a dictionary with one or more output names (the ones shown in the algorithm description) as keys and the file paths of those outputs as values:

```
>>> 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
```

You can then load the output in the project as any common layer:

```
>>> buffered_layer = myresult['OUTPUT']
>>> QgsProject.instance().addMapLayer(buffered_layer)
```

To immediately load the processing outputs in the project, you can use the `runAndLoadResults()` method instead of `run()`.

```
>>> processing.runAndLoadResults("native:buffer", {parameters:values})
```

If you want to open an algorithm dialog from the console you can use the `createAlgorithmDialog` method. The only mandatory parameter is the algorithm id, but you can also define the dictionary of parameters so that the dialog will be filled automatically:

```
>>> 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()
```

The `execAlgorithmDialog` method opens the dialog immediately:

```
>>> 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 Creating scripts and running them from the toolbox

You can create your own algorithms by writing Python code. Processing scripts extend `QgsProcessingAlgorithm`, so you need to add some extra lines of code to implement mandatory functions. You can find *Create new script* (clean sheet) and *Create New Script from Template* (template that includes code for mandatory functions of `QgsProcessingAlgorithm`) under the *Scripts* dropdown menu on the top of the Processing toolbox. The Processing Script Editor will open, and that's where you should type your code. Saving the script from there in the `scripts` folder (the default folder when you open the save file dialog) with a `.py` extension should create the corresponding algorithm.

The name of the algorithm (the one you will see in the toolbox) is defined within the code.

Let's have a look at the following code, which defines a Processing algorithm that performs a buffer operation with a user defined buffer distance on a vector layer that is specified by the user, after first smoothing the layer.

```
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'

```

(наставак на следећој страни)

(настављено са претходне стране)

```
    def __init__(self):
        super().__init__()

    def name(self):
        return "algTest"

    def displayName(self):
        return "algTest script"

    def createInstance(self):
        return type(self)()

    def initAlgorithm(self, config=None):
        self.addParameter(QgsProcessingParameterFeatureSource(
            self.INPUT_VECTOR, "Input vector"))
        self.addParameter(QgsProcessingParameterNumber(
            self.INPUT_BUFFERDIST, "Buffer distance",
            QgsProcessingParameterNumber.Double,
            100.0))
        self.addParameter(QgsProcessingParameterFeatureSink(
            self.OUTPUT_BUFFER, "Output buffer"))

    def processAlgorithm(self, parameters, context, feedback):
        #DO SOMETHING
        algresult = processing.run("native:smoothgeometry",
            {'INPUT': parameters[self.INPUT_VECTOR],
             'ITERATIONS':2,
             'OFFSET':0.25,
             'MAX_ANGLE':180,
             'OUTPUT': 'memory:'},
            context=context, feedback=feedback, is_child_algorithm=True)
        smoothed = algresult['OUTPUT']
        algresult = processing.run('native:buffer',
            {'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}
```

After doing the necessary imports, the following `QgsProcessingAlgorithm` functions are specified:

- `name()`: The id of the algorithm (lowercase).
- `displayName()`: A human readable name for the algorithm.
- `createInstance()`: Create a new instance of the algorithm class.
- `initAlgorithm()`: Configure the parameterDefinitions and outputDefinitions.

  Here you describe the parameters and output of the algorithm. In this case, a feature source for the input, a feature sink for the result and a number for the buffer distance.
- `processAlgorithm()`: Do the work.

Here we first run the `smoothgeometry` algorithm to smooth the geometry, and then we run the `buffer` algorithm on the smoothed output. To be able to run algorithms from within another algorithm we have to set the `is_child_algorithm` argument to `True`. You can see how input and output parameters are used as parameters to the `smoothgeometry` and `buffer` algorithms.

There are a number of different parameter types available for input and output. You can find the full list at *Ulazni i izlazni tipovi za algoritme Obrade*.

The first parameter to the constructors is the name of the parameter, and the second is the description of the parameter (for the user interface). The rest of the constructor parameters are parameter type specific.

The input can be turned into QGIS classes using the `parameterAs…` functions of `QgsProcessingAlgorithm`. For instance to get the number provided for the buffer distance as a double:

```
self.parameterAsDouble(parameters, self.INPUT_BUFFERDIST, context)).
```

The `processAlgorithm()` function should return a dictionary containing values for every output defined by the algorithm. This allows access to these outputs from other algorithms, including other algorithms contained within the same model.

Well behaved algorithms should define and return as many outputs as makes sense. Non-feature outputs, such as numbers and strings, are very useful when running your algorithm as part of a larger model, as these values can be used as input parameters for subsequent algorithms within the model. Consider adding numeric outputs for things like the number of features processed, the number of invalid features encountered, the number of features output, etc. The more outputs you return, the more useful your algorithm becomes!

Feedback

The `feedback` object passed to `processAlgorithm()` should be used for user feedback / interaction. You can use the `setProgress()` function of the `feedback` object to update the progress bar (0 to 100) to inform the user about the progress of the algorithm. This is very useful if your algorithm takes a long time to complete.

The `feedback` object provides an `isCanceled()` method that should be monitored to enable cancellation of the algorithm by the user. The `pushInfo()` method of `feedback` can be used to send information to the user, and `reportError()` is handy for pushing non-fatal errors to users.

Algorithms should avoid using other forms of providing feedback to users, such as print statements or logging to `QgsMessageLog`, and should always use the feedback object instead. This allows verbose logging for the algorithm, and is also thread-safe (which is important, given that algorithms are typically run in a background thread).

Handling errors

If your algorithm encounters an error which prevents it from executing, such as invalid input values or some other condition from which it cannot or should not recover, then you should raise a `QgsProcessingException`. E.g.:

```
if feature['value'] < 20:
  raise QgsProcessingException('Invalid input value {}, must be >= 20'.
↪format(feature['value']))
```

Try to avoid raising `QgsProcessingException` for non-fatal errors (e.g. when a feature has a null geometry), and instead just report these errors via `feedback.reportError()` and skip the feature. This helps make your algorithm „model-friendly”, as it avoids halting the execution of an entire algorithm when a non-fatal error is encountered.

Documenting your scripts

As in the case of models, you can create additional documentation for your scripts, to explain what they do and how to use them.

`QgsProcessingAlgorithm` provides the `helpString()`, `shortHelpString()` and `helpUrl()` functions for that purpose. Specify / override these to provide more help to the user.

`shortDescription()` is used in the tooltip when hovering over the algorithm in the toolbox.

23.7.3 Pre- and post-execution script hooks

Scripts can also be used as pre- and post-execution hooks that are run before and after an algorithm is run, respectively. This can be used to automate tasks that should be performed whenever an algorithm is executed.

The syntax is identical to the syntax explained above, but an additional global variable named `alg` is available, representing the algorithm that has just been (or is about to be) executed.

In the *General* group of the processing options dialog, you will find two entries named *Pre-execution script* and *Post-execution script* where the filenames of the scripts to be run in each case can be entered.

23.8 Using processing from the command line

QGIS comes with a tool called `QGIS Processing Executor` which allows you to run Processing algorithms and models (built-in or provided by plugins) directly from the command line without starting QGIS Desktop itself.

From a command line tool, run `qgis_process` and you should get:

```
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.
```

(наставак на следећој страни)

(настављено са претходне стране)

```
                    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.
```

Белешка

Only installed plugins that advertise `hasProcessingProvider=yes` in their `metadata.txt` file are recognized and can be activated or loaded by qgis_process tool.

Путоказ

Before calling qgis_process on a system without window manager (e.g. a headless server), you should set:

```
export QT_QPA_PLATFORM=offscreen
```

The command `list` can be used to get a list of all available providers and algorithms.

```
qgis_process list
```

The command `help` can be used to get further information about commands or algorithms.

```
qgis_process help qgis:regularpoints
```

The command `run` can be used to run an algorithm or model. Specify the name of the algorithm or a path to a model as first parameter.

```
qgis_process run native:buffer -- INPUT=source.shp DISTANCE=2 OUTPUT=buffered.shp
```

Where a parameter accepts a list of values, set the same variable multiple times.

```
qgis_process run native:mergevectorlayers -- LAYERS=input1.shp LAYERS=input2.shp␣
↪OUTPUT=merged.shp
```

While running an algorithm a text-based feedback bar is shown, and the operation can be cancelled via `CTRL+C`.

The `run` command also supports further parameters.

- `--json` will format stdout output in a JSON structured way.
- `--ellipsoid` will set the ellipsoid to the specified one.
- `--distance_units` will use the specified distance units.
- `--area_units` will use the specified area units.
- `--project_path` will load the specified project for running the algorithm.

Complex input parameters, i.e. parameter types which are themselves specified as a dictionary type object for algorithms, are supported by qgis_process. To indicate that parameters will be specified via stdin, the qgis_process command must follow the format (with a trailing – in place of the usual arguments list).

```
qgis_process run algorithmId -
```

The JSON object must contain an „inputs” key, which is a map of the input parameter values. E.g.

```
echo "{'inputs': {'INPUT': 'my_shape.shp', 'DISTANCE': 5}}" | qgis_process run␣
↪native:buffer -
```

Additionally, extra settings like the distance units, area units, ellipsoid and project path can be included in this JSON object:

```
{
 'ellipsoid': 'EPSG:7019',
 'distance_units': 'feet',
 'area_units': 'ha',
 'project_path': 'C:/temp/my_project.qgs'
 'inputs': {'DISTANCE': 5, 'SEGMENTS': 8 ... }
}
```

Specifying input parameters via stdin implies automatically the `JSON` output format for results.

23.9 Writing new Processing algorithms as Python scripts

There are two options for writing Processing algorithms using Python.

- *Extending* `QgsProcessingAlgorithm`
- *Using the @alg decorator*

Within QGIS, you can use *Create new script* in the *Scripts* menu at the top of the *Processing Toolbox* to open the *Processing Script Editor* where you can write your code. To simplify the task, you can start with a script template by using *Create new script from template* from the same menu. This opens a template that extends `QgsProcessingAlgorithm`.

If you save the script in the `scripts` folder (the default location) with a `.py` extension, the algorithm will become available in the *Processing Toolbox*.

23.9.1 Extending QgsProcessingAlgorithm

The following code

1. takes a vector layer as input
2. counts the number of features
3. does a buffer operation
4. creates a raster layer from the result of the buffer operation
5. returns the buffer layer, raster layer and number of features

```
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
13 class ExampleProcessingAlgorithm(QgsProcessingAlgorithm):
```

(наставак на следећој страни)

(настављено са претходне стране)

```
    """
    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'),
```

(наставак на следећој страни)

(настављено са претходне стране)

```
                types=[QgsProcessing.TypeVectorAnyGeometry]
            )
        )
        self.addParameter(
            QgsProcessingParameterVectorDestination(
                'BUFFER_OUTPUT',
                self.tr('Buffer output'),
            )
        )
        # 'OUTPUT' is the recommended name for the main output
        # parameter.
        self.addParameter(
            QgsProcessingParameterRasterDestination(
                'OUTPUT',
                self.tr('Raster output')
            )
        )
        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
```

(наставак на следећој страни)

(настављено са претходне стране)

```
        # using self.parameterAsDouble.
        bufferdist = self.parameterAsDouble(parameters, 'BUFFERDIST',
                                            context)
        rastercellsize = self.parameterAsDouble(parameters, 'CELLSIZE',
                                                context)
        if feedback.isCanceled():
            return {}
        buffer_result = processing.run(
            'native:buffer',
            {
                # Here we pass on the original parameter values of INPUT
                # and BUFFER_OUTPUT to the buffer algorithm.
                'INPUT': parameters['INPUT'],
                'OUTPUT': parameters['BUFFER_OUTPUT'],
                'DISTANCE': bufferdist,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10
            },
            # Because the buffer algorithm is being run as a step in
            # 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 cancellation requests.
            context=context,
            feedback=feedback)

        # Check for cancellation
        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 {}

```

(наставак на следећој страни)

(настављено са претходне стране)

```
        # Return the results
        return {'OUTPUT': rasterized_result['OUTPUT'],
                'BUFFER_OUTPUT': buffer_result['OUTPUT'],
                'NUMBEROFFEATURES': numfeatures}
```

Processing algorithm standard functions:

- **createInstance (mandatory)**
  Must return a new copy of your algorithm. If you change the name of the class, make sure you also update the value returned here to match!
- **name (mandatory)**
  Returns the unique algorithm name, used for identifying the algorithm.
- **displayName (mandatory)**
  Returns the translated algorithm name.
- **group**
  Returns the name of the group this algorithm belongs to.
- **groupId**
  Returns the unique ID of the group this algorithm belongs to.
- **shortHelpString**
  Returns a localised short help string for the algorithm.
- **initAlgorithm (mandatory)**
  Here we define the inputs and outputs of the algorithm.

  `INPUT` and `OUTPUT` are recommended names for the main input and main output parameters, respectively.

  If a parameter depends on another parameter, `parentParameterName` is used to specify this relationship (could be the field / band of a layer or the distance units of a layer).
- **processAlgorithm (mandatory)**
  This is where the processing takes place.

  Parameters are retrieved using special purpose functions, for instance `parameterAsSource` and `parameterAsDouble`.

  `processing.run` can be used to run other processing algorithms from a processing algorithm. The first parameter is the name of the algorithm, the second is a dictionary of the parameters to the algorithm. `is_child_algorithm` is normally set to `True` when running an algorithm from within another algorithm. `context` and `feedback` inform the algorithm about the environment to run in and the channel for communicating with the user (catching cancel request, reporting progress, providing textual feedback). When using the (parent) algorithm's parameters as parameters to „child" algorithms, the original parameter values should be used (e.g. `parameters['OUTPUT']`).

  It is good practice to check the feedback object for cancellation as much as is sensibly possible! Doing so allows for responsive cancellation, instead of forcing users to wait for unwanted processing to occur.

  The algorithm should return values for all the output parameters it has defined as a dictionary. In this case, that's the buffer and rasterized output layers, and the count of features processed. The dictionary keys must match the original parameter/output names.

23.9.2 The @alg decorator

Using the @alg decorator, you can create your own algorithms by writing the Python code and adding a few extra lines to supply additional information needed to make it a proper Processing algorithm. This simplifies the creation of algorithms and the specification of inputs and outputs.

One important limitation with the decorator approach is that algorithms created in this way will always be added to a user's Processing Scripts provider – it is not possible to add these algorithms to a custom provider, e.g. for use in plugins.

The following code uses the @alg decorator to

1. use a vector layer as input
2. count the number of features
3. do a buffer operation
4. create a raster layer from the result of the buffer operation
5. returns the buffer layer, raster layer and number of features

```
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,
```

(наставак на следећој страни)

(настављено са претходне стране)

```
                                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}
```

As you can see, it involves two algorithms ('native:buffer' and 'qgis:rasterize'). The last one ('qgis:rasterize') creates a raster layer from the buffer layer that was generated by the first one ('native:buffer').

The part of the code where this processing takes place is not difficult to understand if you have read the previous chapter. The first lines, however, need some additional explanation. They provide the information that is needed to turn your code into an algorithm that can be run from any of the GUI components, like the toolbox or the model designer.

These lines are all calls to the `@alg` decorator functions that help simplify the coding of the algorithm.

- The @alg decorator is used to define the name and location of the algorithm in the Toolbox.
- The @alg.input decorator is used to define the inputs of the algorithm.
- The @alg.output decorator is used to define the outputs of the algorithm.

For existing parameters and their correspondence, read *Ulazni i izlazni tipovi za algoritme Obrade*.

23.9.3 Handing algorithm output

When you declare an output representing a layer (raster or vector), the algorithm will try to add it to QGIS once it is finished.

- Raster layer output: QgsProcessingParameterRasterDestination / alg.RASTER_LAYER_DEST.
- Vector layer output: QgsProcessingParameterVectorDestination / alg.VECTOR_LAYER_DEST.

So even if the `processing.run()` method does not add the layers it creates to the user's current project, the two output layers (buffer and raster buffer) will be loaded, since they are saved to the destinations entered by the user (or to temporary destinations if the user does not specify destinations).

If a layer is created as output of an algorithm, it should be declared as such. Otherwise, you will not be able to properly use the algorithm in the modeler, since what is declared will not match what the algorithm really creates.

You can return strings, numbers and more by specifying them in the result dictionary (as demonstrated for „NUMBEROFFEATURES"), but they should always be explicitly defined as outputs from your algorithm. We encourage algorithms to output as many useful values as possible, since these can be valuable for use in later algorithms when your algorithm is used as part of a model.

23.9.4 Communicating with the user

If your algorithm takes a long time to process, it is a good idea to inform the user about the progress. You can use `feedback` (`QgsProcessingFeedback`) for this.

The progress text and progressbar can be updated using two methods: `setProgressText(text)` and `setProgress(percent)`.

You can provide more information by using `pushCommandInfo(text)`, `pushDebugInfo(text)`, `pushInfo(text)` and `reportError(text)`.

If your script has a problem, the correct way of handling it is to raise a `QgsProcessingException`. You can pass a message as an argument to the constructor of the exception. Processing will take care of handling it and communicating with the user, depending on where the algorithm is being executed from (toolbox, modeler, Python console, …)

23.9.5 Documenting your scripts

You can document your scripts by overloading the `helpString()` and `helpUrl()` methods of `QgsProcessingAlgorithm`.

23.9.6 Flags

You can override the `flags()` method of `QgsProcessingAlgorithm` to tell QGIS more about your algorithm. You can for instance tell QGIS that the script shall be hidden from the modeler, that it can be canceled, that it is not thread safe, and more.

Савет

By default, Processing runs algorithms in a separate thread in order to keep QGIS responsive while the processing task runs. If your algorithm is regularly crashing, you are probably using API calls which are not safe to do in a background thread. Try returning the QgsProcessingAlgorithm.FlagNoThreading flag from your algorithm's flags() method to force Processing to run your algorithm in the main thread instead.

23.9.7 Best practices for writing script algorithms

Here's a quick summary of ideas to consider when creating your script algorithms and, especially, if you want to share them with other QGIS users. Following these simple rules will ensure consistency across the different Processing elements such as the toolbox, the modeler or the batch processing interface.

- Do not load resulting layers. Let Processing handle your results and load your layers if needed.
- Always declare the outputs your algorithm creates.
- Do not show message boxes or use any GUI element from the script. If you want to communicate with the user, use the methods of the feedback object (`QgsProcessingFeedback`) or throw a `QgsProcessingException`.

There are already many processing algorithms available in QGIS. You can find code on the QGIS repo.

23.10 Ulazni i izlazni tipovi za algoritme Obrade

Ovo je lista ulaznih i izlaznih tipova koji su podržani u Obradi, sa odgovarajućim konstantama alg dekoratora i njihovim nazivom u dizajneru modela, ako je izložen.

23.10.1 Ulazni tipovi

Naziv ulaza	Klasa	Konstanta algoritma	Opis
Sloj napomena	QgsProcessingParameterAnnotationLayer	alg.ANNOTATION_LAYER	Sloj napomena
Površina	QgsProcessingParameterArea		Numerički parametar koji predstavlja meru površine
Konfiguracija autentifikacije	QgsProcessingParameterAuthConfig	alg.AUTH_CFG	Omogućava korisnicima da izaberu neku od dostupnih konfiguracija autentifikacije ili da naprave nove
Rasterski kanal	QgsProcessingParameterBand	alg.BAND	Kanal rasterskog sloja
Logička vrednost	QgsProcessingParameterBoolean	alg.BOOL	Logička vrednost
Naziv veze	QgsProcessingParameterProviderConnection		Izbor iz dostupnih registrovanih veza za određenog provajdera podataka
Boja	QgsProcessingParameterColor	alg.COLOR	Boja
Koordinatna operacija	QgsProcessingParameterCoordinateOperation	alg.COORDINATE_OPERAT	Koordinatna operacija (za CRS transformacije)
CRS	QgsProcessingParameterCrs	alg.CRS	Koordinatni referentni sistem
Šema baze podataka	QgsProcessingParameterDatabaseSchema	alg.DATABASE_SCHEMA	Šema baze podataka
Tabela baze podataka	QgsProcessingParameterDatabaseTable	alg.DATABASE_TABLE	Tabela baze podataka
Datum	ProcessingDateTimeParDate	alg.DATE	Isključivo datum
Datum i vreme	ProcessingDateTimeParDateTime	alg.DATETIME	Datum i vreme
Rastojanje	QgsProcessingParameterDistance	alg.DISTANCE	Numerički parametar dvostruke preciznosti za vrednosti rastojanja
DXF slojevi	QgsProcessingParameterDxfLayers		Lista vektorskih slojeva za izvoz u DXF datoteku
Enum	QgsProcessingParameterEnum	alg.ENUM	Nabrajanje, koje omogućava izbor iz skupa unapred definisanih vrednosti
Expression	QgsProcessingParameterExpression	alg.EXPRESSION	Izraz
Obuhvat	QgsProcessingParameterExtent	alg.EXTENT	Prostorni obuhvat definisan sa xmin, xmax, ymin, ymax
	QgsProcessingParameterField	alg.FIELD	Polje u atributskoj tabeli vektorskog sloja
Agregati polja	QgsProcessingParameterAggregate		Parametar za konfiguracije „agregata“, koje se sastoje od definicije željenih izlaznih polja, tipova i agregata kojima se ona popunjavaju

наставак на следећој страни

Табела 23.1 – настављено са претходне стране

Naziv ulaza	Klasa	Konstanta algoritma	Opis
Mapiranje polja	QgsProcessingParameterFieldMapping		Parametar za konfiguracije „mapiranja polja“, koje se sastoje od definicije željenih izlaznih polja, tipova i izraza kojima se ona popunjavaju
Datoteka/folder	QgsProcessingParameterFile	alg.FILE	Ime postojeće datoteke
	QgsProcessingParameterFileDestination	alg.FILE_DEST	Ime novokreirane izlazne datoteke
	QgsProcessingParameterFolderDestination	alg.FOLDER_DEST	Folder (odredišni folder)
Geometrija	QgsProcessingParameterGeometry	alg.GEOMETRY	Geometrija
	ProcessingNumberParameterInteger	alg.INT	Ceo broj
Raspored za štampu	QgsProcessingParameterLayout	alg.LAYOUT	Raspored
Stavka rasporeda za štampu	QgsProcessingParameterLayoutItem	alg.LAYOUT_ITEM	Stavka rasporeda
Sloj karte	QgsProcessingParameterMapLayer	alg.MAPLAYER	Sloj karte
Tema karte	QgsProcessingParameterMapTheme	alg.MAP_THEME	Tema karte projekta
Matrica	QgsProcessingParameterMatrix	alg.MATRIX	Matrica
Mesh grupe skupova podataka	QgsProcessingParameterMeshDatasetGroups		Grupe skupova podataka Mesh sloja
Vreme Mesh skupa podataka	QgsProcessingParameterMeshDatasetTime		Indeks skupa podataka iz Mesh sloja prema podešavanju vremena
Mesh sloj	QgsProcessingParameterMeshLayer	alg.MESH_LAYER	Mesh sloj
Višestruki ulaz	QgsProcessingParameterMultipleLayers	alg.MULTILAYER	Skup slojeva
Broj	ProcessingNumberParameterDouble	alg.NUMBER	Numerička vrednost
Tačka	QgsProcessingParameterPoint	alg.POINT	Tačka
Atribut oblaka tačaka	QgsProcessingParameterPointCloudAttribute		Atribut iz sloja oblaka tačaka
	QgsProcessingParameterPointCloudDestination	alg.POINTCLOUD_LAYER_	Parametar odredišta za sloj oblaka tačaka, za zadavanje odredišne putanje sloja oblaka tačaka koji algoritam kreira
Sloj oblaka tačaka	QgsProcessingParameterPointCloudLayer	alg.POINTCLOUD_LAYER	Sloj oblaka tačaka
	QgsProcessingParameterProviderConnection	alg.PROVIDER_CONNECTI	Dostupna veza za provajdera baze podataka

наставак на следећој страни

Табела 23.1 – настављено са претходне стране

Naziv ulaza	Klasa	Konstanta algoritma	Opis
Opseg	QgsProcessingParameterRange	alg.RANGE	Opseg brojeva
Rasterski sloj	QgsProcessingParameterRasterLayer	alg.RASTER_LAYER	Rasterski sloj
	QgsProcessingParameterRasterDestination	alg.RASTER_LAYER_DEST	Parametar odredišta za rasterski sloj, za zadavanje odredišne putanje rasterskog sloja koji algoritam kreira
Razmera	QgsProcessingParameterScale	alg.SCALE	Razmera karte
	QgsProcessingParameterFeatureSink	alg.SINK	Ponor objekata
Niska	QgsProcessingParameterString	alg.STRING	Tekstualna niska
Slojevi za kreiranje TIN-a	QgsProcessingParameterTinInputLayers		Izbor više slojeva za kreiranje TIN-a sa temenima i/ili prelomnim linijama
Vreme	ProcessingDateTimeParTime	alg.TIME	Isključivo vreme
Vektorski objekti	QgsProcessingParameterFeatureSource	alg.SOURCE	Izvor objekata
Vektorski sloj	QgsProcessingParameterVectorLayer	alg.VECTOR_LAYER	Vektorski sloj
	QgsProcessingParameterVectorDestination	alg.VECTOR_LAYER_DEST	Parametar odredišta za vektorski sloj, za zadavanje odredišne putanje vektorskog sloja koji algoritam kreira
Slojevi za upis vektorskih pločica	QgsProcessingParameterVectorTileDestination		Parametar odredišta za sloj vektorskih pločica, za zadavanje odredišne putanje sloja vektorskih pločica koji algoritam kreira
Zapremina	QgsProcessingParameterVolume		Numerički parametar koji predstavlja meru zapremine

23.10.2 Izlazni tipovi

Klasa	Konstanta algoritma	Opis
QgsProcessingOutputBoolean	alg.BOOL	Logička vrednost
QgsProcessingOutputNumber	alg.DISTANCE	Numerički parametar dvostruke preciznosti za vrednosti rastojanja
QgsProcessingOutputFile	alg.FILE	Ime postojeće datoteke
QgsProcessingOutputFolder	alg.FOLDER	Folder
QgsProcessingOutputHtml	alg.HTML	HTML
QgsProcessingOutputNumber	alg.INT	Ceo broj
QgsProcessingOutputLayerDefinition	alg.LAYERDEF	Definicija sloja
QgsProcessingOutputMapLayer	alg.MAPLAYER	Sloj karte
QgsProcessingOutputMultipleLayers	alg.MULTILAYER	Skup slojeva
QgsProcessingOutputNumber	alg.NUMBER	Numerička vrednost
QgsProcessingOutputPointCloudLayer	alg.POINTCLOUD_LAYER	Sloj oblaka tačaka

наставак на следећој страни

Табела 23.2 – настављено са претходне стране

Klasa	Konstanta algoritma	Opis
QgsProcessingOutputRasterLayer	alg.RASTER_LAYER	Rasterski sloj
QgsProcessingOutputString	alg.STRING	Tekstualna niska
QgsProcessingOutputVectorLayer	alg.VECTOR_LAYER	Vektorski sloj
QgsProcessingOutputVectorTileLayer		Sloj vektorskih pločica

23.11 Configuring external applications

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.

This section will show you how to configure the Processing framework to include these additional applications, and it will explain some particular features of the algorithms based on them. Once you have correctly configured the system, you will be able to execute external algorithms from any component like the toolbox or the model designer, just like you do with any other algorithm.

By default, algorithms that rely on an external application not shipped with QGIS are not enabled. You can enable them in the Processing settings dialog if they are installed on your system.

23.11.1 A note for Windows users

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 A note on file formats

When using external software, opening a file in QGIS does not mean that it can be opened and processed in that other software. In most cases, other software can read what you have opened in QGIS, but in some cases, that might not be true. When using databases or uncommon file formats, whether for raster or vector layers, problems might arise. If that happens, try to use well-known file formats that you are sure are understood by both programs, and check the console output (in the log panel) to find out what is going wrong.

You might for instance get trouble and not be able to complete your work if you call an external algorithm with a GRASS raster layers as input. For this reason, such layers will not appear as available to algorithms.

You should, however, not have problems with vector layers, since QGIS automatically converts from the original file format to one accepted by the external application before passing the layer to it. This adds extra processing time, which might be significant for large layers, so do not be surprised if it takes more time to process a layer from a DB connection than a layer from a Shapefile format dataset of similar size.

Providers not using external applications can process any layer that you can open in QGIS, since they open it for analysis through QGIS.

All raster and vector output formats produced by QGIS can be used as input layers. Some providers do not support certain formats, but all can export to common formats that can later be transformed by QGIS automatically. As for input layers, if a conversion is needed, that might increase the processing time.

23.11.3 A note on vector layer selections

External applications may also be made aware of the selections that exist in vector layers within QGIS. However, that requires rewriting all input vector layers, just as if they were originally in a format not supported by the external application. Only when no selection exists, or the *Use only selected features* option is not enabled in the processing general configuration, can a layer be directly passed to an external application.

In other cases, exporting only selected features is needed, which causes longer execution times.

23.11.4 Using third-party Providers

R scripts and libraries

To enable R in Processing you need to install the **Processing R Provider** plugin and configure R for QGIS.

Configuration is done in *Provider* ► *R* in the *Processing* tab of *Settings* ► *Options*.

Depending on your operating system, you may have to use *R folder* to specify where your R binaries are located.

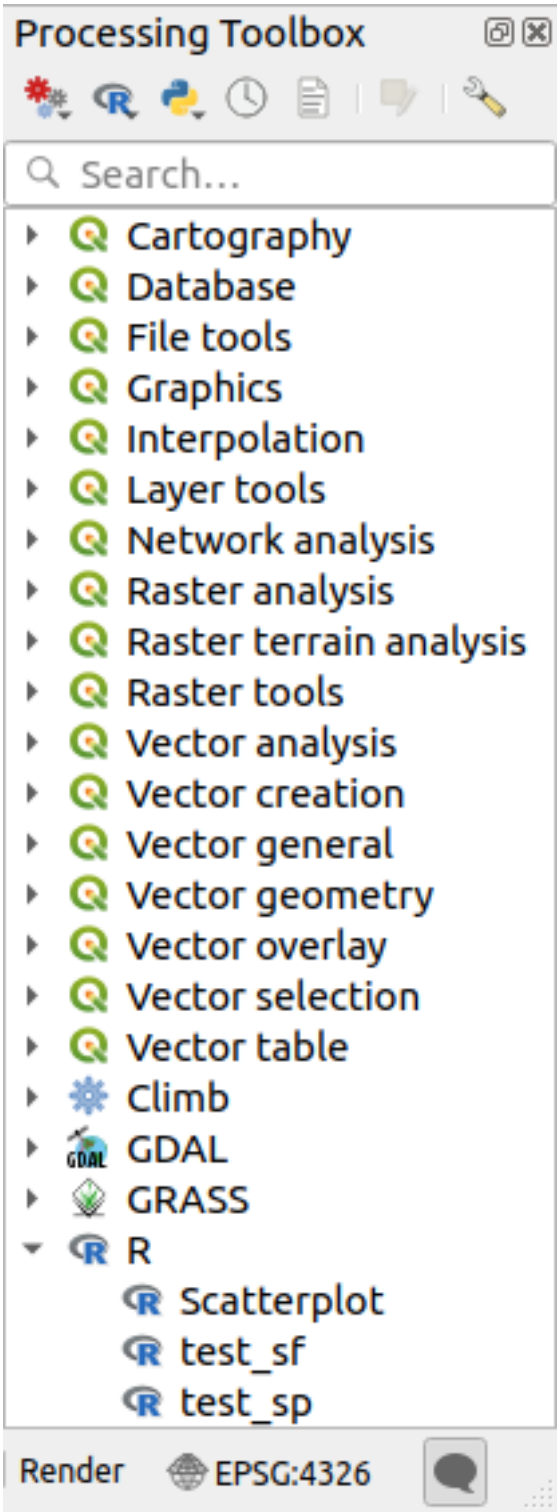
Белешка

On **Windows** the R executable file is normally in a folder (`R-<version>`) under `C:\Program Files\R\`. Specify the folder and **NOT** the binary!

On **Linux** you just have to make sure that the R folder is in the PATH environment variable. If `R` in a terminal window starts R, then you are ready to go.

After installing the **Processing R Provider** plugin, you will find some example scripts in the *Processing Toolbox*:

- *Scatterplot* runs an R function that produces a scatter plot from two numerical fields of the provided vector layer.
- *test_sf* does some operations that depend on the `sf` package and can be used to check if the R package `sf` is installed. If the package is not installed, R will try to install it (and all the packages it depends on) for you, using the *Package repository* specified in *Provider* ► *R* in the Processing options. The default is https://cran.r-project.org/. Installing may take some time…
- *test_sp* can be used to check if the R package `sp` is installed. If the package is not installed, R will try to install it for you.



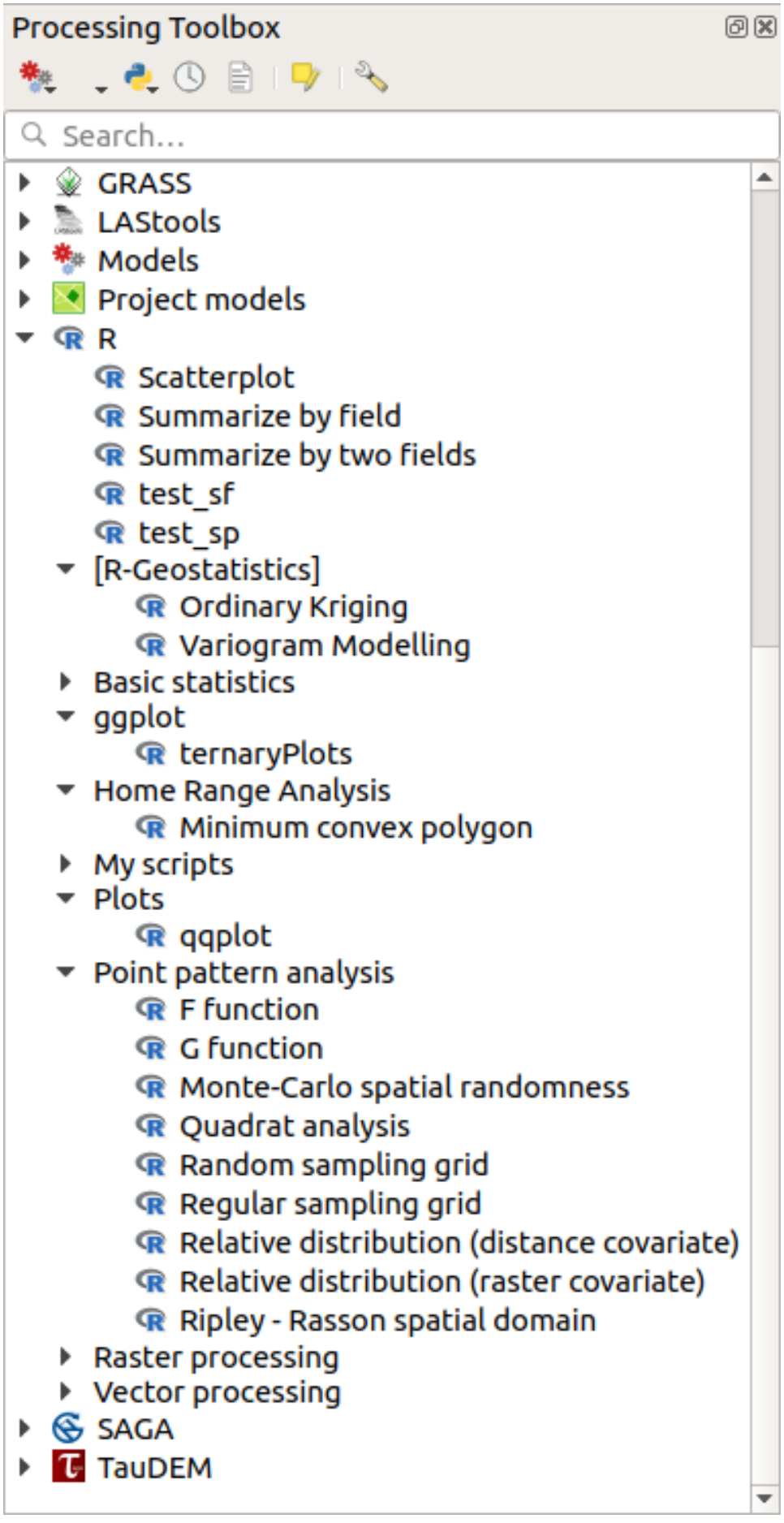
If you have R configured correctly for QGIS, you should be able to run these scripts.

Adding R scripts from the 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).

A set of example R scripts is available in the QGIS Repository. Perform the following steps to load and enable them using the *QGIS Resource Sharing* plugin.

1. Add the *QGIS Resource Sharing* plugin (you may have to enable *Show also experimental plugins* in the Plugin Manager *Settings*)
2. Open it (*Plugins* ► *Resource Sharing* ► *Resource Sharing*)
3. Choose the *Settings* tab
4. Click *Reload repositories*
5. Choose the *All* tab
6. Select *QGIS R script collection* in the list and click on the *Install* button
7. The collection should now be listed in the *Installed* tab
8. Close the plugin
9. Open the *Processing Toolbox*, and if everything is OK, the example scripts will be present under R, in various groups (only some of the groups are expanded in the screenshot below).

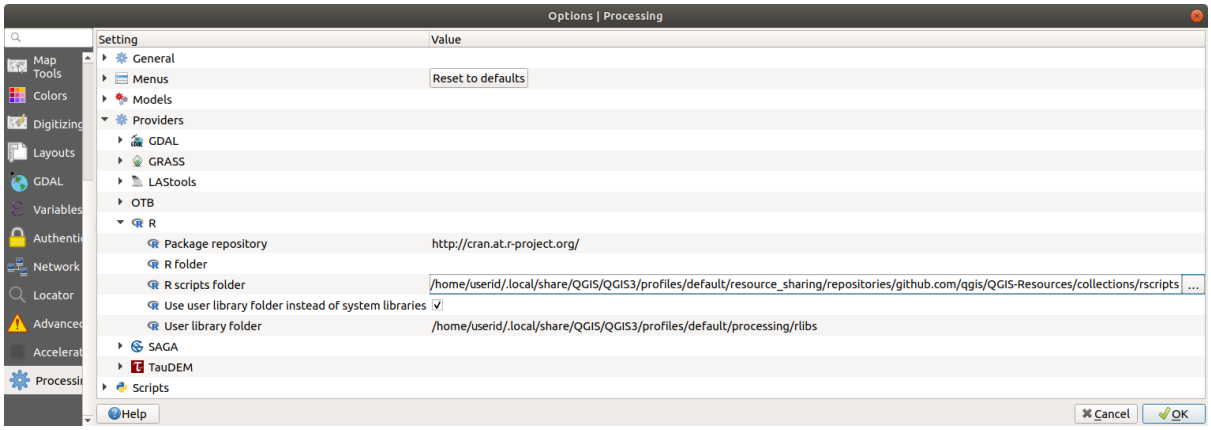


Слика23.35: The *Processing Toolbox* with some R scripts shown

The scripts at the top are the example scripts from the *Processing R Provider* plugin.

10. If, for some reason, the scripts are not available in the *Processing Toolbox*, you can try to:
    1. Open the Processing settings (*Settings* ► *Options* ► *Processing* tab)
    2. Go to *Providers* ► *R* ► *R scripts folder*
       - On Ubuntu, set the path to (or, better, include in the path):

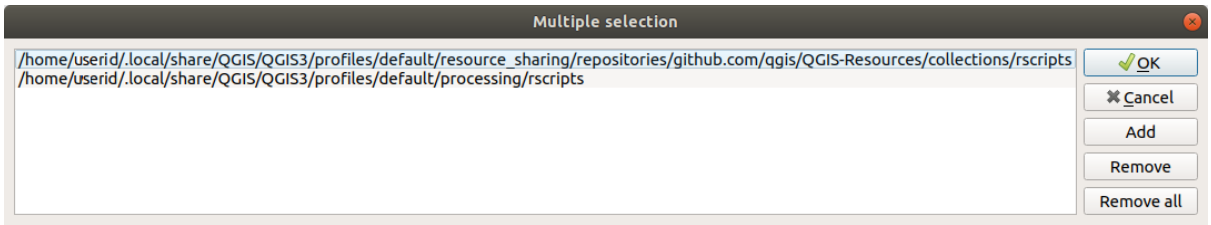
         /home/<user>/.local/share/QGIS/QGIS3/profiles/default/resource_sharing/repositories/github.com/qgis/QGIS-
         -Resources/collections/rscripts



       - On Windows, set the path to (or, better, include in the path):

         C:\Users\<user>\AppData\Roaming\QGIS\QGIS3\profiles\default\resource_sharing\repositories\github.com\qg
         -Resources\collections\rscripts

       To edit, double-click. You can then choose to just paste / type the path, or you can navigate to the directory by using the … button and press the *Add* button in the dialog that opens. It is possible to provide several directories here. They will be separated by a semicolon („;”).



If you would like to get all the R scripts from the QGIS 2 on-line collection, you can select *QGIS R script collection (from QGIS 2)* instead of *QGIS R script collection*. You will probably find that scripts that depend on vector data input or output will not work.

Creating R scripts

You can write scripts and call R commands, as you would do from R. This section shows you the syntax for using R commands in QGIS, and how to use QGIS objects (layers, tables) in them.

To add an algorithm that calls an R function (or a more complex R script that you have developed and you would like to have available from QGIS), you have to create a script file that performs the R commands.

R script files have the extension `.rsx`, and creating them is pretty easy if you just have a basic knowledge of R syntax and R scripting. They should be stored in the R scripts folder. You can specify the folder (*R scripts folder*) in the *R* settings group in Processing settings dialog).

Let's have a look at a very simple script file, which calls the R method `spsample` to create a random grid within the boundary of the polygons in a given polygon layer. This method belongs to the `maptools` package. Since almost all

the algorithms that you might like to incorporate into QGIS will use or generate spatial data, knowledge of spatial packages like `maptools` and `sp`/`sf`, is very useful.

```
##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)
```

The first lines, which start with a double Python comment sign (`##`), define the display name and group of the script, and tell QGIS about its inputs and outputs.

Белешка

To find out more about how to write your own R scripts, have a look at the R Intro section in the training manual and consult the *QGIS R Syntax* section.

When you declare an input parameter, QGIS uses that information for two things: creating the user interface to ask the user for the value of that parameter, and creating a corresponding R variable that can be used as R function input.

In the above example, we have declared an input of type `vector`, named `Vector_layer`. When executing the algorithm, QGIS will open the layer selected by the user and store it in a variable named `Vector_layer`. So, the name of a parameter is the name of the variable that you use in R for accessing the value of that parameter (you should therefore avoid using reserved R words as parameter names).

Spatial parameters such as vector and raster layers are read using the `st_read()` (or `readOGR`) and `brick()` (or `readGDAL`) commands (you do not have to worry about adding those commands to your description file – QGIS will do it), and they are stored as `sf` (or `Spatial*DataFrame`) objects.

Table fields are stored as strings containing the name of the selected field.

Vector files can be read using the `readOGR()` command instead of `st_read()` by specifying `##load_vector_using_rgdal`. This will produce a `Spatial*DataFrame` object instead of an `sf` object.

Raster files can be read using the `readGDAL()` command instead of `brick()` by specifying `##load_raster_using_rgdal`.

If you are an advanced user and do not want QGIS to create the object for the layer, you can use `##pass_filenames` to indicate that you prefer a string with the filename. In this case, it is up to you to open the file before performing any operation on the data it contains.

With the above information, it is possible to understand the first lines of the R script (the first line not starting with a Python comment character).

```
library(sp)
spatpoly = as(Vector_layer, "Spatial")
pts=spsample(polyg,numpoints,type="random")
```

The `spsample` function is provided by the *sp* library, so the first thing we do is to load that library. The variable `Vector_layer` contains an `sf` object. Since we are going to use a function (`spsample`) from the *sp* library, we must convert the `sf` object to a `SpatialPolygonsDataFrame` object using the `as` function.

Then we call the `spsample` function with this object and the `numpoints` input parameter (which specifies the number of points to generate).

Since we have declared a vector output named `Output`, we have to create a variable named `Output` containing an `sf` object.

We do this in two steps. First we create a `SpatialPolygonsDataFrame` object from the result of the function, using the *SpatialPointsDataFrame* function, and then we convert that object to an `sf` object using the `st_as_sf` function (of the *sf* library).

You can use whatever names you like for your intermediate variables. Just make sure that the variable storing your final result has the defined name (in this case `Output`), and that it contains a suitable value (an `sf` object for vector layer output).

In this case, the result obtained from the `spsample` method had to be converted explicitly into an `sf` object via a `SpatialPointsDataFrame` object, since it is itself an object of class `ppp`, which can not be returned to QGIS.

If your algorithm generates raster layers, the way they are saved will depend on whether or not you have used the `##dontuserasterpackage` option. If you have used it, layers are saved using the `writeGDAL()` method. If not, the `writeRaster()` method from the `raster` package will be used.

If you have used the `##pass_filenames` option, outputs are generated using the `raster` package (with `writeRaster()`).

If your algorithm does not generate a layer, but a text result in the console instead, you have to indicate that you want the console to be shown once the execution is finished. To do so, just start the command lines that produce the results you want to print with the > ('greater than') sign. Only output from lines prefixed with > are shown. For instance, here is the description file of an algorithm that performs a normality test on a given field (column) of the attributes of a vector layer:

```
##layer=vector
##field=field layer
##nortest=group
library(nortest)
>lillie.test(layer[[field]])
```

The output of the last line is printed, but the output of the first is not (and neither are the outputs from other command lines added automatically by QGIS).

If your algorithm creates any kind of graphics (using the `plot()` method), add the following line (`output_plots_to_html` used to be `showplots`):

```
##output_plots_to_html
```

This will cause QGIS to redirect all R graphical outputs to a temporary file, which will be opened once R execution has finished.

Both graphics and console results will be available through the processing results manager.

For more information, please check the R scripts in the official QGIS collection (you download and install them using the *QGIS Resource Sharing* plugin, as explained in *Adding R scripts from the QGIS collection*). Most of them are rather simple and will greatly help you understand how to create your own scripts.

Белешка

The `sf`, `rgdal` and `raster` libraries are loaded by default, so you do not have to add the corresponding `library()` commands. However, other libraries that you might need have to be explicitly loaded by typing: `library(ggplot2)` (to load the `ggplot2` library). If the package is not already installed on your machine, Processing will try to download and install it. In this way the package will also become available in R Standalone. **Be aware** that if the package has to be downloaded, the script may take a long time to run the first time.

R libraries installed when running sf_test

The R script `sp_test` tries to load the R packages `sp` and `raster`.

The R script *sf_test* tries to load `sf` and `raster`. If these two packages are not installed, R may try to load and install them (and all the libraries that they depend on).

The following R libraries end up in `~/.local/share/QGIS/QGIS3/profiles/default/processing/rscripts` after `sf_test` has been run from the Processing Toolbox on Ubuntu with version 2.0 of the *Processing R Provider* plugin and a fresh install of *R* 3.4.4 (*apt* package `r-base-core` only):

```
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.

By default, the Processing framework tries to configure its GRASS connector to use the GRASS distribution that ships along with QGIS. This should work without problems for most systems, but if you experience problems, you might have to configure the GRASS connector manually. Also, if you want to use a different GRASS installation, you can change the setting to point to the folder where the other version is installed. GRASS 7 is needed for algorithms to work correctly.

If you are running Linux, you just have to make sure that GRASS is correctly installed, and that it can be run without problem from a terminal window.

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

To use LAStools in QGIS, you need to download and install LAStools on your computer and install the LAStools plugin (available from the official repository) in QGIS.

On Linux platforms, you will need Wine to be able to run some of the tools.

LAStools is activated and configured in the Processing options (*Settings* ► *Options*, *Processing* tab, *Providers* ► *LAStools*), where you can specify the location of LAStools (*LAStools folder*) and Wine (*Wine folder*). On Ubuntu,

the default Wine folder is `/usr/bin`.

OTB Applications

OTB applications are fully supported within the QGIS Processing framework.

OTB (Orfeo ToolBox) is an image processing library for remote sensing data. It also provides applications that provide image processing functionalities. The list of applications and their documentation are available in OTB CookBook

Белешка

Note that OTB is not distributed with QGIS and needs to be installed separately. Binary packages for OTB can be found on the download page.

To configure QGIS processing to find the OTB library:

1. Open the processing settings: *Settings* ► *Options* ► *Processing*
2. You can see `OTB` under *Providers* menu:
    1. Expand the *OTB* entry
    2. Set the *OTB folder*. This is the location of your OTB installation.
    3. Set the *OTB application folder*. This is the location of your OTB applications (`<PATH_TO_OTB_INSTALLATION>/lib/otb/applications`)
    4. Click *OK* to save the settings and close the dialog.

If settings are correct, OTB algorithms will be available in the *Processing Toolbox*.

Documentation of OTB settings available in QGIS Processing

- **OTB folder**: This is the directory where OTB is available.
- **OTB application folder**: This is the location(s) of OTB applications.

  Multiple paths are allowed.
- **Logger level** (optional): Level of logger to use by OTB applications.

  The level of logging controls the amount of detail printed during algorithm execution. Possible values for logger level are `INFO`, `WARNING`, `CRITICAL`, `DEBUG`. This value is `INFO` by default. This is an advanced user configuration.
- **Maximum RAM to use** (optional): by default, OTB applications use all available system RAM.

  You can, however, instruct OTB to use a specific amount of RAM (in MB) using this option. A value of 256 is ignored by the OTB processing provider. This is an advanced user configuration.
- **Geoid file** (optional): Path to the geoid file.

  This option sets the value of the elev.dem.geoid and elev.geoid parameters in OTB applications. Setting this value globally enables users to share it across multiple processing algorithms. Empty by default.
- **SRTM tiles folder** (optional): Directory where SRTM tiles are available.

  SRTM data can be stored locally to avoid downloading of files during processing. This option sets the value of elev.dem.path and elev.dem parameters in OTB applications. Setting this value globally enables users to share it across multiple processing algorithms. Empty by default.

Compatibility and Troubleshoot

Starting from OTB 6.6.1, new releases of OTB are made compatible with at least the latest QGIS version available at that time.

If you have issues with OTB applications in QGIS Processing, please open an issue on the OTB bug tracker, using the `qgis` label.

Additional information about OTB and QGIS can be found in OTB Cookbook.

ГЛАВА 24

Provajderi i algoritmi obrade

Algoritmi obrade i njihovi parametri (kako su predstavljeni u korisničkom interfejsu) dokumentovani su ovde.

24.1 Dobavljač QGIS algoritma

Provajder QGIS algoritama implementira razne operacije analize i geoobrade koristeći uglavnom samo QGIS API. Tako da će skoro svi algoritmi ovog provajdera raditi „odmah po instalaciji“ bez ikakve dodatne konfiguracije

Ovaj provajder uključuje neke algoritme iz dodataka, a takođe dodaje i sopstvene algoritme.

24.1.1 3D pločice

Pretvaranje B3DM u GLTF

`Dodato u 3.34`

Pretvara datoteke iz zastarelog `.B3DM` formata u `.GLTF` ili `.GLB`.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni B3DM	`INPUT`	[3D pločica]	Ulazna datoteka koja se pretvara.
Izlazna datoteka	`OUTPUT`	[3D pločica] Podrazumevano: `[Save to temporary file]`	Zadajte izlaznu datoteku 3D pločice. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazna datoteka	`OUTPUT`	[3D pločica]	Izlazni sloj u `.GLTF` formatu.

Python kod

ID algoritma: `native:b3dmtogltf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvaranje GLTF u vektorske objekte

`Dodato u 3.34`

Pretvara sadržaj `.GLTF` ili `.GLB` datoteke u standardne formate vektorskih slojeva.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni GLTF	`INPUT`	[3D pločica]	Ulazna datoteka koja se pretvara.
Izlazni poligoni Opciono	`OUTPUT_POLYGONS`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj u koji se smeštaju poligonski objekti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Izlazne linije Opciono	`OUTPUT_LINES`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj u koji se smeštaju linijski objekti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni poligoni	`OUTPUT_POLYGONS`	[vektor: poligon]	Vektorski sloj koji sadrži poligonske objekte iz ulaznog sloja.
Izlazne linije	`OUTPUT_LINES`	[vektor: linija]	Vektorski sloj koji sadrži linijske objekte iz ulaznog sloja.

Python kod

ID algoritma: `native:gltftovector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.2 Kartografija

Poravnavanje tačaka prema objektima

Izračunava rotaciju potrebnu da se tačkasti objekti poravnaju sa njima najbližim objektom iz drugog referentnog sloja. U izlazni sloj dodaje se novo polje koje se popunjava uglom (u stepenima, u smeru kazaljke na satu) do najbližeg referentnog objekta.

Opciono se simbologija izlaznog sloja može podesiti da automatski koristi izračunato polje rotacije za rotiranje simbola markera. Po želji se može zadati najveće rastojanje koje se koristi pri poravnavanju tačaka, kako bi se izbeglo poravnavanje izolovanih tačaka prema udaljenim objektima.

Путоказ

Ovaj algoritam namenjen je slučajevima kao što je poravnavanje tačkastih simbola zgrada prema pravcu najbližeg puta.

☑ Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Tačkasti objekti za koje se izračunava rotacija
Referentni sloj	`REFERENCE_LAYER`	[vektor: geometrija]	Sloj u kome se traži najbliži objekat za izračunavanje rotacije
Najveće rastojanje koje se uzima u obzir Opciono	`MAX_DISTANCE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se u okviru ovog rastojanja ne pronađe referentni objekat, tačkastom objektu se ne dodeljuje rotacija.
Naziv polja sa uglom	`FIELD_NAME`	[niska] Podrazumevano: 'rotation'	Polje u koje se čuva vrednost rotacije.
Automatski primeni simbologiju	`APPLY_SYMBOLOGY`	[logička vrednost] Podrazumevano: True	Rotira simbol markera objekata koristeći vrednost polja sa uglom

наставак на следећој страни

Табела 24.5 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Poravnati sloj	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Save to temporary file]`	Zadajte rotirani izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Poravnati sloj	`OUTPUT`	[vektor: tačka]	Tačkasti sloj sa dodatim poljem rotacije. Ako se učita u QGIS, podrazumevano mu se primenjuje simbologija ulaznog sloja, sa rotacijom simbola markera definisanom podacima.

Python kod

ID algoritma: `native:angletonearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Objedinjavanje baza stilova

Objedinjuje više QGIS baza stilova u jednu bazu stilova. Ako u različitim izvornim bazama postoje stavke istog tipa i istog naziva, biće preimenovane tako da imaju jedinstvene nazive u objedinjenoj izlaznoj bazi.

See also

Kreiranje baze stilova iz projekta

Parametri

Oznaka	Назив	Tip	Opis
Ulazne baze podataka	`INPUT`	[datoteka] [lista]	Datoteke koje sadrže QGIS stavke stila

наставак на следећој страни

Табела 24.7 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Objekti koji se objedinjuju	`OBJECTS`	[nabrajanje] [lista]	Tipovi stavki stila iz ulaznih baza koje želite da smestite u novu bazu. To mogu biti: • 0 — *Simboli* • 1 — *Rampe boja* • 2 — *Formati teksta* • 3 — *Podešavanja oznaka*
Izlazna baza stilova	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Izlazna `.XML` datoteka koja objedinjuje izabrane stavke stila. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Broj rampi boja	`COLORRAMPS`	[numerički: ceo broj]	Broj rampi boja
Broj podešavanja oznaka	`LABELSETTINGS`	[numerički: ceo broj]	Broj podešavanja oznaka
Izlazna baza stilova	`OUTPUT`	[datoteka]	Izlazna `.XML` datoteka koja objedinjuje izabrane stavke stila
Broj simbola	`SYMBOLS`	[numerički: ceo broj]	Broj simbola
Broj formata teksta	`TEXTFORMATS`	[numerički: ceo broj]	Broj formata teksta

Python kod

ID algoritma: `native:combinestyles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje kategorizovanog iscrtavača iz stilova

Postavlja iscrtavač vektorskog sloja na kategorizovani iscrtavač koristeći odgovarajuće simbole iz baze stilova. Ako se ne zada datoteka stila, koriste se simboli iz korisnikove tekuće *biblioteke simbola*.

Zadati izraz ili polje koristi se za kreiranje kategorija iscrtavača. Svaka kategorija pojedinačno se uparuje sa simbolima koji postoje u zadatoj QGIS XML bazi stilova. Kad god se pronađe odgovarajući naziv simbola, simbol kategorije biće postavljen na taj upareni simbol.

Po želji, izlazi mogu biti i tabele sa listama kategorija koje nisu mogle da se upare sa simbolima i simbola koji nisu upareni sa kategorijama.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	INPUT	[vektor: bilo koji]	Vektorski sloj na koji se primenjuje kategorizovani stil
Kategorizuj pomoću izraza	FIELD	[izraz]	Polje ili izraz za kategorizaciju objekata
Baza stilova (ostavite prazno da biste koristili sačuvane simbole)	STYLE	[datoteka]	Datoteka (.XML) sa simbolima koji se primenjuju na kategorije ulaznog sloja. Datoteka se može dobiti alatkom *Deljenje simbola* u Upravljaču stilovima. Ako se ne zada datoteka, koristi se lokalna QGIS biblioteka simbola.
Uparuj nazive simbola uz razlikovanje malih i velikih slova	CASE_SENSITIVE	[logička vrednost] Podrazumevano: False	Ako je True (označeno), poređenje naziva kategorija i simbola razlikuje mala i velika slova
Zanemari nealfanumeričke znakove pri uparivanju	TOLERANT	[logička vrednost] Podrazumevano: False	Ako je True (označeno), nealfanumerički znakovi u nazivima kategorija i simbola biće zanemareni, čime se dozvoljava veća tolerancija pri uparivanju.
Neuparene kategorije Opciono	NON_MATCHING_CAT	[vektor: tabela] Podrazumevano: [Skip output]	Izlazna tabela za kategorije koje se ne uparuju ni sa jednim simbolom u bazi. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Neupareni nazivi simbola Opciono	NON_MATCHING_SYM	[vektor: tabela] Podrazumevano: [Skip output]	Izlazna tabela za simbole iz zadate baze stilova koji se ne uparuju ni sa jednom kategorijom. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Neuparene kategorije	NON_MATCHING_CAT	[vektor: tabela]	Izlistava kategorije koje nisu mogle da se upare ni sa jednim simbolom u zadatoj bazi stilova
Neupareni nazivi simbola	NON_MATCHING_SYM	[vektor: tabela]	Izlistava simbole iz zadate baze stilova koji nisu mogli da se upare ni sa jednom kategorijom
Kategorizovani sloj	OUTPUT	[isto kao ulaz]	Ulazni vektorski sloj sa primenjenim kategorizovanim stilom. Novi sloj se ne pravi.

Python kod

ID algoritma: `native:categorizeusingstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje baze stilova iz projekta

Izdvaja sve objekte stila (simbole, rampe boja, formate teksta i podešavanja oznaka) iz QGIS projekta.

Izdvojeni simboli čuvaju se u QGIS bazu stilova (format `XML`), kojom se može upravljati i koja se može uvesti kroz dijalog *Upravljač stilovima*.

See also

Objedinjavanje baza stilova

Parametri

Oznaka	Назив	Tip	Opis
Ulazni projekat (ostavite prazno za tekući) Opciono	`INPUT`	[datoteka]	Datoteka QGIS projekta iz koje se izdvajaju stavke stila
Objekti koji se izdvajaju	`OBJECTS`	[nabrajanje] [lista]	Tipovi stavki stila iz ulaznog projekta koje želite da smestite u novu bazu. To mogu biti: • 0 — *Simboli* • 1 — *Rampe boja* • 2 — *Formati teksta* • 3 — *Podešavanja oznaka*
Izlazna baza stilova	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Zadajte izlaznu `.XML` datoteku za izabrane stavke stila. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Broj rampi boja	`COLORRAMPS`	[numerički: ceo broj]	Broj rampi boja
Broj podešavanja oznaka	`LABELSETTINGS`	[numerički: ceo broj]	Broj podešavanja oznaka
Izlazna baza stilova	`OUTPUT`	[datoteka]	Izlazna `.XML` datoteka za izabrane stavke stila

наставак на следећој страни

Табела 24.12 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj simbola	SYMBOLS	[numerički: ceo broj]	Broj simbola
Broj formata teksta	TEXTFORMATS	[numerički: ceo broj]	Broj formata teksta

Python kod

ID algoritma: native:stylefromproject

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz atlas rasporeda kao slike

Izvozi atlas rasporeda za štampu kao datoteke slika (npr. PNG ili JPEG slike).

Ako je postavljen sloj pokrivenosti, podešavanja atlasa izabranog rasporeda koja ovaj algoritam izlaže biće prepisana. U tom slučaju, prazan izraz za filtriranje ili sortiranje isključuje ta podešavanja.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Atlas raspored	LAYOUT	[raspored]	Raspored koji se izvozi
Sloj pokrivenosti Opciono	COVERAGE_LAYER	[vektor: geometrija]	Sloj koji se koristi za generisanje atlasa
Izraz za filtriranje	FILTER_EXPRESSION	[izraz]	Izraz koji se koristi za filtriranje objekata atlasa
Izraz za sortiranje Opciono	SORTBY_EXPRESSION	[izraz]	Izraz koji se koristi za sortiranje objekata atlasa
Obrni redosled sortiranja	SORTBY_REVERSE	[logička vrednost] Podrazumevano: False	Određuje da li se sortiranje obrće. Koristi se kada je zadat izraz za sortiranje.
Izraz za naziv izlazne datoteke	FILENAME_EXPRESSION	[izraz] Podrazumevano: 'output_'||@atlas_feature	Izraz koji se koristi za generisanje naziva datoteka
Izlazni folder	FOLDER	[folder]	Odredišni folder u kome će se generisati slike

Napredni parametri

Oznaka	Назив	Tip	Opis
Slojevi karte koji se dodeljuju otključanim stavkama karte Opciono	`LAYERS`	[nabrajanje] [sloj]	Slojevi koji se prikazuju u stavkama karte čiji sadržaj nije zaključan
Format slike	`EXTENSION`	[nabrajanje] Podrazumevano: png	Format datoteke generisanih izlaza. Lista dostupnih formata razlikuje se u zavisnosti od operativnog sistema i instaliranih drajvera.
DPI Opciono	`DPI` Podrazumevano: Nije postavljeno	[numerički: decimalni]	DPI izlaznih datoteka. Ako se ne postavi, koristi se vrednost iz podešavanja rasporeda za štampu.
Generiši world datoteku	`GEOREFERENCE`	[logička vrednost] Podrazumevano: True	Određuje da li se generiše world datoteka
Izvezi RDF metapodatke	`INCLUDE_METADATA`	[logička vrednost] Podrazumevano: True	Određuje da li se generišu RDF metapodaci (naslov, autor, …)
Omogući antialijasing	`ANTIALIAS`	[logička vrednost] Podrazumevano: True	Određuje da li se omogućava antialijasing

Izlazi

Oznaka	Назив	Tip	Opis
Datoteka slike	`OUTPUT`	[datoteka]	Datoteke slika koje generiše atlas raspored

Python kod

ID algoritma: `native:atlaslayouttoimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz atlas rasporeda kao PDF (više datoteka)

Izvozi atlas rasporeda za štampu u više PDF datoteka.

Ako je postavljen sloj pokrivenosti, podešavanja atlasa izabranog rasporeda koja ovaj algoritam izlaže biće prepisana. U tom slučaju, prazan izraz za filtriranje ili sortiranje isključuje ta podešavanja.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Atlas raspored	`LAYOUT`	[raspored]	Raspored koji se izvozi

наставак на следећој страни

Табела 24.16 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Sloj pokrivenosti Opciono	COVERAGE_LAYER	[vektor: geometrija]	Sloj koji se koristi za generisanje atlasa
Izraz za filtriranje	FILTER_EXPRESSION	[izraz]	Izraz koji se koristi za filtriranje objekata atlasa
Izraz za sortiranje Opciono	SORTBY_EXPRESSION	[izraz]	Izraz koji se koristi za sortiranje objekata atlasa
Obrni redosled sortiranja	SORTBY_REVERSE	[logička vrednost] Podrazumevano: False	Određuje da li se sortiranje obrće. Koristi se kada je zadat izraz za sortiranje.
Naziv izlazne datoteke Opciono	OUTPUT_FILENAME	[izraz]	Šablon naziva izlaznih PDF datoteka.
Izlazni folder	OUTPUT_FOLDER	[folder]	Odredišni folder za izlazne PDF datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Slojevi karte koji se dodeljuju otključanim stavkama karte Opciono	LAYERS	[nabrajanje] [sloj]	Slojevi koji se prikazuju u stavkama karte čiji sadržaj nije zaključan
DPI Opciono	DPI Podrazumevano: Nije postavljeno	[numerički: decimalni]	DPI izlaznih datoteka. Ako se ne postavi, koristi se vrednost iz podešavanja rasporeda za štampu.
Uvek izvezi kao vektore	FORCE_VECTOR	[logička vrednost] Podrazumevano: False	Određuje da li se vektorski podaci ostavljaju kao vektori
Uvek izvezi kao raster	FORCE_RASTER	[logička vrednost] Podrazumevano: False	Prinudno rasterizuje sve stavke na karti. Ovaj parametar ima prednost nad parametrom FORCE_VECTOR.
Dodaj podatke o georeferenciranju	GEOREFERENCE	[logička vrednost] Podrazumevano: True	Određuje da li se generiše world datoteka
Izvezi RDF metapodatke	INCLUDE_METADATA	[logička vrednost] Podrazumevano: True	Određuje da li se generišu RDF metapodaci (naslov, autor, …)
Onemogući izvoz rasterskih slojeva u pločicama	DISABLE_TILED	[logička vrednost] Podrazumevano: False	Određuje da li se raster deli na pločice
Pojednostavi geometrije radi smanjenja veličine izlazne datoteke	SIMPLIFY	[logička vrednost] Podrazumevano: True	Određuje da li se geometrije pojednostavljuju radi smanjenja veličine izlazne datoteke
Izvoz teksta	TEXT_FORMAT	[nabrajanje] Podrazumevano: 0	Određuje da li se tekst izvozi kao putanje ili kao tekstualni objekti. *Moguće opcije* su: • 0 - Uvek izvezi tekst kao putanje (preporučeno) • 1 - Uvek izvezi tekst kao tekstualne objekte • 2 - Radije izvezi tekst kao tekstualne objekte — Dodato u 3.40

наставак на следећој страни

Табела 24.17 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Kompresija slike	`IMAGE_COMPRESSION`	[nabrajanje] Podrazumevano: 0	Određuje nivo kompresije slike i koliko je datoteka pogodna za štampu ili postprodukciju u spoljnim programima. Moguće opcije su: • 0 - Sa gubicima (JPEG) • 1 - Bez gubitaka

Izlazi

Oznaka	Назив	Tip	Opis
PDF datoteka	`OUTPUT`	[datoteka]	PDF datoteka koja odgovara izvezenom atlas rasporedu

Python kod

ID algoritma: `native:atlaslayouttomultiplepdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz atlas rasporeda kao PDF (jedna datoteka)

Izvozi atlas rasporeda za štampu kao jednu PDF datoteku.

Ako je postavljen sloj pokrivenosti, podešavanja atlasa izabranog rasporeda koja ovaj algoritam izlaže biće prepisana. U tom slučaju, prazan izraz za filtriranje ili sortiranje isključuje ta podešavanja.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Atlas raspored	`LAYOUT`	[raspored]	Raspored koji se izvozi
Sloj pokrivenosti Opciono	`COVERAGE_LAYER`	[vektor: geometrija]	Sloj koji se koristi za generisanje atlasa
Izraz za filtriranje	`FILTER_EXPRESSION`	[izraz]	Izraz koji se koristi za filtriranje objekata atlasa
Izraz za sortiranje Opciono	`SORTBY_EXPRESSION`	[izraz]	Izraz koji se koristi za sortiranje objekata atlasa
Obrni redosled sortiranja	`SORTBY_REVERSE`	[logička vrednost] Podrazumevano: False	Određuje da li se sortiranje obrće. Koristi se kada je zadat izraz za sortiranje.
PDF datoteka	`OUTPUT`	[datoteka] Podrazumevano: [Save to temporary file]	Naziv (sa putanjom) izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Slojevi karte koji se dodeljuju otključanim stavkama karte Opciono	`LAYERS`	[nabrajanje] [sloj]	Slojevi koji se prikazuju u stavkama karte čiji sadržaj nije zaključan
DPI Opciono	`DPI` Podrazumevano: Nije postavljeno	[numerički: decimalni]	DPI izlaznih datoteka. Ako se ne postavi, koristi se vrednost iz podešavanja rasporeda za štampu.
Uvek izvezi kao vektore	`FORCE_VECTOR`	[logička vrednost] Podrazumevano: False	Određuje da li se vektorski podaci ostavljaju kao vektori
Uvek izvezi kao raster	`FORCE_RASTER`	[logička vrednost] Podrazumevano: False	Prinudno rasterizuje sve stavke na karti. Ovaj parametar ima prednost nad parametrom `FORCE_VECTOR`.
Dodaj podatke o georeferenciranju	`GEOREFERENCE`	[logička vrednost] Podrazumevano: True	Određuje da li se generiše world datoteka
Izvezi RDF metapodatke	`INCLUDE_METADATA`	[logička vrednost] Podrazumevano: True	Određuje da li se generišu RDF metapodaci (naslov, autor, …)
Onemogući izvoz rasterskih slojeva u pločicama	`DISABLE_TILED`	[logička vrednost] Podrazumevano: False	Određuje da li se raster deli na pločice
Pojednostavi geometrije radi smanjenja veličine izlazne datoteke	`SIMPLIFY`	[logička vrednost] Podrazumevano: True	Određuje da li se geometrije pojednostavljuju radi smanjenja veličine izlazne datoteke
Izvoz teksta	`TEXT_FORMAT`	[nabrajanje] Podrazumevano: 0	Određuje da li se tekst izvozi kao putanje ili kao tekstualni objekti. *Moguće opcije* su: • 0 - Uvek izvezi tekst kao putanje (preporučeno) • 1 - Uvek izvezi tekst kao tekstualne objekte • 2 - Radije izvezi tekst kao tekstualne objekte — `Dodato u 3.40`
Kompresija slike	`IMAGE_COMPRESSION`	[nabrajanje] Podrazumevano: 0	Određuje nivo kompresije slike i koliko je datoteka pogodna za štampu ili postprodukciju u spoljnim programima. Moguće opcije su: • 0 - Sa gubicima (JPEG) • 1 - Bez gubitaka

Izlazi

Oznaka	Назив	Tip	Opis
PDF datoteka	`OUTPUT`	[datoteka]	PDF datoteka koja odgovara izvezenom atlas rasporedu

Python kod

ID algoritma: `native:atlaslayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz rasporeda za štampu kao slike

Izvozi raspored za štampu kao datoteku slike (npr. PNG ili JPEG slike)

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Raspored za štampu	`LAYOUT`	[raspored]	Raspored koji se izvozi
Datoteka slike	`OUTPUT`	[datoteka] Podrazumevano: [Save to temporary file]	Naziv (sa putanjom) izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Slojevi karte koji se dodeljuju otključanim stavkama karte Opciono	`LAYERS`	[nabrajanje] [sloj]	Slojevi koji se prikazuju u stavkama karte čiji sadržaj nije zaključan
DPI Opciono	`DPI` Podrazumevano: Nije postavljeno	[numerički: decimalni]	DPI izlaznih datoteka. Ako se ne postavi, koristi se vrednost iz podešavanja rasporeda za štampu.
Generiši world datoteku	`GEOREFERENCE`	[logička vrednost] Podrazumevano: True	Određuje da li se generiše world datoteka
Izvezi RDF metapodatke	`INCLUDE_METADATA`	[logička vrednost] Podrazumevano: True	Određuje da li se generišu RDF metapodaci (naslov, autor, …)
Omogući antialijasing	`ANTIALIAS`	[logička vrednost] Podrazumevano: True	Određuje da li se omogućava antialijasing

Izlazi

Oznaka	Назив	Tip	Opis
Datoteka slike	OUTPUT	[datoteka]	Datoteka slike koja odgovara izvezenom rasporedu za štampu

Python kod

ID algoritma: native:printlayouttoimage

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz rasporeda za štampu kao PDF

Izvozi raspored za štampu kao PDF datoteku.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Raspored za štampu	LAYOUT	[raspored]	Raspored koji se izvozi
PDF datoteka	OUTPUT	[datoteka] Podrazumevano: [Save to temporary file]	Naziv (sa putanjom) izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Slojevi karte koji se dodeljuju otključanim stavkama karte Opciono	LAYERS	[nabrajanje] [sloj]	Slojevi koji se prikazuju u stavkama karte čiji sadržaj nije zaključan
DPI Opciono	DPI Podrazumevano: Nije postavljeno	[numerički: decimalni]	DPI izlaznih datoteka. Ako se ne postavi, koristi se vrednost iz podešavanja rasporeda za štampu.
Uvek izvezi kao vektore	FORCE_VECTOR	[logička vrednost] Podrazumevano: False	Određuje da li se vektorski podaci ostavljaju kao vektori
Uvek izvezi kao raster	FORCE_RASTER	[logička vrednost] Podrazumevano: False	Prinudno rasterizuje sve stavke na karti. Ovaj parametar ima prednost nad parametrom FORCE_VECTOR.
Dodaj podatke o georeferenciranju	GEOREFERENCE	[logička vrednost] Podrazumevano: True	Određuje da li se generiše world datoteka

наставак на следећој страни

Табела 24.26 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izvezi RDF metapodatke	`INCLUDE_METADATA`	[logička vrednost] Podrazumevano: True	Određuje da li se generišu RDF metapodaci (naslov, autor, …)
Onemogući izvoz rasterskih slojeva u pločicama	`DISABLE_TILED`	[logička vrednost] Podrazumevano: False	Određuje da li se raster deli na pločice
Pojednostavi geometrije radi smanjenja veličine izlazne datoteke	`SIMPLIFY`	[logička vrednost] Podrazumevano: True	Određuje da li se geometrije pojednostavljuju radi smanjenja veličine izlazne datoteke
Izvoz teksta	`TEXT_FORMAT`	[nabrajanje] Podrazumevano: 0	Određuje da li se tekst izvozi kao putanje ili kao tekstualni objekti. *Moguće opcije* su: • 0 - Uvek izvezi tekst kao putanje (preporučeno) • 1 - Uvek izvezi tekst kao tekstualne objekte • 2 - Radije izvezi tekst kao tekstualne objekte — `Dodato u 3.40`
Kompresija slike	`IMAGE_COMPRESSION`	[nabrajanje] Podrazumevano: 0	Određuje nivo kompresije slike i koliko je datoteka pogodna za štampu ili postprodukciju u spoljnim programima. Moguće opcije su: • 0 - Sa gubicima (JPEG) • 1 - Bez gubitaka
Izvezi slojeve kao zasebne PDF datoteke	`SEPARATE_LAYERS`	[logička vrednost] Podrazumevano: False	Ako je True, praviće se zasebna PDF datoteka po sloju po stavci karte u rasporedu. Dodatno se mogu praviti zasebne PDF datoteke za druge složene stavke rasporeda, čime se dobija skup PDF datoteka koje sadrže logičke atomske komponente rasporeda.

Izlazi

Oznaka	Назив	Tip	Opis
PDF datoteka	`OUTPUT`	[datoteka]	PDF datoteke koje odgovaraju izvezenom rasporedu za štampu

Python kod

ID algoritma: `native:printlayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje oznaka

Izdvaja podatke o oznakama sa iscrtane karte pri zadatom obuhvatu i razmeri.

Ako se zada tema karte, iscrtana karta odgovaraće vidljivosti i simbologiji te teme. Ako se ostavi prazno, koriste se svi vidljivi slojevi iz projekta. Izdvojeni podaci o oznakama obuhvataju: položaj (dat kao geometrije tačaka), pridruženi naziv sloja i ID objekta, tekst oznake, rotaciju (u stepenima, u smeru kazaljke na satu), poravnanje više redova i detalje o fontu.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Obuhvat karte	`EXTENT`	[obuhvat]	Obuhvat karte iz koje se izdvajaju oznake Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Razmera karte	`SCALE`	[razmera]	Izdvojene oznake biće iscrtane koristeći svojstva postavljena na ovoj razmeri.
Tema karte Opciono	`MAP_THEME`	[tema karte]	Tema karte koja prikazuje slojeve iz kojih se izdvajaju oznake. Ako se ne postavi, izdvajaju se oznake trenutno vidljivih slojeva.
Uključi nepostavljene oznake	`INCLUDE_UNPLACED`	[logička vrednost] Podrazumevano: True	Zadajte da li se izdvajaju sve preklapajuće oznake, uključujući i one u sukobu (dakle nepostavljene).
Izdvojene oznake	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za obuhvate. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Rezolucija karte (u DPI) Opciono	`DPI` Podrazumevano: 96.0	[numerički: decimalni]	

Izlazi

Oznaka	Назив	Tip	Opis
Izdvojene oznake	`OUTPUT`	[vektor: tačka]	Tačkasti vektorski sloj koji predstavlja preuzete oznake. Svaki objekat ima atribute koji identifikuju njegov izvor (sloj, ID objekta) i dodeljena svojstva označavanja (tekst, font, veličina, rotacija, …). Na sloj se primenjuje i podrazumevani stil sa označavanjem i praznim simbolom. **Упозорење** Pošto neka od generisanih polja imaju naziv duži od 10 znakova, korišćenje ESRI shapefile formata (`.SHP`) za čuvanje izlaza može dovesti do neočekivanog iscrtavanja pri učitavanju sloja u QGIS.

Python kod

ID algoritma: `native:extractlabels`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Obuhvat karte rasporeda za štampu u sloj

Pravi poligonski sloj koji sadrži obuhvat stavke (ili stavki) karte u rasporedu za štampu, sa atributima koji zadaju veličinu karte (u jedinicama rasporeda, tj. jedinicama *referentne karte*), razmeru i rotaciju.

Ako se zada parametar stavke karte, izvozi se samo odgovarajući obuhvat karte. Ako se ne zada, izvoze se svi obuhvati karata iz rasporeda.

Opciono se može zadati određeni izlazni CRS. Ako se ne zada, koristi se originalni CRS stavke karte.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Raspored za štampu	`LAYOUT`	[nabrajanje]	Raspored za štampu u tekućem projektu
Stavka karte Opciono	`MAP`	[nabrajanje] Podrazumevano: *Sve stavke karte*	Stavke karte čije podatke želite da izdvojite. Ako se nijedna ne zada, obrađuju se sve stavke karte.
Obuhvat	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za obuhvate. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Zameni CRS Opciono	`CRS`	[crs] Podrazumevano: *CRS rasporeda*	Izaberite CRS sloja u kome će podaci biti prijavljeni.

Izlazi

Oznaka	Назив	Tip	Opis
Visina karte	`HEIGHT`	[numerički: decimalni]	
Obuhvat	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj koji sadrži obuhvate svih ulaznih stavki karte iz rasporeda
Rotacija karte	`ROTATION`	[numerički: decimalni]	
Razmera karte	`SCALE`	[numerički: decimalni]	
Širina karte	`WIDTH`	[numerički: decimalni]	

Python kod

ID algoritma: `native:printlayoutmapextenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavljanje stila sloja

Primenjuje zadati stil na sloj. Stil mora biti definisan u `QML` datoteci.

Ne kreiraju se novi izlazi: stil se odmah dodeljuje sloju.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[sloj]	Ulazni sloj na koji želite da primenite stil
Datoteka stila	`STYLE`	[datoteka]	Putanja do `.qml` datoteke stila

Izlazi

Oznaka	Назив	Tip	Opis
	`OUTPUT`	[isto kao ulaz]	Ulazni sloj sa dodeljenim novim stilom. Novi sloj se ne kreira.

Python kod

ID algoritma: `native:setlayerstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

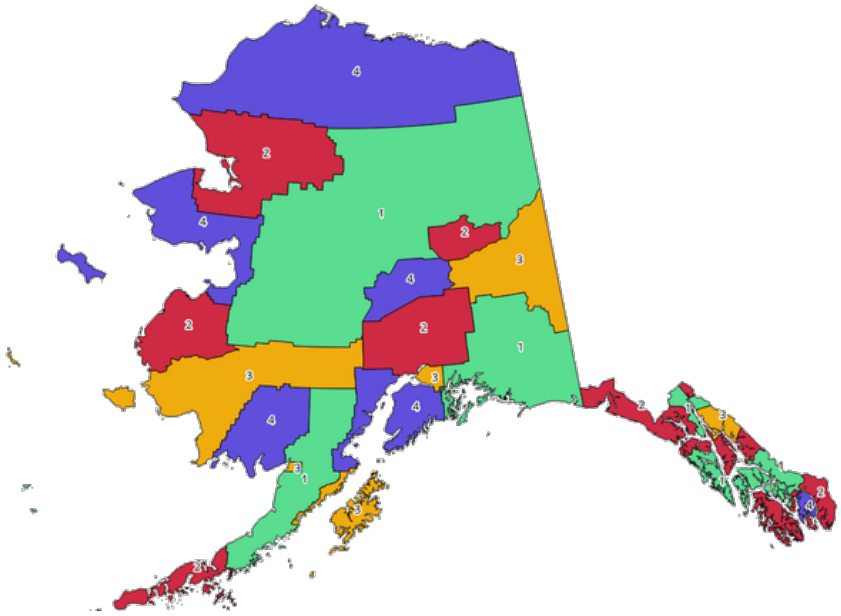
Topološko bojenje

Dodeljuje indeks boje poligonskim objektima tako da nijedan susedni poligon nema isti indeks boje, uz minimizovanje potrebnog broja boja.

Algoritam omogućava izbor metoda koji se koristi pri dodeli boja.

Po želji se može zadati najmanji broj boja. Indeks boje čuva se u novom atributu pod nazivom **color_id**.

Sledeći primer prikazuje algoritam sa izabrane četiri različite boje; kao što vidite, svaka klasa boje ima isti broj objekata.



Слика24.1: Primer topoloških boja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski sloj
Najmanji broj boja	`MIN_COLORS`	[numerički: ceo broj] Podrazumevano: 4	Najmanji broj boja koje se dodeljuju. Minimum 1, maksimum 1000.
Najmanje rastojanje između objekata	`MIN_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Sprečava da bliski (ali nedodirujući) objekti dobiju iste boje. Minimum 0.0.
Uravnoteži dodelu boja	`BALANCE`	[nabrajanje] Podrazumevano: 0	Opcije su: • 0 — Po broju objekata Pokušava da dodeli boje tako da broj objekata dodeljenih svakom pojedinačnom indeksu boje bude uravnotežen. • 1 — Po dodeljenoj površini Dodeljuje boje tako da ukupna površina objekata dodeljenih svakoj boji bude uravnotežena. Ovaj režim može pomoći da veliki objekti ne učine jednu od boja dominantnijom na obojenoj karti. • 2 — Po rastojanju između boja Dodeljuje boje tako da se maksimizuje rastojanje između objekata iste boje. Ovaj režim pomaže da raspodela boja po karti bude ravnomernija.

наставак на следећој страни

Табела 24.36 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obojeno	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Obojeno	`OUTPUT`	[vektor: poligon]	Poligonski vektorski sloj sa dodatom kolonom `color_id`

Python kod

ID algoritma: `qgis:topologicalcoloring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Prenos napomena iz glavnog sloja

Prenosi sve *napomene* iz glavnog sloja napomena u projektu u novi sloj napomena. Položaj stavki se zatim može podesiti unutar niza slojeva.

Parametri

Oznaka	Назив	Tip	Opis
Naziv novog sloja	`LAYER_NAME`	[niska] Podrazumevano: 'Annotations'	Naziv sloja napomena koji se kreira

Izlazi

Oznaka	Назив	Tip	Opis
Naziv novog sloja	`OUTPUT`	[sloj]	Sloj sa stavkama iz glavnog sloja napomena

Python kod

ID algoritma: `native:transferannotationsfrommain`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

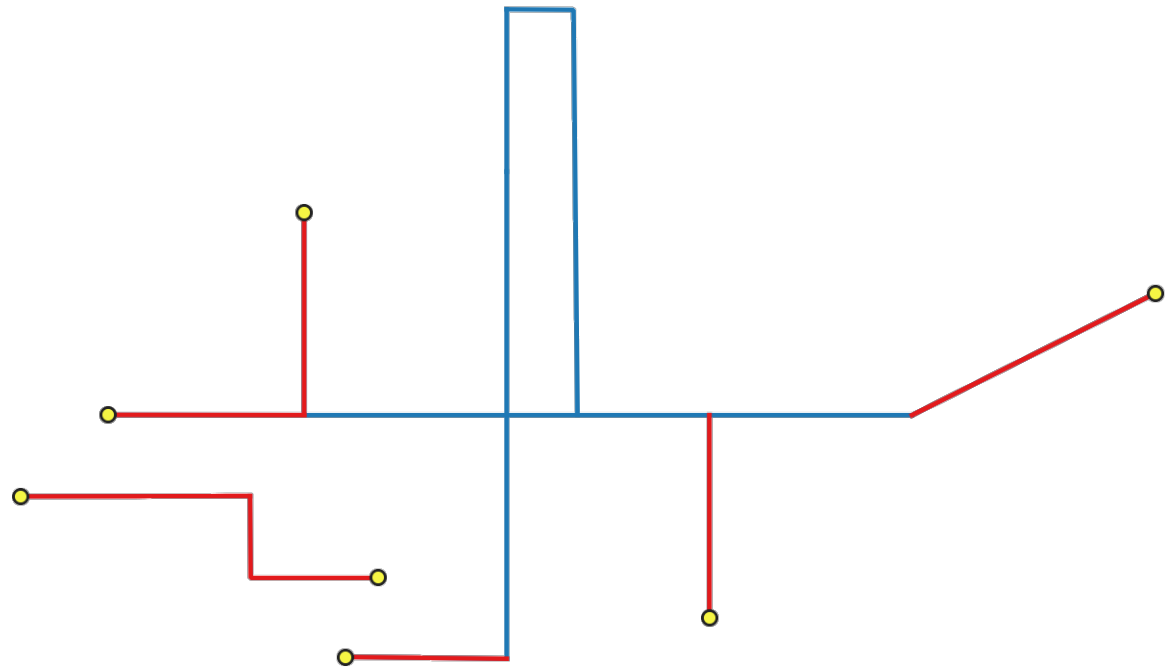
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.3 Check Geometry

Dangle-end lines

`Dodato u 3.42`

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.



Слика24.2: Identifying dangle-end features (in red) and vertices (yellow).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.

наставак на следећој страни

Табела 24.40 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dangle-end errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Dangle-end features Opciono	`OUTPUT`	[vektor: linija] Podrazumevano: `[Skip output]`	Line layer containing the identified dangle-end line features. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Dangle-end errors	`ERRORS`	[vektor: tačka]	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`	[vektor: linija]	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 kod

Algorithm ID: `native:checkgeometrydangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Degenerate polygons

`Dodato u 3.42`

Checks the polygons with less than 3 points, which are degenerate polygons. Degenerate polygons are errors.

Белешка

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)`.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Degenerate polygons errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the centroid of the degenerate polygons. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Degenerate polygons features Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Specification of the output layer containing the degenerate polygons. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Degenerate polygons errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:checkgeometrydegeneratepolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Duplicated geometries

`Dodato u 3.42`

Checks for duplicated geometries in a vector layer, and reports them as errors.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification

наставак на следећој страни

Табела 24.46 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Duplicate geometries errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Duplicate geometries Opciono	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Skip output]`	Specification of the output layer containing the geometries that were found to be duplicated. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Duplicate geometries errors	`ERRORS`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer containing the geometries that were found to be duplicated. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometryduplicate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Duplicated vertices

`Dodato u 3.42`

Checks for duplicated vertices in line or polygon geometries, and reports them as errors.

See also

Delete duplicated vertices, *Ukloni duplirana temena*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification

наставак на следећој страни

Табела 24.49 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Duplicated vertices errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Duplicated vertices features Opciono	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Skip output]`	Specification of the output layer containing the features with duplicated vertices. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Duplicated vertices errors	`ERRORS`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer containing the features with duplicated vertices. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometryduplicatenodes`

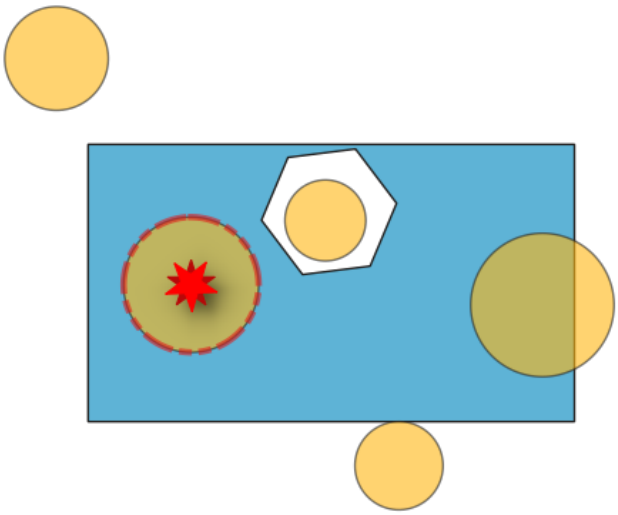
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Features inside polygon

`Dodato u 3.42`

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.



Слика24.3: Reporting errors on features inside a polygon.

See also

Brisanje objekata

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Point layer representing the contained features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.52 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Contained features Opciono	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Skip output]`	Specification of the output layer containing features within the checked polygons. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Errors from contained features	`ERRORS`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer containing features within the checked polygons. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometrycontained`

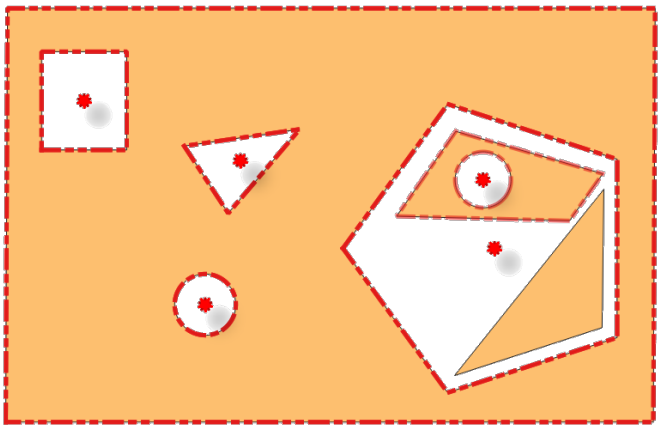
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Holes

`Dodato u 3.42`

Detects holes in polygon geometries and reports them as errors.



Слика24.4: Reporting errors for holes on a polygon feature.

See also

Fill holes

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification

наставак на следећој страни

Табела 24.55 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Holes errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Polygons with holes Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Polygon layer with the features containing holes. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Holes errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:checkgeometryhole`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Lines intersecting each other

`Dodato u 3.42`

Checks intersections between line geometries within a layer. Intersections between two different lines are errors.

See also

Presek linija, *Presek*, *Self-intersections*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Layer with the geometries to check.

наставак на следећој страни

Табела 24.58 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Intersection errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the intersections location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Intersecting feature Opciono	`OUTPUT`	[vektor: linija] Podrazumevano: `[Skip output]`	Specification of the output layer for features intersecting each other. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Intersection errors	`ERRORS`	[vektor: tačka]	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`	[vektor: linija]	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 kod

Algorithm ID: `native:checkgeometrylineintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Lines intersecting other layer

`Dodato u 3.42`

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.

See also

Presek linija, *Presek*, *Self-intersections*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Check layer	`CHECK_LAYER`	[vektor: linija, poligon]	Layer with the geometries to check against.
Line intersecting other layer errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output point layer containing the locations of intersection errors. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Line intersecting other layer features Opciono	`OUTPUT`	[line] Podrazumevano: `[Skip output]`	Line layer with the input features that intersect features in the check layer. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Line intersecting other layer errors	`ERRORS`	[vektor: tačka]	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`	[vektor: linija]	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 kod

Algorithm ID: `native:checkgeometrylinelayerintersection`

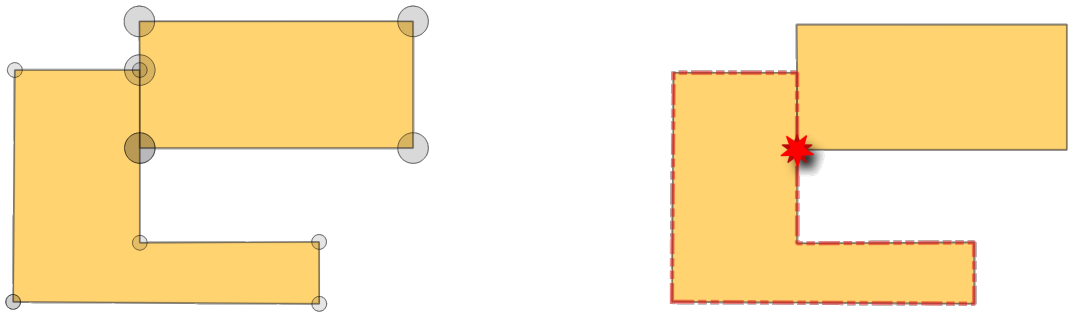
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Missing vertices along borders

`Dodato u 3.42`

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.



Слика24.5: Reporting errors for missing vertices on polygon features.

See also

Add missing vertices along borders

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Missing vertices errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Missing vertices features Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Polygon layer with the features whose vertices are missing. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Missing vertices errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:checkgeometrymissingvertex`

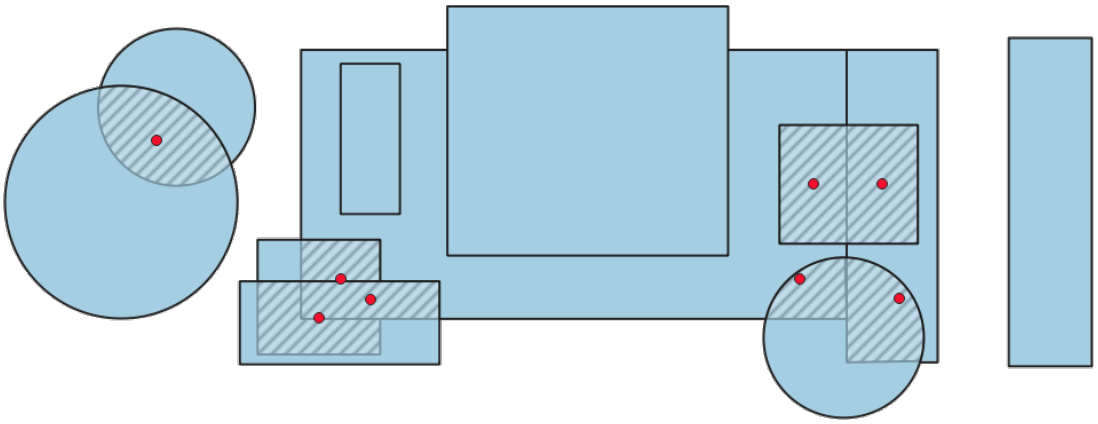
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Overlaps

`Dodato u 3.42`

Calculates overlapping areas in polygon geometries, and reports areas smaller than a given minimum as errors.



Слика24.6: Hashed polygons indicate overlapping areas smaller than the specified minimum.

See also

Delete overlaps, *Provera pokrivenosti*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Minimum overlap area	`MIN_OVERLAP_AREA`	[numerički: decimalni]	Minimum area of the overlap to be reported as an error. If set to 0, all overlaps are reported.
Overlap errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing centroid points of the overlapping areas. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.67 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Overlap features Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Specification of the output layer for features containing overlaps. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Overlap errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	Output layer containing the overlapping areas. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometryoverlap`

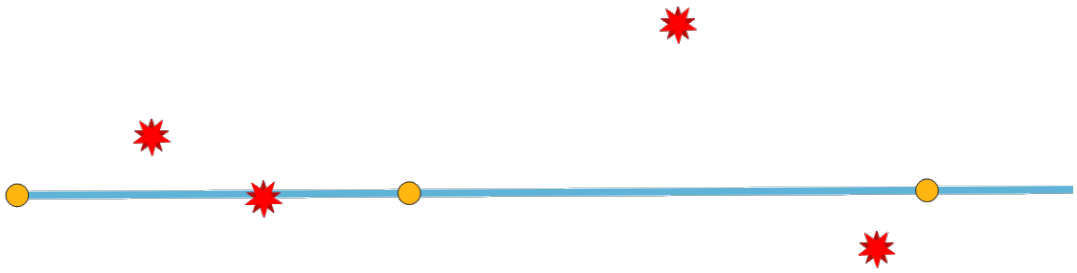
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Points outside lines

`Dodato u 3.42`

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.



Слика24.7: Reporting errors on points not covered by a line.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing points not covered by a line. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Points not covered by a line	`ERRORS`	[vektor: tačka]	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 kod

Algorithm ID: `native:checkgeometrypointcoveredbyline`

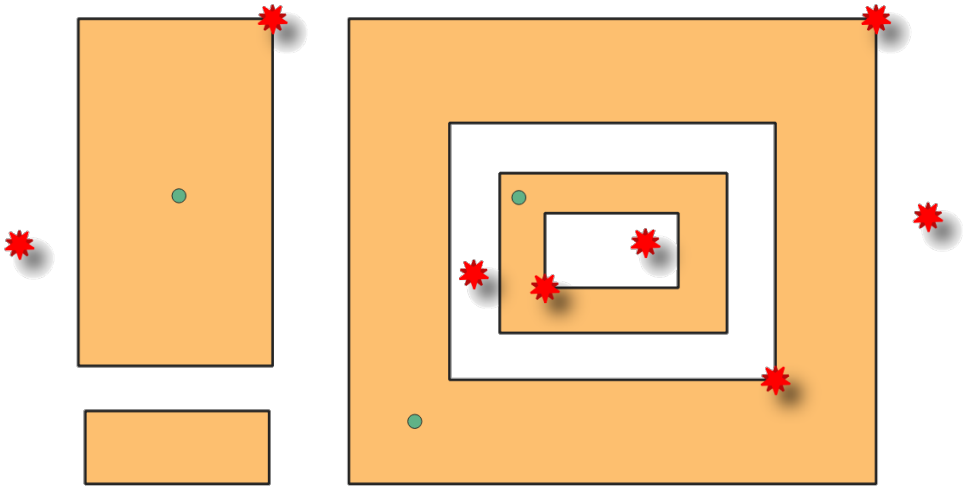
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Points outside polygons

`Dodato u 3.42`

Checks if points from the input layer are in polygons from the selected polygon layers. Points that are not fully inside polygons are errors.



Слика24.8: Reporting errors on points outside polygons.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing points outside the polygons. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Points outside polygons errors	`ERRORS`	[vektor: tačka]	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 kod

Algorithm ID: `native:checkgeometrypointinpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

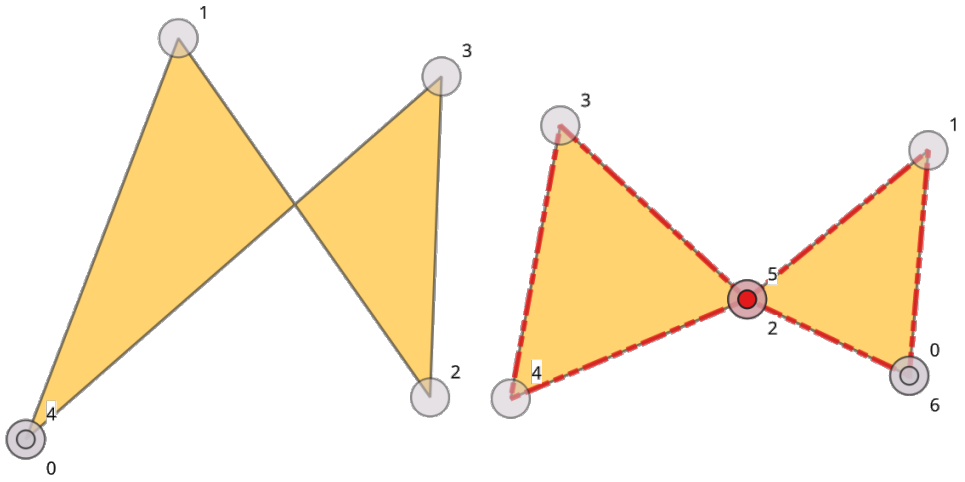
Self-contacts

`Dodato u 3.42`

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.

Белешка

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)`.



Слика24.9: Self-intersection vs self-contact.

See also

Self-intersections

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Self contact error points	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.76 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Self contact features Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Line or polygon layer containing self-contact features. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Self contact error points	`ERRORS`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	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 kod

Algorithm ID: `native:checkgeometryselfcontact`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

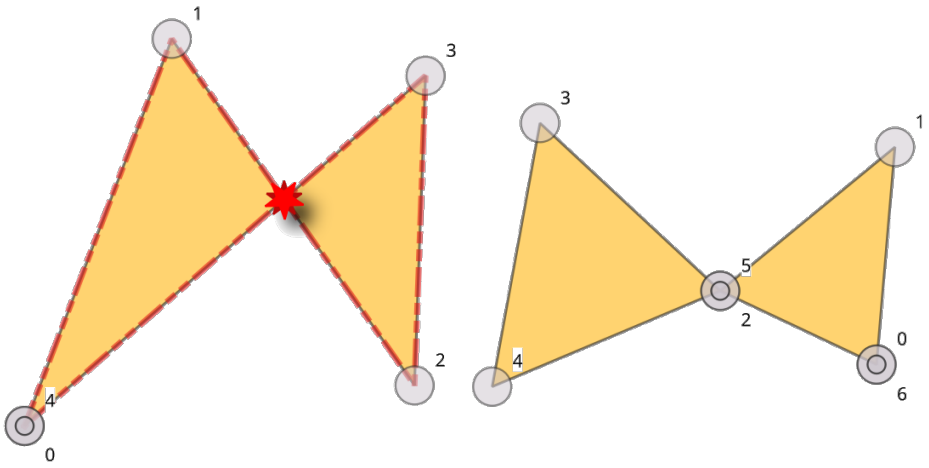
Self-intersections

`Dodato u 3.42`

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.

Белешка

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)`.



Слика24.10: Self-intersection vs self-contact.

See also

Split self-intersecting geometries, *Self-contacts*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification
Self-intersecting errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Self-intersecting features Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Line or polygon layer containing self-intersected features. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Self-intersecting errors	`ERRORS`	[vektor: tačka]	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`	[vektor: linija, poligon]	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 kod

Algorithm ID: `native:checkgeometryselfintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Sliver polygons

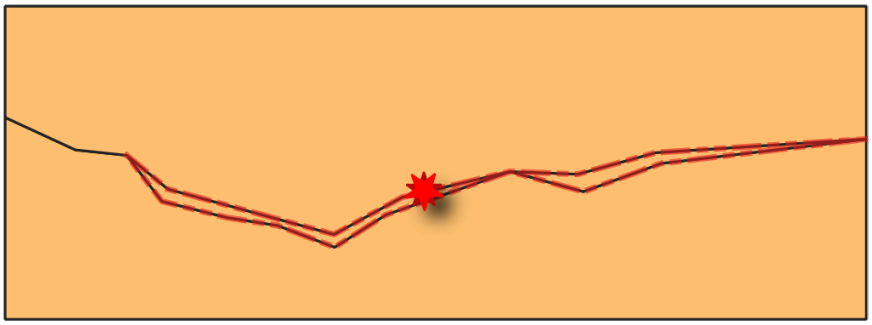
`Dodato u 3.42`

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.

Белешка

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)`.



Слика24.11: Reporting errors on features with sliver polygons.

See also

Fix small polygons

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Maximum thinness	`MAX_THINNESS`	[numerički: decimalni] 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`	[numerički: decimalni] Podrazumevano: 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`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the centroid points of sliver polygon features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.82 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Sliver polygon features Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Polygon layer containing sliver polygon features. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Sliver polygon errors	`ERRORS`	[vektor: tačka]	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 polygon features	`OUTPUT`	[vektor: poligon]	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 kod

Algorithm ID: `native:checkgeometrysliverpolygon`

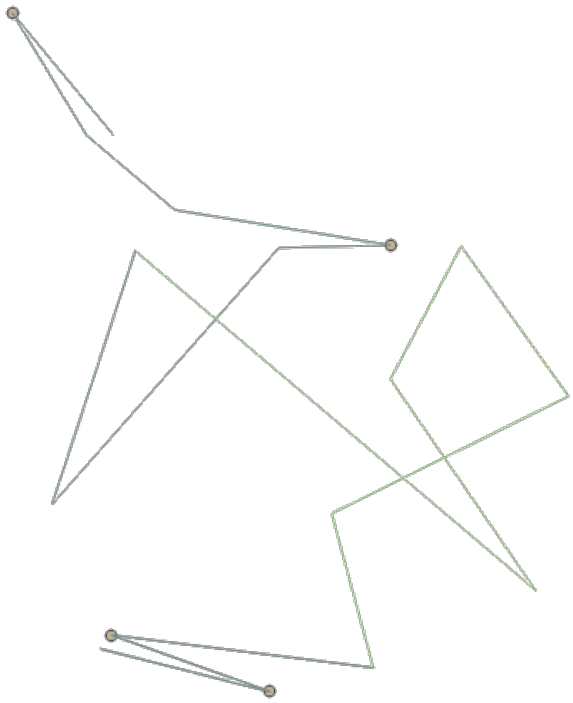
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Small angles

`Dodato u 3.42`

Compares the angles within line or polygon geometries to a specified threshold, and reports as error any angle below that value.



Слика24.12: Reporting errors on a line feature for angles lower than 15°.

See also

Delete small angles

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Minimum angle	`MIN_ANGLE`	[numerički: decimalni]	Allowed minimum angle between adjacent segments in a feature's geometry. Any angle below this threshold is reported as error.
Small angle errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Small angle errors	`ERRORS`	[vektor: tačka]	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 kod

Algorithm ID: `native:checkgeometryangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

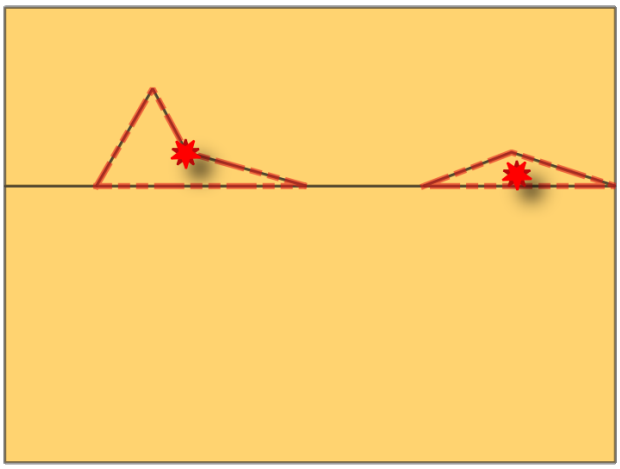
Small polygons

`Dodato u 3.42`

Detects polygon features whose area is below a specified value as errors.

Белешка

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)`.



Слика24.13: Reporting errors on features with small polygons (below given area threshold).

See also

Fix small polygons

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Area threshold	`AREATHRESHOLD`	[numerički: decimalni] Podrazumevano: 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`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing centroid points of small polygons. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.88 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Small polygons features Opciono	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Skip output]`	Specification of the output layer for features containing small polygon features. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Small polygons errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:checkgeometryarea`

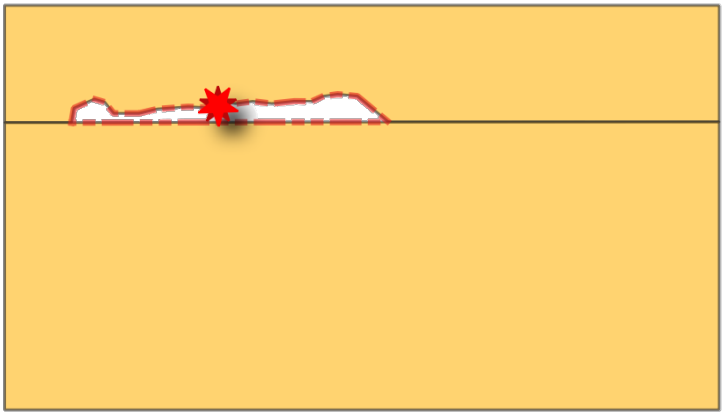
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Small gaps

`Dodato u 3.42`

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.



Слика24.14: Reporting errors on polygon features for gaps smaller than the specified threshold.

See also

Fill gaps, *Provera pokrivenosti*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Gap threshold	`GAP_THRESHOLD`	[numerički: decimalni] Podrazumevano: 0.0	Maximum area of gaps to be reported as errors, in map units. If set to 0, all the gaps are reported.

наставак на следећој страни

Табела 24.91 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Allowed gaps layer Opciono	`ALLOWED_GAPS_LAYER`	[vektor: poligon]	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 Opciono	`ALLOWED_GAPS_BUFFER`	[numerički: decimalni] Podrazumevano: 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.
Neighbors layer	`NEIGHBORS`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Specification of the output table representing the gap ID and the unique ID of its neighbor features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Gap errors Opciono	`ERRORS`	[vektor: tačka] Podrazumevano: `[Skip output]`	Specification of the output layer containing the centroid points of the gaps. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Gap features	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the gap geometries. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Neighbors layer	`NEIGHBORS`	[vektor: tabela]	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.
Gap errors	`ERRORS`	[vektor: tačka]	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`	[vektor: poligon]	Output layer containing the gap geometries. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometrygap`

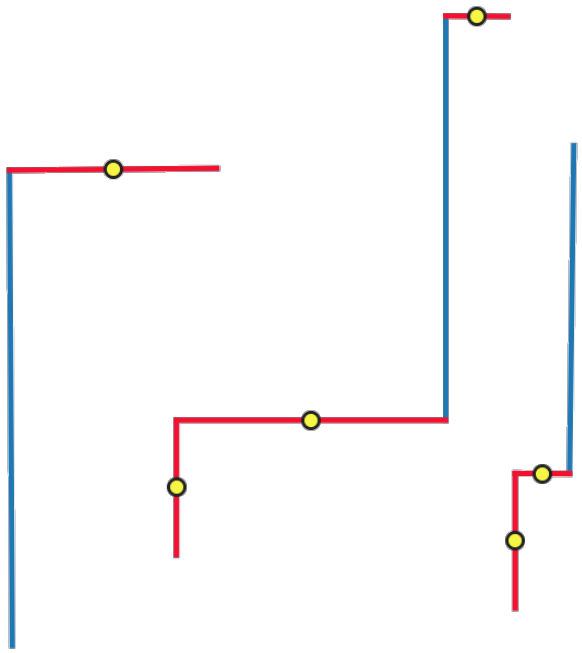
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Small segments

`Dodato u 3.42`

Calculates length of individual segments in line or polygon geometries, and reports segments shorter than a minimum length as errors.



Слика24.15: Reporting errors (in red) on single line segments shorter than the given threshold.

See also

Brisanje objekata

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
Minimum segment length	`MIN_SEGMENT_LENG`	[numerički: decimalni] Podrazumevano: 0.0	Minimum length of segments to be considered valid, in map units.
Short segments errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing centroid of short segments. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.94 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Short segments features Opciono	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Skip output]`	Specification of the output layer for features containing short segments. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Short segments errors	`ERRORS`	[vektor: tačka]	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.
Short segments features	`OUTPUT`	[vektor: isto kao ulaz]	Output layer containing, for each identified short segment, the feature it belongs to. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometrysegmentlength`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Strictly multipart

`Dodato u 3.42`

Checks if multipart geometries have more than one part. Multipart geometries with only one part are errors.

See also

Convert to strictly multipart

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification.
One-part geometry errors	`ERRORS`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing centroid of the multipart geometries that have only one part. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
One-part geometry features Opciono	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Skip output]`	Specification of the output layer for features containing multipart geometries that have only one part. *One of*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
One-part geometry errors	`ERRORS`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer containing features that are multipart but have only one part. Available fields are the same as in the `ERRORS` output.

Python kod

Algorithm ID: `native:checkgeometrymultipart`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.4 Baza podataka

Izvoz u PostgreSQL

Izvozi vektorski sloj u PostgreSQL bazu podataka, praveći novu relaciju. Ako relacija istog naziva postoji, može se ukloniti pre pravljenja nove relacije. Pre toga se mora napraviti veza između QGIS-a i PostgreSQL baze podataka (pogledajte npr. *Creating a stored Connection*).

Parametri

Oznaka	Назив	Tip	Opis
Sloj za uvoz	`INPUT`	[vektor: bilo koji]	Vektorski sloj koji se dodaje u bazu podataka
Baza podataka (naziv veze)	`DATABASE`	[niska]	Naziv veze sa bazom podataka (ne naziv baze podataka). Postojeće veze biće prikazane u padajućoj listi.
Šema (naziv šeme) Opciono	`SCHEMA`	[niska] Podrazumevano: 'public'	Naziv šeme za čuvanje podataka. Može biti nova ili već postojati.
Tabela u koju se uvozi (ostavite prazno za korišćenje naziva sloja) Opciono	`TABLENAME`	[niska] Podrazumevano: ''	Definiše naziv tabele za uvezenu vektorsku datoteku. Ako se ništa ne doda, koristiće se naziv sloja.
Polje primarnog ključa Opciono	`PRIMARY_KEY`	[polje tabele: bilo koji]	Postavlja polje primarnog ključa iz postojećeg polja u vektorskom sloju. Kolona sa **jedinstvenim** vrednostima može se koristiti kao primarni ključ baze podataka.
Kolona geometrije	`GEOMETRY_COLUMN`	[niska] Podrazumevano: 'geom'	Definiše naziv kolone geometrije u novoj PostgreSQL tabeli. Informacija o geometriji objekata čuva se u ovoj koloni.
Kodiranje Opciono	`ENCODING`	[niska] Podrazumevano: 'UTF-8'	Definiše kodiranje izlaznog sloja
Prepiši	`OVERWRITE`	[logička vrednost] Podrazumevano: True	Ako navedena tabela postoji, postavljanje ove opcije na `True` obezbeđuje da bude obrisana i da se napravi nova tabela pre dodavanja objekata. Ako je ova opcija `False` i tabela postoji, algoritam će izbaciti izuzetak („relation already exists“).
Napravi prostorni indeks	`CREATEINDEX`	[logička vrednost] Podrazumevano: True	Zadaje da li se pravi prostorni indeks ili ne
Pretvori nazive polja u mala slova	`LOWERCASE_NAMES`	[logička vrednost] Podrazumevano: True	Pretvara nazive polja ulaznog vektorskog sloja u mala slova
Ukloni ograničenje dužine na znakovnim poljima	`DROP_STRING_LENG`	[logička vrednost] Podrazumevano: False	Da li ograničenja dužine na znakovnim poljima treba ukloniti ili ne
Napravi jednodelne geometrije umesto višedelnih	`FORCE_SINGLEPART`	[logička vrednost] Podrazumevano: False	Da li objekti izlaznog sloja treba da budu jednodelni umesto višedelnih. Podrazumevano se postojeća informacija o geometriji čuva.

Izlazi

Algoritam nema izlaz.

Python kod

ID algoritma: `native:importintopostgis`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u SpatiaLite

Izvozi vektorski sloj u SpatiaLite bazu podataka. Pre toga se mora napraviti veza između QGIS-a i SpatiaLite baze podataka (pogledajte npr. *SpatiaLite Layers*).

Parametri

Oznaka	Назив	Tip	Opis
Sloj za uvoz	`INPUT`	[vektor: bilo koji]	Vektorski sloj koji se dodaje u bazu podataka
Datoteka baze podataka	`DATABASE`	[vektor: bilo koji]	SQLite/SpatiaLite datoteka baze podataka na koju se povezuje
Tabela u koju se uvozi (ostavite prazno za korišćenje naziva sloja) Opciono	`TABLENAME`	[niska] Podrazumevano: ”	Definiše naziv tabele za uvezenu vektorsku datoteku. Ako se ništa ne zada, koristiće se naziv sloja.
Polje primarnog ključa Opciono	`PRIMARY_KEY`	[polje tabele: bilo koji]	Koristi polje iz ulaznog vektorskog sloja kao primarni ključ
Kolona geometrije	`GEOMETRY_COLUMN`	[niska] Podrazumevano: ’geom’	Definiše naziv kolone geometrije u novoj SpatiaLite tabeli. Informacija o geometriji objekata čuva se u ovoj koloni.
Kodiranje Opciono	`ENCODING`	[niska] Podrazumevano: ’UTF-8’	Definiše kodiranje izlaznog sloja
Prepiši	`OVERWRITE`	[logička vrednost] Podrazumevano: True	Ako navedena tabela postoji, postavljanje ove opcije na `True` obezbeđuje da bude obrisana i da se napravi nova tabela pre dodavanja objekata sloja. Ako je ova opcija `False` i tabela postoji, algoritam će izbaciti izuzetak („table already exists“).
Napravi prostorni indeks	`CREATEINDEX`	[logička vrednost] Podrazumevano: True	Zadaje da li se pravi prostorni indeks ili ne
Pretvori nazive polja u mala slova	`LOWERCASE_NAMES`	[logička vrednost] Podrazumevano: True	Pretvori nazive polja ulaznog vektorskog sloja u mala slova

наставак на следећој страни

Табела 24.101 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ukloni ograničenje dužine na znakovnim poljima	DROP_STRING_LENG	[logička vrednost] Podrazumevano: False	Da li ograničenja dužine na znakovnim poljima treba ukloniti ili ne
Napravi jednodelne geometrije umesto višedelnih	FORCE_SINGLEPART	[logička vrednost] Podrazumevano: False	Da li objekti izlaznog sloja treba da budu jednodelni umesto višedelnih. Podrazumevano se postojeća informacija o geometriji čuva.

Izlazi

Algoritam nema izlaz.

Python kod

ID algoritma: `native:importintospatialite`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pakovanje slojeva

Dodaje slojeve u GeoPackage.

Ako GeoPackage postoji i `Overwrite existing GeoPackage` je uključeno, biće prepisan (uklonjen i ponovo napravljen). Ako GeoPackage postoji i `Overwrite existing GeoPackage` nije uključeno, sloj će biti dodat.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	LAYERS	[vektor: bilo koji] [lista]	(Vektorski) slojevi koji se uvoze u GeoPackage. Rasterski slojevi nisu podržani. Ako se doda rasterski sloj, biće izbačen `QgsProcessingException`.
Prepiši postojeći GeoPackage	OVERWRITE	[logička vrednost] Podrazumevano: False	Ako navedeni GeoPackage postoji, postavljanje ove opcije na `True` obezbeđuje da bude obrisan i da se napravi novi pre dodavanja slojeva. Ako je postavljeno na `False`, slojevi će biti dodati.
Sačuvaj stilove slojeva u GeoPackage	SAVE_STYLES	[logička vrednost] Podrazumevano: True	Sačuvaj stilove slojeva.
Sačuvaj samo izabrane objekte	SELECTED_FEATURES_ONI	[logička vrednost] Podrazumevano: False	Ako sloj ima izbor, postavljanje ove opcije na `True` daće čuvanje samo izabranih objekata. Za slojeve bez izbora čuvaju se svi objekti.

наставак на следећој страни

Табела 24.102 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izvezi povezane slojeve prateći relacije definisane u projektu	`EXPORT_RELATED_I`	[logička vrednost] Podrazumevano: False	Ako ulazni sloj ima *relacije* postavljene u projektu, postavljanje ove opcije na `True` daće izvoz i njegovih povezanih slojeva. Ako sloj ima izabrane objekte, biće izvezeni samo njihovi povezani objekti, osim ako je povezani sloj takođe bio ulazni sloj.
Odredišni GeoPackage	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Zadajte gde se čuva GeoPackage datoteka. Jedno od: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Slojevi u novom paketu	`OUTPUT_LAYERS`	[niska] [lista]	Lista slojeva dodatih u GeoPackage.

Python kod

ID algoritma: `native:package`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

PostgreSQL izvrši i učitaj SQL

Omogućava izvršavanje SQL upita nad bazom podataka na PostgreSQL bazi povezanoj sa QGIS-om i učitava rezultat. Algoritam **neće** napraviti novi sloj: namenjen je za pokretanje upita nad samim slojem.

Primer

1. Postavi sve vrednosti postojećeg polja na fiksnu vrednost. Niska SQL upita biće:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   U gornjem primeru, sve vrednosti polja `field_to_update` tabele `your_table` biće postavljene na `20`.

2. Napravi novu kolonu `area` i izračunaj površinu svakog objekta pomoću PostGIS funkcije `ST_AREA`.

   ```
   -- Create the new column "area" on the table your_table"
   ALTER TABLE your_table ADD COLUMN area double precision;
   -- Update the "area" column and calculate the area of each feature:
   UPDATE your_table SET area=ST_AREA(geom);
   ```

See also

PostgreSQL izvrši SQL, *Izvršavanje SQL-a*, *SpatiaLite izvrši SQL*

Parametri

Oznaka	Назив	Tip	Opis
Baza podataka (naziv veze)	`DATABASE`	[niska]	Veza sa bazom podataka (ne naziv baze podataka). Postojeće veze biće prikazane u padajućoj listi.
SQL upit	`SQL`	[niska]	Definiše SQL upit, na primer `'UPDATE my_table SET field=10'`.
Naziv polja jedinstvenog ID-ja	`ID_FIELD`	[niska] Podrazumevano: id	Postavlja polje primarnog ključa (kolonu u tabeli rezultata)
Naziv polja geometrije Opciono	`GEOMETRY_FIELD`	[niska] Podrazumevano: 'geom'	Naziv kolone geometrije (kolone u tabeli rezultata)

Izlazi

Oznaka	Назив	Tip	Opis
SQL sloj	`OUTPUT`	[vektor: bilo koji]	Rezultujući vektorski sloj koji se učitava u QGIS.

Python kod

ID algoritma: `qgis:postgisexecuteandloadsql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

PostgreSQL izvrši SQL

Omogućava izvršavanje SQL upita nad bazom podataka na PostgreSQL bazi povezanoj sa QGIS-om. Algoritam **neće** napraviti novi sloj: namenjen je za pokretanje upita nad samim slojem.

Primer

1. Postavi sve vrednosti postojećeg polja na fiksnu vrednost. Niska SQL upita biće:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   U gornjem primeru, sve vrednosti polja `field_to_update` tabele `your_table` biće postavljene na `20`.

2. Napravi novu kolonu `area` i izračunaj površinu svakog objekta pomoću PostGIS funkcije `ST_AREA`.

   ```
   -- Create the new column "area" on the table your_table"
   ALTER TABLE your_table ADD COLUMN area double precision;
   -- Update the "area" column and calculate the area of each feature:
   UPDATE your_table SET area=ST_AREA(geom);
   ```

See also

PostgreSQL izvrši i učitaj SQL, *Izvršavanje SQL-a*, *SpatiaLite izvrši SQL*

Parametri

Oznaka	Назив	Tip	Opis
Baza podataka (naziv veze)	`DATABASE`	[niska]	Veza sa bazom podataka (ne naziv baze podataka). Postojeće veze biće prikazane u padajućoj listi.
SQL upit	`SQL`	[niska]	Definiše SQL upit, na primer `'UPDATE my_table SET field=10'`.

Izlazi

Ne pravi se izlaz. SQL upit se izvršava na mestu.

Python kod

ID algoritma: `native:postgisexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

SpatiaLite izvrši SQL

Omogućava izvršavanje SQL upita nad bazom podataka na SpatiaLite bazi. Baza podataka se određuje ulaznim slojem ili datotekom. Algoritam **neće** napraviti novi sloj: namenjen je za pokretanje upita nad samim slojem.

See also

PostgreSQL izvrši SQL, *Izvršavanje SQL-a*

Za neke primere SQL upita pogledajte *Primeri PostgreSQL upita*.

Parametri

Oznaka	Назив	Tip	Opis
Sloj baze podataka (ili datoteka)	`DATABASE`	[datoteka]	SQLite/SpatiaLite datoteka baze podataka na koju se povezuje
SQL upit	`SQL`	[niska] Podrazumevano: ”	Definiše SQL upit, na primer `'UPDATE my_table SET field=10'`.

Izlazi

Ne pravi se izlaz. SQL upit se izvršava na mestu.

Python kod

ID algoritma: `native:spatialiteexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

SpatiaLite izvrši SQL (registrovana BP)

Omogućava izvršavanje SQL upita nad bazom podataka na SpatiaLite bazi povezanoj sa QGIS-om. Algoritam **neće** napraviti novi sloj: namenjen je za pokretanje upita nad samim slojem.

See also

PostgreSQL izvrši SQL, *Izvršavanje SQL-a*

Za neke primere SQL upita pogledajte *Primeri PostgreSQL upita*.

Parametri

Oznaka	Назив	Tip	Opis
Baza podataka	`DATABASE`	[nabrajanje] Podrazumevano: nije postavljeno	Izaberite SQLite/SpatiaLite bazu podataka povezanu sa trenutnom sesijom
SQL upit	`SQL`	[niska] Podrazumevano: ”	Definiše SQL upit, na primer `'UPDATE my_table SET field=10'`.

Izlazi

Ne pravi se izlaz. SQL upit se izvršava na mestu.

Python kod

ID algoritma: `native:spatialiteexecutesqlregistered`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.5 Alati za datoteke

Preuzmi datoteku preko HTTP(S)

Preuzima URL u sistem datoteka pomoću HTTP(S) GET ili POST zahteva.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
URL	`URL`	[niska]	URL datoteke koja se preuzima.

наставак на следећој страни

Табела 24.109 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Odredište datoteke	`OUTPUT`	[niska] Podrazumevano: `[Save to temporary file]`	Specifikacija odredišta datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	HTTP metod koji se koristi za zahtev. Opcije su: • 0 — GET • 1 — POST
Podaci Opciono	`DATA`	[niska]	Podaci koji se dodaju u telo zahteva ako je zahtev POST.

Izlazi

Oznaka	Назив	Tip	Opis
Odredište datoteke	`OUTPUT`	[niska]	Lokacija preuzete datoteke

Python kod

ID algoritma: `native:filedownloader`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

HTTP(S) POST/GET zahtev

`Dodato u 3.40`

Izvršava HTTP(S) POST/GET zahtev i vraća HTTP statusni kod i podatke odgovora. Ako dođe do greške, vraća se kod greške i poruka. Opciono, rezultat se može upisati u datoteku na disku. Podrazumevano će algoritam upozoriti na greške. Opciono, algoritam se može podesiti da HTTP greške tretira kao neuspehe.

Parametri

Oznaka	Назив	Tip	Opis
URL ili putanja datoteke	`URL`	[niska]	URL ili putanja datoteke koja se otvara.
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	HTTP metod koji se koristi za zahtev. Opcije su: • 0 — GET • 1 — POST

наставак на следећој страни

Табела 24.112 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
POST podaci Opciono	`DATA`	[niska]	Podaci koji se dodaju u telo zahteva ako je zahtev POST.
Autentifikacija Opciono	`AUTH_CONFIG`	[authconfig] Podrazumevano: Bez autentifikacije	Konfiguracija autentifikacije koja se prosleđuje
Tretiraj HTTP greške kao neuspehe	`FAIL_ON_ERROR`	[logička vrednost] Podrazumevano: False	Ako je postavljeno, algoritam će prijaviti neuspeh pri nailasku na HTTP grešku.
Odredište datoteke Opciono	`OUTPUT`	[niska] Podrazumevano: `[Skip Output]`	Rezultat se može upisati u datoteku umesto da bude vraćen kao niska. Specifikacija odredišta datoteke. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Odredište datoteke	`OUTPUT`	[datoteka]	Preuzeta datoteka sa vraćenim rezultatom
HTTP status	`STATUS_CODE`	[niska]	HTTP status
Kod mrežne greške	`ERROR_CODE`	[niska]	Kod greške kada dođe do neuspeha (kao i pri HTTP statusu 404 itd.)
Poruka mrežne greške	`ERROR_MESSAGE`	[niska]	Niska koja sadrži poruku greške u slučaju neuspeha
Podaci rezultata	`RESULT_DATA`	[niska]	Niska koja sadrži podatke rezultata u slučaju uspeha

Python kod

ID algoritma: `native:httprequest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Otvori datoteku ili URL

Otvara datoteke u njihovoj podrazumevanoj pridruženoj aplikaciji, ili URL-ove u podrazumevanom veb pregledaču korisnika.

Parametri

Oznaka	Назив	Tip	Opis
URL ili putanja datoteke	`URL`	[niska]	URL ili putanja datoteke koja se otvara.

Izlazi

Algoritam nema izlaz.

Python kod

ID algoritma: `native:openurl`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

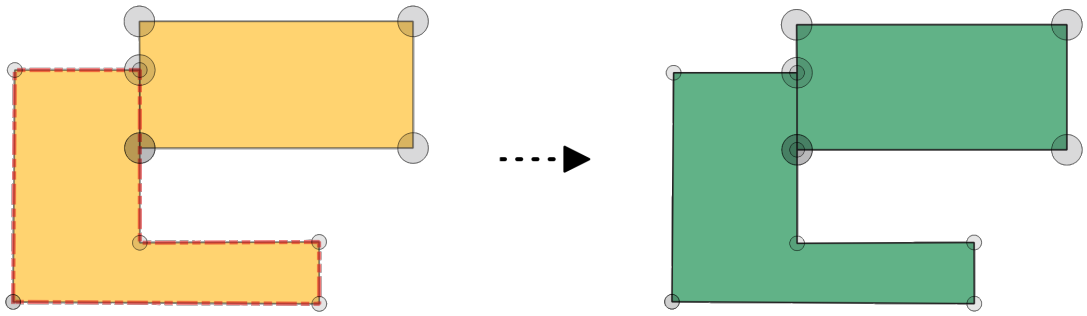
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.6 Fix Geometry

Add missing vertices along borders

`Dodato u 3.42`

Adds the missing vertices along polygons borders, based on an error layer from the *Missing vertices along borders* algorithm.



Слика24.16: Fixing missing vertices along polygon borders.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification in input layer, as set in the check algorithm.

наставак на следећој страни

Табела 24.115 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
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`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing border vertices	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing border vertices	`REPORT`	[vektor: tačka]	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`	[vektor: poligon]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometrymissingvertex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Convert to strictly multipart

`Dodato u 3.42`

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.

See also

Unapredi u višedelno

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Strictly-multipart layer	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing multipart	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the centroids of the fixed geometries. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing multipart	`REPORT`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometrymultipart`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Delete duplicated vertices

`Dodato u 3.42`

Deletes duplicated vertices from the input geometries, based on errors reported by the *Duplicated vertices* algorithm.

See also

Ukloni duplirana temena

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing duplicate vertices	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing duplicate vertices	`REPORT`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometryduplicatenodes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

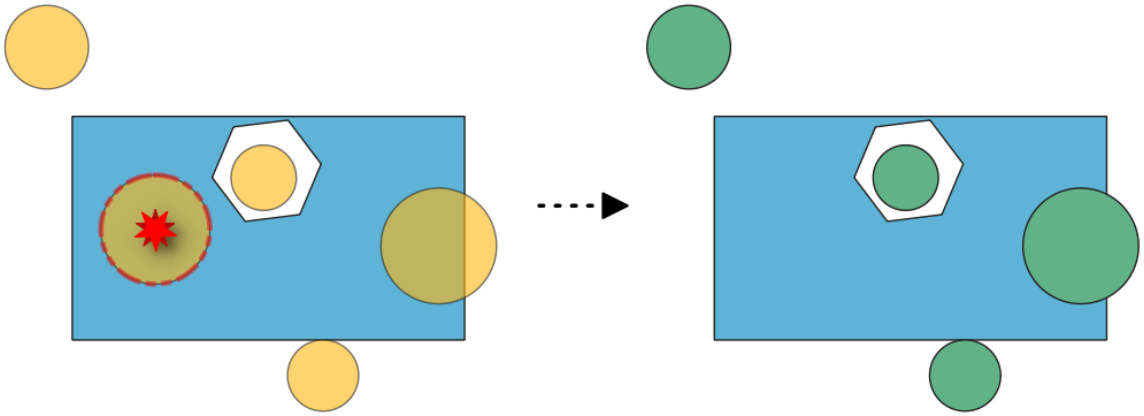
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Brisanje objekata

`Dodato u 3.42`

Deletes error features based on an error layer from some check geometry algorithms, such as:

- *Features inside polygon*
- *Degenerate polygons*
- *Small segments*
- *Duplicated geometries*



Слика24.17: Fixing errors from the „*Features inside polygon*” algorithm check.

See also

Brisanje duplih geometrija, *Uklanjanje geometrija*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Cleaned layer	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from deleting features	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from deleting features	`REPORT`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer with features removed based on detected errors. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometrydeletefeatures`

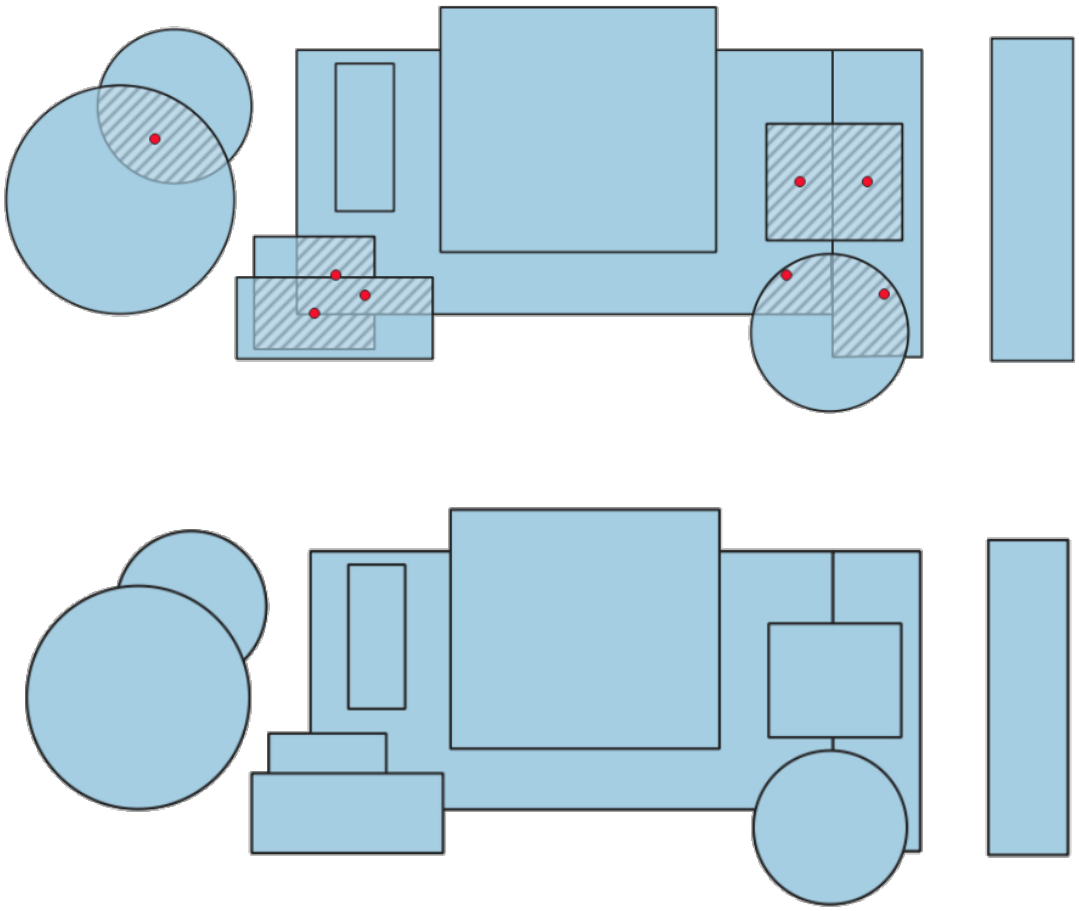
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Delete overlaps

`Dodato u 3.42`

Deletes overlapping areas based on an error layer from the *Overlaps* algorithm.



Слика24.18: Deleting overlapping areas.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of overlap feature unique identifier	`OVERLAP_FEATURE_UNIQ`	[polje tabele: bilo koji]	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`	[polje tabele: bilo koji]	Field storing the error value for each feature, as set in the check algorithm.

наставак на следећој страни

Табела 24.126 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
No-overlap layer	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing overlaps	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing overlaps	`REPORT`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:fixgeometryoverlap`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

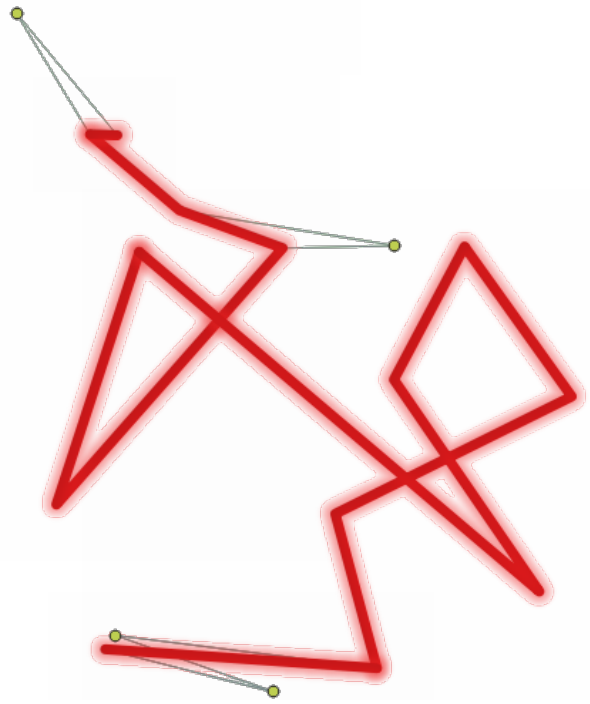
Delete small angles

`Dodato u 3.42`

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.

Пажња

This algorithm removes the vertex at the reported small angles, generating new segments that may form a new small angle.



Слика24.19: Fixing reported errors on a line feature for angles lower than 15°.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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 angle fixed layer	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.129 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Report layer from fixing small angles	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing small angles	`REPORT`	[vektor: tačka]	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.

наставак на следећој страни

Табела 24.131 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Small angle fixed layer	`OUTPUT`	[isto kao ulaz]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometryangle`

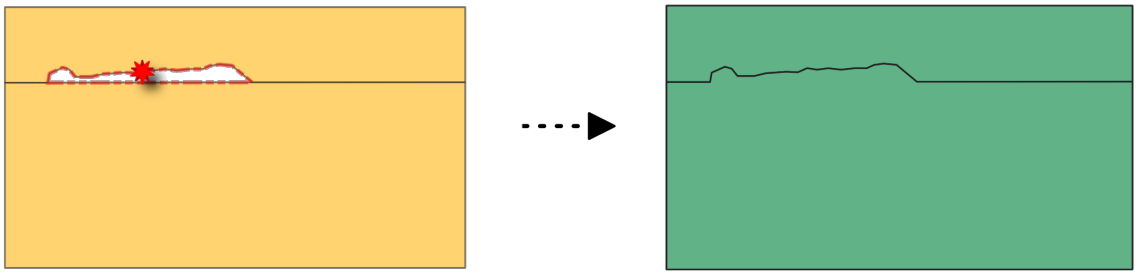
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fill gaps

`Dodato u 3.42`

Fills the gaps based on a gap and neighbors layer from the *Small gaps* algorithm. Three different fixing methods are available.



Слика24.20: Fixing gap between polygon features using the `Add to largest neighbouring area` method.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Neighbors layer	`NEIGHBORS`	[vektor: tabela]	Relational table with the unique ID of the gap and its neighbors in input layer, generated by the check algorithm.
Gaps layer	`GAPS`	[vektor: poligon]	Polygon layer with the gaps geometries, generated by the check algorithm.

наставак на следећој страни

Табела 24.132 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Metod	`METHOD`	[nabrajanje] Podrazumevano: 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`	[polje tabele: bilo koji] 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`	[polje tabele: bilo koji]	Field storing the error ID for each gap, as output by the check algorithm.
Gaps-filled layer	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing gaps	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing gaps	`REPORT`	[vektor: tačka]	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`	[vektor: poligon]	Output layer of polygons without gaps. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometrygap`

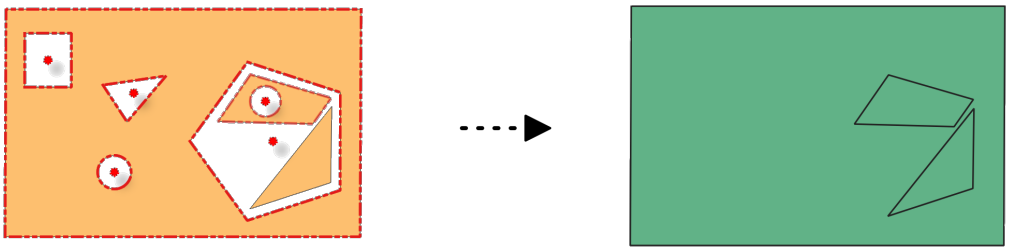
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fill holes

`Dodato u 3.42`

Deletes holes in polygon geometries based on an error layer from the *Holes* algorithm.



Слика24.21: Deleting holes in a polygon feature.

See also

Obriši rupe

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing holes	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing holes	`REPORT`	[vektor: tačka]	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`	[vektor: poligon]	Output layer of polygons without holes. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometryhole`

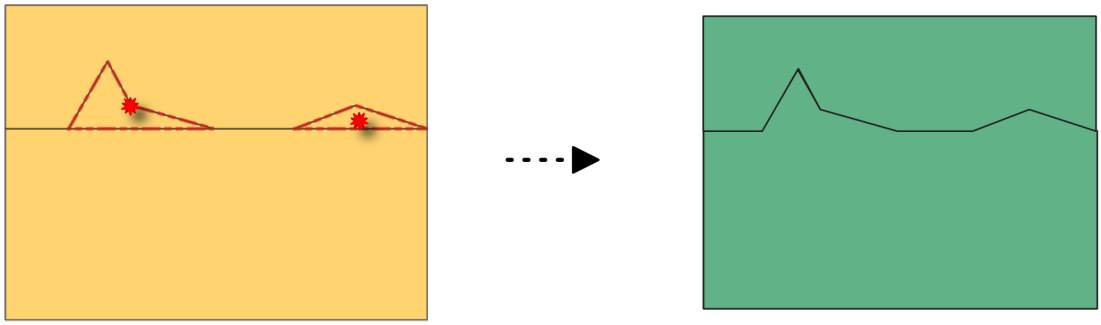
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fix small polygons

`Dodato u 3.42`

Merges neighboring polygons according to the chosen method, based on an error layer from the *Small polygons* or *Sliver polygons* algorithm.



Слика24.22: Before and after fixing small polygons (method: merge with neighboring polygon with largest area).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm.
Metod	`METHOD`	[nabrajanje] Podrazumevano: 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.
Field to consider when merging polygons with the identical attribute method Opciono	`MERGE_ATTRIBUTE`	[polje tabele: bilo koji]	Field in the input layer to consider when merging polygons with the identical attribute method.
Field of original feature unique identifier	`UNIQUE_ID`	[polje tabele: bilo koji]	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.

наставак на следећој страни

Табела 24.138 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
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`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from merging small polygons	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from merging small polygons	`REPORT`	[vektor: tačka]	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`	[vektor: poligon]	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 kod

Algorithm ID: `native:fixgeometryarea`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

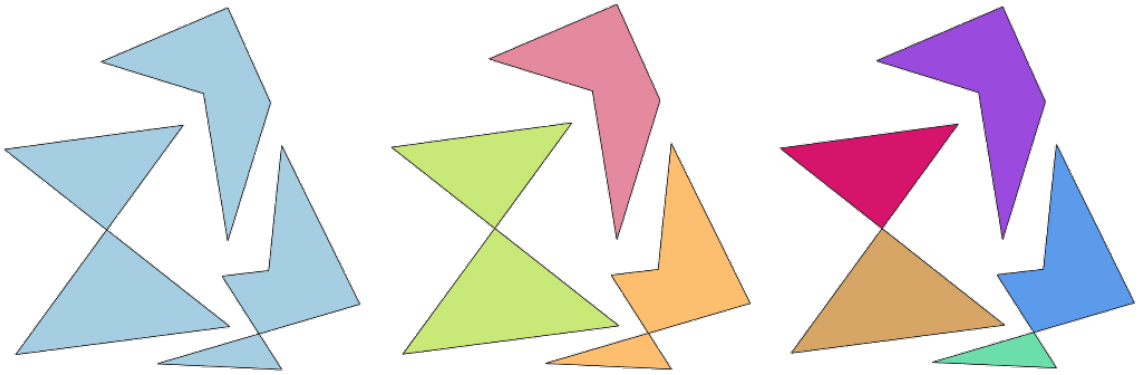
Split self-intersecting geometries

`Dodato u 3.42`

Splits self-intersecting geometries based on an error layer from the *Self-intersections* algorithm.

Белешка

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)`.



Слика24.23: Fixing self-intersection using various methods: as multi-part (middle), and multiple single-part geometries (right).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vektor: tačka]	Point layer with the errors location, generated by the check algorithm
Metod	`METHOD`	[nabrajanje]	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`	[polje tabele: bilo koji]	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.

наставак на следећој страни

Табела 24.141 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
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`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Report layer from fixing self-intersections	`REPORT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tolerancija	`TOLERANCE`	[numerički: ceo broj] Podrazumevano: 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.

Izlazi

Oznaka	Назив	Tip	Opis
Report layer from fixing self--intersections	`REPORT`	[vektor: tačka]	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`	[vektor: isto kao ulaz]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python kod

Algorithm ID: `native:fixgeometryselfintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.7 GPS

Pretvori GPS podatke

Koristi GPSBabel alatku za pretvaranje datoteke GPS podataka iz niza formata u standardni GPX format.

Parametri

Oznaka	Назив	Tip	Opis
Ulazna datoteka	`INPUT`	[datoteka]	Datoteka koja sadrži podatke za pretvaranje
Format	`FORMAT`	[nabrajanje]	Format datoteke za pretvaranje, iz ove liste.
Tip objekta	`FEATURE_TYPE`	[nabrajanje] Podrazumevano: 0	Tip podataka za pretvaranje • 0 — Tačke puta • 1 — Rute • 2 — Tragovi
Izlaz	`OUTPUT`	[vektor: bilo koji] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne GPX datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT_LAYER`	[vektor: bilo koji]	Izlazni sloj sa podacima u standardnom GPX formatu

Python kod

ID algoritma: `native:convertgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvori tip GPX objekta

Koristi GPSBabel alatku za pretvaranje GPX objekata iz jednog tipa u drugi (npr. pretvaranje svih objekata tačaka puta u rutu).

Parametri

Oznaka	Назив	Tip	Opis
Ulazna datoteka	`INPUT`	[datoteka]	Datoteka koja sadrži podatke za pretvaranje
Pretvaranje	`CONVERSION`	[nabrajanje] Podrazumevano: 0	Tip pretvaranja koji se primenjuje • 0 — Tačke puta iz rute • 1 — Tačke puta iz traga • 2 — Rute iz tačaka puta • 3 — Tragovi iz tačaka puta
Izlaz	`OUTPUT`	[vektor: tačka ili linija] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[vektor: bilo koji]	Izlazni sloj sa pretvorenim GPX objektima

Python kod

ID algoritma: `native:convertgpxfeaturetype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preuzmi GPS podatke sa uređaja

Koristi GPSBabel alatku za preuzimanje podataka sa GPS uređaja u standardni GPX format.

Parametri

Oznaka	Назив	Tip	Opis
Uređaj	`DEVICE`	[nabrajanje] Podrazumevano:*Garmin serial*	GPS uređaj korišćen za pravljenje podataka. Mora biti deklarisan u dijalogu *GPS podešavanja*.
Port	`PORT`	[nabrajanje]	Port na koji je uređaj povezan. Dostupni portovi zavise od operativnog sistema.
Tip objekta	`FEATURE_TYPE`	[nabrajanje] Podrazumevano: 0	Tip podataka za pretvaranje • 0 — Tačke puta • 1 — Rute • 2 — Tragovi
Izlaz	`OUTPUT`	[vektor: bilo koji] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT_LAYER`	[vektor: bilo koji]	Izlazni sloj sa podacima u standardnom GPX formatu

Python kod

ID algoritma: `native:downloadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Otpremi GPS podatke na uređaj

Koristi GPSBabel alatku za otpremanje podataka na GPS uređaj iz standardnog GPX formata.

Parametri

Oznaka	Назив	Tip	Opis
Ulazna datoteka	`INPUT`	[datoteka]	`.GPX` datoteka koja sadrži podatke za otpremanje
Uređaj	`DEVICE`	[nabrajanje] Podrazumevano:*Garmin serial*	GPS uređaj na koji želite da otpremite podatke. Mora biti deklarisan u dijalogu *GPS podešavanja*.
Port	`PORT`	[nabrajanje]	Port na koji je uređaj povezan. Dostupni portovi zavise od operativnog sistema.
Tip objekta	`FEATURE_TYPE`	[nabrajanje] Podrazumevano: 0	Tip podataka za otpremanje • 0 — Tačke puta • 1 — Rute • 2 — Tragovi

Izlazi

Ne pruža se izlaz. Ako je uspešno, podaci se učitavaju na uređaj.

Python kod

ID algoritma: `native:uploadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.8 Interpolacija

Toplotna karta (procena gustine jezgrom)

Pravi rasterski sloj gustine (toplotnu kartu) od ulaznog tačkastog vektorskog sloja pomoću procene gustine jezgrom.

Gustina se računa na osnovu broja tačaka na nekom mestu, pri čemu veći broj grupisanih tačaka daje veće vrednosti. Toplotne karte omogućavaju lako prepoznavanje *žarišta* i grupisanja tačaka.

Parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Tačkasti vektorski sloj koji se koristi za toplotnu kartu

наставак на следећој страни

Табела 24.151 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Полупречник	`RADIUS`	[numerički: decimalni] Podrazumevano: 100.0	Полупречник претраге топлотне карте (или пропусни опсег језгра) у јединицама карте. Полупречник задаје растојање око тачке на којем ће се осетити утицај те тачке. Веће вредности дају веће заглађивање, док мање вредности могу приказати финије детаље и варијације у густини тачака.
Veličina izlaznog rastera	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 0.1	Veličina piksela izlaznog rasterskog sloja u jedinicama sloja. U grafičkom okruženju, veličina se može zadati brojem redova (`Broj redova`) / kolona (`Broj kolona`) **ili** veličinom piksela (`Veličina piksela X` / `Veličina piksela Y`). Povećanje broja redova ili kolona smanjiće veličinu ćelije i povećati veličinu datoteke izlaznog rastera. Vrednosti u poljima `Redovi`, `Kolone`, `Veličina piksela X` i `Veličina piksela Y` ažuriraće se istovremeno - udvostručavanje broja redova udvostručiće broj kolona, a veličina ćelije će se prepoloviti. Obuhvat izlaznog rastera ostaće (približno) isti.
Полупречник из поља Opciono	`RADIUS_FIELD`	[polje tabele: numerički]	Поставља полупречник претраге за сваки објекат из атрибутског поља у улазном слоју.
Težina iz polja Opciono	`WEIGHT_FIELD`	[polje tabele: numerički]	Омогућава да улазни објекти буду пондерисани атрибутским пољем. Ово се може користити за повећање утицаја који одређени објекти имају на резултујућу топлотну карту.
Oblik jezgra	`KERNEL`	[nabrajanje] Podrazumevano: *0*	Контролише брзину којом се утицај тачке смањује како се растојање од тачке повећава. Различита језгра опадају различитим брзинама, па triweight језгро даје објектима већу тежину на растојањима ближим тачки него Епанечниковљево језгро. Због тога triweight даје „оштрија“ жаришта, а Епанечниковљево „глатија“ жаришта. Dostupno je mnogo oblika (za više informacija pogledajte stranicu na Vikipediji): • 0 — Kvartično • 1 — Trougaono • 2 — Uniformno • 3 — Triweight • 4 — Epanečnikovljevo

наставак на следећој страни

Табела 24.151 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Odnos opadanja (samo za trougaona jezgra) Opciono	`DECAY`	[numerički: decimalni] Podrazumevano: *0.0*	Može se koristiti sa trougaonim jezgrima za dodatnu kontrolu načina na koji toplota od objekta opada sa rastojanjem od objekta. • Vrednost 0 (=minimum) označava da će toplota biti koncentrisana u centru zadatog poluprečnika i potpuno ugašena na ivici. • Vrednost 0.5 označava da će pikseli na ivici poluprečnika dobiti upola manje toplote od piksela u centru poluprečnika pretrage. • Vrednost 1 znači da je toplota ravnomerno raspoređena po celom krugu poluprečnika pretrage. (Ovo je ekvivalentno „Uniformnom“ jezgru.) • Vrednost veća od 1 označava da je toplota veća ka ivici poluprečnika pretrage nego u centru.
Skaliranje izlazne vrednosti	`OUTPUT_VALUE`	[nabrajanje] Podrazumevano: *0*	Omogućava promenu vrednosti izlaznog rastera toplotne karte. Jedno od: • 0 — Sirovo • 1 — Skalirano
Toplotna karta	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa vrednostima gustine jezgra. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

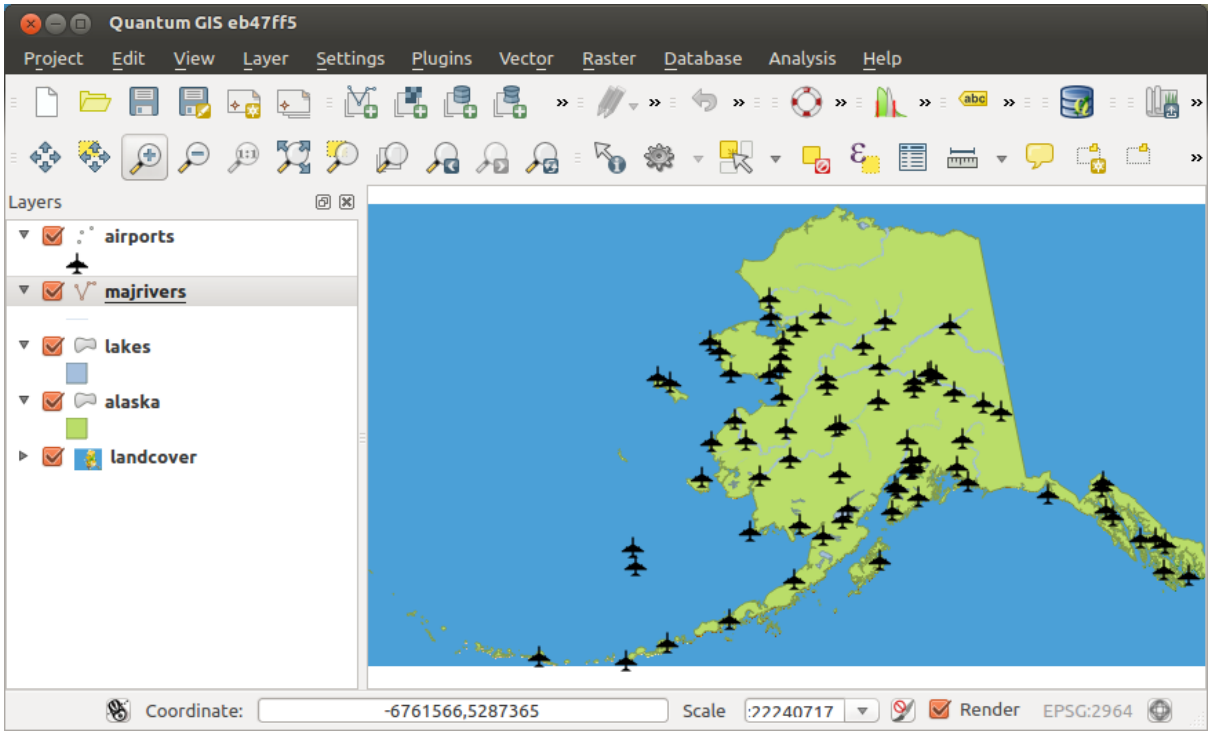
Izlazi

Oznaka	Назив	Tip	Opis
Toplotna karta	`OUTPUT`	[raster]	Rasterski sloj sa vrednostima gustine jezgra

Primer: Pravljenje toplotne karte

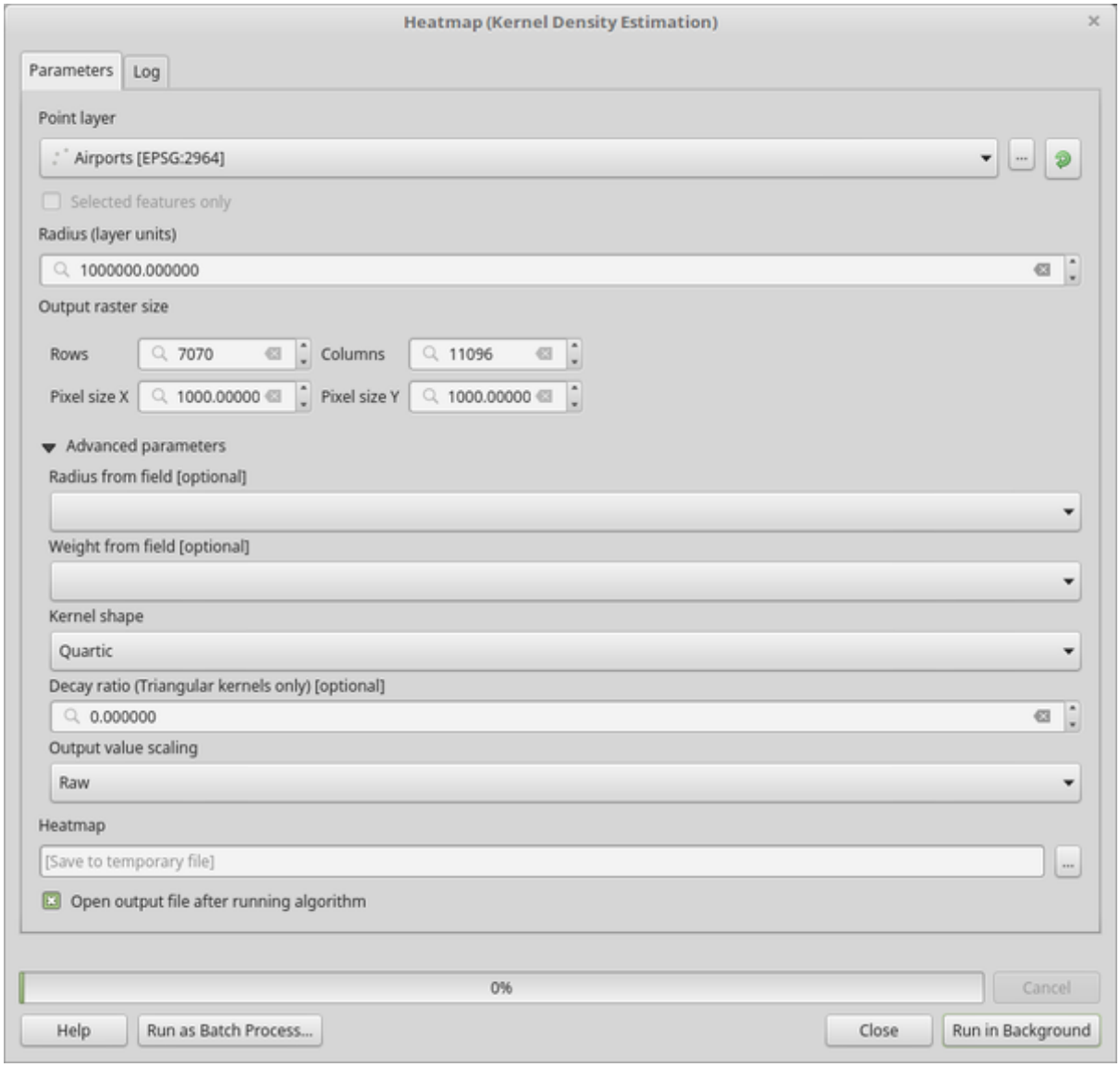
Za sledeći primer koristićemo tačkasti vektorski sloj `airports` iz QGIS uzorka podataka (pogledajte *Preuzimanje primernih podataka*). Još jedan odličan QGIS tutorijal o pravljenju toplotnih karata možete pronaći na http://qgistutorials.com.

Na Сл.24.24 prikazani su aerodromi Aljaske.



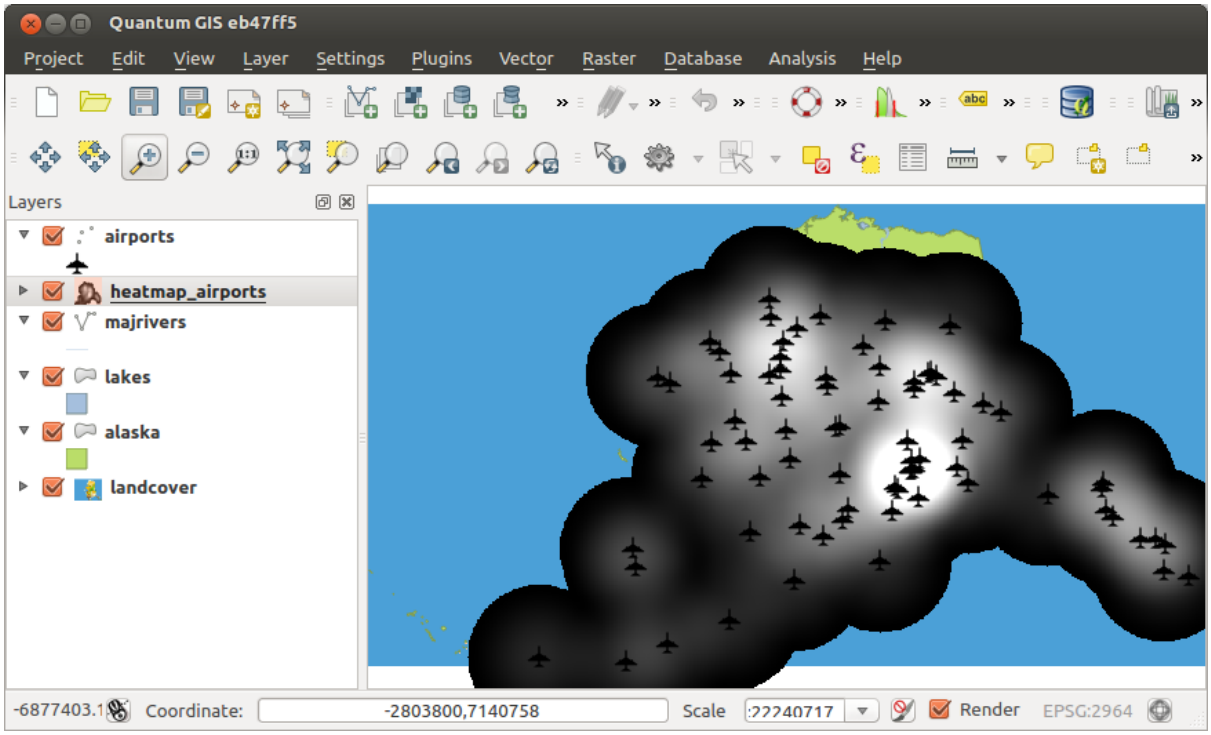
Слика24.24: Aerodromi Aljaske

1. Otvorite algoritam *Toplotna karta (procena gustine jezgrom)* iz QGIS grupe *Interpolacija*
2. U polju *Tačkasti sloj* izaberite `airports` iz liste tačkastih slojeva učitanih u trenutnom projektu.
3. Promenite *Poluprečnik* na `1000000` metara.
4. Promenite *Veličina piksela X* na `1000`. *Veličina piksela Y*, *Redovi* i *Kolone* biće automatski ažurirani.
5. Kliknite na *Pokreni* da biste napravili i učitali toplotnu kartu aerodroma (pogledajte Сл.24.26).



Слика24.25: Dijalog toplotne karte

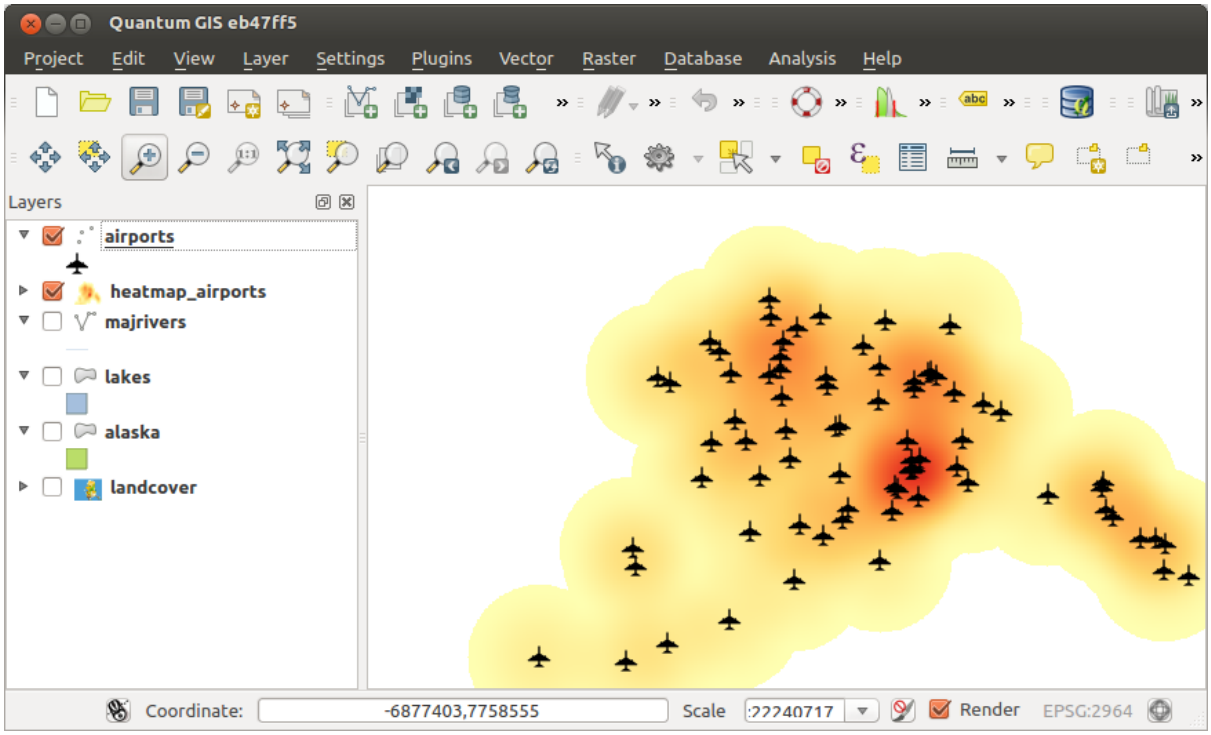
QGIS će generisati toplotnu kartu i dodati je u prozor karte. Podrazumevano je toplotna karta u sivim tonovima, gde svetlije oblasti prikazuju veće koncentracije aerodroma. Toplotna karta se sada može stilizovati u QGIS-u radi poboljšanja izgleda.



Слика24.26: Toplotna karta nakon učitavanja izgleda kao siva površina

1. Otvorite dijalog svojstava sloja `heatmap_airports` (izaberite sloj `heatmap_airports`, otvorite kontekstni meni desnim tasterom miša i izaberite *Svojstva*).
2. Izaberite karticu *Simbologija*.
3. Promenite *Tip iscrtavanja* na „Jednokanalna pseudoboja“.
4. Izaberite odgovarajuću *Rampu boja* , na primer `YlOrRd`.
5. Kliknite na dugme *Klasifikuj*.
6. Pritisnite *OK* da biste ažurirali sloj.

Konačni rezultat prikazan je na Сл.24.27.



Слика24.27: Stilizovana toplotna karta aerodroma Aljaske

Python kod

ID algoritma: `qgis:heatmapkerneldensityestimation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

IDW interpolacija

Generiše interpolaciju tačkastog vektorskog sloja metodom inverznog ponderisanja rastojanja (IDW).

Uzoračke tačke se ponderišu tokom interpolacije tako da uticaj jedne tačke u odnosu na drugu opada sa rastojanjem od nepoznate tačke koju želite da napravite.

IDW metod interpolacije ima i neke nedostatke: kvalitet rezultata interpolacije može opasti ako je raspored uzoračkih tačaka neravnomeran.

Osim toga, maksimalne i minimalne vrednosti na interpoliranoj površini mogu se javiti samo u uzoračkim tačkama.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj(evi)	`INTERPOLATION_DATA`	[niska]	Vektorski sloj(evi) i polje(a) koji se koriste za interpolaciju, kodirani u nisku (za više detalja pogledajte klasu `ParameterInterpolationData` u InterpolationWidgets). Za sastavljanje niske podataka interpolacije obezbeđeni su sledeći elementi grafičkog okruženja: • **Vektorski sloj** [vektor: bilo koji] • **Atribut interpolacije** [tablefield: numerički]: Atribut koji se koristi u interpolaciji • **Koristi Z koordinatu za interpolaciju** [logička vrednost]: Koristi sačuvane Z vrednosti sloja (Podrazumevano: False) Za svaku dodatu kombinaciju sloj-polje može se izabrati tip: • *Tačke*: Uzoračke lokacije visine koje sadrže vrednosti za interpolaciju (npr. visina terena, Z). Ove tačke su primarni ulaz za računanje izlazne površine. • *Strukturne linije*: Linije koje predstavljaju poznate neprekidne osobine površine koje usmeravaju oblik ili strukturu terena (npr. grebeni, ose puteva, nasipi). One opisuju pravac terena uz pretpostavku neprekidnosti duž linije. • *Prelomne linije*: Linije koje predstavljaju poznate diskontinuitete površine ili oštre granice (npr. litice, obale reka, ivice iskopa, obrisi jezera/bara). One označavaju gde se površina može naglo promeniti. Potrebne su ako postoje značajne promene visine duž linije. U niski su elementi sloj-polje razdvojeni sa `'::|::'`. Podelementi elemenata sloj-polje razdvojeni su sa `'::~::'`.
Koeficijent rastojanja P	`DISTANCE_COEFFICIENT`	[numerički: decimalni] Podrazumevano: 2.0	Postavlja koeficijent rastojanja za interpolaciju. Minimum: 0.0, maksimum: 100.0.

наставак на следећој страни

Табела 24.153 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Obuhvat izlaznog rasterskog sloja. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina izlaznog rastera	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 0.1	Veličina piksela izlaznog rasterskog sloja u jedinicama sloja. U grafičkom okruženju, veličina se može zadati brojem redova (`Broj redova`) / kolona (`Broj kolona`) **ili** veličinom piksela (`Veličina piksela X` / `Veličina piksela Y`). Povećanje broja redova ili kolona smanjiće veličinu ćelije i povećati veličinu datoteke izlaznog rastera. Vrednosti u poljima `Redovi`, `Kolone`, `Veličina piksela X` i `Veličina piksela Y` ažuriraće se istovremeno - udvostručavanje broja redova udvostručiće broj kolona, a veličina ćelije će se prepoloviti. Obuhvat izlaznog rastera ostaće (približno) isti.
Interpolirano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Rasterski sloj interpoliranih vrednosti. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano	`OUTPUT`	[raster]	Rasterski sloj interpoliranih vrednosti

Python kod

ID algoritma: `qgis:idwinterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

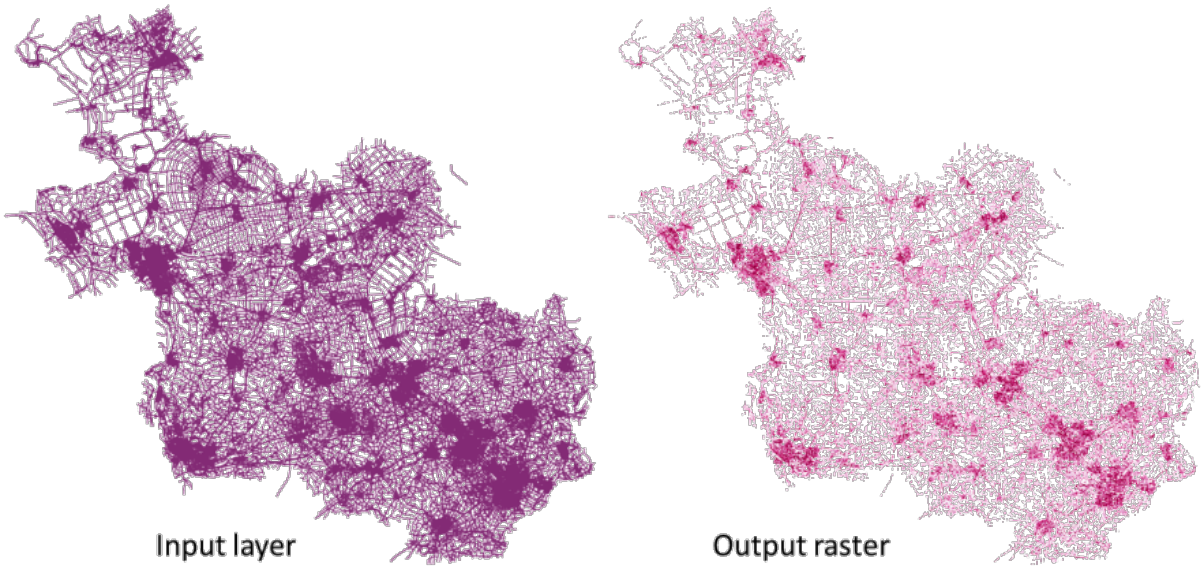
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Gustina linija

Za svaku rastersku ćeliju računa meru gustine linijskih objekata unutar kružnog okruženja. Ova mera se dobija sabiranjem svih segmenata linija koji se seku sa kružnim okruženjem i deljenjem tog zbira površinom tog okruženja. Na segmente linija može se primeniti faktor ponderisanja.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.



Слика24.28: Primer gustine linija. Izvor ulaznog sloja: Roads Overijssel - Holandija (OSM).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni linijski sloj	`INPUT`	[vektor: linija]	Ulazni vektorski sloj koji sadrži linijske objekte
Polje težine	`WEIGHT`	[polje tabele: numerički]	Polje sloja koje sadrži faktor težine koji se koristi tokom računanja
Poluprečnik pretrage	`RADIUS`	[numerički: decimalni] Podrazumevano: 10.0	Poluprečnik kružnog okruženja. Jedinice se mogu zadati ovde.
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 10.0	Veličina piksela izlaznog rasterskog sloja u jedinicama sloja. Raster ima kvadratne piksele.
Raster gustine linija	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Izlaz kao rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Raster gustine linija	`OUTPUT`	[raster]	Izlazni rasterski sloj gustine linija.

Python kod

ID algoritma: `native:linedensity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

TIN interpolacija

Generiše interpolaciju tačkastog vektorskog sloja metodom nepravilne triangulisane mreže (TIN).

Metodom TIN možete napraviti površinu sastavljenu od trouglova najbližih susednih tačaka. Za to se oko izabranih uzoračkih tačaka prave opisane kružnice, a njihovi preseci se povezuju u mrežu nepreklapajućih i što je moguće kompaktnijih trouglova. Rezultujuće površine nisu glatke.

Algoritam pravi i rasterski sloj interpoliranih vrednosti i vektorski linijski sloj sa granicama triangulacije.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj(evi)	`INTERPOLATION_DATA`	[niska]	Vektorski sloj(evi) i polje(a) koji se koriste za interpolaciju, kodirani u nisku (za više detalja pogledajte klasu `ParameterInterpolationData` u InterpolationWidgets). Za sastavljanje niske podataka interpolacije obezbeđeni su sledeći elementi grafičkog okruženja: • **Vektorski sloj** [vektor: bilo koji] • **Atribut interpolacije** [tablefield: numerički]: Atribut koji se koristi u interpolaciji • **Koristi Z koordinatu za interpolaciju** [logička vrednost]: Koristi sačuvane Z vrednosti sloja (Podrazumevano: False) Za svaku dodatu kombinaciju sloj-polje može se izabrati tip: • *Tačke*: Uzoračke lokacije visine koje sadrže vrednosti za interpolaciju (npr. visina terena, Z). Ove tačke su primarni ulaz za računanje izlazne površine. • *Strukturne linije*: Linije koje predstavljaju poznate neprekidne osobine površine koje usmeravaju oblik ili strukturu terena (npr. grebeni, ose puteva, nasipi). One opisuju pravac terena uz pretpostavku neprekidnosti duž linije. • *Prelomne linije*: Linije koje predstavljaju poznate diskontinuitete površine ili oštre granice (npr. litice, obale reka, ivice iskopa, obrisi jezera/bara). One označavaju gde se površina može naglo promeniti. Potrebne su ako postoje značajne promene visine duž linije. U niski su elementi sloj-polje razdvojeni sa `'::|::'`. Podelementi elemenata sloj-polje razdvojeni su sa `'::~::'`.
Metod interpolacije	`METHOD`	[nabrajanje] Podrazumevano: 0	Postavlja metod interpolacije koji se koristi. Jedno od: • *Linearni* • *Clough-Toucher (kubni)*

наставак на следећој страни

Табела 24.158 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Obuhvat izlaznog rasterskog sloja. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina izlaznog rastera	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 0.1	Veličina piksela izlaznog rasterskog sloja u jedinicama sloja. U grafičkom okruženju, veličina se može zadati brojem redova (`Broj redova`) / kolona (`Broj kolona`) **ili** veličinom piksela (`Veličina piksela X` / `Veličina piksela Y`). Povećanje broja redova ili kolona smanjiće veličinu ćelije i povećati veličinu datoteke izlaznog rastera. Vrednosti u poljima `Redovi`, `Kolone`, `Veličina piksela X` i `Veličina piksela Y` ažuriraće se istovremeno - udvostručavanje broja redova udvostručiće broj kolona, a veličina ćelije će se prepoloviti. Obuhvat izlaznog rastera ostaće (približno) isti.
Interpolirano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Izlazna TIN interpolacija kao rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Triangulacija	`TRIANGULATION`	[vektor: linija] Podrazumevano: `[Skip output]`	Izlazni TIN kao vektorski sloj. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano	`OUTPUT`	[raster]	Izlazna TIN interpolacija kao rasterski sloj
Triangulacija	`TRIANGULATION`	[vektor: linija]	Izlazni TIN kao vektorski sloj.

Python kod

ID algoritma: `qgis:tininterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.9 Alatke za slojeve

Izvoz informacij(a) o slojevima

Pravi poligonski sloj sa objektima koji odgovaraju obuhvatu izabranih slojev(a).

Dodatni detalji o sloju (CRS, naziv provajdera, putanja datoteke, naziv sloja, filter podskupa, sažetak i pripisivanje) pridružuju se svakom objektu kao atributi.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[sloj] [lista]	Ulazni slojevi o kojima se preuzimaju informacije.
Izlaz	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja sa informacijama. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[vektor: poligon]	Poligonski vektorski sloj koji prikazuje obuhvat ulaznih slojeva i pridružene informacije u atributima.

Python kod

ID algoritma: `native:exportlayersinformation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u tabelarni dokument

Izvozi atribute izabranih vektorskih slojeva u tabelarni dokument ili ih opciono dodaje u postojeći tabelarni dokument kao dodatne listove.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[vektor: bilo koji] [lista]	Ulazni vektorski slojevi. Izlazni tabelarni dokument sadržaće po jedan list za svaki sloj, sa atributima tog sloja.
Koristi aliase polja kao zaglavlja kolona	`USE_ALIAS`	[logička vrednost] Podrazumevano: False	Koristi aliase polja iz atributske tabele za tabelarni dokument.
Izvezi formatirane vrednosti umesto sirovih	`FORMATTED_VALUES`	[logička vrednost] Podrazumevano: False	Ako je `True`, u tabelarni dokument izvozi formatirane, ljudima čitljive vrednosti (npr. iz *mape vrednosti ili relacije vrednosti*).
Prepiši postojeći tabelarni dokument	`OVERWRITE`	[logička vrednost] Podrazumevano: True	Ako zadati tabelarni dokument postoji, postavljanje ove opcije na `True` prepisaće postojeći dokument. Ako je opcija `False` a dokument postoji, slojevi će biti dodati kao dodatni listovi.
Odredišni tabelarni dokument	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Izlazni tabelarni dokument sa listom za svaki sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Odredišni tabelarni dokument	`OUTPUT`	[datoteka]	Tabelarni dokument sa listom za svaki sloj.
Slojevi unutar tabelarnog dokumenta	`OUTPUT_LAYERS`	[lista]	Lista listova dodatih u tabelarni dokument.

Python kod

ID algoritma: `native:exporttospreadsheet`

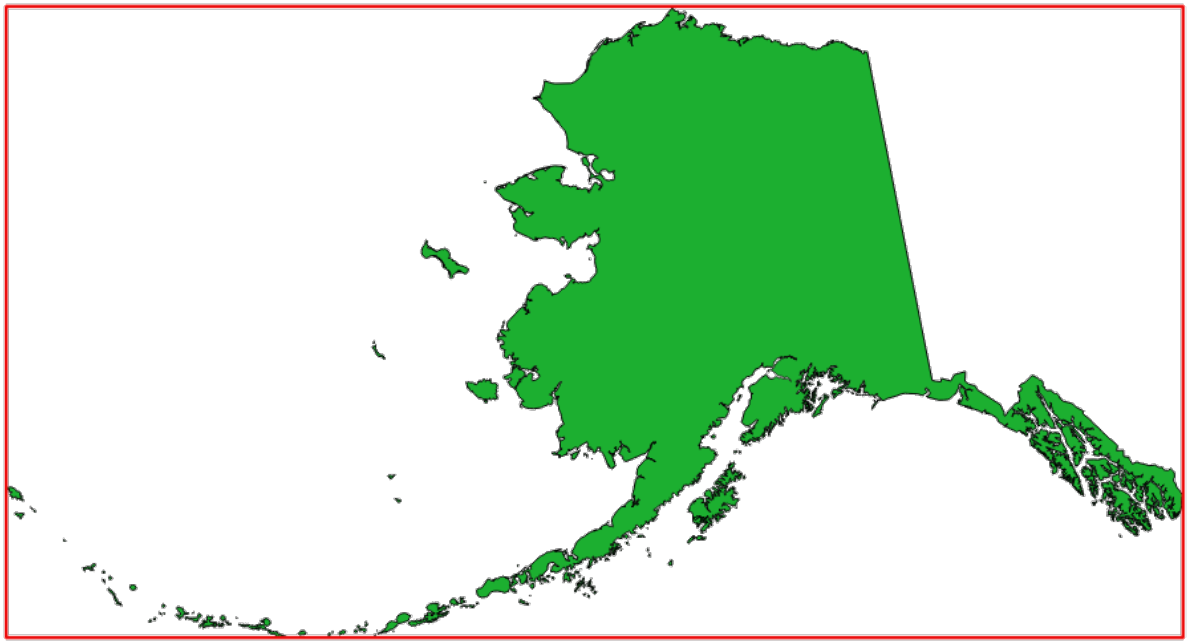
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje obuhvata sloja

Generiše vektorski sloj sa najmanjim graničnim pravougaonikom (pravougaonik sa orijentacijom S-J) koji pokriva sve ulazne objekte.

Izlazni sloj sadrži jedan granični pravougaonik za ceo ulazni sloj.



Слика24.29: Crveno je granični pravougaonik izvornog sloja

Podrazumevani meni: *Vektor ► Istraživačke alatke*

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Ulazni sloj
Obuhvat	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte poligonski vektorski sloj za izlazni obuhvat. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Obuhvat	`OUTPUT`	[vektor: poligon]	Izlazni (poligonski) vektorski sloj sa obuhvatom (najmanji granični pravougaonik)

Python kod

ID algoritma: `native:polygonfromlayerextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.10 Mesh

Izvoz izolinija

Pravi izolinije kao vektorski sloj od skalarnog skupa podataka mreže.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Prirast između nivoa izolinija Opciono	`INCREMENT`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Interval između generisanih nivoa.
Minimalni nivo izolinije Opciono	`MINIMUM`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Početne vrednosti nivoa izolinija. Nijedan generisani nivo neće biti niži od ove vrednosti.
Maksimalni nivo izolinije Opciono	`MAXIMUM`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Maksimalne vrednosti izolinija, tj. nijedan generisani nivo neće biti veći od ove vrednosti.
Lista nivoa izolinija Opciono	`CONTOUR_LEVEL_LIST`	[niska] Podrazumevano: Nije postavljeno	Lista željenih nivoa izolinija (razdvojenih zarezima). Ako je popunjena, polja prirasta, minimuma i maksimuma neće se uzimati u obzir.
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu
Izvezene izolinije	`OUTPUT_LINES`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj koji predstavlja izolinije mrežnog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.166 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izvezeni poligoni izolinija	`OUTPUT_POLYGONS`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski sloj koji predstavlja izolinije mrežnog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izvezene izolinije	`OUTPUT_LINES`	[vektor: linija]	Linijski sloj koji predstavlja izolinije mrežnog sloja.
Izvezeni poligoni izolinija	`OUTPUT_POLYGONS`	[vektor: poligon]	Poligonski sloj koji predstavlja izolinije mrežnog sloja.

Python kod

ID algoritma: `native:meshcontours`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz vrednosti skupa podataka poprečnog preseka na linijama iz mreže

Izdvaja vrednosti skupa podataka mreže iz linija sadržanih u vektorskom sloju.

Svaka linija se diskretizuje pomoću parametra rastojanja rezolucije radi izdvajanja vrednosti na njenim temenima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Linije za izvoz podataka	`INPUT_LINES`	[vektor: linija]	Linije duž kojih će se podaci izdvajati iz skupa podataka mreže

наставак на следећој страни

Табела 24.168 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Rezolucija segmentacije linije	`RESOLUTION`	[numerički: decimalni] Podrazumevano: 10.0	Rastojanje između tačaka na linijama duž kojih će se podaci izdvajati iz skupa podataka mreže. Jedinice se mogu izabrati.
Broj cifara za koordinate	`COORDINATES_DIGITS`	[numerički: ceo broj] Podrazumevano: 2	Broj cifara za zaokruživanje vrednosti koordinata
Broj cifara za vrednost skupa podataka	`DATASET_DIGITS`	[numerički: ceo broj] Podrazumevano: 2	Broj cifara za zaokruživanje vrednosti skupa podataka
Izvezena CSV datoteka podataka	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izvezena CSV datoteka podataka	`OUTPUT`	[datoteka]	

Python kod

ID algoritma: `native:meshexportcrosssection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz ivica mreže

Izvozi ivice mrežnog sloja u linijski vektorski sloj, sa vrednostima skupa podataka na ivicama kao vrednostima atributa.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu

наставак на следећој страни

Табела 24.170 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Opcija izvoza vektora	`VECTOR_OPTION`	[nabrajanje]	Tip koordinata za izvoz vektorskih vrednosti. • 0 — Dekartov (x,y) • 1 — Polarni (intenzitet, stepen) • 2 — Dekartov i polarni
Izlazni vektorski sloj	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Specifikacija izlazne datoteke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni vektorski sloj	`OUTPUT`	[vektor: linija]	Izlazni vektorski linijski sloj koji sadrži ivice ulaznog mrežnog sloja sa pridruženim vrednostima skupa podataka

Python kod

ID algoritma: `native:exportmeshedges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz lica mreže

Izvozi lica mrežnog sloja u poligonski vektorski sloj, sa vrednostima skupa podataka na licima kao vrednostima atributa.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu

наставак на следећој страни

Табела 24.172 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Opcija izvoza vektora	`VECTOR_OPTION`	[nabrajanje]	Tip koordinata za izvoz vektorskih vrednosti. • 0 — Dekartov (x,y) • 1 — Polarni (intenzitet, stepen) • 2 — Dekartov i polarni
Izlazni vektorski sloj	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specifikacija izlazne datoteke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni vektorski sloj	`OUTPUT`	[vektor: poligon]	Izlazni vektorski poligonski sloj koji sadrži lica ulaznog mrežnog sloja sa pridruženim vrednostima skupa podataka

Python kod

ID algoritma: `native:exportmeshfaces`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz mreže na rasterskoj mreži

Izvozi vrednosti skupa podataka mrežnog sloja u tačkasti vektorski sloj u mreži, sa vrednostima skupa podataka u toj tački kao vrednostima atributa.

Za podatke o zapremini (3D naslagane vrednosti skupa podataka), izvezene vrednosti skupa podataka se usrednjavaju po licima pomoću metoda definisanog u *svojstvima mrežnog sloja* (podrazumevano je metod višenivovskog usrednjavanja). 1D mreže nisu podržane.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka

наставак на следећој страни

Табела 24.174 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat nad kojim se obrađuju podaci. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Korak mreže Opciono	`GRID_SPACING`	[numerički: decimalni] Podrazumevano: 10.0	Razmak između uzoračkih tačaka koji se koristi
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu
Opcija izvoza vektora	`VECTOR_OPTION`	[nabrajanje]	Tip koordinata za izvoz vektorskih vrednosti. • 0 — Dekartov (x,y) • 1 — Polarni (intenzitet, stepen) • 2 — Dekartov i polarni
Izlazni vektorski sloj	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specifikacija izlazne datoteke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni vektorski sloj	`OUTPUT`	[vektor: tačka]	Izlazni vektorski tačkasti sloj sa vrednostima skupa podataka izračunatim iz preklopljenog lica.

Python kod

ID algoritma: `native:exportmeshongrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz temena mreže

Izvozi temena mrežnog sloja u tačkasti vektorski sloj, sa vrednostima skupa podataka na temenima kao vrednostima atributa.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu
Opcija izvoza vektora	`VECTOR_OPTION`	[nabrajanje]	Tip koordinata za izvoz vektorskih vrednosti. • 0 — Dekartov (x,y) • 1 — Polarni (intenzitet, stepen) • 2 — Dekartov i polarni
Izlazni vektorski sloj	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specifikacija izlazne datoteke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni vektorski sloj	`OUTPUT`	[vektor: tačka]	Izlazni vektorski tačkasti sloj koji sadrži temena ulaznog mrežnog sloja sa pridruženim vrednostima skupa podataka

Python kod

ID algoritma: `native:exportmeshvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz vrednosti vremenskih serija iz tačaka skupa podataka mreže

Izdvaja vrednosti vremenskih serija skupa podataka mreže iz tačaka sadržanih u vektorskom sloju.

Ako se vremenski korak zadrži na podrazumevanoj vrednosti (0 sati), koristi se vremenski korak prva dva skupa podataka iz prve izabrane grupe skupova podataka.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izdvajaju podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Početno vreme	`STARTING_TIME`	[datetime]	Početak vremenskog opsega koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Krajnje vreme	`FINISHING_TIME`	[datetime]	Kraj vremenskog opsega koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka
Vremenski korak (sati) Opciono	`TIME_STEP`	[numerički: decimalni] Podrazumevano: 0	Vreme između dva uzastopna koraka koja se izdvajaju. Zadržite `0` da biste koristili vremenski korak prve izabrane grupe skupova podataka.
Tačke za izvoz podataka	`INPUT_POINTS`	[vektor: tačka]	Vektorski sloj koji sadrži tačke duž kojih će se podaci izdvajati iz skupa podataka mreže
Broj cifara za koordinate	`COORDINATES_DIGITS`	[numerički: ceo broj] Podrazumevano: 2	Broj cifara za zaokruživanje vrednosti koordinata
Broj cifara za vrednost skupa podataka	`DATASET_DIGITS`	[numerički: ceo broj] Podrazumevano: 2	Broj cifara za zaokruživanje vrednosti skupa podataka
Izvezena CSV datoteka podataka	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izvezena CSV datoteka podataka	`OUTPUT`	[datoteka]	`.CSV` datoteka koja sadrži vrednosti vremenskih serija skupa podataka mreže na preklapajućim tačkastim objektima

Python kod

ID algoritma: `native:meshexporttimeseries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterizacija skupa podataka mreže

Pravi rasterski sloj od skupa podataka mreže.

Za podatke o zapremini (3D naslagane vrednosti skupa podataka), izvezene vrednosti skupa podataka se usrednjavaju po licima pomoću metoda definisanog u *svojstvima mrežnog sloja* (podrazumevano je metod višenivovskog usrednjavanja). 1D mreže nisu podržane.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Mrežni sloj iz kojeg se izvoze podaci
Grupe skupova podataka	`DATASET_GROUPS`	[sloj] [lista]	Grupe skupova podataka
Vreme skupa podataka	`DATASET_TIME`	[datetime]	Vremenski opseg koji se uzima u obzir • 0 — Trenutno vreme platna • 1 — Zadati datum/vreme • 2 — Vremenski korak grupe skupova podataka

наставак на следећој страни

Табела 24.180 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat nad kojim se obrađuju podaci. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela izlaznog rasterskog sloja.
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu
Izlazni rasterski sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni rasterski sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj sa vrednostima skupa podataka izračunatim iz mrežnog sloja.

Python kod

ID algoritma: `native:meshrasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Površ u poligon

`Dodato u 3.42`

Izvozi poligonsku datoteku koja sadrži granicu mrežnog sloja. Rezultujući poligon može sadržati rupe i može biti višedelni poligon.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni mrežni sloj	`INPUT`	[mreža]	Ulazni mrežni sloj čija će granica biti izvezena kao poligon
Izlazni koordinatni sistem Opciono	`OUTPUT_CRS`	[crs]	Koordinatni referentni sistem (CRS) za izlazni poligonski sloj. Ako nije zadat, koristiće se CRS ulaznog mrežnog sloja.
Izlazni vektorski sloj	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specifikacija izlazne datoteke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni vektorski sloj	`OUTPUT`	[vektor: poligon]	Rezultujući poligonski sloj koji sadrži granicu mrežnog sloja. Poligon može sadržati rupe i može biti višedelni.

Python kod

ID algoritma: `native:surfacetopolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pravljenje TIN mreže

Pravi TIN mrežni sloj od vektorskih slojeva. TIN mreža se pravi pomoću Delonijeve triangulacije.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`SOURCE_DATA`	[vektor: bilo koji] [lista]	Vektorski slojevi koji se kombinuju za generisanje mrežnog sloja
Vektorski sloj	SAMO GRAFIČKO OKRUŽENJE	[vektor: bilo koji] [lista]	Birač vektorskih slojeva koji se kombinuju za generisanje mrežnog sloja
Vrednost na temenu	SAMO GRAFIČKO OKRUŽENJE	[polje tabele: bilo koji]	Birač polja koje se koristi iz izabranog sloja. Svakom temenu se dodeljuje odgovarajuća vrednost njegovog izvornog objekta.
Koristi Z koordinatu za vrednost na temenu	SAMO GRAFIČKO OKRUŽENJE	[logička vrednost] Podrazumevano: False	Ako je uključeno, Z vrednost tačaka vektorskog sloja ili temena poligona/linija koristiće se za dodelu Z vrednosti temena mrežnog sloja. Dostupno samo ako su ulazni slojevi 3D.
Izlazni format	`MESH_FORMAT`	[nabrajanje] Podrazumevano: 2DM	Izlazni format generisanog sloja • 0 — 2DM • 1 — SELAFIN • 2 — PLY • 3 — Ugrid • 4 — Mike21
Izlazni koordinatni sistem Opciono	`CRS_OUTPUT`	[crs]	Koordinatni referentni sistem koji se dodeljuje izlazu
Izlazna datoteka	`OUTPUT_MESH`	[mreža] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazna datoteka	`OUTPUT_MESH`	[mreža]	Izlazni mrežni sloj sa vrednostima skupa podataka izračunatim iz vektorskih slojeva.

Python kod

ID algoritma: `native:tinmeshcreation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.11 Alati za metapodatke

Dodaj metapodatke istorije

`Dodato u 3.42`

Dodaje novi unos istorije u metapodatke sloja.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Ulazni sloj (vektorski, rasterski itd.) u koji se dodaje unos istorije.
Unos istorije	`HISTORY`	[niska]	Tekst koji se dodaje kao novi unos u metapodatke istorije sloja. Biće dodat na sve postojeće unose istorije.

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[isto kao ulaz]	Rezultujući sloj sa ažuriranom istorijom u njegovim **Svojstvima metapodataka**.

Python kod

ID algoritma: `native:addhistorymetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kopiraj metapodatke sloja

`Dodato u 3.42`

Kopira metapodatke sa izvornog sloja na ciljni sloj. Svi postojeći metapodaci u ciljnom sloju biće zamenjeni metapodacima izvornog sloja. Ovo uključuje sva polja metapodataka, kao što su istorija, apstrakt i druga svojstva.

Parametri

Oznaka	Назив	Tip	Opis
Izvorni sloj	`SOURCE`	[sloj]	Sloj sa kojeg se kopiraju metapodaci.
Ciljni sloj	`TARGET`	[sloj]	Sloj u koji se lepe metapodaci. Svi postojeći metapodaci u ovom sloju biće zamenjeni.
Sačuvaj metapodatke kao podrazumevane	`DEFAULT`	[logička vrednost] Podrazumevano: False	Ako je uključeno, informacija o metapodacima biće sačuvana uz sloj, pa će podrazumevano biti dostupna u kasnijim projektima.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[sloj]	Ciljni sloj sa metapodacima zamenjenim metapodacima izvornog sloja. Ovo uključuje sva polja metapodataka, kao što su istorija, apstrakt i druga svojstva.

Python kod

ID algoritma: `native:copylayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvezi metapodatke sloja

`Dodato u 3.42`

Izvozi metapodatke sloja u QMD datoteku.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Sloj čiji se metapodaci izvoze.
Izlaz	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazna datoteka	`OUTPUT`	[datoteka]	`.qmd` datoteka koja sadrži izvezene metapodatke. Ova datoteka se može koristiti za uvoz metapodataka u drugi sloj.

Python kod

ID algoritma: `native:exportlayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi metapodatke sloja

`Dodato u 3.42`

Primenjuje metapodatke na sloj iz `.qmd` datoteke.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Sloj na koji se primenjuju metapodaci.
Datoteka metapodataka	`METADATA`	[datoteka]	`.qmd` datoteka koja sadrži metapodatke koji se primenjuju.
Sačuvaj metapodatke kao podrazumevane	`DEFAULT`	[logička vrednost] Podrazumevano: False	Ako je uključeno, informacija o metapodacima biće sačuvana uz sloj, pa će podrazumevano biti dostupna u kasnijim projektima.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[isto kao ulaz]	Ulazni sloj sa metapodacima zamenjenim metapodacima iz `.qmd` datoteke.

Python kod

ID algoritma: `native:setlayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi polja metapodataka

`Dodato u 3.42`

Postavlja razna polja metapodataka za sloj.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Sloj čija se polja metapodataka ažuriraju
Identifikator Opciono	`IDENTIFIER`	[niska]	Jedinstveni identifikator sloja
Identifikator roditelja Opciono	`PARENT_IDENTIFIER`	[niska]	Identifikator nadređenog resursa, ako je primenljivo
Naslov Opciono	`TITLE`	[niska]	Naslov sloja
Tip Opciono	`TYPE`	[niska]	Tip podataka sačuvanih u sloju

наставак на следећој страни

Табела 24.195 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Jezik Opciono	`LANGUAGE`	[niska]	Jezik metapodataka
Kodiranje Opciono	`ENCODING`	[niska]	Kodiranje znakova koje se koristi u metapodacima
Apstrakt Opciono	`ABSTRACT`	[niska]	Kratak opis ili apstrakt sloja
Koordinatni referentni sistem Opciono	`CRS`	[crs]	Koordinatni referentni sistem sloja
Naknade Opciono	`FEES`	[niska]	Informacije o eventualnim naknadama povezanim sa pristupom sloju
Zanemari prazna polja	`IGNORE_EMPTY`	[logička vrednost] Podrazumevano: False	Ako je uključeno, neće se ažurirati polja metapodataka koja nisu popunjena

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[isto kao ulaz]	Ulazni sloj sa ažuriranim navedenim poljima metapodataka.

Python kod

ID algoritma: `native:setmetadatafields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Ažuriraj metapodatke sloja

`Dodato u 3.42`

Kopira sva nepopunjena polja metapodataka sa izvornog sloja na ciljni sloj. Prazna ulazna polja ostavlja nepromenjena u ciljnom.

Parametri

Oznaka	Назив	Tip	Opis
Izvorni sloj	`SOURCE`	[sloj]	Sloj sa kojeg se kopiraju nepopunjena polja metapodataka
Ciljni sloj	`TARGET`	[sloj]	Sloj u koji se lepe nepopunjena polja metapodataka. Prazna polja u izvornom sloju neće prepisati postojeće metapodatke u ciljnom sloju.

Izlazi

Oznaka	Назив	Tip	Opis
Ciljni sloj	`OUTPUT`	[sloj]	Ciljni sloj sa ažuriranim metapodacima.

Python kod

ID algoritma: `native:updatelayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.12 Alati za modelovanje

Упозорење

Ovi alati su dostupni samo u dizajneru modela. Nisu dostupni u kutiji sa alatkama za obradu.

Izračunaj izraz

`Dodato u 3.34`

Računa rezultat QGIS izraza i uklanja potrebu za višestrukim korišćenjem istog izraza kroz model kada je isti rezultat potreban više puta. Osim toga, omogućava slučajeve upotrebe koji inače ne bi bili mogući. Na primer, možete jednom generisati vrednost vremenske oznake i koristiti je više puta unutar modela; ako bi se vremenska oznaka ponovo računala svaki put, vrednosti bi se razlikovale tokom rada modela.

Parametri

Oznaka	Назив	Tip	Opis
Ulaz	`INPUT`	[izraz]	Izraz koji se računa

Izlazi

Oznaka	Назив	Tip	Opis
Vrednost	`OUTPUT`	[Vrednost rezultata]	Izračunata vrednost rezultata; tip podataka izlaza zavisiće od konkretnog izraza korišćenog u algoritmu.

Python kod

ID algoritma: `native:calculateexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uslovna grana

Dodaje uslovnu granu u model, omogućavajući da se delovi modela izvršavaju na osnovu rezultata izračunavanja izraza. Uglavnom korišćenjem zavisnosti između alatki za kontrolu toka modela.

Parametri

Oznaka	Назив	Tip	Opis
Polje	`BRANCH`	[niska]	Naziv uslova
Polje	`CONDITION`	[izraz]	Izraz koji se izračunava

Izlazi

Nijedan.

Python kod

ID algoritma: `native:condition`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi direktorijum

Pravi novi direktorijum u sistemu datoteka. Direktorijumi će se praviti rekurzivno, praveći sve potrebne nadređene direktorijume radi konstruisanja pune zadate putanje direktorijuma. Neće se prijaviti greška ako direktorijum već postoji.

Parametri

Oznaka	Назив	Tip	Opis
Putanja direktorijuma	`PATH`	[niska]	Putanja foldera koji se pravi

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[folder]	Napravljeni folder

Python kod

ID algoritma: `native:createdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filter objekata

Filtrira objekte iz ulaznog sloja i preusmerava ih na jedan ili više izlaza. Ako ne znate nazive nijednog atributa zajedničkog za sve moguće ulazne slojeve, filtriranje je moguće samo po geometriji objekta i opštim mehanizmima zapisa, kao što su `$id` i `uuid`.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni sloj.
Izlazi i filteri (jedan ili više)	`OUTPUT_<name of the filter>`	[isto kao ulaz]	Izlazni slojevi sa filterima (onoliko koliko ima filtera).

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz (jedan ili više)	`native:filter_1:OUTP of filter>`	[isto kao ulaz]	Izlazni slojevi sa filtriranim objektima (onoliko koliko ima filtera).

Python kod

ID algoritma: `native:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filtriraj slojeve po tipu

Filtrira slojeve po njihovom tipu. Ulazni slojevi biće usmereni na različite izlaze u zavisnosti od toga da li su vektorski ili rasterski sloj.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[sloj]	Generički sloj karte

Izlazi

Oznaka	Назив	Tip	Opis
Vektorski objekti Opciono	`VECTOR`	[vektor]	Vektorski sloj ulaza, ako je kompatibilan
Rasterski sloj Opciono	`RASTER`	[raster]	Rasterski sloj ulaza, ako je kompatibilan

Python kod

ID algoritma: `native:filterlayersbytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Učitaj sloj u projekat

Učitava sloj u trenutni projekat.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Sloj koji se učitava u legendu
Naziv učitanog sloja	`NAME`	[niska]	Naziv učitanog sloja

Izlazi

Oznaka	Назив	Tip	Opis
Sloj	`OUTPUT`	[isto kao ulaz]	(Preimenovani) učitani sloj

Python kod

ID algoritma: `native:loadlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbaci izuzetak

Izbacuje izuzetak i otkazuje izvršavanje modela. Poruka izuzetka se može prilagoditi, a opciono se može zadati i uslov zasnovan na izrazu. Ako se koristi uslov izraza, izuzetak će biti izbačen samo ako je rezultat izraza tačan. Netačan rezultat znači da izuzetak neće biti izbačen i da izvršavanje modela može neprekidno da se nastavi.

Parametri

Oznaka	Назив	Tip	Opis
Poruka	`MESSAGE`	[niska]	Poruka koja se prikazuje
Uslov Opciono	`CONDITION`	[izraz]	Izraz koji se izračunava ako je tačan

Izlazi

Poruka u panelu dnevnika.

Python kod

ID algoritma: `native:raiseexception`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbaci poruku

Izbacuje informativnu poruku u dnevnik. Poruka se može prilagoditi, a opciono se može zadati i uslov zasnovan na izrazu. Ako se koristi uslov izraza, poruka će biti zabeležena samo ako je rezultat izraza tačan. Netačan rezultat znači da poruka neće biti zabeležena.

Parametri

Oznaka	Назив	Tip	Opis
Informativna poruka	`MESSAGE`	[niska]	Poruka koja se prikazuje
Uslov Opciono	`CONDITION`	[izraz]	Izraz koji se izračunava ako je tačan

Izlazi

Poruka u panelu dnevnika.

Python kod

ID algoritma: `native:raisemessage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbaci upozorenje

Izbacuje poruku upozorenja u dnevnik. Poruka upozorenja se može prilagoditi, a opciono se može zadati i uslov zasnovan na izrazu. Ako se koristi uslov izraza, upozorenje će biti zabeleženo samo ako je rezultat izraza tačan. Netačan rezultat znači da upozorenje neće biti zabeleženo.

Parametri

Oznaka	Назив	Tip	Opis
Poruka	`MESSAGE`	[niska]	Poruka koja se prikazuje
Uslov Opciono	`CONDITION`	[izraz]	Izraz koji se izračunava ako je tačan

Izlazi

Poruka u panelu dnevnika.

Python kod

ID algoritma: `native:raisewarning`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preimenuj sloj

Preimenuje sloj.

Parametri

Oznaka	Назив	Tip	Opis
Sloj	`INPUT`	[sloj]	Sloj koji se preimenuje
Novi naziv	`NAME`	[niska]	Novi naziv sloja

Izlazi

Oznaka	Назив	Tip	Opis
Sloj	`OUTPUT`	[isto kao ulaz]	(Preimenovani) izlazni sloj

Python kod

ID algoritma: `native:renamelayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Sačuvaj dnevnik u datoteku

Čuva dnevnik izvršavanja modela u datoteku. Opciono, dnevnik se može sačuvati u HTML formatiranoj verziji.

Parametri

Oznaka	Назив	Tip	Opis
Koristi HTML	`USE_HTML`	[logička vrednost] Podrazumevano: False	Koristi HTML formatiranje

Izlazi

Oznaka	Назив	Tip	Opis
Datoteka	`OUTPUT`	[niska]	Odredište dnevnika

Python kod

ID algoritma: `native:savelog`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi promenljivu projekta

Postavlja promenljivu izraza za trenutni projekat.

Parametri

Oznaka	Назив	Tip	Opis
Naziv promenljive	`NAME`	[niska]	Naziv promenljive
Vrednost promenljive	`VALUE`	[niska]	Vrednost koja se čuva

Izlazi

Nijedan.

Python kod

ID algoritma: `native:setprojectvariable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje niski

Spaja dve niske u jednu u dizajneru modela obrade.

Parametri

Oznaka	Назив	Tip	Opis
Ulaz 1	`INPUT_1`	[niska]	Prva niska
Ulaz 2	`INPUT_2`	[niska]	Druga niska

Izlazi

Oznaka	Назив	Tip	Opis
Spajanje	`CONCATENATION`	[niska]	Spojena niska

Python kod

ID algoritma: `native:stringconcatenation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Bafer promenljivog rastojanja

Упозорење

Ovaj algoritam je zastareo i može biti uklonjen u svakom trenutku. Umesto njega koristite algoritam *Bafer*.

Računa oblast bafera za sve objekte u ulaznom sloju.

Veličina bafera za zadati objekat definiše se atributom, pa omogućava da različiti objekti imaju različite veličine bafera.

See also

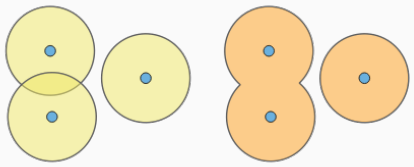
Bafer

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Polje rastojanja	`DISTANCE`	[polje tabele: numerički]	Atribut za poluprečnik rastojanja bafera
Segmenti	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: *5*	Kontroliše broj segmenata linije koji se koriste za aproksimaciju četvrtine kruga pri pravljenju zaobljenih pomeraja.

наставак на следећој страни

Табела 24.220 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Stopi rezultat	DISSOLVE	[logička vrednost] Podrazumevano: *False*	Izaberite da stopite konačni bafer, čime se dobija jedan objekat koji pokriva sve ulazne objekte.  Слика24.30: Normalan i stopljen bafer
Stil završetka	END_CAP_STYLE	[nabrajanje] Podrazumevano: Zaobljen	Kontroliše kako se rukuje završecima linija u baferu.  Слика24.31: Zaobljen, ravan i kvadratni stil završetka
Stil spoja	JOIN_STYLE	[nabrajanje] Podrazumevano: Zaobljen	Zadaje da li se pri pomeranju uglova u liniji koriste zaobljeni, oštri ili zasečeni spojevi.
Granica oštrog spoja	MITER_LIMIT	[numerički: decimalni] Podrazumevano: 2.0	Primenljivo samo za oštre stilove spoja, kontroliše maksimalno rastojanje od krive pomeraja koje se koristi pri pravljenju oštrog spoja.

Izlazi

Oznaka	Назив	Tip	Opis
Bafer	OUTPUT	[vektor: poligon]	Poligonski vektorski sloj bafera.

Python kod

ID algoritma: `qgis:variabledistancebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.13 Analiza mreže

Oblast dostupnosti (iz sloja)

Vraća sve ivice ili delove ivica mreže do kojih se može stići unutar zadatog rastojanja ili vremena, polazeći od tačkastog sloja. Ovo omogućava procenu dostupnosti unutar mreže, npr. do kojih mesta mogu da stignem na putnoj mreži bez utroška većeg od zadate vrednosti (utrošak može biti rastojanje ili vreme).

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski sloj koji predstavlja mrežu	`INPUT`	[vektor: linija]	Linijski vektorski sloj koji predstavlja mrežu koja se pokriva
Vektorski sloj sa početnim tačkama	`START_POINTS`	[vektor: tačka]	Tačkasti vektorski sloj čiji se objekti koriste kao početne tačke za generisanje oblasti dostupnosti
Tip putanje koja se računa	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Tip putanje koja se računa. Jedno od: • 0 — Najkraća • 1 — Najbrža
Trošak putovanja (rastojanje za „Najkraća“, vreme za „Najbrža“)	`TRAVEL_COST`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost se procenjuje kao rastojanje (u jedinicama sloja mreže) pri traženju *Najkraće* putanje i kao vreme (u satima) za *Najbržu* putanju.
Oblast dostupnosti (linije)	`OUTPUT_LINES`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj za oblast dostupnosti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Oblast dostupnosti (granični čvorovi)	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Skip output]`	Zadajte izlazni tačkasti sloj za granične čvorove oblasti dostupnosti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.222 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Objekti kojima se ne može trasirati `Dodato u 3.38`	`OUTPUT_NON_ROUTABLE`	[vektor: tačka] Podrazumevano: `[Skip output]`	Zadajte izlaz koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže). *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Polje pravca Opciono	`DIRECTION_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za zadavanje pravaca ivica mreže. Vrednosti koje se koriste u ovom polju zadaju se sa tri parametra `Vrednost za pravac napred`, `Vrednost za pravac nazad` i `Vrednost za oba pravca`. Pravci napred i nazad odgovaraju jednosmernoj ivici, „oba pravca“ označava dvosmernu ivicu. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevano podešavanje pravca (zadato parametrom `Podrazumevani pravac`).
Vrednost za pravac napred Opciono	`VALUE_FORWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem napred
Vrednost za pravac nazad Opciono	`VALUE_BACKWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem nazad
Vrednost za oba pravca Opciono	`VALUE_BOTH`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje dvosmernih ivica
Podrazumevani pravac	`DEFAULT_DIRECTION`	[nabrajanje] Podrazumevano: 2	Ako objekat nema postavljenu vrednost u polju pravca ili ako polje pravca nije postavljeno, koristi se ova vrednost pravca. Jedno od: • 0 — Pravac napred • 1 — Pravac nazad • 2 — Oba pravca
Polje brzine Opciono	`SPEED_FIELD`	[polje tabele: numerički]	Polje koje daje vrednost brzine (u `km/h`) za ivice mreže pri traženju najbrže putanje. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevana vrednost brzine (zadata parametrom `Podrazumevana brzina`).

наставак на следећој страни

Табела 24.223 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Podrazumevana brzina (km/h)	`DEFAULT_SPEED`	[numerički: decimalni] Podrazumevano: 50.0	Vrednost koja se koristi za računanje vremena putovanja ako za ivicu nije zadata vrednost brzine u navedenom polju
Tolerancija topologije	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Dve linije sa čvorovima bližim od zadate tolerancije smatraju se povezanim
Uključi tačke gornje/donje granice	`INCLUDE_BOUNDS`	[logička vrednost] Podrazumevano: False	Pravi izlazni tačkasti sloj sa po dve tačke za svaku ivicu na granicama oblasti dostupnosti. Jedna tačka je početak te ivice, druga je kraj.
Maksimalno rastojanje tačke od mreže Opciono `Dodato u 3.38`	`POINT_TOLERANCE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Zadaje opciono ograničenje rastojanja od tačaka do sloja mreže. Ako je tačka udaljenija od mreže od ovog rastojanja, tretiraće se kao tačka kojoj se ne može trasirati. Ako nije postavljeno, krajnje tačke biće zalepljene za najbližu tačku na sloju mreže, bez obzira na to koliko su zapravo udaljene od mreže.

Izlazi

Oznaka	Назив	Tip	Opis
Oblast dostupnosti (granični čvorovi)	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti sloj sa graničnim čvorovima oblasti dostupnosti.
Oblast dostupnosti (linije)	`OUTPUT_LINES`	[vektor: linija]	Linijski sloj koji predstavlja delove mreže koje početne tačke mogu opslužiti za zadati trošak.
Objekti kojima se ne može trasirati `Dodato u 3.38`	`OUTPUT_NON_ROUTABLE`	[vektor: tačka]	Opcioni izlaz koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže).

Python kod

ID algoritma: `native:serviceareafromlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Oblast dostupnosti (iz tačke)

Vraća sve ivice ili delove ivica mreže do kojih se može stići unutar zadatog rastojanja ili vremena, polazeći od tačkastog objekta. Ovo omogućava procenu dostupnosti unutar mreže, npr. do kojih mesta mogu da stignem na putnoj mreži bez utroška većeg od zadate vrednosti (utrošak može biti rastojanje ili vreme).

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski sloj koji predstavlja mrežu	`INPUT`	[vektor: linija]	Linijski vektorski sloj koji predstavlja mrežu koja se pokriva
Tip putanje koja se računa	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Tip putanje koja se računa. Jedno od: • 0 — Najkraća • 1 — Najbrža
Početna tačka	`START_POINT`	[koordinate]	Koordinata tačke oko koje se računa oblast dostupnosti. Pritisnite dugme … pored opcije i kliknite na platno da biste parametar popunili koordinatom tačke na koju ste kliknuli.
Trošak putovanja (rastojanje za „Najkraća“, vreme za „Najbrža“)	`TRAVEL_COST`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost se procenjuje kao rastojanje (u jedinicama sloja mreže) pri traženju *Najkraće* putanje i kao vreme (u satima) za *Najbržu* putanju.
Oblast dostupnosti (linije) Opciono	`OUTPUT_LINES`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj za oblast dostupnosti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Oblast dostupnosti (granični čvorovi) Opciono	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Skip output]`	Zadajte izlazni tačkasti sloj za granične čvorove oblasti dostupnosti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Polje pravca Opciono	`DIRECTION_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za zadavanje pravaca ivica mreže. Vrednosti koje se koriste u ovom polju zadaju se sa tri parametra `Vrednost za pravac napred`, `Vrednost za pravac nazad` i `Vrednost za oba pravca`. Pravci napred i nazad odgovaraju jednosmernoj ivici, „oba pravca“ označava dvosmernu ivicu. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevano podešavanje pravca (zadato parametrom `Podrazumevani pravac`).
Vrednost za pravac napred Opciono	`VALUE_FORWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem napred
Vrednost za pravac nazad Opciono	`VALUE_BACKWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem nazad
Vrednost za oba pravca Opciono	`VALUE_BOTH`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje dvosmernih ivica
Podrazumevani pravac	`DEFAULT_DIRECTION`	[nabrajanje] Podrazumevano: 2	Ako objekat nema postavljenu vrednost u polju pravca ili ako polje pravca nije postavljeno, koristi se ova vrednost pravca. Jedno od: • 0 — Pravac napred • 1 — Pravac nazad • 2 — Oba pravca
Polje brzine Opciono	`SPEED_FIELD`	[polje tabele: numerički]	Polje koje daje vrednost brzine (u `km/h`) za ivice mreže pri traženju najbrže putanje. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevana vrednost brzine (zadata parametrom `Podrazumevana brzina`).
Podrazumevana brzina (km/h)	`DEFAULT_SPEED`	[numerički: decimalni] Podrazumevano: 50.0	Vrednost koja se koristi za računanje vremena putovanja ako za ivicu nije zadata vrednost brzine u navedenom polju
Tolerancija topologije	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Dve linije sa čvorovima bližim od zadate tolerancije smatraju se povezanim

Maksimalno rastojanje tačke od mreže Opciono `Dodato u 3.38`	`POINT_TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Zadaje opciono ograničenje rastojanja od početne tačke do sloja mreže. Ako je tačka udaljenija od mreže od ovog rastojanja, prijaviće se greška. Ako nije postavljeno, tačka će biti zalepljena za najbližu tačku na sloju mreže, bez obzira na to koliko je zapravo udaljena od mreže.
Uključi tačke gornje/donje granice	`INCLUDE_BOUNDS`	[logička vrednost] Podrazumevano: False	Pravi izlazni tačkasti sloj sa po dve tačke za svaku ivicu na granicama oblasti dostupnosti. Jedna tačka je početak te ivice, druga je kraj.

Izlazi

Oznaka	Назив	Tip	Opis
Oblast dostupnosti (granični čvorovi)	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti sloj sa graničnim čvorovima oblasti dostupnosti.
Oblast dostupnosti (linije)	`OUTPUT_LINES`	[vektor: linija]	Linijski sloj koji predstavlja delove mreže koje početna tačka može opslužiti za zadati trošak.

Python kod

ID algoritma: `native:serviceareafrompoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Najkraća putanja (sloj do tačke)

Računa optimalne (najkraće ili najbrže) rute od više početnih tačaka definisanih vektorskim slojem do zadate krajnje tačke.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski sloj koji predstavlja mrežu	`INPUT`	[vektor: linija]	Linijski vektorski sloj koji predstavlja mrežu koja se pokriva

наставак на следећој страни

Табела 24.228 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip putanje koja se računa	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Tip putanje koja se računa. Jedno od: • 0 — Najkraća • 1 — Najbrža
Vektorski sloj sa početnim tačkama	`START_POINTS`	[vektor: tačka]	Tačkasti vektorski sloj čiji se objekti koriste kao početne tačke ruta
Krajnja tačka (x, y)	`END_POINT`	[koordinate]	Tačkasti objekat koji predstavlja krajnju tačku ruta Pritisnite dugme … pored opcije i kliknite na platno da biste parametar popunili koordinatom tačke na koju ste kliknuli.
Najkraća putanja	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj za najkraće putanje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Objekti kojima se ne može trasirati Opciono `Dodato u 3.38`	`OUTPUT_NON_ROUTABLE`	[vektor: tačka] Podrazumevano: `[Skip output]`	Zadajte izlaz koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže). *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Polje pravca Opciono	`DIRECTION_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za zadavanje pravaca ivica mreže. Vrednosti koje se koriste u ovom polju zadaju se sa tri parametra `Vrednost za pravac napred`, `Vrednost za pravac nazad` i `Vrednost za oba pravca`. Pravci napred i nazad odgovaraju jednosmernoj ivici, „oba pravca“ označava dvosmernu ivicu. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevano podešavanje pravca (zadato parametrom `Podrazumevani pravac`).
Vrednost za pravac napred Opciono	`VALUE_FORWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem napred

наставак на следећој страни

Табела 24.229 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Vrednost za pravac nazad Opciono	VALUE_BACKWARD	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem nazad
Vrednost za oba pravca Opciono	VALUE_BOTH	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje dvosmernih ivica
Podrazumevani pravac	DEFAULT_DIRECTION	[nabrajanje] Podrazumevano: 2	Ako objekat nema postavljenu vrednost u polju pravca ili ako polje pravca nije postavljeno, koristi se ova vrednost pravca. Jedno od: • 0 — Pravac napred • 1 — Pravac nazad • 2 — Oba pravca
Polje brzine Opciono	SPEED_FIELD	[polje tabele: numerički]	Polje koje daje vrednost brzine (u km/h) za ivice mreže pri traženju najbrže putanje. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevana vrednost brzine (zadata parametrom Podrazumevana brzina).
Podrazumevana brzina (km/h)	DEFAULT_SPEED	[numerički: decimalni] Podrazumevano: 50.0	Vrednost koja se koristi za računanje vremena putovanja ako za ivicu nije zadata vrednost brzine u navedenom polju
Tolerancija topologije	TOLERANCE	[numerički: decimalni] Podrazumevano: 0.0	Dve linije sa čvorovima bližim od zadate tolerancije smatraju se povezanim

Maksimalno rastojanje tačke od mreže Opciono Dodato u 3.38	POINT_TOLERANCE	[numerički: decimalni] Podrazumevano: Nije postavljeno	Zadaje opciono ograničenje rastojanja od početnih i krajnjih tačaka do sloja mreže. Ako je početni objekat udaljeniji od mreže od ovog rastojanja, tretiraće se kao objekat kojem se ne može trasirati. Ako je krajnja tačka udaljenija od mreže od ovog rastojanja, prijaviće se greška. Ako nije postavljeno, tačke će biti zalepljene za najbližu tačku na sloju mreže, bez obzira na to koliko su zapravo udaljene od mreže.

Izlazi

Oznaka	Назив	Tip	Opis
Najkraća putanja	OUTPUT	[vektor: linija]	Linijski sloj najkraće ili najbrže putanje od svake početne tačke do krajnje tačke
Objekti kojima se ne može trasirati Dodato u 3.38	OUTPUT_NON_ROUTABLE	[vektor: tačka]	Opcioni izlazni sloj koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže).

Python kod

ID algoritma: `native:shortestpathlayertopoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Najkraća putanja (tačka do sloja)

Računa optimalne (najkraće ili najbrže) rute između zadate početne tačke i više krajnjih tačaka definisanih tačkastim vektorskim slojem.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski sloj koji predstavlja mrežu	`INPUT`	[vektor: linija]	Linijski vektorski sloj koji predstavlja mrežu koja se pokriva
Tip putanje koja se računa	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Tip putanje koja se računa. Jedno od: • 0 — Najkraća • 1 — Najbrža
Početna tačka	`START_POINT`	[koordinate]	Tačkasti objekat koji predstavlja početnu tačku ruta Pritisnite dugme … pored opcije i kliknite na platno da biste parametar popunili koordinatom tačke na koju ste kliknuli.
Vektorski sloj sa krajnjim tačkama	`END_POINTS`	[vektor: tačka]	Tačkasti vektorski sloj čiji se objekti koriste kao krajnje tačke ruta
Najkraća putanja	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj za najkraće putanje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.231 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Objekti kojima se ne može trasirati Opciono Dodato u 3.38	OUTPUT_NON_ROUTABLE	[vektor: tačka] Podrazumevano: [Skip output]	Zadajte izlaz koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže). *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Polje pravca Opciono	DIRECTION_FIELD	[polje tabele: bilo koji]	Polje koje se koristi za zadavanje pravaca ivica mreže. Vrednosti koje se koriste u ovom polju zadaju se sa tri parametra Vrednost za pravac napred, Vrednost za pravac nazad i Vrednost za oba pravca. Pravci napred i nazad odgovaraju jednosmernoj ivici, „oba pravca“ označava dvosmernu ivicu. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevano podešavanje pravca (zadato parametrom Podrazumevani pravac).
Vrednost za pravac napred Opciono	VALUE_FORWARD	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem napred
Vrednost za pravac nazad Opciono	VALUE_BACKWARD	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem nazad
Vrednost za oba pravca Opciono	VALUE_BOTH	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje dvosmernih ivica
Podrazumevani pravac	DEFAULT_DIRECTION	[nabrajanje] Podrazumevano: 2	Ako objekat nema postavljenu vrednost u polju pravca ili ako polje pravca nije postavljeno, koristi se ova vrednost pravca. Jedno od: • 0 — Pravac napred • 1 — Pravac nazad • 2 — Oba pravca
Polje brzine Opciono	SPEED_FIELD	[polje tabele: numerički]	Polje koje daje vrednost brzine (u km/h) za ivice mreže pri traženju najbrže putanje. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevana vrednost brzine (zadata parametrom Podrazumevana brzina).

наставак на следећој страни

Табела 24.232 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Podrazumevana brzina (km/h)	`DEFAULT_SPEED`	[numerički: decimalni] Podrazumevano: 50.0	Vrednost koja se koristi za računanje vremena putovanja ako za ivicu nije zadata vrednost brzine u navedenom polju
Tolerancija topologije	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Dve linije sa čvorovima bližim od zadate tolerancije smatraju se povezanim
Maksimalno rastojanje tačke od mreže Opciono `Dodato u 3.38`	`POINT_TOLERANCE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Zadaje opciono ograničenje rastojanja od početnih i krajnjih tačaka do sloja mreže. Ako je početna tačka udaljenija od mreže od ovog rastojanja, prijaviće se greška. Ako je krajnji objekat udaljeniji od mreže od ovog rastojanja, tretiraće se kao objekat kojem se ne može trasirati. Ako nije postavljeno, tačke će biti zalepljene za najbližu tačku na sloju mreže, bez obzira na to koliko su zapravo udaljene od mreže.

Izlazi

Oznaka	Назив	Tip	Opis
Najkraća putanja	`OUTPUT`	[vektor: linija]	Linijski sloj najkraće ili najbrže putanje od svake početne tačke do krajnje tačke
Objekti kojima se ne može trasirati `Dodato u 3.38`	`OUTPUT_NON_ROUTABLE`	[vektor: tačka]	Opcioni izlazni sloj koji će se koristiti za čuvanje ulaznih objekata kojima nije bilo moguće trasirati putanju (npr. onih koji su previše udaljeni od sloja mreže).

Python kod

ID algoritma: `native:shortestpathpointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Najkraća putanja (tačka do tačke)

Računa optimalnu (najkraću ili najbržu) rutu između zadate početne tačke i zadate krajnje tačke.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski sloj koji predstavlja mrežu	`INPUT`	[vektor: linija]	Linijski vektorski sloj koji predstavlja mrežu koja se pokriva
Tip putanje koja se računa	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Tip putanje koja se računa. Jedno od: • 0 — Najkraća • 1 — Najbrža
Početna tačka (x, y)	`START_POINT`	[koordinate]	Tačkasti objekat koji predstavlja početnu tačku ruta. Pritisnite dugme … pored opcije i kliknite na platno da biste parametar popunili koordinatom tačke na koju ste kliknuli.
Krajnja tačka (x, y)	`END_POINT`	[koordinate]	Tačkasti objekat koji predstavlja krajnju tačku ruta. Pritisnite dugme … pored opcije i kliknite na platno da biste parametar popunili koordinatom tačke na koju ste kliknuli.
Najkraća putanja	`OUTPUT`	[vektor: linija]	Zadajte izlazni linijski sloj za najkraće putanje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Polje pravca Opciono	`DIRECTION_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za zadavanje pravaca ivica mreže. Vrednosti koje se koriste u ovom polju zadaju se sa tri parametra `Vrednost za pravac napred`, `Vrednost za pravac nazad` i `Vrednost za oba pravca`. Pravci napred i nazad odgovaraju jednosmernoj ivici, „oba pravca“ označava dvosmernu ivicu. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevano podešavanje pravca (zadato parametrom `Podrazumevani pravac`).

наставак на следећој страни

Табела 24.235 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Vrednost za pravac napred Opciono	`VALUE_FORWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem napred
Vrednost za pravac nazad Opciono	`VALUE_BACKWARD`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje ivica sa pravcem nazad
Vrednost za oba pravca Opciono	`VALUE_BOTH`	[niska] Podrazumevano: ” (prazna niska)	Vrednost postavljena u polju pravca za prepoznavanje dvosmernih ivica
Podrazumevani pravac	`DEFAULT_DIRECTION`	[nabrajanje] Podrazumevano: 2	Ako objekat nema postavljenu vrednost u polju pravca ili ako polje pravca nije postavljeno, koristi se ova vrednost pravca. Jedno od: • 0 — Pravac napred • 1 — Pravac nazad • 2 — Oba pravca
Polje brzine Opciono	`SPEED_FIELD`	[polje tabele: numerički]	Polje koje daje vrednost brzine (u `km/h`) za ivice mreže pri traženju najbrže putanje. Ako objekat nema vrednost u ovom polju, ili polje nije postavljeno, koristi se podrazumevana vrednost brzine (zadata parametrom `Podrazumevana brzina`).
Podrazumevana brzina (km/h)	`DEFAULT_SPEED`	[numerički: decimalni] Podrazumevano: 50.0	Vrednost koja se koristi za računanje vremena putovanja ako za ivicu nije zadata vrednost brzine u navedenom polju
Tolerancija topologije	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Dve linije sa čvorovima bližim od zadate tolerancije smatraju se povezanim
Maksimalno rastojanje tačke od mreže Opciono `Dodato u 3.38`	`POINT_TOLERANCE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Zadaje opciono ograničenje rastojanja od početnih i krajnjih tačaka do sloja mreže. Ako je bilo koja tačka udaljenija od mreže od ovog rastojanja, prijaviće se greška. Ako nije postavljeno, tačke će biti zalepljene za najbližu tačku na sloju mreže, bez obzira na to koliko su zapravo udaljene od mreže.

Izlazi

Oznaka	Назив	Tip	Opis
Najkraća putanja	`OUTPUT`	[vektor: linija]	Linijski sloj najkraće ili najbrže putanje od svake početne tačke do krajnje tačke

Python kod

ID algoritma: `native:shortestpathpointtopoint`

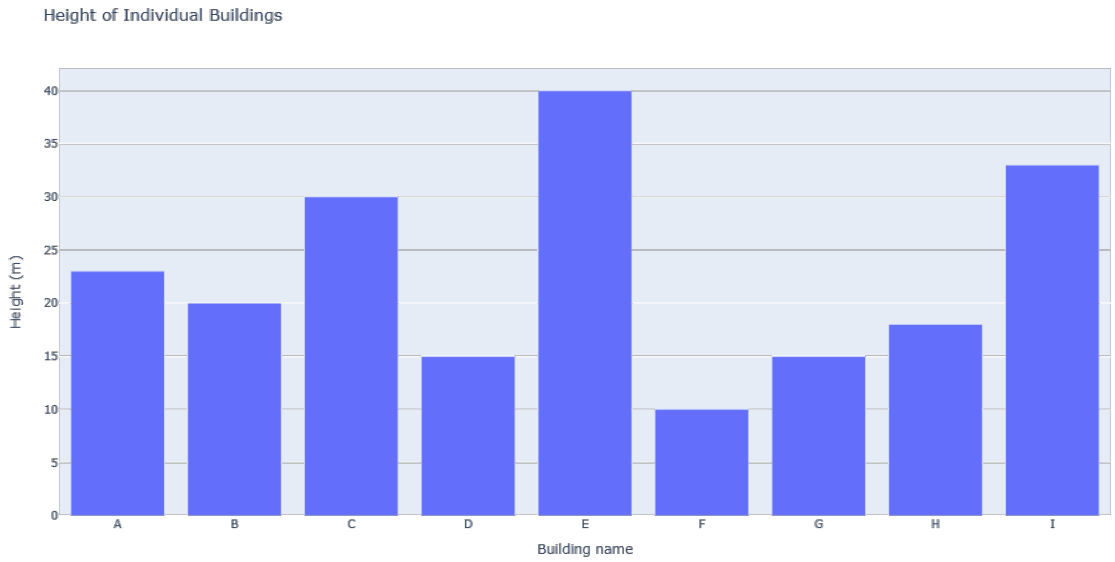
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.14 Grafikoni

Stubičasti grafikon

Pravi stubičasti grafikon od kategorije i polja sloja.



Слика24.32: Primer stubičastog grafikona

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Naziv polja kategorije	`NAME_FIELD`	[polje tabele: bilo koji]	Kategoričko polje koje se koristi za grupisanje stubića (X osa)
Polje vrednosti	`VALUE_FIELD`	[polje tabele: numerički]	Vrednost koja se koristi za grafikon (Y osa).
Naslov Opciono `Dodato u 3.42`	`TITLE`	[niska] Podrazumevano: „”	Naslov grafikona
Naslov X ose Opciono `Dodato u 3.42`	`XAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja kategorije. Uz jedan razmak, naslov ose se sakriva.
Naslov Y ose Opciono `Dodato u 3.42`	`YAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja vrednosti. Uz jedan razmak, naslov ose se sakriva.

наставак на следећој страни

Табела 24.237 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Stubičasti grafikon	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Stubičasti grafikon	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada ► Pregledač rezultata*.

Python kod

ID algoritma: `qgis:barplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kutijasti dijagram

Pravi kutijasti dijagram od polja kategorije i numeričkog polja sloja.



Слика24.33: Primer kutijastog dijagrama

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje naziva kategorije	`NAME_FIELD`	[polje tabele: bilo koji]	Kategoričko polje koje se koristi za grupisanje kutija (X osa)
Polje vrednosti	`VALUE_FIELD`	[polje tabele: numerički]	Vrednost koja se koristi za grafikon (Y osa).
Dodatne statističke linije	`MSD`	[nabrajanje] Podrazumevano: 0	Dodatne statističke informacije koje se dodaju grafikonu. Jedno od: • 0 — Prikaži srednju vrednost • 1 — Prikaži standardnu devijaciju • 2 — Ne prikazuj srednju vrednost i standardnu devijaciju
Naslov Opciono `Dodato u 3.42`	`TITLE`	[niska] Podrazumevano: „”	Naslov grafikona
Naslov X ose Opciono `Dodato u 3.42`	`XAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja kategorije. Uz jedan razmak, naslov ose se ne prikazuje.
Naslov Y ose Opciono `Dodato u 3.42`	`YAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja vrednosti. Uz jedan razmak, naslov ose se ne prikazuje.
Kutijasti dijagram	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Kutijasti dijagram	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada* ► *Pregledač rezultata*.

Python kod

ID algoritma: `qgis:boxplot`

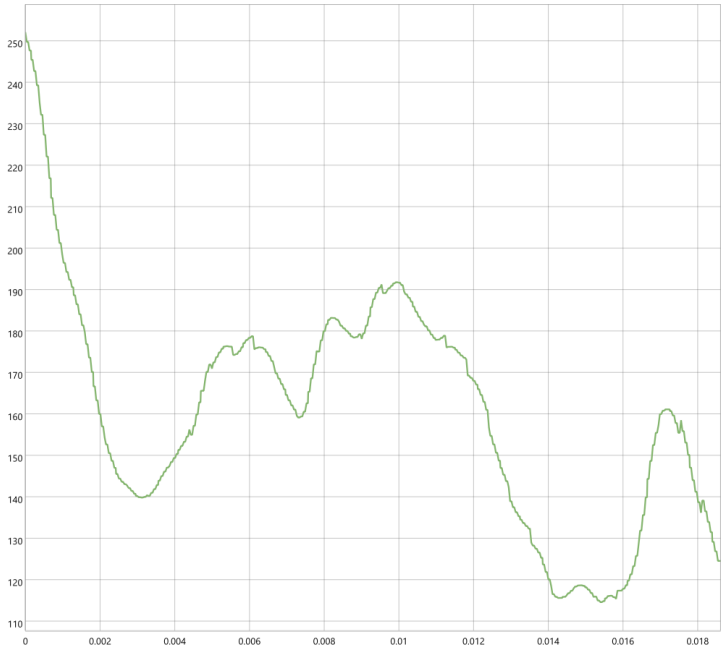
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Generiši sliku visinskog profila

`Dodato u 3.42`

Pravi sliku visinskog profila od liste slojeva karte i opcionog sloja terena.



Слика24.34: Primer visinskog profila

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Kriva profila	`CURVE`	[geometrija: linija]	LineString geometrija (WKT ili GeoJSON) koja predstavlja krivu profila duž koje se generiše visinski profil
Slojevi karte	`MAP_LAYERS`	[sloj] [lista]	Lista slojeva karte koji se uključuju u visinski profil
Širina grafikona (u pikselima)	`WIDTH`	[numerički: ceo broj] Podrazumevano: 400 Minimalna vrednost: 0	Širina izlaznog grafikona u pikselima.
Visina grafikona (u pikselima)	`HEIGHT`	[numerički: ceo broj] Podrazumevano: 300 Minimalna vrednost: 0	Visina izlaznog grafikona u pikselima.
Sloj terena Opciono	`TERRAIN_LAYER`	[raster]	Sloj terena (npr. DEM) koji se koristi za podatke o visini. Ako nije zadat, podaci o visini biće izvedeni iz slojeva karte.

Napredni parametri

Oznaka	Назив	Tip	Opis
Minimalno rastojanje grafikona (X osa) Opciono	`MINIMUM_DISTANCE`	[numerički: decimalni]	Minimalno rastojanje (X osa) koje se prikazuje na grafikonu. Ako nije zadato, grafikon će se automatski skalirati.
Maksimalno rastojanje grafikona (X osa) Opciono	`MAXIMUM_DISTANCE`	[numerički: decimalni]	Maksimalno rastojanje (X osa) koje se prikazuje na grafikonu. Ako nije zadato, grafikon će se automatski skalirati.
Minimalna visina grafikona (Y osa) Opciono	`MINIMUM_ELEVATION`	[numerički: decimalni]	Minimalna visina (Y osa) koja se prikazuje na grafikonu. Ako nije zadato, grafikon će se automatski skalirati.
Maksimalna visina grafikona (Y osa) Opciono	`MAXIMUM_ELEVATION`	[numerički: decimalni]	Maksimalna visina (Y osa) koja se prikazuje na grafikonu. Ako nije zadato, grafikon će se automatski skalirati.
Boja teksta grafikona Opciono	`TEXT_COLOR`	[boja]	Boja teksta na grafikonu (npr. natpisi osa, naslovi).
Boja pozadine grafikona Opciono	`BACKGROUND_COLOR`	[boja]	Boja pozadine grafikona.
Boja ivice grafikona Opciono	`BORDER_COLOR`	[boja]	Boja ivice grafikona.
Tolerancija profila	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 5.0 Minimalna vrednost: 0	Definiše koliko objekat (vektorska tačka, linija, poligon ili oblak tačaka) može biti udaljen od linije profila da bi bio uključen u rezultate. Koristi jedinice karte i ne utiče na druge tipove slojeva.
DPI grafikona	`DPI`	[numerički: ceo broj] Podrazumevano: 96 Minimalna vrednost: 0	Rezolucija izlazne slike u tačkama po inču (DPI).
Izlazna slika	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku slike za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazna slika	`OUTPUT`	[datoteka]	Generisana slika visinskog profila u zadatom formatu.

Python kod

ID algoritma: `native:generateelevationprofileimage`

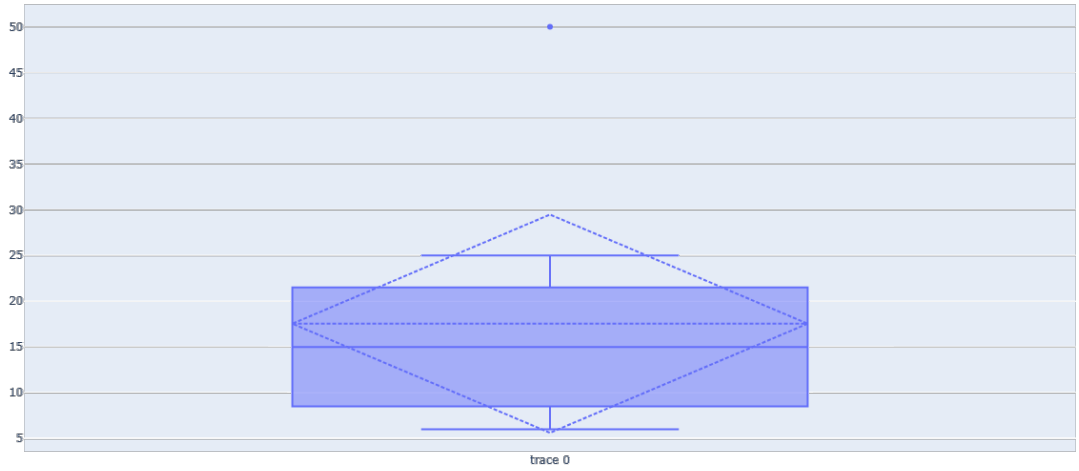
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Grafikon srednje vrednosti i standardne devijacije

Pravi kutijasti dijagram sa vrednostima srednje vrednosti i standardne devijacije.



Слика24.35: Primer grafikona srednje vrednosti i standardne devijacije

Parametri

Oznaka	Назив	Tip	Opis
Ulazna tabela	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje naziva kategorije	`NAME_FIELD`	[polje tabele: bilo koji]	Kategoričko polje koje se koristi za grupisanje kutija (X osa)
Polje vrednosti	`VALUE_FIELD`	[polje tabele: numerički]	Vrednost koja se koristi za grafikon (Y osa).
Grafikon	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Grafikon	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada* ► *Pregledač rezultata*.

Python kod

ID algoritma: `qgis:meanandstandarddeviationplot`

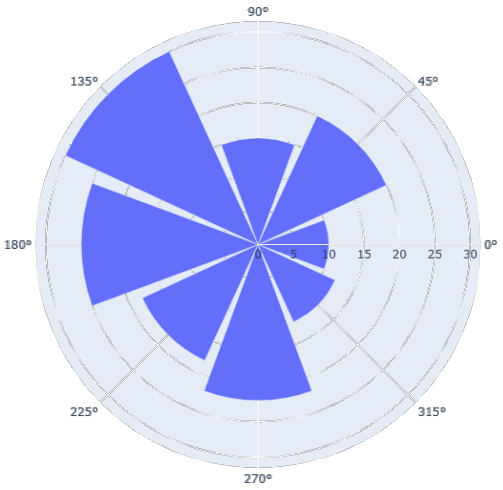
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Polarni grafikon

Generiše polarni grafikon na osnovu vrednosti ulaznog vektorskog sloja.

Kao parametri se moraju uneti dva polja: jedno koje definiše kategoriju svakog objekta (za grupisanje objekata) i drugo sa promenljivom koja se prikazuje (ova mora biti numerička).



Слика24.36: Primer polarnog grafikona

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje naziva kategorije	`NAME_FIELD`	[polje tabele: bilo koji]	Kategoričko polje koje se koristi za grupisanje objekata (X osa)
Polje vrednosti	`VALUE_FIELD`	[polje tabele: numerički]	Vrednost koja se koristi za grafikon (Y osa).
Polarni grafikon	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Polarni grafikon	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada ► Pregledač rezultata*.

Python kod

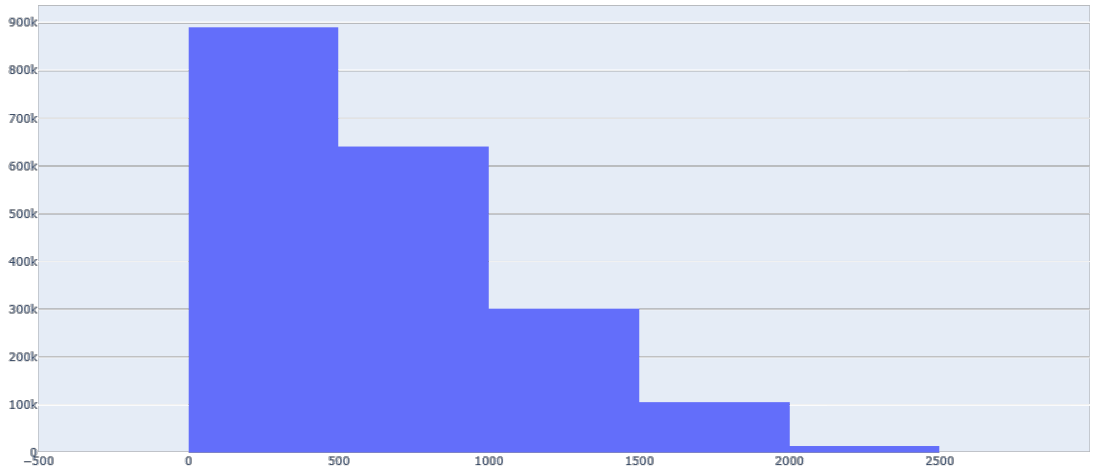
ID algoritma: `qgis:polarplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Histogram rasterskog sloja

Generiše histogram sa vrednostima rasterskog sloja.



Слика24.37: Primer histograma rasterskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal]	Rasterski kanal koji se koristi za histogram
broj pregrada	`BINS`	[numerički: ceo broj] Podrazumevano: 10	Broj pregrada koji se koristi u histogramu (X osa). Minimum 2.

наставак на следећој страни

Табела 24.248 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Histogram	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Histogram	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada ► Pregledač rezultata*.

Python kod

ID algoritma: `qgis:rasterlayerhistogram`

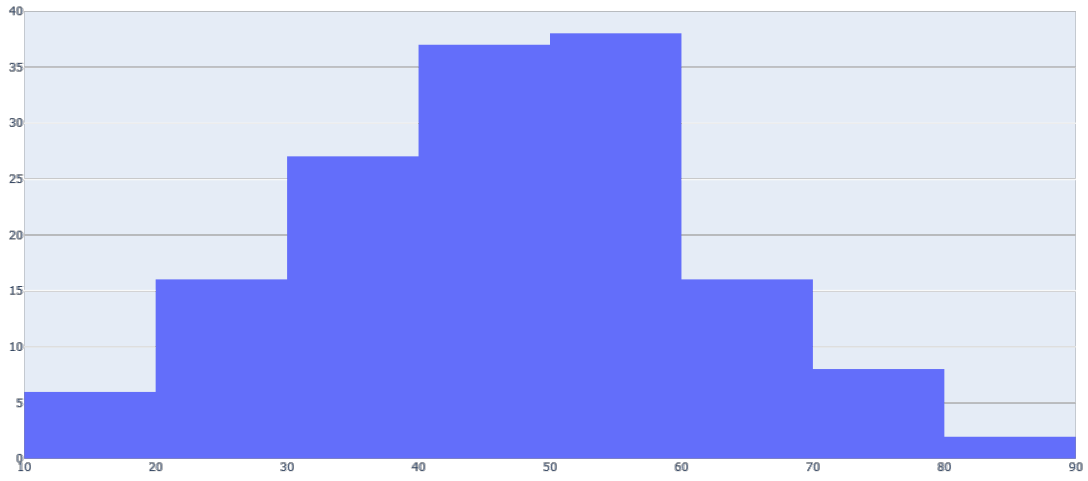
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Histogram vektorskog sloja

Generiše histogram sa vrednostima atributa vektorskog sloja.

Atribut koji se koristi za računanje histograma mora biti numerički.



Слика24.38: Primer histograma vektorskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Atribut	`FIELD`	[polje tabele: numerički]	Vrednost koja se koristi za grafikon (Y osa).
broj pregrada	`BINS`	[numerički: ceo broj] Podrazumevano: 10	Broj pregrada koji se koristi u histogramu (X osa). Minimum 2.
Histogram	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Histogram	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada* ► *Pregledač rezultata*.

Python kod

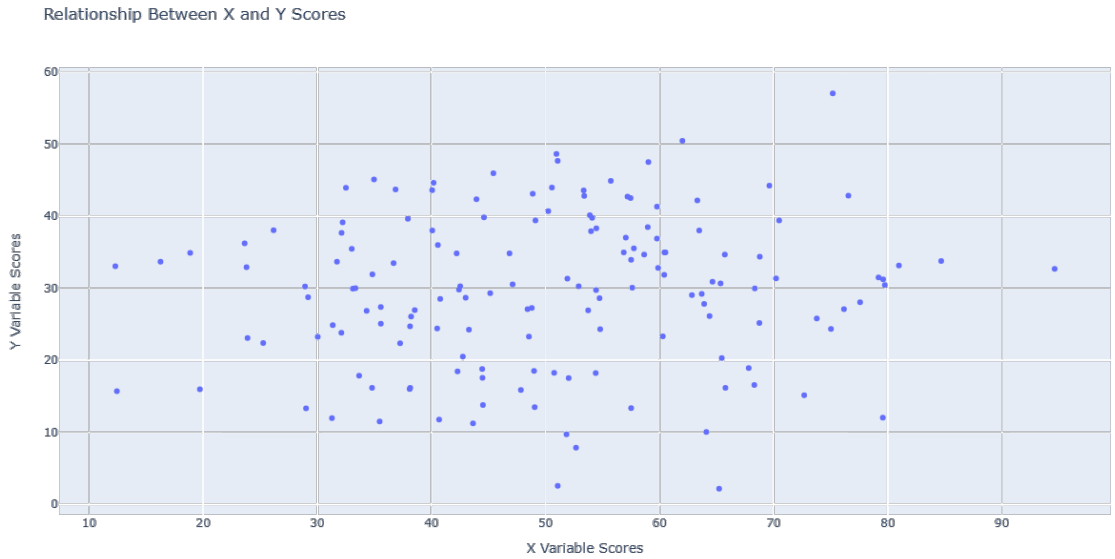
ID algoritma: `qgis:vectorlayerhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Dijagram rasturanja vektorskog sloja

Pravi jednostavan `X` - `Y` dijagram rasturanja za vektorski sloj.



Слика24.39: Primer dijagrama rasturanja vektorskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
X atribut	`XFIELD`	[polje tabele: numerički]	Polje koje se koristi za X osu
Y atribut	`YFIELD`	[polje tabele: numerički]	Polje koje se koristi za Y osu
Tekst na prelazu miša Opciono `Dodato u 3.42`	`HOVERTEXT`	[izraz] Podrazumevano: „”	Tekst koji se prikazuje pri prelasku mišem preko tačaka podataka. Može se izabrati iz polja ili zasnovati na izrazu.
Naslov Opciono `Dodato u 3.42`	`TITLE`	[niska] Podrazumevano: „”	Naslov grafikona
Naslov X ose Opciono `Dodato u 3.42`	`XAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja X atributa. Uz jedan razmak, naslov ose se ne prikazuje.
Naslov Y ose Opciono `Dodato u 3.42`	`YAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja Y atributa. Uz jedan razmak, naslov ose se ne prikazuje.
Koristi logaritamsku razmeru za X osu `Dodato u 3.42`	`XAXIS_LOG`	[logička vrednost] Podrazumevano: False	Kada je omogućeno, koristi logaritamsku razmeru za X osu
Koristi logaritamsku razmeru za Y osu `Dodato u 3.42`	`YAXIS_LOG`	[logička vrednost] Podrazumevano: False	Kada je omogućeno, koristi logaritamsku razmeru za Y osu

наставак на следећој страни

Табела 24.252 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dijagram rasturanja	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Dijagram rasturanja	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada ► Pregledač rezultata*.

Python kod

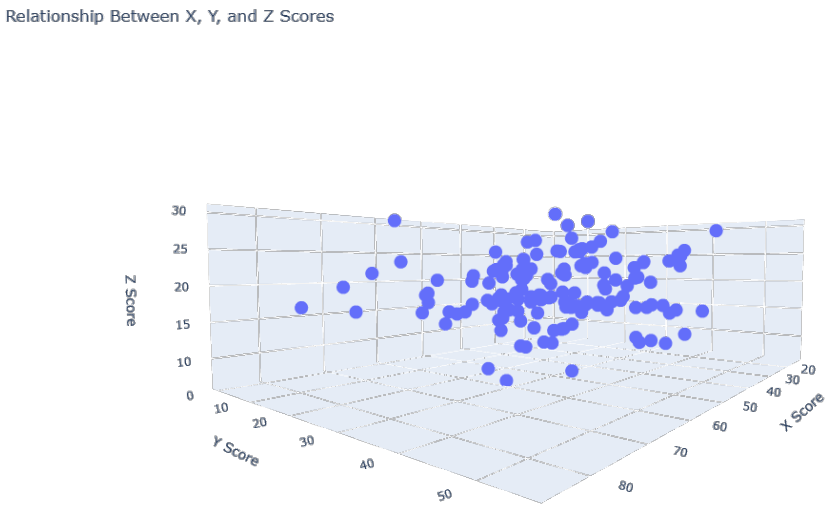
ID algoritma: `qgis:vectorlayerscatterplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

3D dijagram rasturanja vektorskog sloja

Pravi 3D dijagram rasturanja za vektorski sloj.



Слика24.40: Primer 3D dijagrama rasturanja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
X atribut	`XFIELD`	[polje tabele: numerički]	Polje koje se koristi za X osu
Y atribut	`YFIELD`	[polje tabele: numerički]	Polje koje se koristi za Y osu
Z atribut	`ZFIELD`	[polje tabele: numerički]	Polje koje se koristi za Z osu
Naslov Opciono `Dodato u 3.42`	`TITLE`	[niska] Podrazumevano: „”	Naslov grafikona
Naslov X ose Opciono `Dodato u 3.42`	`XAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja X atributa.
Naslov Y ose Opciono `Dodato u 3.42`	`YAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja Y atributa.
Naslov Z ose Opciono `Dodato u 3.42`	`ZAXIS_TITLE`	[niska] Podrazumevano: „”	Ako je prazno, koristi se naziv polja Z atributa.
3D dijagram rasturanja	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za grafikon. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
3D dijagram rasturanja	`OUTPUT`	[html]	HTML datoteka sa grafikonom. Dostupna u *Obrada* ► *Pregledač rezultata*.

Python kod

ID algoritma: `qgis:scatter3dplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.15 Pretvaranje oblaka tačaka

Пажња

Pokretanje ovih algoritama zahteva QGIS instaliran sa PDAL >= 2.5.0 (pogledajte meni *Pomoć* ► *O programu*).

Pretvaranje formata

`Dodato u 3.32`

Pretvara oblak tačaka u drugi format datoteke, npr. pravi kompresovanu `.LAZ`.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se pretvara
Pretvoreno	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka koja se koristi kao izlaz. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka u izmenjenom formatu datoteke. Trenutno podržani formati su `.LAS`, `.LAZ`, `.COPC.LAZ` i `.VPC`.

Python kod

ID algoritma: `pdal:convertformat`

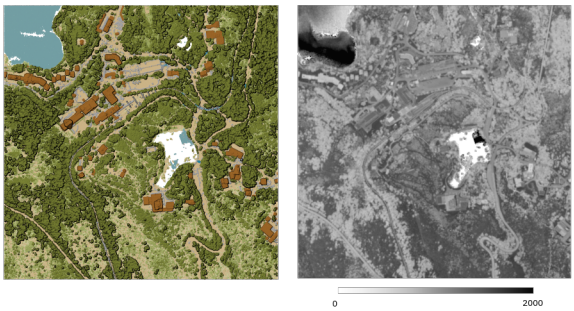
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u raster

`Dodato u 3.32`

Izvozi podatke oblaka tačaka u 2D rastersku mrežu sa ćelijom zadate rezolucije, upisujući vrednosti iz navedenog atributa.



Слика24.41: Rasterski izlaz sa korišćenjem atributa Intensity tačaka

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se izvozi
Atribut	`ATTRIBUTE`	[polje] [nabrajanje]	Polje sloja oblaka tačaka iz kojeg se preuzimaju vrednosti
Rezolucija rastera gustine	`RESOLUTION`	[numerički: decimalni] Podrazumevano: 1.0	Veličina ćelije izlaznog rastera
Veličina pločice za paralelna izvršavanja	`TILE_SIZE`	[numerički: ceo broj] Podrazumevano: 1000	
Izvezeno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte rastersku datoteku u koju se izvoze podaci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka

наставак на следећој страни

Табела 24.259 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
X koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_X`	[numerički: decimalni]	
Y koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_Y`	[numerički: decimalni]	

Izlazi

Oznaka	Назив	Tip	Opis
Izvezeno	`OUTPUT`	[raster]	Izlazni rasterski sloj u koji se izvoze objekti sloja oblaka tačaka. Trenutno podržan format je `.TIF`.

Python kod

ID algoritma: `pdal:exportraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

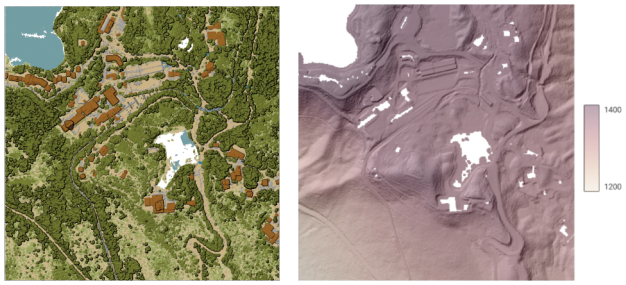
Izvoz u raster (pomoću triangulacije)

`Dodato u 3.32`

Izvozi podatke oblaka tačaka u 2D rastersku mrežu pomoću triangulacije tačaka, a zatim interpolacijom vrednosti ćelija iz trouglova.

Белешка

Korišćenje ovog algoritma može biti sporije ako radite sa velikim skupom podataka. Ako je vaš oblak tačaka gust, tačke tla možete izvesti kao raster pomoću algoritma *Izvoz u raster*.



Слика24.42: Rasterski izlaz terena dobijen triangulacijom oblaka tačaka

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se izvozi
Rezolucija rastera gustine	`RESOLUTION`	[numerički: decimalni] Podrazumevano: 1.0	Veličina ćelije izlaznog rastera
Veličina pločice za paralelna izvršavanja	`TILE_SIZE`	[numerički: ceo broj] Podrazumevano: 1000	
Izvezeno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte rastersku datoteku u koju se izvoze podaci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka

наставак на следећој страни

Табела 24.262 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
X koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_X`	[numerički: decimalni]	
Y koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_Y`	[numerički: decimalni]	

Izlazi

Oznaka	Назив	Tip	Opis
Izvezeno (pomoću triangulacije)	`OUTPUT`	[raster]	Izlazni rasterski sloj u koji se izvoze objekti sloja oblaka tačaka. Trenutno podržan format je `.TIF`.

Python kod

ID algoritma: `pdal:exportrastertin`

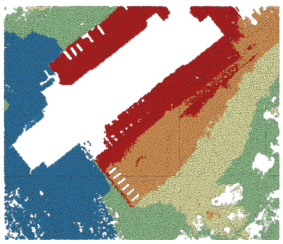
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u vektor

`Dodato u 3.32`

Izvozi podatke oblaka tačaka u vektorski sloj sa 3D tačkama (GeoPackage), opciono sa dodatnim atributima.



Слика24.43: Izvoz oblaka tačaka (tačke tla) u vektorski sloj stilizovan na osnovu nadmorske visine

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se izvozi
Atribut Opciono	`ATTRIBUTE`	[polje] [lista]	Jedno ili više polja sloja oblaka tačaka koja se izvoze zajedno sa tačkama.
Izvezeno	`OUTPUT`	[vektor] Podrazumevano: `[Save to temporary file]`	Zadajte vektorsku datoteku u koju se podaci izvoze. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Izvezeno	`OUTPUT`	[vektor]	Izlazni vektorski sloj u koji se izvoze objekti sloja oblaka tačaka. Trenutno podržan format je `.GPKG`.

Python kod

ID algoritma: `pdal:exportvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.16 Upravljanje podacima oblaka tačaka

Пажња

Pokretanje ovih algoritama zahteva QGIS instaliran sa PDAL >= 2.5.0 (pogledajte meni *Pomoć* ► *O programu*).

Dodela projekcije

`Dodato u 3.32`

Dodeljuje koordinatni referentni sistem sloju oblaka tačaka, ako nedostaje ili je pogrešan. Pravi se novi sloj.

See also

Reprojektovanje

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka kojem se dodeljuje CRS
Željeni CRS	`CRS`	[crs]	CRS koji se primenjuje na sloj
Izlazni sloj	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka koja se koristi kao izlaz. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka sa novim CRS-om. Trenutno podržani formati su `.LAS`, `.LAZ`, `.COPC.LAZ` i `.VPC`.

Python kod

ID algoritma: `pdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

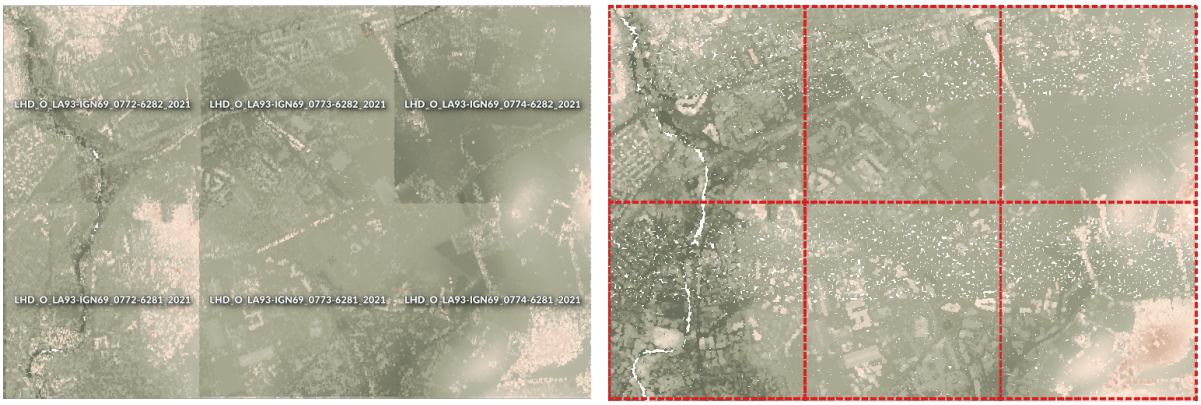
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi virtuelni oblak tačaka (VPC)

`Dodato u 3.32`

Pravi *virtuelni oblak tačaka (VPC)* od ulaznih podataka oblaka tačaka.

Ako opcione parametre ostavite neuključene, VPC datoteka će biti napravljena vrlo brzo jer će algoritam čitati samo metapodatke ulaznih datoteka. Ako je bilo koji od opcionih parametara postavljen, algoritam će pročitati sve tačke, što može potrajati.



Слика24.44: Pravljenje virtuelnog oblaka tačaka sa pregledom iz skupa pločica oblaka tačaka

See also

Spajanje

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[oblak tačaka] [lista]	Ulazni slojevi oblaka tačaka koji se kombinuju unutar sloja virtuelnog oblaka tačaka
Izračunaj granični poligon	`BOUNDARY`	[logička vrednost] Podrazumevano: False	Postavite na True da biste prikazali tačne granice podataka (umesto samo pravougaonog obuhvata)
Izračunaj statistiku	`STATISTICS`	[logička vrednost] Podrazumevano: False	Postavite na True da biste razumeli opsege vrednosti različitih atributa

наставак на следећој страни

Табела 24.269 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Napravi pregledni oblak tačaka	`OVERVIEW`	[logička vrednost] Podrazumevano: False	Generiše jedan „prorećen“ oblak tačaka svih ulaznih podataka (koristeći samo svaku 1000. tačku iz originalnih podataka). Pregledni oblak tačaka biće napravljen pored VPC datoteke - na primer, za `mydata.vpc`, pregledni oblak tačaka bi se zvao `mydata-overview.copc.laz`.
Virtuelni oblak tačaka	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka u koju se podaci ugrađuju. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Virtuelni oblak tačaka	`OUTPUT`	[raster]	Izlazni sloj oblaka tačaka koji kombinuje sve ulazne podatke, kao virtuelna datoteka.

Python kod

ID algoritma: `pdal:virtualpointcloud`

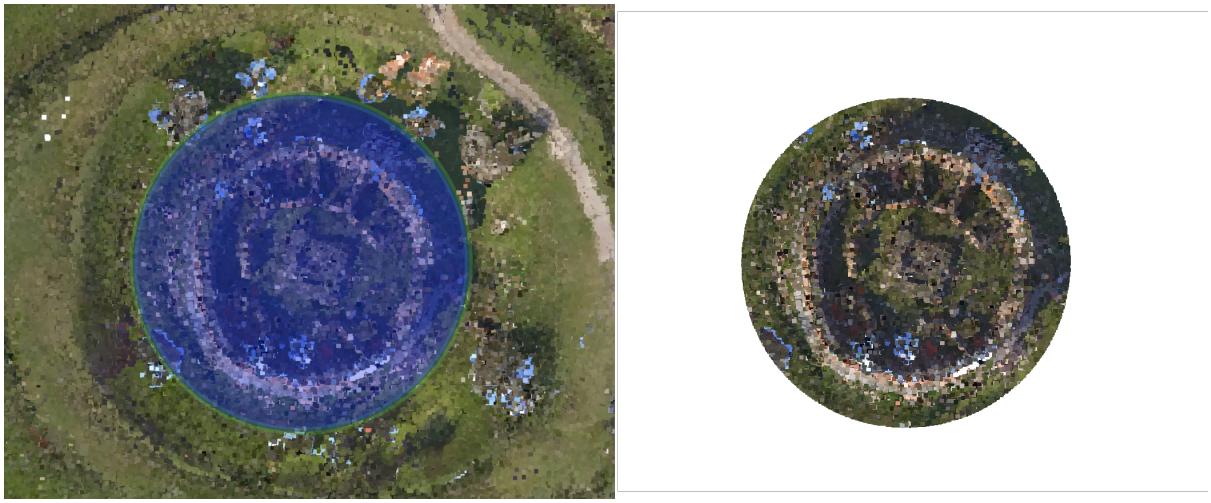
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Isecanje

`Dodato u 3.32`

Iseca sloj oblaka tačaka poligonskim slojem tako da rezultujući oblak tačaka sadrži samo tačke unutar poligona.



Слика24.45: Isecanje ulaznog sloja oblaka tačaka poligonskim pokrivačem

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se iseca
Poligoni za isecanje	`OVERLAY`	[vektor: poligon]	Poligonski vektorski sloj koji se koristi kao pokrivač za isecanje tačaka
Isečeno	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka u koju se izvoze isečene tačke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno	`OUTPUT`	[oblak tačaka]	Izlazni oblak tačaka čiji su objekti tačke unutar poligonskog sloja pokrivača.

Python kod

ID algoritma: `pdal:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi COPC

`Dodato u 3.32`

Pravi indeks za sve ulazne datoteke oblaka tačaka u paketnom režimu.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[oblak tačaka] [lista]	Ulazni slojevi oblaka tačaka za koje se pravi indeks
Izlazni direktorijum Opciono	`OUTPUT`	[folder] Podrazumevano: `[Skip output]`	Zadajte folder u kojem se prave nove datoteke. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT`	[folder]	Izlazni folder koji sadrži slojeve oblaka tačaka sa pratećim COPC indeksnim datotekama.

Python kod

ID algoritma: `pdal:createcopc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Informacije

`Dodato u 3.32`

Ispisuje osnovne metapodatke iz ulazne datoteke oblaka tačaka.

Primer izlaznih informacija:

```
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
```

(наставак на следећој страни)

(настављено са претходне стране)

```
- 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
```

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka iz kojeg se izdvajaju informacije o metapodacima
Informacije o sloju	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku za čuvanje informacija o metapodacima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Informacije o sloju	`OUTPUT`	[html]	`HTML` datoteka za čuvanje informacija o metapodacima.

Python kod

ID algoritma: `pdal:info`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje

`Dodato u 3.32`

Spaja više datoteka oblaka tačaka u jednu.

See also

Napravi virtuelni oblak tačaka (VPC)

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[oblak tačaka] [lista]	Ulazni slojevi oblaka tačaka koji se spajaju u jedan
Spojeno	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni oblak tačaka koji spaja ulazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Spojeno	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka koji spaja sve ulazne datoteke.

Python kod

ID algoritma: `pdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Reprojektovanje

`Dodato u 3.32`

Reprojektuje oblak tačaka u drugi koordinatni referentni sistem (CRS).

See also

Dodela projekcije

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se reprojektuje u drugi CRS
Ciljni CRS	`CRS`	[crs]	CRS koji se primenjuje na sloj
Reprojektovano	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte reprojektovanu datoteku oblaka tačaka. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Koordinatna operacija Opciono	`OPERATION`	[datum]	*Transformacija datuma* koja se koristi za reprojektovanje podataka između izvornog i ciljnog sistema.

Izlazi

Oznaka	Назив	Tip	Opis
REPROJEKTOVANO	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka u ciljnom CRS-u.

Python kod

ID algoritma: `pdal:reproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Proređivanje (po poluprečniku uzorkovanja)

`Dodato u 3.32`

Pravi prorećenu verziju oblaka tačaka uzorkovanjem po rastojanju tačaka (smanjuje broj tačaka unutar određenog poluprečnika).

See also

Proređivanje (preskakanjem tačaka)

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka od kojeg se pravi prorećena verzija
Poluprečnik uzorkovanja (u jedinicama karte)	`SAMPLING_RADIUS`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje unutar kojeg se tačke uzorkuju u jednu tačku
Prorećeno (po poluprečniku)	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni oblak tačaka sa smanjenim brojem tačaka. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Prorećeno (po poluprečniku)	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka sa smanjenim brojem tačaka.

Python kod

ID algoritma: `pdal:thinbyradius`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Proređivanje (preskakanjem tačaka)

`Dodato u 3.32`

Pravi prorećenu verziju oblaka tačaka zadržavajući samo svaku N-tu tačku (smanjuje broj tačaka preskakanjem bliskih tačaka).

See also

Proređivanje (po poluprečniku uzorkovanja)

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka od kojeg se pravi prorećena verzija
Broj tačaka koje se preskaču	`POINTS_NUMBER`	[numerički: ceo broj] Podrazumevano: 1	Zadrži samo svaku N-tu tačku u ulaznom sloju
Prorećeno (decimacijom)	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni oblak tačaka sa smanjenim brojem tačaka. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka

наставак на следећој страни

Табела 24.288 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Proređeno (decimacijom)	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka sa smanjenim brojem tačaka.

Python kod

ID algoritma: `pdal:thinbydecimate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pločice

Pravi pločice od ulaznih datoteka oblaka tačaka, preporučeno za najbolje performanse (pri prikazu ili analizi) sa takvim skupovima podataka u QGIS-u.

See also

Napravi virtuelni oblak tačaka (VPC), *Napravi COPC*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[oblak tačaka] [lista]	Ulazni slojevi oblaka tačaka od kojih se prave pločice
Dužina pločice	`LENGTH`	[numerički: decimalni] Podrazumevano: 1000.0	Veličina ivice svake napravljene pločice
Izlazni direktorijum	`OUTPUT`	[folder] Podrazumevano: `[Save to temporary folder]`	Zadajte folder u kojem se čuvaju napravljene pločice. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodeli CRS Opciono	`CRS`	[crs]	CRS koji se primenjuje na sloj

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT`	[folder]	Izlazni folder koji sadrži pločice napravljene od ulaznih datoteka.

Python kod

ID algoritma: `pdal:tile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.17 Izdvajanje iz oblaka tačaka

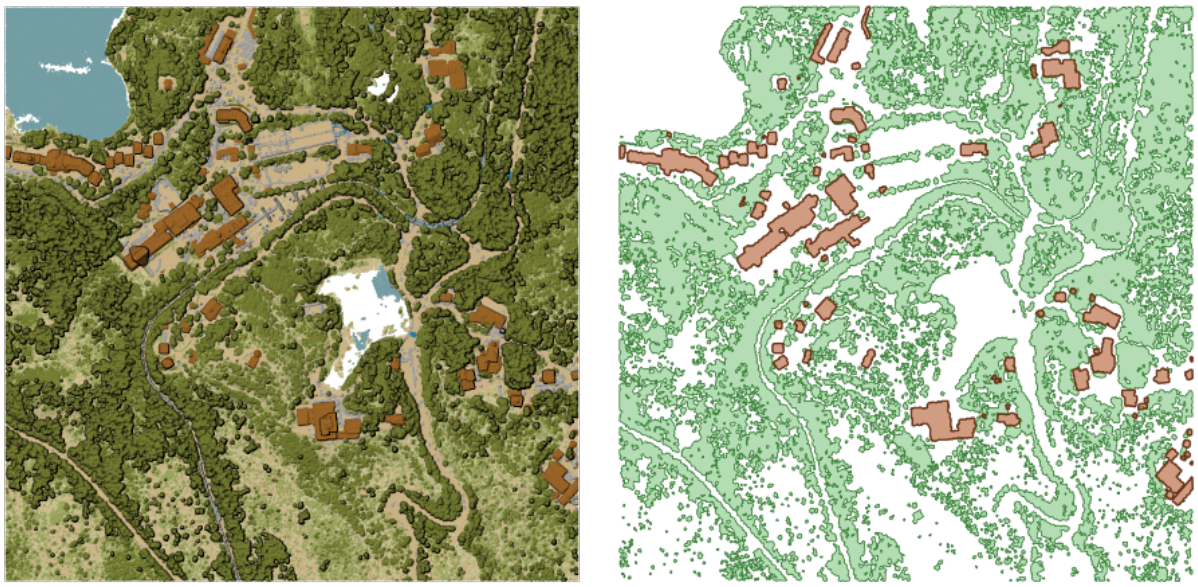
Пажња

Pokretanje ovih algoritama zahteva QGIS instaliran sa PDAL >= 2.5.0 (pogledajte meni *Pomoć* ► *O programu*).

Granica

`Dodato u 3.32`

Izvozi poligonsku datoteku koja sadrži granicu sloja oblaka tačaka. Može sadržati rupe i može biti višedelni poligon.



Слика24.46: Izdvajanje poligona visoke vegetacije i zgrada iz ulaznog sloja oblaka tačaka

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka za koji se izračunava granica
Rezolucija ćelija koje se koriste za izračunavanje granice Opciono	`RESOLUTION`	[numerički: decimalni]	Rezolucija ćelija koje se koriste za izračunavanje granice
Najmanji broj tačaka u ćeliji da bi se ćelija smatrala zauzetom Opciono	`THRESHOLD`	[numerički: ceo broj]	Najmanji broj tačaka u ćeliji da bi se ćelija smatrala zauzetom
Granica	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka koja se koristi kao izlaz. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Granica	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj sa granicom oblaka tačaka. Trenutno podržan format je `.GPKG`.

Python kod

ID algoritma: `pdal:boundary`

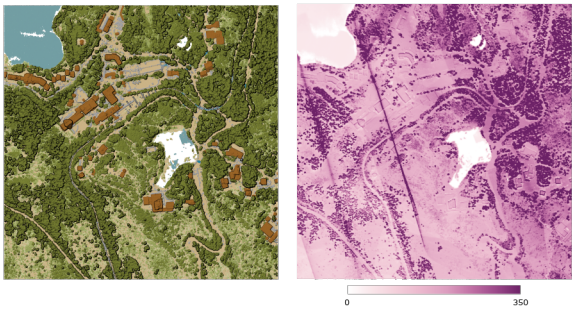
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Gustina

`Dodato u 3.32`

Izvozi rastersku datoteku na osnovu broja tačaka unutar svake rasterske ćelije - korisno za proveru kvaliteta skupova podataka oblaka tačaka.



Слика24.47: Gustina tačaka (broj tačaka na 2x2 m) kao raster

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se izvozi
Rezolucija rastera gustine	`RESOLUTION`	[numerički: decimalni] Podrazumevano: 1.0	Veličina ćelije izlaznog rastera
Veličina pločice za paralelna izvršavanja	`TILE_SIZE`	[numerički: ceo broj] Podrazumevano: 1000	Veličina pločica na koje se podaci dele radi paralelnih izvršavanja
Gustina	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte rastersku datoteku u koju se izvoze podaci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka

наставак на следећој страни

Табела 24.297 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
X koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_X`	[numerički: decimalni]	
Y koordinata početka pločice za paralelna izvršavanja Opciono	`ORIGIN_Y`	[numerički: decimalni]	

Izlazi

Oznaka	Назив	Tip	Opis
Gustina	`OUTPUT`	[raster]	Izlazni rasterski sloj sa brojem tačaka unutar svake ćelije. Trenutno podržan format je `.TIF`.

Python kod

ID algoritma: `pdal:density`

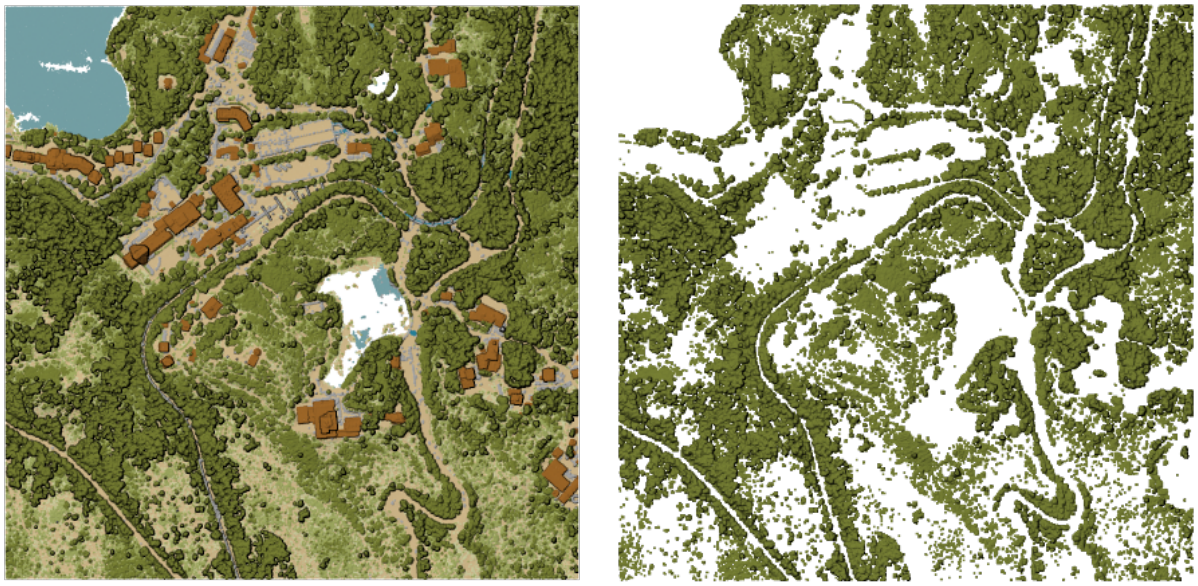
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filter

`Dodato u 3.32`

Izdvaja tačke iz ulaznog oblaka tačaka koje odgovaraju PDAL izrazu i/ili se nalaze unutar pravougaonika za isecanje.



Слика24.48: Filtriranje klase visoke vegetacije iz ulaznog sloja oblaka tačaka

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[oblak tačaka]	Ulazni sloj oblaka tačaka koji se izvozi
Izraz za filtriranje Opciono	`FILTER_EXPRESSION`	[izraz]	*PDAL izraz* za izbor podskupa objekata u podacima oblaka tačaka
Obuhvat isecanja Opciono	`FILTER_EXTENT`	[obuhvat]	Obuhvat karte za izbor podskupa objekata u podacima oblaka tačaka Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Filtrirano	`OUTPUT`	[oblak tačaka] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku oblaka tačaka u koju se izvoze podaci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Filtrirano	`OUTPUT`	[oblak tačaka]	Izlazni sloj oblaka tačaka sa filtriranim objektima ulaznog sloja oblaka tačaka. Trenutno podržani formati su `.LAS`, `.LAZ`, `.COPC.LAZ` i `.VPC`.

Python kod

ID algoritma: `pdal:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.18 Analiza rastera

Procentualni rang steka ćelija od vrednosti

Računa procentualni rang vrednosti po ćeliji za stek rastera na osnovu jedne ulazne vrednosti i upisuje ih u izlazni raster.

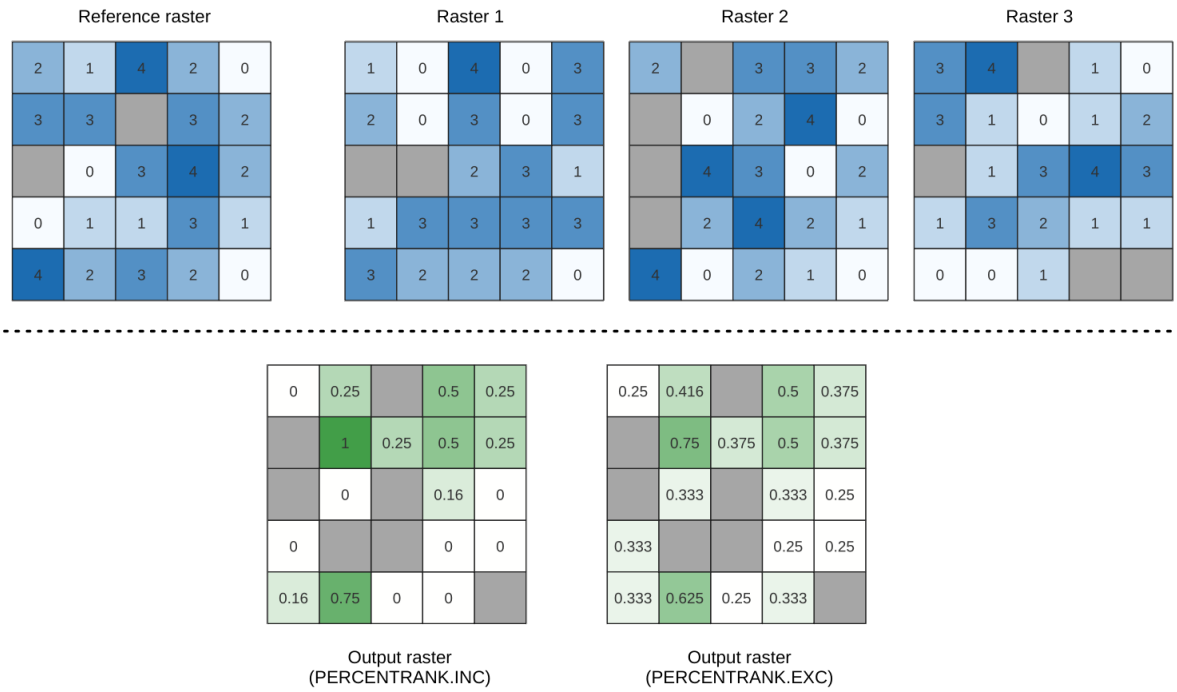
Na svakoj lokaciji ćelije, zadata vrednost se rangira među odgovarajućim vrednostima u steku svih preklopljenih i sortiranih vrednosti ćelija iz ulaznih rastera. Za vrednosti izvan raspodele vrednosti steka, algoritam vraća NoData jer se vrednost ne može rangirati među vrednostima ćelija.

Postoje dva metoda za računanje percentila:

- Uključiva linearna interpolacija (PERCENTRANK.INC)
- Isključiva linearna interpolacija (PERCENTRANK.EXC)

Metodi linearne interpolacije vraćaju jedinstveni procentualni rang za različite vrednosti. Oba metoda interpolacije prate svoje odgovarajuće metode koje implementiraju LibreOffice ili Microsoft Excel.

Obuhvat i rezolucija izlaznog rastera definisani su referentnim rasterom. Ulazni rasterski slojevi koji ne odgovaraju veličini ćelije referentnog rasterskog sloja biće preuzorkovani pomoću preuzorkovanja najbližeg suseda. NoData vrednosti u bilo kom od ulaznih slojeva daće NoData ćeliju u izlazu ako parametar „Zanemari NoData vrednosti“ nije postavljen. Tip podataka izlaznog rastera uvek će biti `Float32`.



Слика24.49: Procentualno rangiranje vrednosti = 1. `NoData` ćelije (sive) se zanemaruju.

See also

Percentil steka ćelija, *Procentualni rang steka ćelija iz rasterskog sloja*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod za računanje percentila: • 0 — Uključiva linearna interpolacija (PERCENTRANK.INC) • 1 — Isključiva linearna interpolacija (PERCENTRANK.EXC)
Vrednost	`VALUE`	[numerički: decimalni] Podrazumevano: 10.0	Vrednost koja se rangira među odgovarajućim vrednostima u steku svih preklopljenih i sortiranih vrednosti ćelija iz ulaznih rastera
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: True	Ako nije označeno, sve NoData ćelije u ulaznim slojevima daće NoData ćeliju u izlaznom rasteru
Referentni sloj	`REFERENCE_LAYER`	[raster]	Referentni sloj za pravljenje izlaznog sloja (obuhvat, CRS, dimenzije piksela)

наставак на следећој страни

Табела 24.301 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`OUTPUT_NODATA_VALUE`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:cellstackpercentrankfromvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Percentil steka ćelija

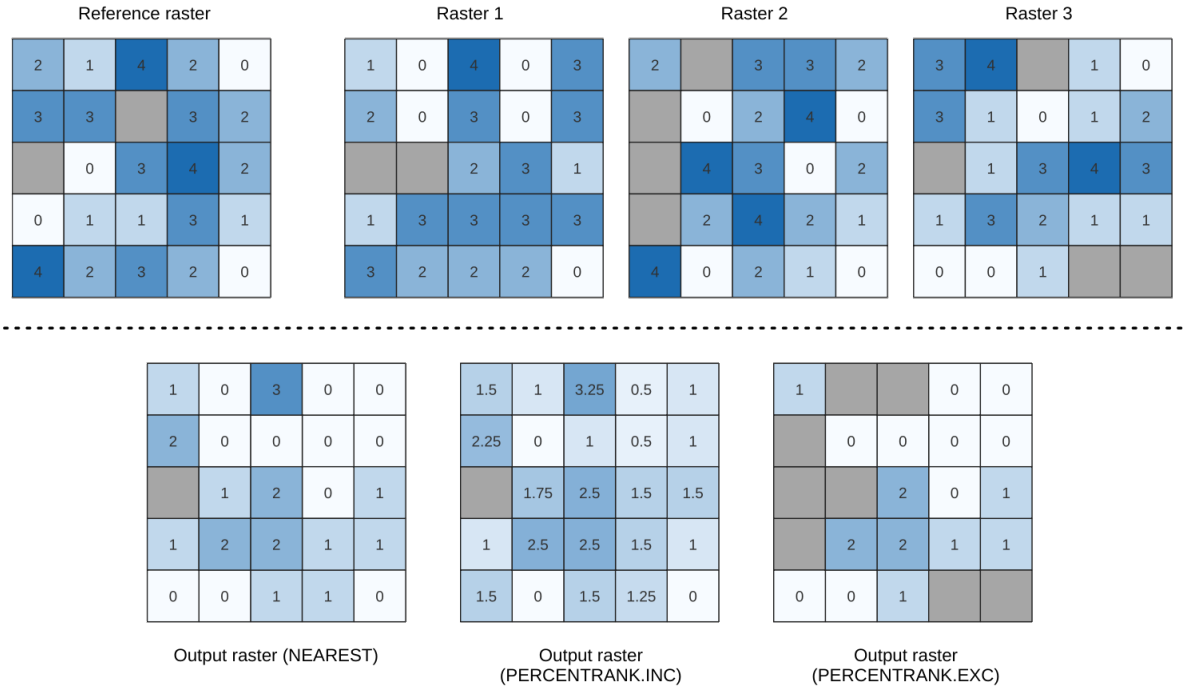
Računa vrednost percentila po ćeliji za stek rastera i upisuje rezultate u izlazni raster. Percentil koji se vraća određuje se ulaznom vrednošću percentila (opseg između 0 i 1). Na svakoj lokaciji ćelije, zadati percentil se dobija pomoću odgovarajuće vrednosti iz steka svih preklopljenih i sortiranih vrednosti ćelija ulaznih rastera.

Postoje tri metoda za računanje percentila:

- Najbliži rang: vraća vrednost koja je najbliža zadatom percentilu
- Uključiva linearna interpolacija (PERCENTRANK.INC)
- Isključiva linearna interpolacija (PERCENTRANK.EXC)

Metodi linearne interpolacije vraćaju jedinstvene vrednosti za različite percentile. Oba metoda interpolacije prate svoje odgovarajuće metode koje implementiraju LibreOffice ili Microsoft Excel.

Obuhvat i rezolucija izlaznog rastera definisani su referentnim rasterom. Ulazni rasterski slojevi koji ne odgovaraju veličini ćelije referentnog rasterskog sloja biće preuzorkovani pomoću preuzorkovanja najbližeg suseda. NoData vrednosti u bilo kom od ulaznih slojeva daće NoData ćeliju u izlazu ako parametar „Zanemari NoData vrednosti“ nije postavljen. Tip podataka izlaznog rastera uvek će biti `Float32`.



Слика24.50: Percentil = 0.25. `NoData` ćelije (sive) se zanemaruju.

See also

Percentil steka ćelija, *Procentualni rang steka ćelija iz rasterskog sloja*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod za računanje percentila: • 0 — Najbliži rang: vraća vrednost koja je najbliža zadatom percentilu • 1 — Uključiva linearna interpolacija (PERCENTILE.INC) • 2 — Isključiva linearna interpolacija (PERCENTILE.EXC)
Percentil	`VALUE`	[numerički: decimalni] Podrazumevano: 0.25	Vrednost koja se rangira među odgovarajućim vrednostima u steku svih preklopljenih i sortiranih vrednosti ćelija iz ulaznih rastera. Između 0 i 1.
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: True	Ako nije označeno, sve NoData ćelije u ulaznim slojevima daće NoData ćeliju u izlaznom rasteru
Referentni sloj	`REFERENCE_LAYER`	[raster]	Referentni sloj za pravljenje izlaznog sloja (obuhvat, CRS, dimenzije piksela)
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`OUTPUT_NODATA_VALUE`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

наставак на следећој страни

Табела 24.306 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:cellstackpercentile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Procentualni rang steka ćelija iz rasterskog sloja

Računa procentualni rang vrednosti po ćeliji za stek rastera na osnovu ulaznog rastera vrednosti i upisuje ih u izlazni raster.

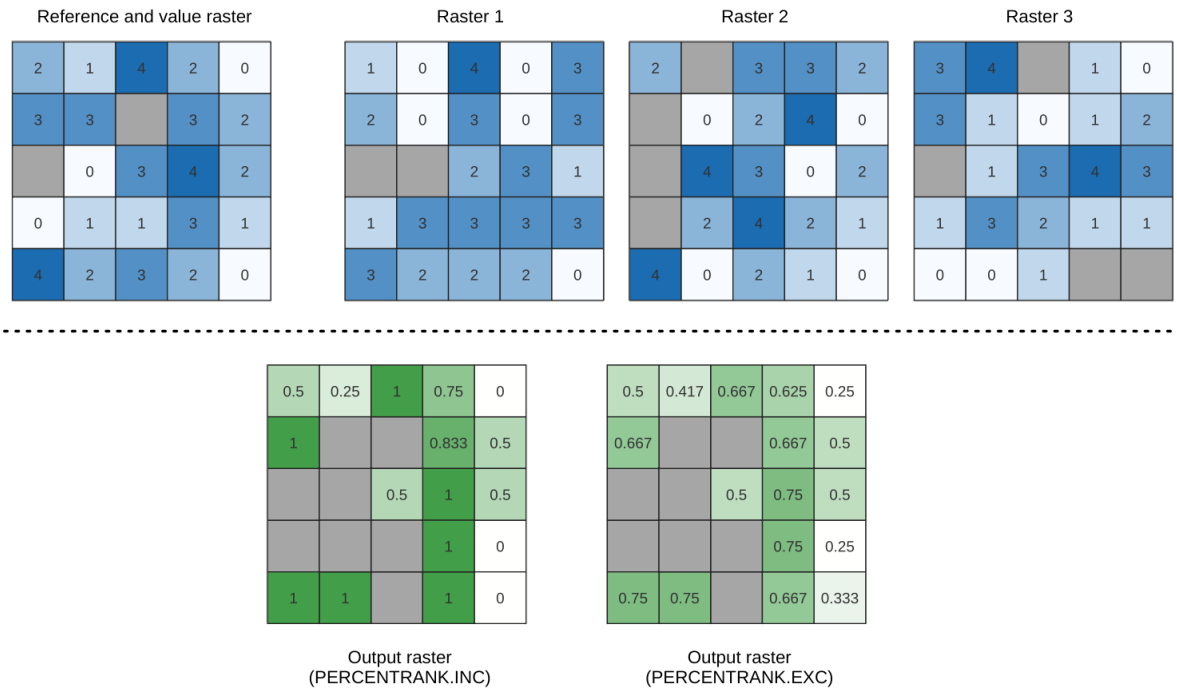
Na svakoj lokaciji ćelije, trenutna vrednost rastera vrednosti se rangira među odgovarajućim vrednostima u steku svih preklopljenih i sortiranih vrednosti ćelija ulaznih rastera. Za vrednosti izvan raspodele vrednosti steka, algoritam vraća NoData jer se vrednost ne može rangirati među vrednostima ćelija.

Postoje dva metoda za računanje percentila:

- Uključiva linearna interpolacija (PERCENTRANK.INC)
- Isključiva linearna interpolacija (PERCENTRANK.EXC)

Metodi linearne interpolacije vraćaju jedinstvene vrednosti za različite percentile. Oba metoda interpolacije prate svoje odgovarajuće metode koje implementiraju LibreOffice ili Microsoft Excel.

Obuhvat i rezolucija izlaznog rastera definisani su referentnim rasterom. Ulazni rasterski slojevi koji ne odgovaraju veličini ćelije referentnog rasterskog sloja biće preuzorkovani pomoću preuzorkovanja najbližeg suseda. NoData vrednosti u bilo kom od ulaznih slojeva daće NoData ćeliju u izlazu ako parametar „Zanemari NoData vrednosti“ nije postavljen. Tip podataka izlaznog rastera uvek će biti `Float32`.



Слика24.51: Rangiranje ćelija rasterskog sloja vrednosti. `NoData` ćelije (sive) se zanemaruju.

See also

Percentil steka ćelija, *Procentualni rang steka ćelija od vrednosti*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Rasterski sloj vrednosti	`INPUT_VALUE_RASTEF`	[raster]	Sloj čije se vrednosti rangiraju među stekom svih preklopljenih slojeva
Kanal rastera vrednosti	`VALUE_RASTER_BAND`	[numerički: ceo broj] Podrazumevano: 1	Kanal „rasterskog sloja vrednosti“ sa kojim se poredi
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod za računanje percentila: • 0 — Uključiva linearna interpolacija (PERCENTRANK.INC) • 1 — Isključiva linearna interpolacija (PERCENTRANK.EXC)
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: True	Ako nije označeno, sve NoData ćelije u ulaznim slojevima daće NoData ćeliju u izlaznom rasteru
Referentni sloj	`REFERENCE_LAYER`	[raster]	Referentni sloj za pravljenje izlaznog sloja (obuhvat, CRS, dimenzije piksela)

наставак на следећој страни

Табела 24.307 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`OUTPUT_NODATA_VALUE`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:cellstackpercentrankfromrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Statistika ćelija

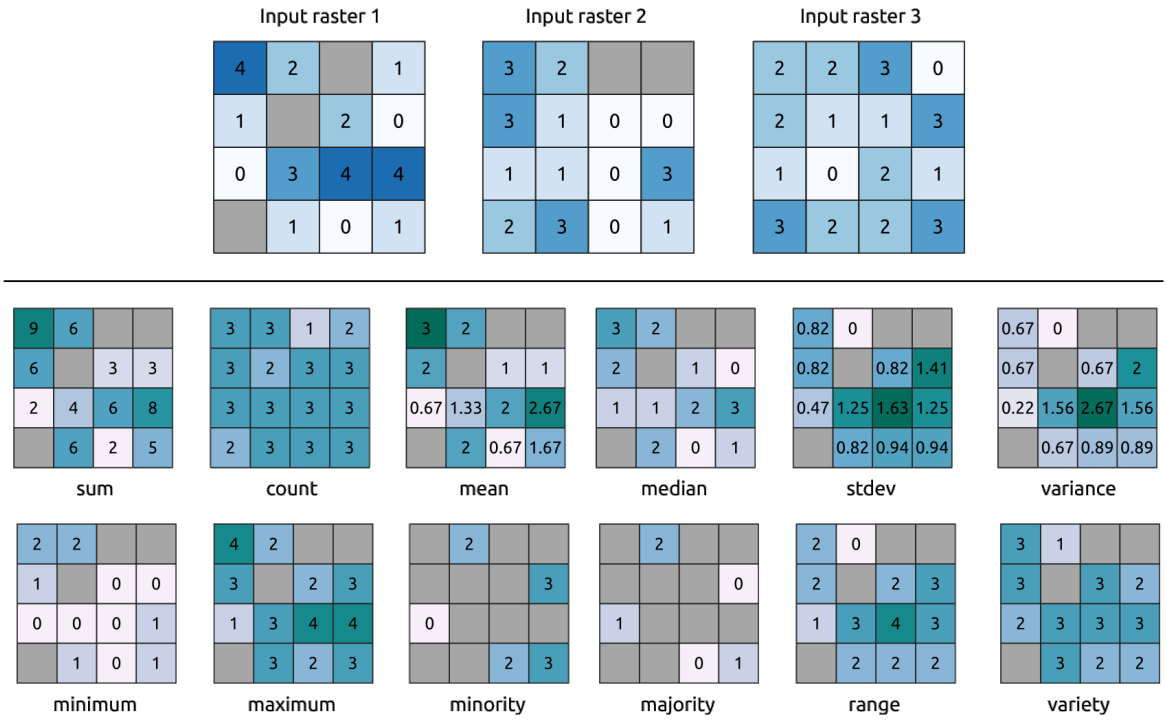
Računa statistiku po ćeliji na osnovu ulaznih rasterskih slojeva i za svaku ćeliju upisuje rezultujuću statistiku u izlazni raster. Na svakoj lokaciji ćelije, izlazna vrednost se definiše kao funkcija svih preklopljenih vrednosti ćelija ulaznih rastera.

Podrazumevano, NoData ćelija u BILO KOM od ulaznih slojeva daće NoData ćeliju u izlaznom rasteru. Ako je označena opcija *Zanemari NoData vrednosti*, tada će se NoData ulazi zanemariti u računanju statistike. To može dati NoData izlaz za lokacije gde su sve ćelije NoData.

Parametar *Referentni sloj* zadaje postojeći rasterski sloj koji se koristi kao referenca pri pravljenju izlaznog rastera. Izlazni raster imaće isti obuhvat, CRS i dimenzije piksela kao ovaj sloj.

Detalji izračunavanja: Ulazni rasterski slojevi koji ne odgovaraju veličini ćelije referentnog rasterskog sloja biće preuzorkovani pomoću `preuzorkovanja najbližeg suseda`. Tip podataka izlaznog rastera biće postavljen na najsloženiji tip podataka prisutan u ulaznim skupovima podataka, osim pri korišćenju funkcija `Mean`, `Standard deviation` i `Variance` (tip podataka je uvek `Float32` ili `Float64` u zavisnosti od ulaznog float tipa) ili `Count` i `Variety` (tip podataka je uvek `Int32`).

- `Count`: statistika broja uvek daje broj ćelija bez NoData vrednosti na trenutnoj lokaciji ćelije.
- `Median`: ako je broj ulaznih slojeva paran, medijana se računa kao aritmetička sredina dve srednje vrednosti uređenih ulaznih vrednosti ćelija.
- `Minority/Majority`: ako se ne može pronaći jedinstvena manjina ili većina, rezultat je NoData, osim ako su sve ulazne vrednosti ćelija jednake.



Слика24.52: Primer sa svim statističkim funkcijama. `NoData` ćelije (sive) se uzimaju u obzir.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Ulazni rasterski slojevi
Statistika	`STATISTIC`	[nabrajanje] Podrazumevano: 0	Dostupne statistike. Opcije: • 0 — Zbir • 1 — Broj • 2 — Srednja vrednost • 3 — Medijana • 4 — Standardna devijacija • 5 — Varijansa • 6 — Minimum • 7 — Maksimum • 8 — Manjina (najređa vrednost) • 9 — Većina (najčešća vrednost) • 10 — Raspon (max - min) • 11 — Raznolikost (broj jedinstvenih vrednosti)
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: True	Računaj statistiku i za sve stekove ćelija, zanemarujući pojavu NoData.
Referentni sloj	`REF_LAYER`	[raster]	Referentni sloj od kojeg se pravi izlazni sloj (obuhvat, CRS, dimenzije piksela)
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost Opciono	`OUTPUT_NO_DATA_VALU`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Izlazni raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat

наставак на следећој страни

Табела 24.312 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:cellstatistics`

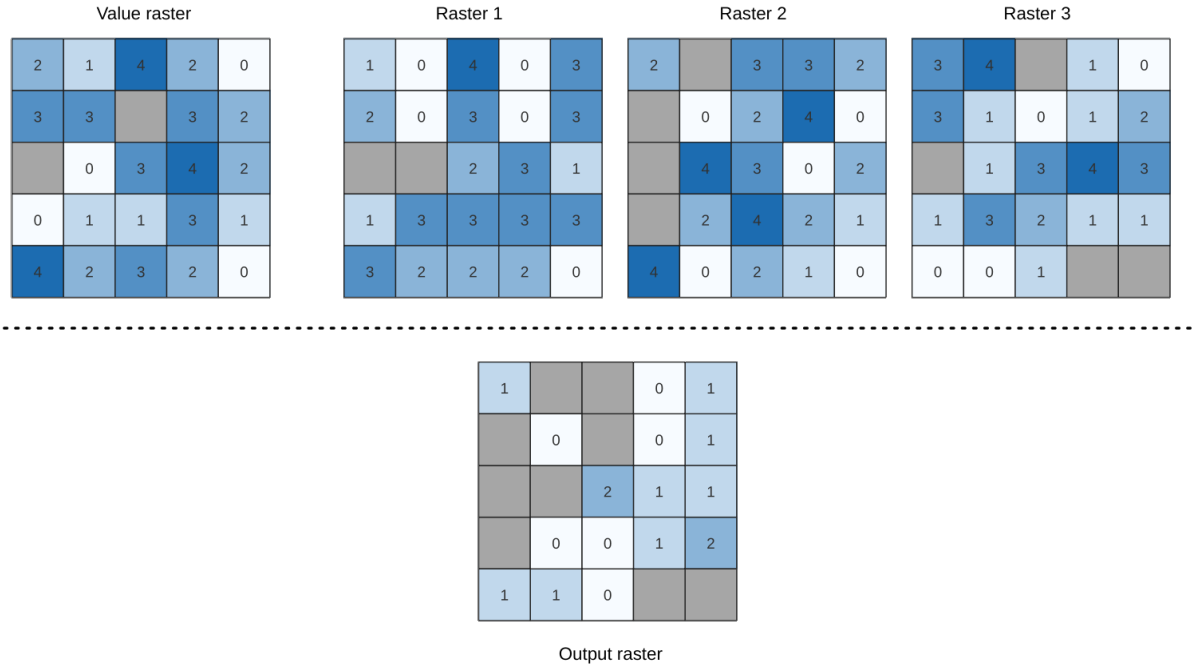
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Učestalost jednaka

Izračunava po ćeliji učestalost (broj puta) kada su vrednosti ulaznog steka rastera jednake vrednosti sloja vrednosti. Obuhvat i rezolucija izlaznog rastera definisani su ulaznim rasterskim slojem i uvek su tipa `Int32`.

Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera - koristite GDAL za korišćenje drugih kanala u analizi. Izlazna NoData vrednost može se postaviti ručno.



Слика24.53: Za svaku ćeliju u izlaznom rasteru, vrednost predstavlja broj puta kada su odgovarajuće ćelije u listi rastera iste kao raster vrednosti. `NoData` ćelije (sive) se uzimaju u obzir.

See also

Učestalost veće od, *Učestalost manja od*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster vrednosti	`INPUT_VALUE_RASTER`	[raster]	Ulazni sloj vrednosti služi kao referentni sloj za uzoračke slojeve
Kanal rastera vrednosti	`INPUT_VALUE_RASTER`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Izaberite kanal koji želite da koristite kao uzorak
Ulazni rasterski slojevi	`INPUT_RASTERS`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Ako nije označeno, sve NoData ćelije u rasteru vrednosti ili steku slojeva podataka daće NoData ćeliju u izlaznom rasteru
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost Opciono	`OUTPUT_NO_DATA_VALU`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Broj ćelija sa istim pojavama vrednosti	`FOUND_LOCATIONS_CC`	[numerički: ceo broj]	
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju

наставак на следећој страни

Табела 24.315 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Srednja učestalost na lokacijama ispravnih ćelija	`MEAN_FREQUENCY_PEF`	[numerički: decimalni]	
Broj pojava vrednosti	`OCCURRENCE_COUNT`	[numerički: ceo broj]	
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:equaltofrequency`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fazifikuj raster (Gausova pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Gausove pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači nepripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Funkcija Gausove pripadnosti definiše se kao $\mu(x) = e^{-f1*(x-f2)^2}$, gde je *f1* rasipanje, a *f2* središnja tačka.



Слика24.54: Primer fazifikacije rastera (Gausova pripadnost) (središnja tačka=300, rasipanje=0.00001).

See also

Fazifikuj raster (velika pripadnost), *Fazifikuj raster (linearna pripadnost)*, *Fazifikuj raster (bliska pripadnost)*, *Fazifikuj raster (stepena pripadnost)*, *Fazifikuj raster (mala pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.
Središnja tačka funkcije	`FUZZYMIDPOINT`	[numerički: decimalni] Podrazumevano: 10.0	Središnja tačka Gausove funkcije
Rasipanje funkcije	`FUZZYSPREAD`	[numerički: decimalni] Podrazumevano: 0.01	Rasipanje Gausove funkcije
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

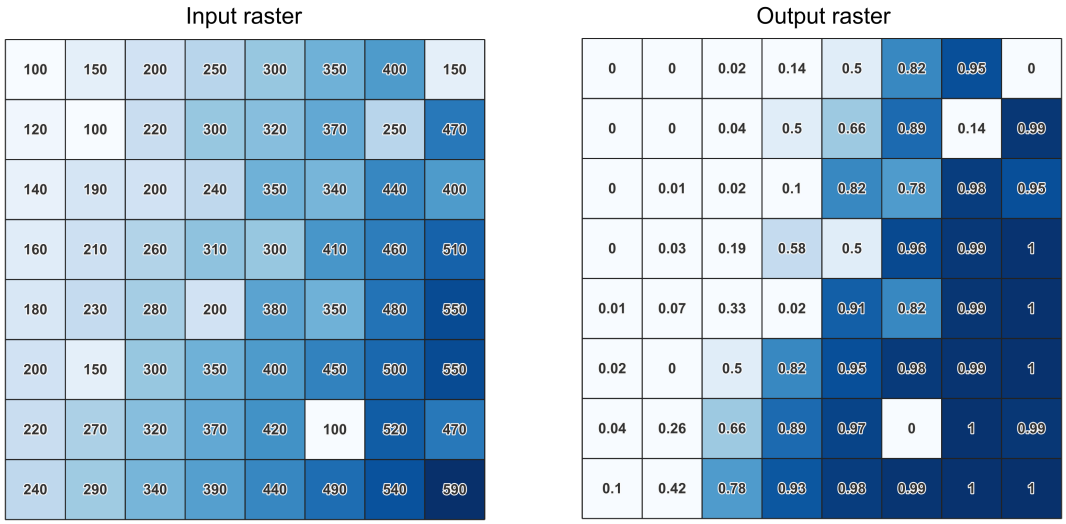
ID algoritma: `native:fuzzifyrastergaussianmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fazifikuj raster (velika pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Velike pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači ne-pripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Funkcija velike pripadnosti definiše se kao $\mu(x) = \frac{1}{1+\left(\frac{x}{f2}\right)^{-f1}}$, gde je *f1* rasipanje, a *f2* središnja tačka.



Слика24.55: Primer fazifikacije rastera (velika pripadnost) (središnja tačka=300, rasipanje=10).

See also

Fazifikuj raster (Gausova pripadnost), *Fazifikuj raster (linearna pripadnost)*, *Fazifikuj raster (bliska pripadnost)*, *Fazifikuj raster (stepena pripadnost)*, *Fazifikuj raster (mala pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.

наставак на следећој страни

Табела 24.319 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Središnja tačka funkcije	`FUZZYMIDPOINT`	[numerički: decimalni] Podrazumevano: 50.0	Središnja tačka velike funkcije
Rasipanje funkcije	`FUZZYSPREAD`	[numerički: decimalni] Podrazumevano: 5.0	Rasipanje velike funkcije
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:fuzzifyrasterlargemembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fazifikuj raster (linearna pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Linearne pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači nepripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Linearna funkcija definiše se kao $\mu(X)\begin{cases}0 & x \le a \\ \frac{x-a}{b-a} & a < x < b \\ 1 & x \ge b\end{cases}$, gde je a donja granica, a b gornja granica. Ova jednačina dodeljuje vrednosti pripadnosti pomoću linearne transformacije za vrednosti piksela između donje i gornje granice. Vrednostima piksela manjim od donje granice dodeljuje se pripadnost 0, dok se vrednostima piksela većim od gornje granice dodeljuje pripadnost 1.

Input raster

100	150	200	250	300	350	400	150
120	100	220	300	320	370	250	470
140	190	200	240	350	340	440	400
160	210	260	310	300	410	460	510
180	230	280	200	380	350	480	550
200	150	300	350	400	450	500	550
220	270	320	370	420	100	520	470
240	290	340	390	440	490	540	590

Output raster

0	0.1	0.2	0.31	0.41	0.51	0.61	0.1
0.04	0	0.24	0.41	0.45	0.55	0.31	0.76
0.08	0.18	0.2	0.29	0.51	0.49	0.69	0.61
0.12	0.22	0.33	0.43	0.41	0.63	0.73	0.84
0.16	0.27	0.37	0.2	0.57	0.51	0.78	0.92
0.2	0.1	0.41	0.51	0.61	0.71	0.82	0.92
0.24	0.35	0.45	0.55	0.65	0	0.86	0.76
0.29	0.39	0.49	0.59	0.69	0.8	0.9	1

Слика24.56: Primer fazifikacije rastera (linearna pripadnost) (donja granica=100, gornja granica=590).

See also

Fazifikuj raster (Gausova pripadnost), *Fazifikuj raster (velika pripadnost)*, *Fazifikuj raster (bliska pripadnost)*, *Fazifikuj raster (stepena pripadnost)*, *Fazifikuj raster (mala pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.
Donja granica rasplinute pripadnosti	`FUZZYLOWBOUND`	[numerički: decimalni] Podrazumevano: 0.0	Donja granica linearne funkcije

наставак на следећој страни

Табела 24.322 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Gornja granica rasplinute pripadnosti	`FUZZYHIGHBOUND`	[numerički: decimalni] Podrazumevano: 1.0	Gornja granica linearne funkcije
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:fuzzifyrasterlinearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fazifikuj raster (bliska pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Bliske pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači nepripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Funkcija bliske pripadnosti definiše se kao $\mu(x) = \frac{1}{1 + f1 * (x - f2)^2}$, gde je *f1* rasipanje, a *f2* središnja tačka.



Слика24.57: Primer fazifikacije rastera (bliska pripadnost) (središnja tačka=300, rasipanje=0.00001).

See also

Fazifikuj raster (Gausova pripadnost), *Fazifikuj raster (velika pripadnost)*, *Fazifikuj raster (linearna pripadnost)*, *Fazifikuj raster (stepena pripadnost)*, *Fazifikuj raster (mala pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.
Središnja tačka funkcije	`FUZZYMIDPOINT`	[numerički: decimalni] Podrazumevano: 50.0	Središnja tačka bliske funkcije
Rasipanje funkcije	`FUZZYSPREAD`	[numerički: decimalni] Podrazumevano: 0.01	Rasipanje bliske funkcije

наставак на следећој страни

Табела 24.325 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:fuzzifyrasternearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

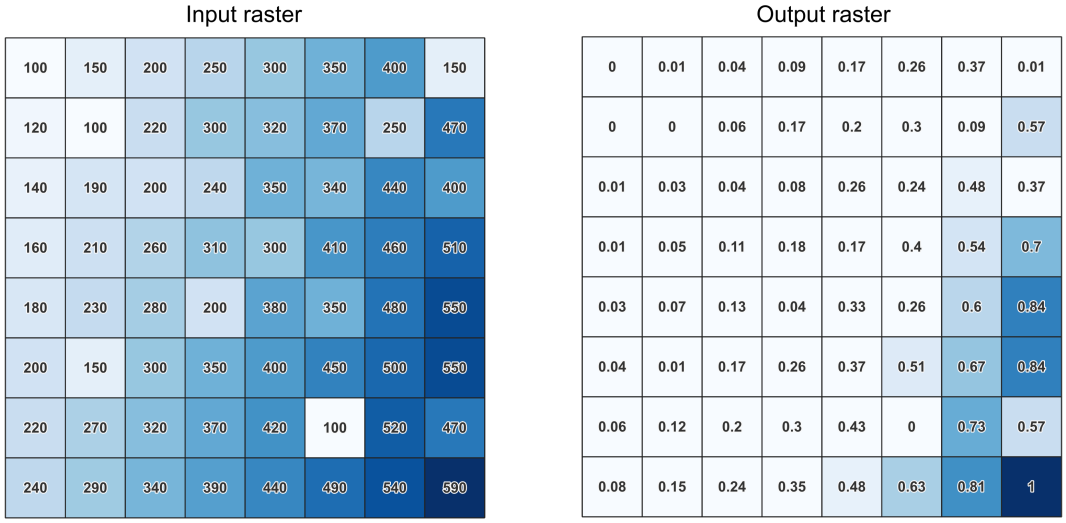
Fazifikuj raster (stepena pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Stepene pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači nepripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Stepena funkcija definiše kao

$$\mu(x)\begin{cases}0 & x \le a \\ \left(\frac{x-a}{b-a}\right)^{f1} & a < x < b \\ 1 & x \ge b\end{cases}$$

, gde je a donja granica, b gornja granica, a $f1$ eksponent. Ova jednačina dodeljuje vred-

nosti pripadnosti pomoću stepene transformacije za vrednosti piksela između donje i gornje granice. Vrednostima piksela manjim od donje granice dodeljuje se pripadnost 0, dok se vrednostima piksela većim od gornje granice dodeljuje pripadnost 1.



Слика24.58: Primer fazifikacije rastera (stepena pripadnost) (donja granica=100, gornja granica=590, eksponent pripadnosti=2).

See also

Fazifikuj raster (Gausova pripadnost), *Fazifikuj raster (velika pripadnost)*, *Fazifikuj raster (linearna pripadnost)*, *Fazifikuj raster (bliska pripadnost)*, *Fazifikuj raster (mala pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.
Donja granica rasplinute pripadnosti	`FUZZYLOWBOUND`	[numerički: decimalni] Podrazumevano: 0.0	Donja granica stepene funkcije
Gornja granica rasplinute pripadnosti	`FUZZYHIGHBOUND`	[numerički: decimalni] Podrazumevano: 1.0	Gornja granica stepene funkcije
Gornja granica rasplinute pripadnosti	`FUZZYEXPONENT`	[numerički: decimalni] Podrazumevano: 2.0	Eksponent stepene funkcije

наставак на следећој страни

Табела 24.328 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

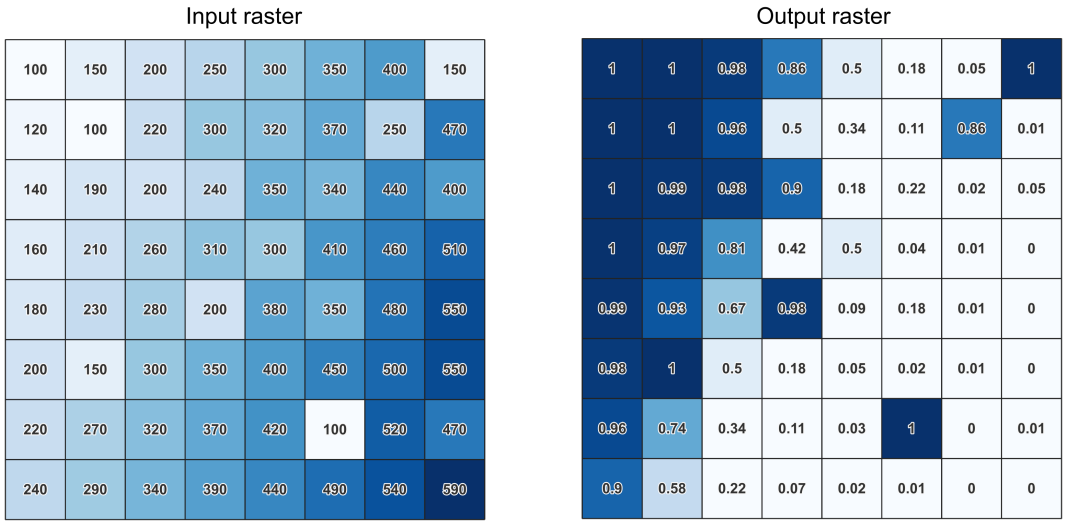
ID algoritma: `native:fuzzifyrasterpowermembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Fazifikuj raster (mala pripadnost)

Pretvara ulazni raster u fazifikovan raster dodeljivanjem vrednosti pripadnosti svakom pikselu, pomoću funkcije Male pripadnosti. Vrednosti pripadnosti su u opsegu od 0 do 1. U fazifikovanom rasteru, vrednost 0 znači nepripadnost definisanom rasplinutom skupu, dok vrednost 1 znači potpunu pripadnost. Funkcija male pripadnosti definiše se kao $\mu(x) = \frac{1}{1+\left(\frac{x}{f2}\right)^{f1}}$, gde je *f1* rasipanje, a *f2* središnja tačka.



Слика24.59: Primer fazifikacije rastera (mala pripadnost) (središnja tačka=300, rasipanje=10).

See also

Fazifikuj raster (Gausova pripadnost), *Fazifikuj raster (velika pripadnost)* *Fazifikuj raster (linearna pripadnost)*, *Fazifikuj raster (bliska pripadnost)*, *Fazifikuj raster (stepena pripadnost)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da fazifikujete.
Središnja tačka funkcije	`FUZZYMIDPOINT`	[numerički: decimalni] Podrazumevano: 50.0	Središnja tačka male funkcije
Rasipanje funkcije	`FUZZYSPREAD`	[numerički: decimalni] Podrazumevano: 5.0	Rasipanje male funkcije
Fazifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATI-ON_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREA-TE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Fazifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HE-IGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TO-TAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:fuzzifyrastersmallmembership`

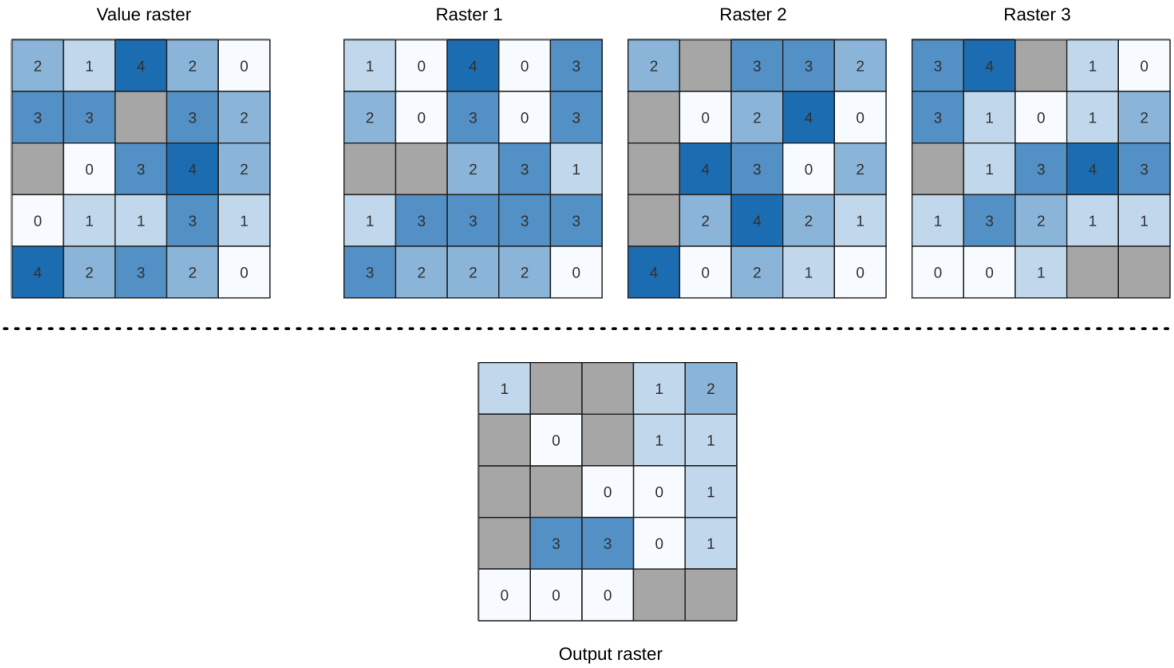
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Učestalost veće od

Izračunava po ćeliji učestalost (broj puta) kada su vrednosti ulaznog steka rastera jednake vrednosti rastera vrednosti. Obuhvat i rezolucija izlaznog rastera definisani su ulaznim rasterskim slojem i uvek su tipa `Int32`.

Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera - koristite GDAL za korišćenje drugih kanala u analizi. Izlazna NoData vrednost može se postaviti ručno.



Слика24.60: Za svaku ćeliju u izlaznom rasteru, vrednost predstavlja broj puta kada su odgovarajuće ćelije u listi rastera veće od rastera vrednosti. `NoData` ćelije (sive) se uzimaju u obzir.

See also

Učestalost jednaka, *Učestalost manja od*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster vrednosti	`INPUT_VALUE_RASTER`	[raster]	Ulazni sloj vrednosti služi kao referentni sloj za uzoračke slojeve
Kanal rastera vrednosti	`INPUT_VALUE_RASTER`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Izaberite kanal koji želite da koristite kao uzorak
Ulazni rasterski slojevi	`INPUT_RASTERS`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Ako nije označeno, sve NoData ćelije u rasteru vrednosti ili steku slojeva podataka daće NoData ćeliju u izlaznom rasteru
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost Opciono	`OUTPUT_NO_DATA_VALU`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Broj ćelija sa istim pojavama vrednosti	`FOUND_LOCATIONS_CC`	[numerički: ceo broj]	
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Srednja učestalost na lokacijama ispravnih ćelija	`MEAN_FREQUENCY_PER`	[numerički: decimalni]	
Broj pojava vrednosti	`OCCURRENCE_COUNT`	[numerički: ceo broj]	
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:greaterthanfrequency`

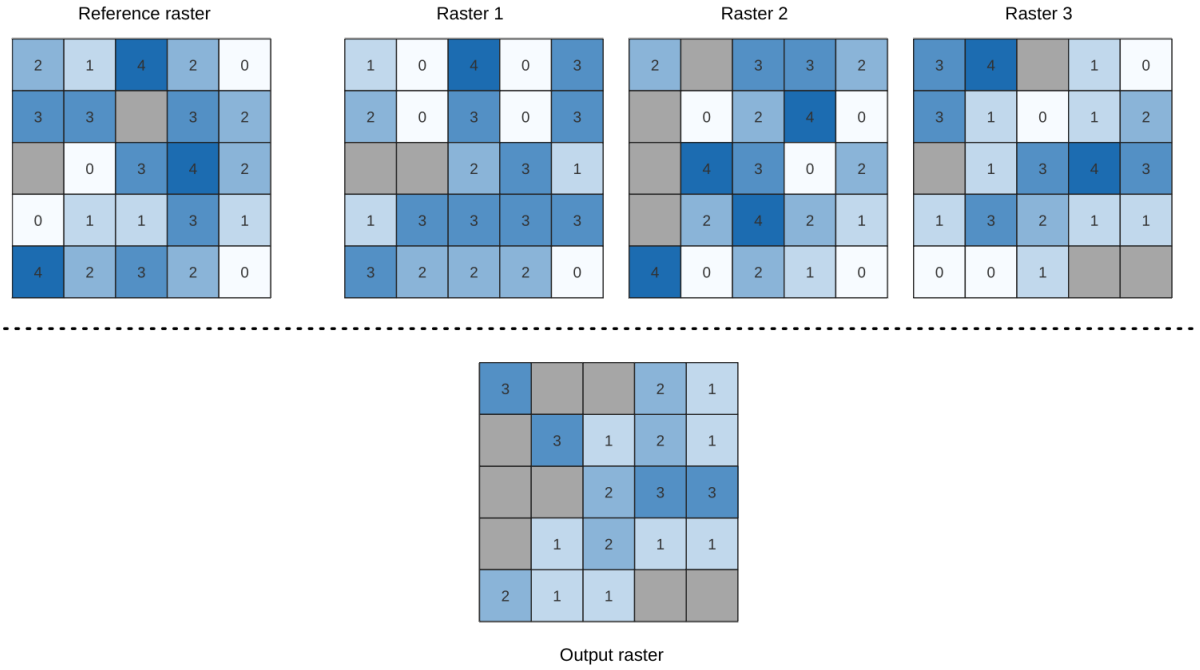
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Najviši položaj u steku rastera

Izračunava po ćeliji položaj rastera sa najvišom vrednošću u steku rastera. Brojanje položaja počinje od 1 i ide do ukupnog broja ulaznih rastera. Redosled ulaznih rastera je bitan za algoritam. Ako više rastera ima najvišu vrednost, za vrednost položaja koristiće se prvi raster.

Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera - koristite GDAL za korišćenje drugih kanala u analizi. Sve NoData ćelije u steku rasterskih slojeva daće NoData ćeliju u izlaznom rasteru, osim ako je označen parametar „zanemari NoData“. Izlazna NoData vrednost može se postaviti ručno. Obuhvat i rezolucija izlaznih rastera definisani su referentnim rasterskim slojem i uvek su tipa `Int32`.



See also

Najniži položaj u steku rastera

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni rasterski slojevi	`INPUT_RASTERS`	[raster] [lista]	Lista rasterskih slojeva sa kojima se poredi
Referentni sloj	`REFERENCE_LAYER`	[raster]	Referentni sloj za pravljenje izlaznog sloja (obuhvat, CRS, dimenzije piksela)
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Ako nije označeno, sve NoData ćelije u steku slojeva podataka daće NoData ćeliju u izlaznom rasteru
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera koji sadrži rezultat. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`OUTPUT_NODATA_VALUE`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:highestpositioninrasterstack`

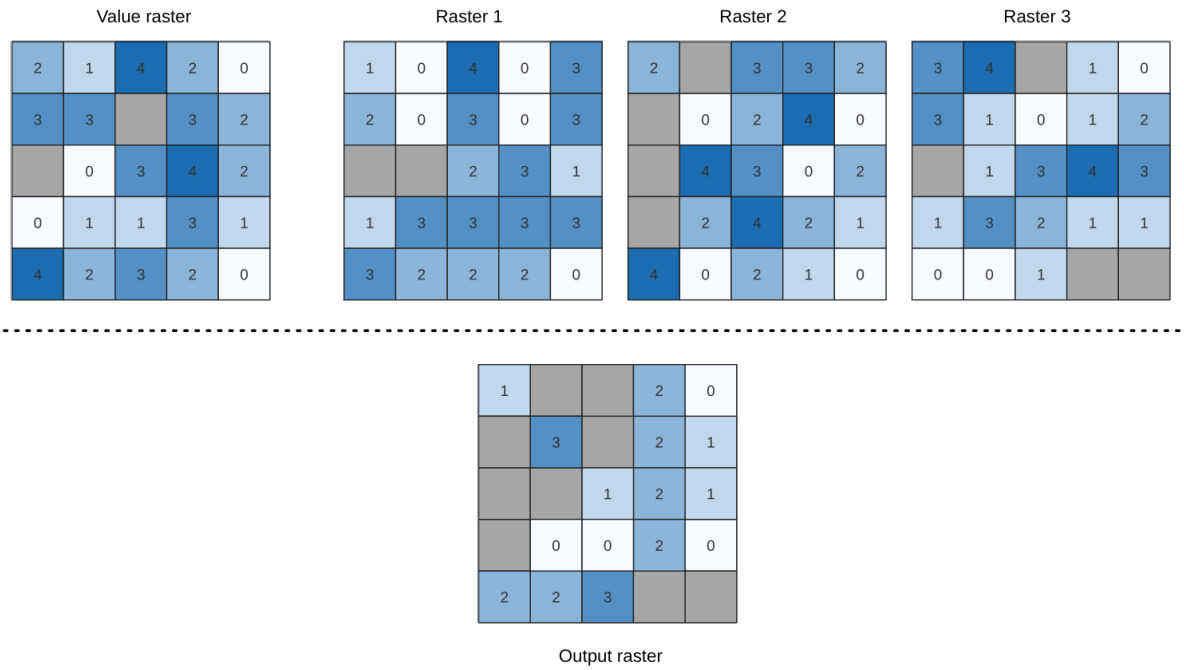
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Učestalost manja od

Izračunava po ćeliji učestalost (broj puta) kada su vrednosti ulaznog steka rastera manje od vrednosti rastera vrednosti. Obuhvat i rezolucija izlaznog rastera definisani su ulaznim rasterskim slojem i uvek su tipa `Int32`.

Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera - koristite GDAL za korišćenje drugih kanala u analizi. Izlazna NoData vrednost može se postaviti ručno.



Слика24.61: Za svaku ćeliju u izlaznom rasteru, vrednost predstavlja broj puta kada su odgovarajuće ćelije u listi rastera manje od rastera vrednosti. `NoData` ćelije (sive) se uzimaju u obzir.

See also

Učestalost jednaka, *Učestalost veće od*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster vrednosti	`INPUT_VALUE_RASTER`	[raster]	Ulazni sloj vrednosti služi kao referentni sloj za uzoračke slojeve
Kanal rastera vrednosti	`INPUT_VALUE_RASTER`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Izaberite kanal koji želite da koristite kao uzorak
Ulazni rasterski slojevi	`INPUT_RASTERS`	[raster] [lista]	Rasterski slojevi koji se izračunavaju. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Ako nije označeno, sve NoData ćelije u rasteru vrednosti ili steku slojeva podataka daće NoData ćeliju u izlaznom rasteru
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost Opciono	OUTPUT_NO_DATA_VALU	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono Dodato u 3.40	CREATION_OPTIONS (za QGIS <= 3.42 ovo je bilo CREATE_OPTIONS)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (|).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	OUTPUT	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	CRS_AUTHID	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	EXTENT	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Broj ćelija sa istim pojavama vrednosti	FOUND_LOCATIONS_CC	[numerički: ceo broj]	
Visina u pikselima	HEIGHT_IN_PIXELS	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	TOTAL_PIXEL_COUNT	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Srednja učestalost na lokacijama ispravnih ćelija	MEAN_FREQUENCY_PEF	[numerički: decimalni]	
Broj pojava vrednosti	OCCURRENCE_COUNT	[numerički: ceo broj]	
Širina u pikselima	WIDTH_IN_PIXELS	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:lessthanfrequency`

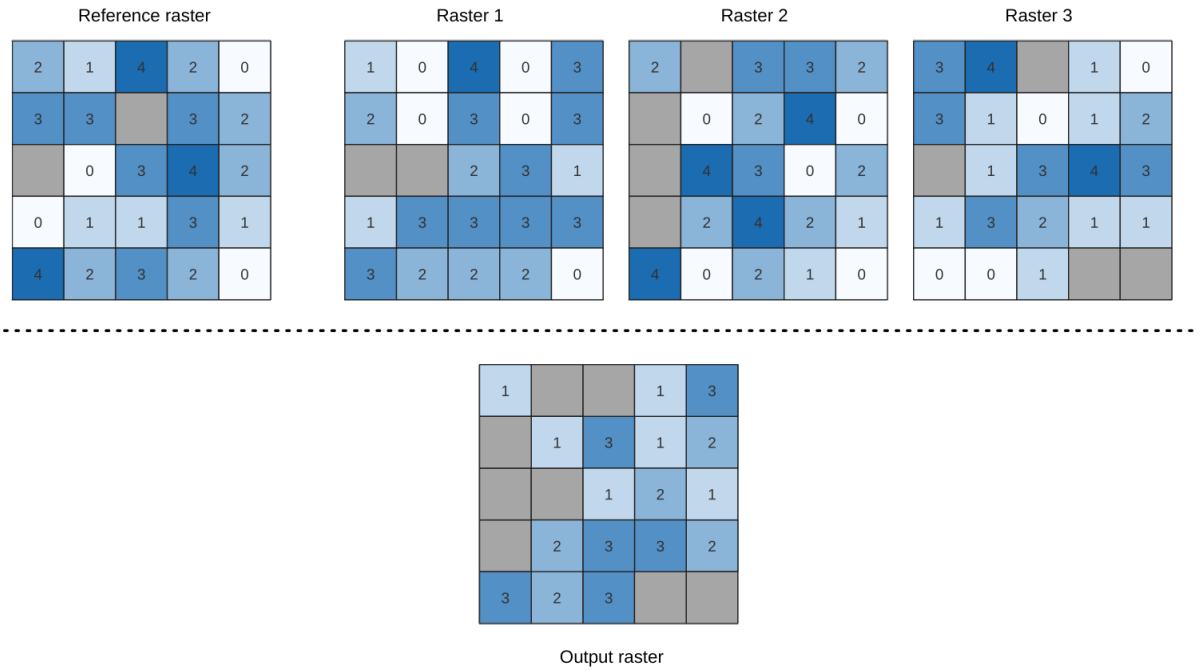
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Najniži položaj u steku rastera

Izračunava po ćeliji položaj rastera sa najnižom vrednošću u steku rastera. Brojanje položaja počinje od 1 i ide do ukupnog broja ulaznih rastera. Redosled ulaznih rastera je bitan za algoritam. Ako više rastera ima najnižu vrednost, za vrednost položaja koristiće se prvi raster.

Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera - koristite GDAL za korišćenje drugih kanala u analizi. Sve NoData ćelije u steku rasterskih slojeva daće NoData ćeliju u izlaznom rasteru, osim ako je označen parametar „zanemari NoData“. Izlazna NoData vrednost može se postaviti ručno. Obuhvat i rezolucija izlaznih rastera definisani su referentnim rasterskim slojem i uvek su tipa `Int32`.



See also

Najviši položaj u steku rastera

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni rasterski slojevi	`INPUT_RASTERS`	[raster] [lista]	Lista rasterskih slojeva sa kojima se poredi
Referentni sloj	`REFERENCE_LAYER`	[raster]	Referentni sloj za pravljenje izlaznog sloja (obuhvat, CRS, dimenzije piksela)
Zanemari NoData vrednosti	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Ako nije označeno, sve NoData ćelije u steku slojeva podataka daće NoData ćeliju u izlaznom rasteru
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera koji sadrži rezultat. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`OUTPUT_NODATA_VALUE`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:lowestpositioninrasterstack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

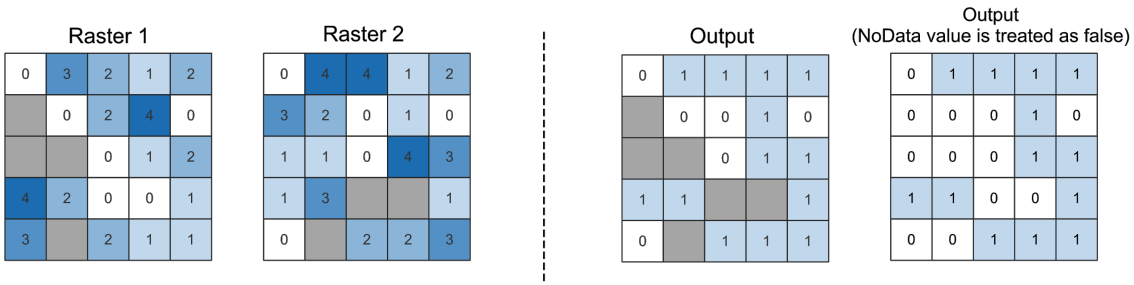
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterski logički AND

Računa logičko `AND` za skup ulaznih rastera. Ako svi ulazni rasteri imaju vrednost različitu od nule za neki piksel, taj piksel biće postavljen na `1` u izlaznom rasteru. Ako bilo koji od ulaznih rastera ima vrednost `0` za piksel, biće postavljen na `0` u izlaznom rasteru.

Parametar referentnog sloja zadaje postojeći rasterski sloj koji se koristi kao referenca pri pravljenju izlaznog rastera. Izlazni raster imaće isti obuhvat, CRS i dimenzije piksela kao ovaj sloj.

Podrazumevano, NoData piksel u BILO KOM od ulaznih slojeva daće NoData piksel u izlaznom rasteru. Ako je označena opcija *Tretiraj NoData vrednosti kao netačne*, tada će se NoData ulazi tretirati isto kao ulazna vrednost `0`.



Слика24.62: Primer rasterskog logičkog AND.

See also

Rasterski logički OR

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Lista ulaznih rasterskih slojeva
Referentni sloj	`REF_LAYER`	[raster]	Referentni sloj od kojeg se pravi izlazni sloj (obuhvat, CRS, dimenzije piksela)
Tretiraj NoData vrednosti kao netačne	`NODATA_AS_FALSE`	[logička vrednost] Podrazumevano: False	Tretiraj NoData vrednosti u ulaznim datotekama kao 0 pri izvršavanju operacije
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera koji sadrži rezultat. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`NO_DATA`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju

наставак на следећој страни

Табела 24.347 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	DATA_TYPE	[nabrajanje] Podrazumevano: 5	Tip podataka izlaznog rastera. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Opcije kreiranja Opciono Dodato u 3.40	CREATION_OPTIONS (za QGIS <= 3.42 ovo je bilo CREATE_OPTIONS)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (|).

Izlazi

Oznaka	Назив	Tip	Opis
Obuhvat	EXTENT	[niska]	Prostorni obuhvat izlaznog rasterskog sloja
Identifikator autoriteta CRS-a	CRS_AUTHID	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Širina u pikselima	WIDTH_IN_PIXELS	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	HEIGHT_IN_PIXELS	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	TOTAL_PIXEL_COUNT	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Broj NoData piksela	NODATA_PIXEL_COUNT	[numerički: ceo broj]	Broj NoData piksela u izlaznom rasterskom sloju
Broj True piksela	TRUE_PIXEL_COUNT	[numerički: ceo broj]	Broj True piksela (vrednost = 1) u izlaznom rasterskom sloju
Broj False piksela	FALSE_PIXEL_COUNT	[numerički: ceo broj]	Broj False piksela (vrednost = 0) u izlaznom rasterskom sloju

наставак на следећој страни

Табела 24.348 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat

Python kod

ID algoritma: `native:rasterbooleanand`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

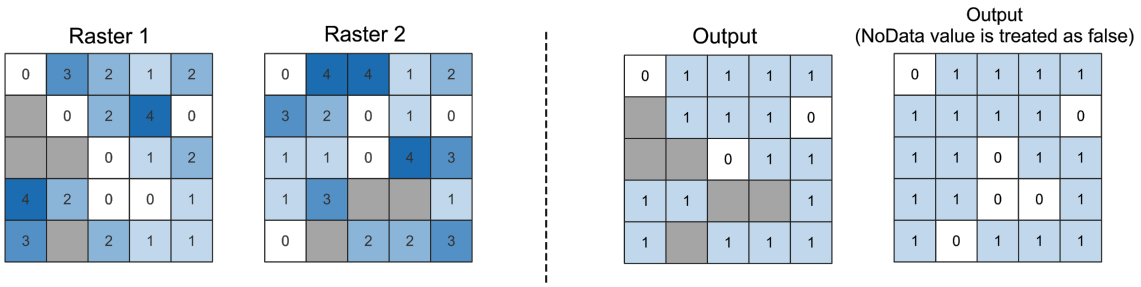
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterski logički OR

Računa logičko OR za skup ulaznih rastera. Ako bilo koji od ulaznih rastera ima vrednost različitu od nule za neki piksel, taj piksel biće postavljen na 1 u izlaznom rasteru. Ako svi ulazni rasteri imaju vrednost 0 za piksel, biće postavljen na 0 u izlaznom rasteru.

Parametar referentnog sloja zadaje postojeći rasterski sloj koji se koristi kao referenca pri pravljenju izlaznog rastera. Izlazni raster imaće isti obuhvat, CRS i dimenzije piksela kao ovaj sloj.

Podrazumevano, NoData piksel u BILO KOM od ulaznih slojeva daće NoData piksel u izlaznom rasteru. Ako je označena opcija *Tretiraj NoData vrednosti kao netačne*, tada će se NoData ulazi tretirati isto kao ulazna vrednost `0`.



Слика24.63: Primer rasterskog logičkog OR.

See also

Rasterski logički AND

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Lista ulaznih rasterskih slojeva
Referentni sloj	`REF_LAYER`	[raster]	Referentni sloj od kojeg se pravi izlazni sloj (obuhvat, CRS, dimenzije piksela)
Tretiraj NoData vrednosti kao netačne	`NODATA_AS_FALSE`	[logička vrednost] Podrazumevano: False	Tretiraj NoData vrednosti u ulaznim datotekama kao 0 pri izvršavanju operacije

наставак на следећој страни

Табела 24.349 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera koji sadrži rezultat. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`NO_DATA`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se koristi za NoData u izlaznom sloju
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Tip podataka izlaznog rastera. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja

наставак на следећој страни

Табела 24.351 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[crs]	Koordinatni referentni sistem izlaznog rasterskog sloja
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Broj NoData piksela	`NODATA_PIXEL_COUNT`	[numerički: ceo broj]	Broj NoData piksela u izlaznom rasterskom sloju
Broj True piksela	`TRUE_PIXEL_COUNT`	[numerički: ceo broj]	Broj True piksela (vrednost = 1) u izlaznom rasterskom sloju
Broj False piksela	`FALSE_PIXEL_COUNT`	[numerički: ceo broj]	Broj False piksela (vrednost = 0) u izlaznom rasterskom sloju
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat

Python kod

ID algoritma: `native:rasterbooleanor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterski kalkulator

Izvršava algebarske operacije pomoću rasterskih slojeva.

Rezultujući sloj imaće vrednosti izračunate prema izrazu. Izraz može sadržati numeričke vrednosti, operatore i reference na bilo koji od slojeva u trenutnom projektu.

наставак на следећој страни

Табела 24.352 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

See also

Rasterski kalkulator (virtuelni), *Rasterski kalkulator*, *Raster Calculator*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Lista ulaznih rasterskih slojeva
Izraz	`EXPRESSION`	[izraz]	*Izraz zasnovan na rasteru* koji se koristi za računanje izlaznog rasterskog sloja.
Izlazni obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Ako obuhvat nije zadat, koristiće se minimalni obuhvat koji pokriva sve izabrane referentne slojeve. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina izlazne ćelije (ostavite prazno za automatsko postavljanje) Opciono	`CELLSIZE`	[numerički: decimalni]	Veličina ćelije izlaznog rasterskog sloja. Ako veličina ćelije nije zadata, koristiće se minimalna veličina ćelije izabranog referentnog sloja(eva). Veličina ćelije biće ista za X i Y osu.
Izlazni CRS Opciono	`CRS`	[crs]	CRS izlaznog rasterskog sloja. Ako izlazni CRS nije zadat, koristiće se CRS prvog referentnog sloja.
Izračunato	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono	`CREATI-ON_OPTIONS`	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izračunato	`OUTPUT`	[raster]	Izlazna rasterska datoteka sa izračunatim vrednostima.

Python kod

ID algoritma: `native:rastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterski kalkulator (virtuelni)

`Dodato u 3.34`

Izvršava algebarske operacije pomoću rasterskih slojeva i generiše rezultat u memoriji.

Rezultujući sloj imaće vrednosti izračunate prema izrazu. Izraz može sadržati numeričke vrednosti, operatore i reference na bilo koji od slojeva u trenutnom projektu.

Virtuelni rasterski sloj je rasterski sloj definisan svojim URI-jem, čiji se pikseli računaju u hodu. To nije nova datoteka na disku; virtuelni sloj je i dalje povezan sa rasterima korišćenim u izračunavanju, što znači da bi ga brisanje ili premeštanje tih rastera pokvarilo. Može se zadati *Naziv sloja*, u suprotnom se koristi izraz izračunavanja kao takav. Uklanjanje virtuelnog sloja iz projekta ga briše, a može se učiniti trajnim u datoteci pomoću kontekstnog menija sloja *Izvezi* ► *Sačuvaj kao…*.

See also

Rasterski kalkulator, *Rasterski kalkulator*, *Raster Calculator*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Lista ulaznih rasterskih slojeva
Izraz	`EXPRESSION`	[izraz]	*Izraz zasnovan na rasteru* koji se koristi za računanje izlaznog rasterskog sloja.

наставак на следећој страни

Табела 24.355 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Ako obuhvat nije zadat, koristiće se minimalni obuhvat koji pokriva sve izabrane referentne slojeve. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina izlazne ćelije (ostavite prazno za automatsko postavljanje) Opciono	`CELLSIZE`	[numerički: decimalni]	Veličina ćelije izlaznog rasterskog sloja. Ako veličina ćelije nije zadata, koristiće se minimalna veličina ćelije izabranog referentnog sloja(eva). Veličina ćelije biće ista za X i Y osu.
Izlazni CRS Opciono	`CRS`	[crs]	CRS izlaznog rasterskog sloja. Ako izlazni CRS nije zadat, koristiće se CRS prvog referentnog sloja.
Naziv izlaznog sloja Opciono	`LAYER_NAME`	[niska]	Naziv koji se dodeljuje generisanom sloju. Ako nije postavljen, koristi se tekst izraza izračunavanja.

Izlazi

Oznaka	Назив	Tip	Opis
Izračunato	`OUTPUT`	[raster]	Izlazni virtuelni rasterski sloj sa izračunatim vrednostima.

Python kod

ID algoritma: `native:virtualrastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Svojstva rasterskog sloja

Vraća osnovna svojstva zadatog rasterskog sloja, uključujući obuhvat, veličinu u pikselima i dimenzije piksela (u jedinicama karte), broj kanala i NoData vrednost.

Ovaj algoritam je namenjen za izdvajanje ovih korisnih svojstava radi korišćenja kao ulaznih vrednosti za druge algoritme u modelu - npr. da omogući prosleđivanje veličina piksela postojećeg rastera GDAL rasterskom algoritmu.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala Opciono	`BAND`	[rasterski kanal] Podrazumevano: Nije postavljeno	Da li se vraćaju i svojstva određenog kanala. Ako je kanal zadat, vraća se i noData vrednost za izabrani kanal.

Izlazi

Oznaka	Назив	Tip	Opis
Broj kanala u rasteru	`BAND_COUNT`	[numerički: ceo broj]	Broj kanala u rasteru
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Obuhvat rasterskog sloja u CRS-u
Kanal ima postavljenu NoData vrednost	`HAS_NODATA_VALUE`	[logička vrednost]	Označava da li rasterski sloj ima postavljenu vrednost za NoData piksele u izabranom kanalu
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u rasterskom sloju
NoData vrednost kanala	`NODATA_VALUE`	[numerički: decimalni]	Vrednost (ako je postavljena) NoData piksela u izabranom kanalu
Veličina piksela (visina) u jedinicama karte	`PIXEL_HEIGHT`	[numerički: ceo broj]	Vertikalna veličina piksela u jedinicama karte
Veličina piksela (širina) u jedinicama karte	`PIXEL_WIDTH`	[numerički: ceo broj]	Horizontalna veličina piksela u jedinicama karte
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj redova u rasterskom sloju
Maksimalna x koordinata	`X_MAX`	[numerički: decimalni]	
Minimalna x koordinata	`X_MIN`	[numerički: decimalni]	
Maksimalna y koordinata	`Y_MAX`	[numerički: decimalni]	
Minimalna y koordinata	`Y_MIN`	[numerički: decimalni]	

Python kod

ID algoritma: `native:rasterlayerproperties`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Statistika rasterskog sloja

Računa osnovnu statistiku iz vrednosti u zadatom kanalu rasterskog sloja. Izlaz se učitava u meniju *Obrada* ► *Pregledač rezultata*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal za koji želite statistiku.
Statistika	`OUTPUT_HTML_FILE`	[html] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Najveća vrednost	`MAX`	[numerički: decimalni]	
Srednja vrednost	`MEAN`	[numerički: decimalni]	
Najmanja vrednost	`MIN`	[numerički: decimalni]	
Statistika	`OUTPUT_HTML_FILE`	[html]	Izlazna datoteka sadrži sledeće informacije: • Analizirana datoteka: putanja rasterskog sloja • Minimalna vrednost: minimalna vrednost rastera • Maksimalna vrednost: maksimalna vrednost rastera • Raspon: razlika između maksimalne i minimalne vrednosti • Zbir: ukupan zbir vrednosti • Srednja vrednost: srednja vrednost vrednosti • Standardna devijacija: standardna devijacija vrednosti • Zbir kvadrata: zbir kvadrata odstupanja svakog posmatranja od ukupne srednje vrednosti
Opseg	`RANGE`	[numerički: decimalni]	
Standardna devijacija	`STD_DEV`	[numerički: decimalni]	

наставак на следећој страни

Табела 24.360 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zbir	`SUM`	[numerički: decimalni]	
Zbir kvadrata	`SUM_OF_SQUARES`	[numerički: decimalni]	

Python kod

ID algoritma: `native:rasterlayerstatistics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izveštaj o jedinstvenim vrednostima rasterskog sloja

Vraća broj i površinu svake jedinstvene vrednosti u zadatom rasterskom sloju. Računanje površine vrši se u jedinici površine CRS-a sloja.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal za koji želite statistiku.
Izveštaj o jedinstvenim vrednostima	`OUTPUT_HTML_FILE`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Tabela jedinstvenih vrednosti	`OUTPUT_TABLE`	[vektor: tabela] Podrazumevano: `[Skip output]`	Specifikacija tabele za jedinstvene vrednosti: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja

наставак на следећој страни

Табела 24.362 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Broj NoData piksela	`NODATA_PIXEL_COUNT`	[numerički: ceo broj]	Broj NoData piksela u izlaznom rasterskom sloju
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Izveštaj o jedinstvenim vrednostima	`OUTPUT_HTML_FILE`	[html]	Izlazna HTML datoteka sadrži sledeće informacije: • Analizirana datoteka: putanja rasterskog sloja • Obuhvat: xmin, ymin, xmax, ymax koordinate obuhvata • Projekcija: projekcija sloja • Širina u pikselima: broj kolona i veličina širine piksela • Visina u pikselima: broj redova i veličina širine piksela • Ukupan broj piksela: broj svih piksela • Broj NoData piksela: broj piksela sa NoData vrednošću
Tabela jedinstvenih vrednosti	`OUTPUT_TABLE`	[vektor: tabela]	Tabela sa tri kolone: • *value*: vrednost piksela • *count*: broj piksela sa ovom vrednošću • *m2* ili *deg2* ili *ft2* ili … : ukupna površina piksela sa ovom vrednošću. Naziv kolone zavisi od jedinice površine CRS-a sloja, a računanje se vrši u toj jedinici.
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:rasterlayeruniquevaluesreport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zonalna statistika rasterskog sloja

Računa statistiku za vrednosti rasterskog sloja, razvrstanu po zonama definisanim u drugom rasterskom sloju.

See also

Zonalna statistika

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal za koji želite da izračunate statistiku.
Sloj zona	`ZONES`	[raster]	Rasterski sloj koji definiše zone. Zone se daju susednim pikselima koji imaju istu vrednost piksela.
Broj kanala zona	`ZONES_BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji definiše zone
Statistika	`OUTPUT_TABLE`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog izveštaja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Referentni sloj Opciono	`REF_LAYER`	[nabrajanje] Podrazumevano: 0	Rasterski sloj koji se koristi za računanje centroida koji će se koristiti kao referenca pri određivanju zona u izlaznom sloju. Jedno od: • 0 — Ulazni sloj: zone se određuju uzorkovanjem vrednosti rasterskog sloja zona u centroidu svakog piksela izvornog rasterskog sloja • 1 — Sloj zona: ulazni rasterski sloj biće uzorkovan u centroidu svakog piksela rasterskog sloja zona

Izlazi

Oznaka	Назив	Tip	Opis
Identifikator autoriteta CRS-a	`CRS_AUTHID`	[niska]	Koordinatni referentni sistem izlaznog rasterskog sloja
Obuhvat	`EXTENT`	[niska]	Prostorni obuhvat izlaznog rasterskog sloja

наставак на следећој страни

Табела 24.365 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Visina u pikselima	`HEIGHT_IN_PIXELS`	[numerički: ceo broj]	Broj redova u izlaznom rasterskom sloju
Broj NoData piksela	`NODATA_PIXEL_COUNT`	[numerički: ceo broj]	Broj NoData piksela u izlaznom rasterskom sloju
Statistika	`OUTPUT_TABLE`	[vektor: tabela]	Izlazni sloj sadrži sledeće informacije **za svaku zonu**: • Površina: površina u kvadratnim rasterskim jedinicama u zoni; • Zbir: ukupan zbir vrednosti piksela u zoni; • Broj: broj piksela u zoni; • Min: minimalna vrednost piksela u zoni; • Max: maksimalna vrednost piksela u zoni; • Mean: srednja vrednost vrednosti piksela u zoni;
Ukupan broj piksela	`TOTAL_PIXEL_COUNT`	[numerički: ceo broj]	Broj piksela u izlaznom rasterskom sloju
Širina u pikselima	`WIDTH_IN_PIXELS`	[numerički: ceo broj]	Broj kolona u izlaznom rasterskom sloju

Python kod

ID algoritma: `native:rasterlayerzonalstats`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Minimum/maksimum rastera

Izdvaja minimalnu i maksimalnu vrednost piksela iz zadatog kanala rasterskog sloja. Ako više piksela deli istu minimalnu ili maksimalnu vrednost, samo jedan od njih biće uključen u izlaz.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Rasterski sloj iz kojeg se izdvajaju minimalna i maksimalna vrednost piksela.
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal rastera koji se analizira. Ako je raster višekanalni, zadajte broj kanala (počev od 1).
Izdvoj	`EXTREMA`	[nabrajanje] Podrazumevano: 0 (Minimum i maksimum)	Izaberite koje ekstremne vrednosti da izdvojite: • 0: Minimum i maksimum • 1: Samo minimum • 2: Samo maksimum

наставак на следећој страни

Табела 24.366 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[vektor: tačka]	Vektorski sloj sa tačkastim objektima na lokacijama minimalnih i/ili maksimalnih vrednosti piksela.

Python kod

ID algoritma: `native:rasterminmax`

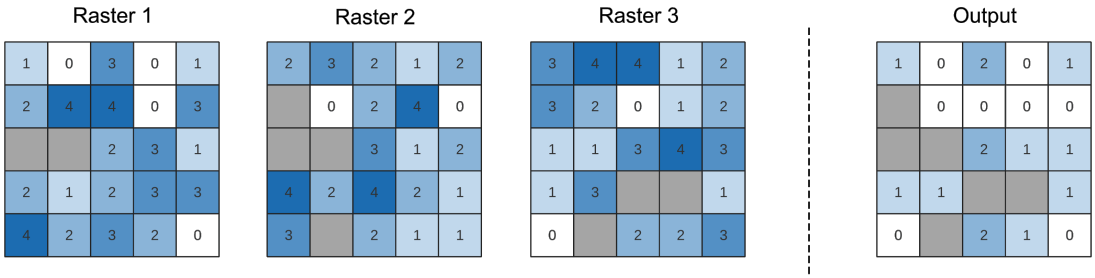
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rang rastera

`Dodato u 3.44`

Vrši analizu po ćeliji u kojoj izlazne vrednosti odgovaraju rangu sortirane liste preklapajućih vrednosti ćelija iz ulaznih slojeva. Izlazni raster biće višekanalni ako je zadato više rangova. Ako se u steku rastera podataka koriste višekanalni rasteri, algoritam će analizu uvek vršiti na prvom kanalu rastera.



Слика24.64: Rezultat ranga rastera za rang 1 (minimalna vrednost). NoData ćelije u bilo kom ulaznom rasteru daju NoData izlaznu ćeliju.

See also

Procentualni rang steka ćelija od vrednosti, *Percentil steka ćelija*, *Procentualni rang steka ćelija iz rasterskog sloja*, *Statistika ćelija*, *Rasterski kalkulator*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT_RASTERS`	[raster] [lista]	Ulazni rasterski sloj(evi) za koje se računa rang.
Rang	`RANKS`	[numerički: ceo broj] [lista] Podrazumevano: [1]	Lista rangova koji se računaju za svaki piksel u sortiranoj listi vrednosti ćelija iz ulaznih rastera. Ćelije čije vrednosti odgovaraju zadatom rangu(ovima) zadržavaju se u izlaznom sloju. Rangovi počinju od 1, što znači da je prvi rang 1. Podržane su i negativne vrednosti.
Rukovanje NoData vrednostima	`NODATA_HANDLING`	[nabrajanje] Podrazumevano: 0	Izaberite kako da se rukuje NoData vrednostima u ulaznom rasteru: • 0 — Isključi NoData iz listi vrednosti: NoData vrednosti neće biti uključene u listu vrednosti piksela pri računanju rangova. • 1 — Prisustvo NoData u listi vrednosti daje NoData izlazne ćelije: izlazni raster imaće NoData vrednosti na odgovarajućim lokacijama.
Rangirano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera koji sadrži vrednosti ranga. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazni obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Ako obuhvat nije zadat, koristiće se minimalni obuhvat koji pokriva sve izabrane referentne slojeve. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina izlazne ćelije Opciono	`CELL_SIZE`	[numerički: decimalni]	Veličina ćelije izlaznog rasterskog sloja. Ako veličina ćelije nije zadata, koristiće se minimalna veličina ćelije izabranog referentnog sloja(eva). Veličina ćelije biće ista za X i Y osu.
Izlazni CRS Opciono	`CRS`	[crs]	CRS izlaznog rasterskog sloja. Ako izlazni CRS nije zadat, koristiće se CRS prvog referentnog sloja.

Izlazi

Oznaka	Назив	Tip	Opis
Rangirano	`OUTPUT`	[raster]	Izlazni rasterski sloj koji sadrži rezultat.

Python kod

ID algoritma: `native:rasterrank`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zapremina površine rastera

Računa zapreminu ispod rasterske površine u odnosu na zadati bazni nivo. Ovo je uglavnom korisno za digitalne modele visina (DEM).

Parametri

Oznaka	Назив	Tip	Opis
ULAZNI sloj	`INPUT`	[raster]	Ulazni raster koji predstavlja površinu
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji definiše površinu.
Bazni nivo	`LEVEL`	[numerički: decimalni] Podrazumevano: 0.0	Definišite baznu ili referentnu vrednost. Ova baza se koristi u računanju zapremine prema parametru `Metod` (pogledajte ispod).
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Definišite metod za računanje zapremine dat razlikom između vrednosti piksela rastera i `Baznog nivoa`. Opcije: • 0 — Broj samo iznad baznog nivoa: zapremini se dodaju samo pikseli iznad baznog nivoa. • 1 — Broj samo ispod baznog nivoa: zapremini se dodaju samo pikseli ispod baznog nivoa. • 2 — Oduzmi zapremine ispod baznog nivoa: pikseli iznad baznog nivoa se dodaju zapremini, pikseli ispod baznog nivoa se oduzimaju od zapremine. • 3 — Dodaj zapremine ispod baznog nivoa: dodaje zapreminu bez obzira na to da li je piksel iznad ili ispod baznog nivoa. Ovo je ekvivalentno sabiranju apsolutnih vrednosti razlike između vrednosti piksela i baznog nivoa.
Izveštaj o zapremini površine	`OUTPUT_HTML_FILE`	[html] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog HTML izveštaja. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Tabela zapremine površine	`OUTPUT_TABLE`	[vektor: tabela] Podrazumevano: `[Skip output]`	Specifikacija izlazne tabele. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zapremina	`VOLUME`	[numerički: decimalni]	Izračunata zapremina
Površina	`AREA`	[numerički: decimalni]	Površina u kvadratnim jedinicama karte
Broj_piksela	`PIXEL_COUNT`	[numerički: ceo broj]	Ukupan broj piksela koji su analizirani
Izveštaj o zapremini površine	`OUTPUT_HTML_FILE`	[html]	Izlazni izveštaj (koji sadrži zapreminu, površinu i broj piksela) u HTML formatu
Tabela zapremine površine	`OUTPUT_TABLE`	[vektor: tabela]	Izlazna tabela (koja sadrži zapreminu, površinu i broj piksela)

Python kod

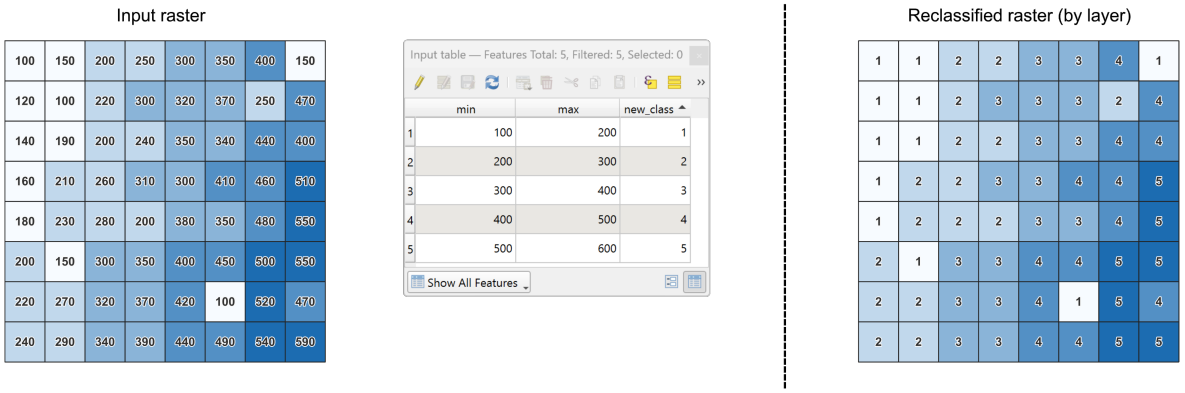
ID algoritma: `native:rastersurfacevolume`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Reklasifikuj po sloju

Reklasifikuje rasterski kanal dodeljivanjem novih vrednosti klasa na osnovu opsega zadatih u vektorskoj tabeli.



Слика24.65: Raster reklasifikovan po sloju (granice opsega: min <= vrednost < max).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Rasterski sloj za reklasifikaciju
Broj kanala	`RASTER_BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako je raster višekanalni, izaberite kanal koji želite da reklasifikujete.

наставак на следећој страни

Табела 24.373 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Sloj koji sadrži granice klasa	`INPUT_TABLE`	[vektor: bilo koji]	Vektorski sloj koji sadrži vrednosti za klasifikaciju.
Polje minimalne vrednosti klase	`MIN_FIELD`	[polje tabele: numerički]	Polje sa minimalnom vrednošću opsega za klasu. Koristite `-inf` za obuhvatanje najniže moguće vrednosti.
Polje maksimalne vrednosti klase	`MAX_FIELD`	[polje tabele: numerički]	Polje sa maksimalnom vrednošću opsega za klasu. Koristite `inf` za obuhvatanje najviše moguće vrednosti.
Polje izlazne vrednosti	`VALUE_FIELD`	[polje tabele: numerički]	Polje sa vrednošću koja se dodeljuje pikselima koji padaju u klasu (između odgovarajućih min i max vrednosti). Koristite `nan` da biste vrednost opsega postavili na NoData.
Reklasifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`NO_DATA`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se primenjuje na NoData vrednosti.
Granice opsega	`RANGE_BOUNDARIES`	[nabrajanje] Podrazumevano: 0	Definiše pravila poređenja za klasifikaciju. Opcije: • 0 — min < vrednost <= max • 1 — min <= vrednost < max • 2 — min <= vrednost <= max • 3 — min < vrednost < max
Koristi NoData kada nijedan opseg ne odgovara vrednosti	`NODATA_FOR_MISSING`	[logička vrednost] Podrazumevano: False	Primenjuje NoData vrednost na vrednosti kanala koje ne padaju ni u jednu klasu. Ako je False, zadržava se originalna vrednost.

наставак на следећој страни

Табела 24.374 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Reklasifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj sa reklasifikovanim vrednostima kanala

Python kod

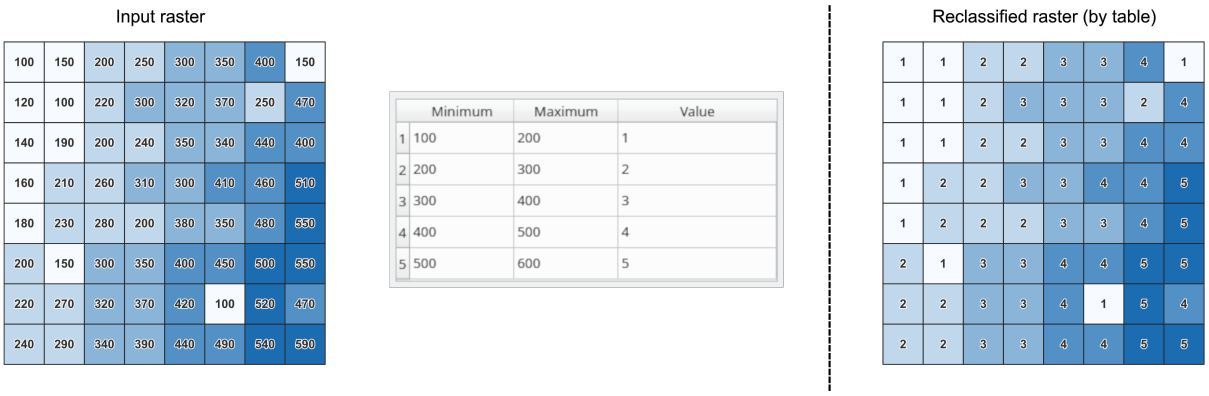
ID algoritma: `native:reclassifybylayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Reklasifikuj po tabeli

Reklasifikuje rasterski kanal dodeljivanjem novih vrednosti klasa na osnovu opsega zadatih u fiksnoj tabeli.



Слика24.66: Raster reklasifikovan po tabeli (granice opsega: min <= vrednost < max).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Rasterski sloj za reklasifikaciju
Broj kanala	`RASTER_BAND`	[rasterski kanal] Podrazumevano: 1	Rasterski kanal za koji želite ponovo da izračunate vrednosti.
Tabela reklasifikacije	`TABLE`	[vektor: tabela]	Tabela sa 3 kolone koju treba popuniti vrednostima za postavljanje granica svake klase (`Minimum` i `Maximum`) i novom `Value` koja se dodeljuje vrednostima kanala koje padaju u klasu. Vrednost `-inf` može se koristiti kao minimum, `inf` kao maksimum, a `nan` se može koristiti za postavljanje izlazne vrednosti na NoData.
Reklasifikovan raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izlazna NoData vrednost	`NO_DATA`	[numerički: decimalni] Podrazumevano: -9999.0	Vrednost koja se primenjuje na NoData vrednosti.
Granice opsega	`RANGE_BOUNDARIES`	[nabrajanje] Podrazumevano: 0	Definiše pravila poređenja za klasifikaciju. Opcije: • 0 — min < vrednost <= max • 1 — min <= vrednost < max • 2 — min <= vrednost <= max • 3 — min < vrednost < max

наставак на следећој страни

Табела 24.377 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Koristi NoData kada nijedan opseg ne odgovara vrednosti	`NODATA_FOR_MISSING`	[logička vrednost] Podrazumevano: False	Primenjuje NoData vrednost na vrednosti kanala koje ne padaju ni u jednu klasu. Ako je False, zadržava se originalna vrednost.
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Reklasifikovan raster	`OUTPUT`	[raster]	Izlazni rasterski sloj sa reklasifikovanim vrednostima kanala

Python kod

ID algoritma: `native:reclassifybytable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

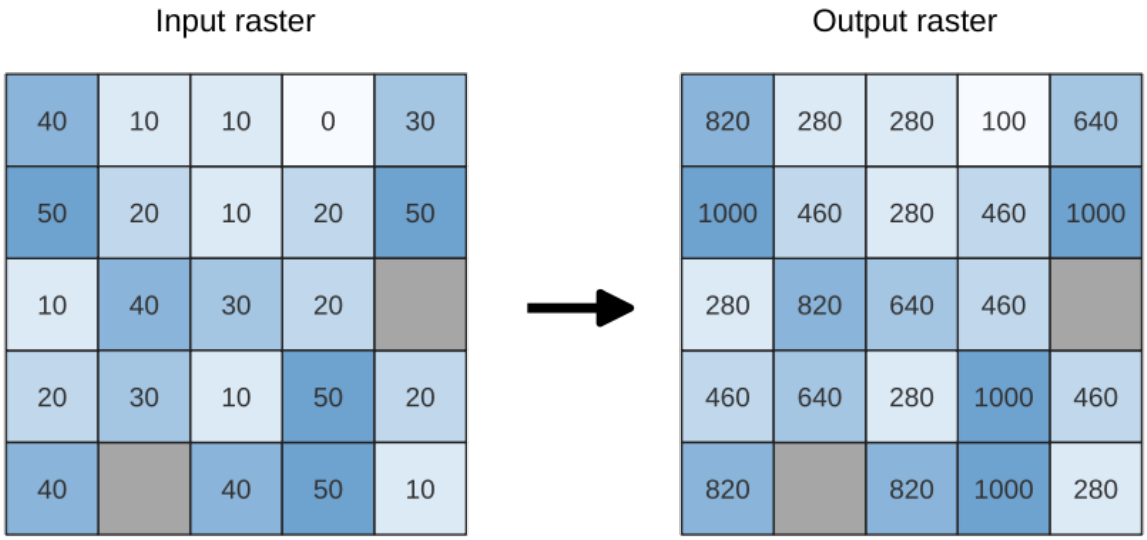
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preskaliraj raster

Preskalira rasterski sloj na novi opseg vrednosti, uz očuvanje oblika (raspodele) histograma rastera (vrednosti piksela). Ulazne vrednosti se mapiraju pomoću linearne interpolacije od minimalne i maksimalne vrednosti piksela izvornog rastera do odredišnog opsega minimalnih i maksimalnih piksela.

Podrazumevano algoritam čuva originalnu NoData vrednost, ali postoji opcija da se ona zameni.



Слика24.67: Preskaliranje vrednosti rasterskog sloja sa [0 - 50] na [100 - 1000]

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Rasterski sloj koji se koristi za preskaliranje
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal.
Nova minimalna vrednost	`MINIMUM`	[numerički: decimalni] Podrazumevana vrednost: 0.0	Minimalna vrednost piksela koja se koristi u preskaliranom sloju
Nova maksimalna vrednost	`MAXIMUM`	[numerički: decimalni] Podrazumevana vrednost: 255.0	Maksimalna vrednost piksela koja se koristi u preskaliranom sloju
Nova NoData vrednost Opciono	`NODATA`	[numerički: decimalni] Podrazumevana vrednost: Nije postavljeno	Vrednost koja se dodeljuje NoData pikselima. Ako nije postavljena, čuvaju se originalne NoData vrednosti.
Preskalirano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Preskalirano	`OUTPUT`	[raster]	Izlazni rasterski sloj sa preskaliranim vrednostima kanala

Python kod

ID algoritma: `native:rescaleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

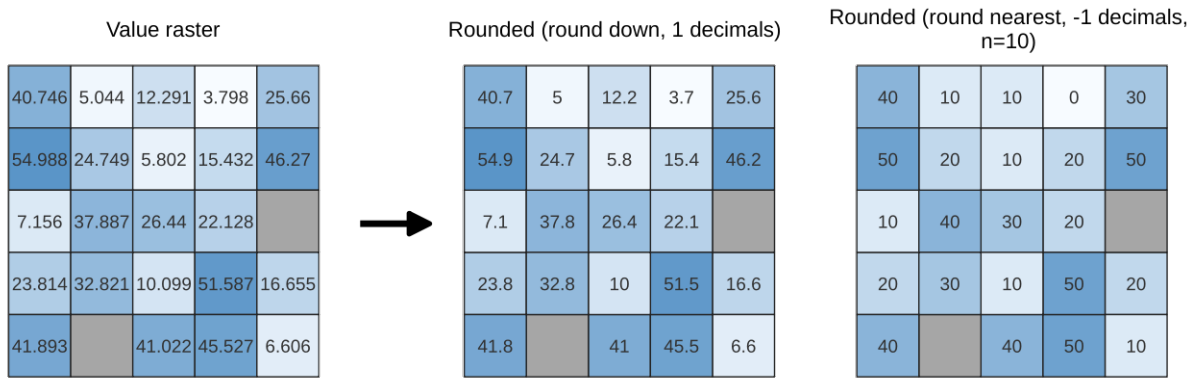
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zaokruži raster

Zaokružuje vrednosti ćelija rasterskog skupa podataka prema zadatom broju decimala.

Alternativno, negativan broj decimalnih mesta može se koristiti za zaokruživanje vrednosti na stepene baze n. Na primer, sa baznom vrednošću n od 10 i decimalnim mestima -1, algoritam zaokružuje vrednosti ćelija na višekratnike broja 10, -2 zaokružuje na višekratnike broja 100, i tako dalje. Mogu se izabrati proizvoljne bazne vrednosti, algoritam primenjuje isti multiplikativni princip. Zaokruživanje vrednosti ćelija na višekratnike baze n može se koristiti za generalizaciju rasterskih slojeva.

Algoritam čuva tip podataka ulaznog rastera. Zato se byte/celobrojni rasteri mogu zaokružiti samo na višekratnike baze n, u suprotnom se izbacuje upozorenje, a raster se kopira kao byte/celobrojni raster.



Слика24.68: Zaokruživanje vrednosti rastera

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Raster koji se obrađuje.
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal rastera
Pravac zaokruživanja	`ROUNDING_DIRECTION`	[lista] Podrazumevano: 1	Kako se bira ciljna zaokružena vrednost. Opcije su: • 0 — Zaokruži naviše • 1 — Zaokruži na najbliže • 2 — Zaokruži naniže
Broj decimalnih mesta	`DECIMAL_PLACES`	[numerički: ceo broj] Podrazumevano: 2	Broj decimalnih mesta na koji se zaokružuje. Koristite negativne vrednosti za zaokruživanje vrednosti ćelija na višekratnik baze n
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Baza n za zaokruživanje na višekratnike broja n	`BASE_N`	[numerički: ceo broj] Podrazumevano: 10	Kada je parametar `DECIMAL_PLACES` negativan, vrednosti rastera se zaokružuju na višekratnike bazne vrednosti n
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Izlazni rasterski sloj sa vrednostima zaokruženim za izabrani kanal.

Python kod

ID algoritma: `native:roundrastervalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uzorkuj vrednosti rastera

Izdvaja vrednosti rastera na lokacijama tačaka. Ako je rasterski sloj višekanalni, uzorkuje se svaki kanal.

Atributska tabela rezultujućeg sloja imaće onoliko novih kolona koliko rasterski sloj ima kanala.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Tačkasti vektorski sloj koji se koristi za uzorkovanje
Rasterski sloj	`RASTERCOPY`	[raster]	Rasterski sloj koji se uzorkuje na zadatim lokacijama tačaka.
Prefiks izlazne kolone	`COLUMN_PREFIX`	[niska] Podrazumevano: 'SAMPLE_'	Prefiks za nazive dodatih kolona.
Uzorkovano Opciono	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj koji sadrži uzorkovane vrednosti. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Uzorkovano	`OUTPUT`	[vektor: tačka]	Izlazni sloj koji sadrži uzorkovane vrednosti.

Python kod

ID algoritma: `native:rastersampling`

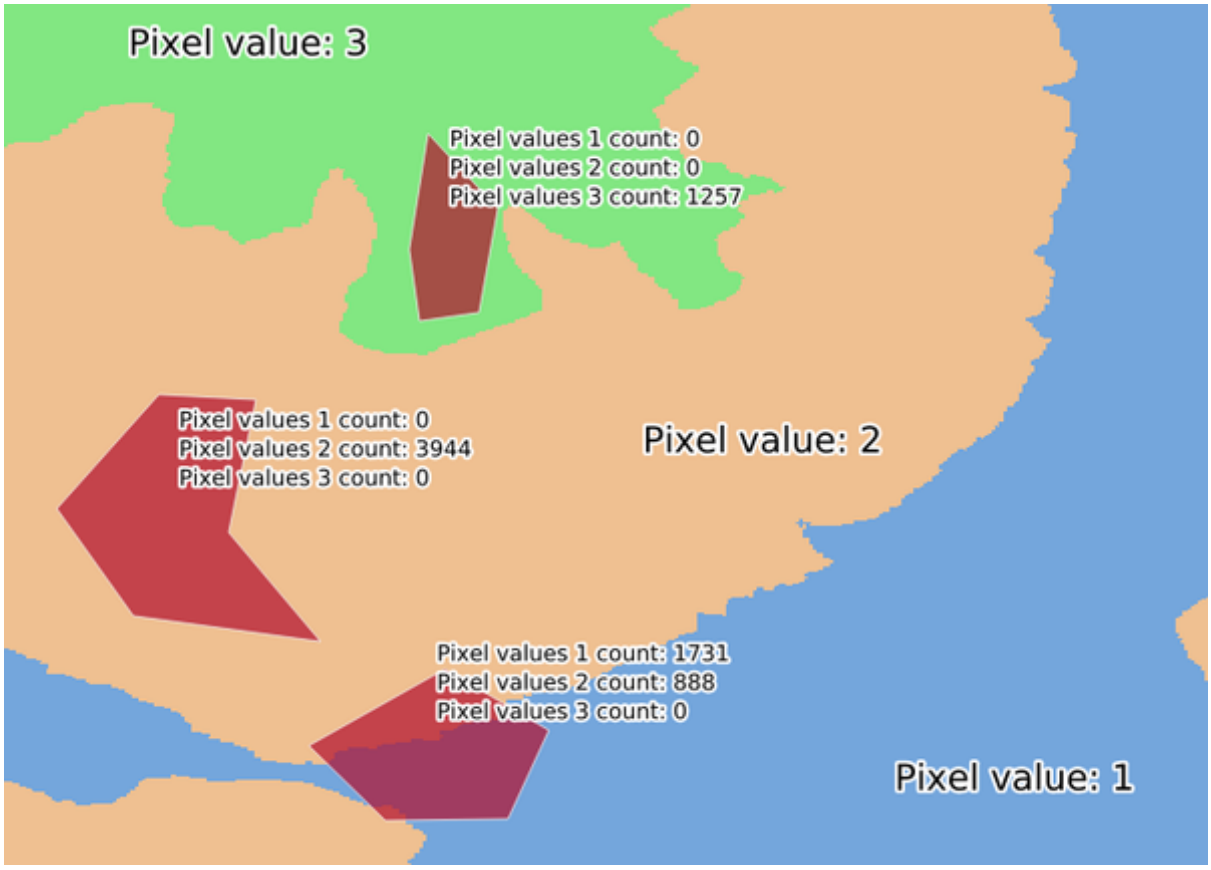
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zonalni histogram

Dodaje polja koja predstavljaju broj svake jedinstvene vrednosti iz rasterskog sloja sadržane unutar poligonskih objekata.

Atributska tabela izlaznog sloja imaće onoliko polja koliko ima jedinstvenih vrednosti rasterskog sloja koje se seku sa poligonom(ima).



Слика24.69: Primer histograma rasterskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj.
Broj kanala	`RASTER_BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal.
Vektorski sloj koji sadrži zone	`INPUT_VECTOR`	[vektor: poligon]	Vektorski poligonski sloj koji definiše zone.
Prefiks izlazne kolone	`COLUMN_PREFIX` Opciono	[niska] Podrazumevano: 'HISTO_'	Prefiks za nazive izlaznih kolona.
Izlazne zone	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski poligonski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazne zone	`OUTPUT`	[vektor: poligon]	Izlazni vektorski poligonski sloj.

Python kod

ID algoritma: `native:zonalhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zonalna tačka minimuma/maksimuma

`Dodato u 3.42`

Izdvaja tačkaste objekte koji odgovaraju minimalnim i maksimalnim vrednostima piksela unutar poligonskih zona.

Izlaz će sadržati po jedan tačkasti objekat za minimalnu i jedan za maksimalnu vrednost rastera za svaki pojedinačni zonalni objekat iz poligonskog sloja. Napravljeni tačkasti sloj biće u istom prostornom referentnom sistemu kao izabrani rasterski sloj.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Poligonski sloj koji definiše zone.
Rasterski sloj	`RASTER`	[raster]	Rasterski sloj iz kojeg se izdvajaju minimalne i maksimalne vrednosti.
Rasterski kanal	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal rasterskog sloja	Ako raster ima više kanala, izaberite kanal za obradu.
Zonalne ekstremne vrednosti	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zonalne ekstremne vrednosti	`OUTPUT`	[vektor: tačka]	Izlazni sloj koji sadrži tačke minimuma i maksimuma za svaku zonu.

Python kod

ID algoritma: `native:zonalminmaxpoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zonalna statistika

Računa statistiku rasterskog sloja za svaki objekat preklapajućeg poligonskog vektorskog sloja.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Vektorski poligonski sloj koji sadrži zone.
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj.
Rasterski kanal	`RASTER_BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal za statistiku.
Prefiks izlazne kolone	`COLUMN_PREFIX`	[niska] Podrazumevano: '_'	Prefiks za nazive izlaznih kolona.
Statistika koja se računa	`STATISTICS`	[nabrajanje] [lista] Podrazumevano: [0,1,2]	Lista statističkih operatora za izlaz. Opcije: • 0 — Broj • 1 — Zbir • 2 — Srednja vrednost • 3 — Medijana • 4 — St. dev. • 5 — Minimum • 6 — Maksimum • 7 — Raspon • 8 — Manjina • 9 — Većina • 10 — Raznolikost • 11 — Varijansa

наставак на следећој страни

Табела 24.387 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zonalna statistika	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski poligonski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zonalna statistika	`OUTPUT`	[vektor: poligon]	Vektorski sloj zona sa dodatom statistikom.

Python kod

ID algoritma: `native:zonalstatisticsfb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.19 Pravljenje rastera

Napravi konstantan rasterski sloj

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen navedenom vrednošću.

Dodatno se može zadati izlazni tip podataka. Algoritam će prekinuti rad ako je uneta vrednost koja se ne može predstaviti izabranim tipom podataka izlaznog rastera.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Konstantna vrednost	`NUMBER`	[numerički: decimalni] Podrazumevano: 1.0	Konstantna vrednost piksela za izlazni rasterski sloj.
Konstanta	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 5	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte • 1 — Integer16 • 2 — Unsigned Integer16 • 3 — Integer32 • 4 — Unsigned Integer32 • 5 — Float32 • 6 — Float64

наставак на следећој страни

Табела 24.390 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Konstanta	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat sa navedenom veličinom piksela i vrednošću.

Python kod

ID algoritma: `native:createconstantrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (binomna raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen binomno raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz N od 10 i verovatnoću od 0.5. Ovo se može promeniti pomoću naprednih parametara za N i verovatnoću. Tip podataka rastera postavljen je na celobrojne tipove (podrazumevano Integer16). Slučajne vrednosti binomne raspodele definisane su kao pozitivni celi brojevi. Raster u pokretnom zarezu predstavljaće konverziju celobrojnih vrednosti u pokretni zarez.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
N	`N`	[numerički: ceo broj] Podrazumevano: 10	
Verovatnoća	`PROBABILITY`	[numerički: decimalni] Podrazumevano: 0.5	

наставак на следећој страни

Табела 24.393 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajnim vrednostima

Python kod

ID algoritma: `native:createrandombinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (eksponencijalna raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen eksponencijalno raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz lambda od 1.0. Ovo se može promeniti pomoću naprednog parametra za lambda. Tip podataka rastera podrazumevano je postavljen na Float32 jer su slučajne vrednosti eksponencijalne raspodele brojevi u pokretnom zarezu.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Float32 • 1 — Float64
Lambda	`LAMBDA`	[numerički: decimalni] Podrazumevano: 1.0	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajnim vrednostima

Python kod

ID algoritma: `native:createrandomexponentialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (gama raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen gama raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz vrednosti alfa i beta od 1.0. Ovo se može promeniti pomoću naprednih parametara za alfa i beta. Tip podataka rastera podrazumevano je postavljen na Float32 jer su slučajne vrednosti gama raspodele brojevi u pokretnom zarezu.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja

наставак на следећој страни

Табела 24.398 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Float32 • 1 — Float64
Alfa	`ALPHA`	[numerički: decimalni] Podrazumevano: 1.0	
Beta	`BETA`	[numerički: decimalni] Podrazumevano: 1.0	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandomgammarasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (geometrijska raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen geometrijski raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz verovatnoću od 0.5. Ovo se može promeniti pomoću naprednog parametra za srednju vrednost. Tip podataka rastera postavljen je na celobrojne tipove (podrazumevano Integer16). Slučajne vrednosti geometrijske raspodele definisane su kao pozitivni celi brojevi. Raster u pokretnom zarezu predstavljaće konverziju celobrojnih vrednosti u pokretni zarez.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Verovatnoća	`PROBABILITY`	[numerički: decimalni] Podrazumevano: 0.5	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandomgeometricrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (negativna binomna raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen negativno binomno raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz parametar raspodele k od 10.0 i verovatnoću od 0.5. Ovo se može promeniti pomoću naprednih parametara za k i verovatnoću. Tip podataka rastera postavljen je na celobrojne tipove (podrazumevano Integer16). Slučajne vrednosti negativne binomne raspodele definisane su kao pozitivni celi brojevi. Raster u pokretnom zarezu predstavljaće konverziju celobrojnih vrednosti u pokretni zarez.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Parametar raspodele k	`K_PARAMETER`	[numerički: ceo broj] Podrazumevano: 10	
Verovatnoća	`PROBABILITY`	[numerički: decimalni] Podrazumevano: 0.5	

наставак на следећој страни

Табела 24.405 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandomnegativebinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (normalna raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen normalno raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz srednju vrednost od 0.0 i standardnu devijaciju od 1.0. Ovo se može promeniti pomoću naprednih parametara za srednju vrednost i standardnu devijaciju. Tip podataka rastera podrazumevano je postavljen na Float32 jer su slučajne vrednosti normalne raspodele brojevi u pokretnom zarezu.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Float32 • 1 — Float64
Srednja vrednost normalne raspodele	`MEAN`	[numerički: decimalni] Podrazumevano: 0.0	
Standardna devijacija normalne raspodele	`STDDEV`	[numerički: decimalni] Podrazumevano: 1.0	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandomnormalrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (Poasonova raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen Poasonovo raspodeljenim slučajnim vrednostima.

Podrazumevano se vrednosti biraju uz srednju vrednost od 1.0. Ovo se može promeniti pomoću naprednog parametra za srednju vrednost. Tip podataka rastera postavljen je na celobrojne tipove (podrazumevano Integer16). Slučajne vrednosti Poasonove raspodele su pozitivni celi brojevi. Raster u pokretnom zarezu predstavljaće konverziju celobrojnih vrednosti u pokretni zarez.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

наставак на следећој страни

Табела 24.410 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 0	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Srednja vrednost	`MEAN`	[numerički: decimalni] Podrazumevano: 1.0	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandompoissonrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi slučajan rasterski sloj (uniformna raspodela)

Generiše rasterski sloj za zadati obuhvat i veličinu ćelije, popunjen slučajnim vrednostima.

Podrazumevano će vrednosti biti u opsegu između minimalne i maksimalne vrednosti navedenog izlaznog tipa rastera. Ovo se može promeniti pomoću naprednih parametara za donju i gornju granicu. Ako granice imaju istu vrednost ili su obe nula (podrazumevano), algoritam će napraviti slučajne vrednosti u punom opsegu vrednosti izabranog tipa podataka rastera. Izbor granica izvan prihvatljivog opsega izlaznog tipa rastera prekinuće algoritam.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Željeni obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: CRS projekta	CRS izlaznog rasterskog sloja
Veličina piksela	`PIXEL_SIZE`	[numerički: decimalni] Podrazumevano: 1.0	Veličina piksela (X=Y) u jedinicama karte.
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip podataka izlaznog rastera	`OUTPUT_TYPE` Podrazumevano: 5	[nabrajanje]	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte • 1 — Integer16 • 2 — Unsigned Integer16 • 3 — Integer32 • 4 — Unsigned Integer32 • 5 — Float32 • 6 — Float64
Donja granica opsega slučajnih brojeva	`LOWER_BOUND`	[numerički: decimalni] Podrazumevano: 0.0	
Gornja granica opsega slučajnih brojeva	`UPPER_BOUND`	[numerički: decimalni] Podrazumevano: 0.0	
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Raster koji pokriva željeni obuhvat, sa veličinom ćelije, popunjen slučajno raspodeljenim vrednostima

Python kod

ID algoritma: `native:createrandomuniformrasterlayer`

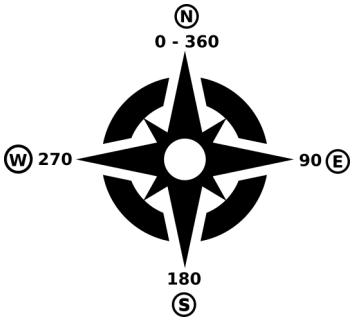
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.20 Analiza rasterskog terena

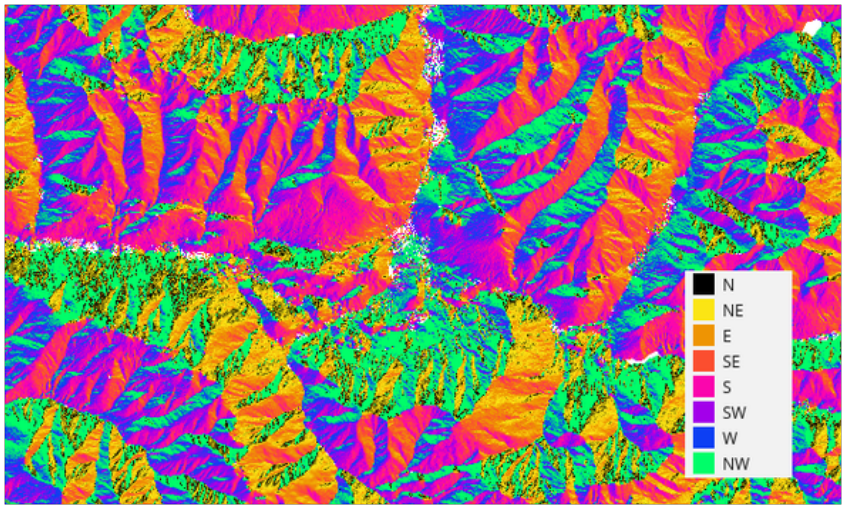
Ekspozicija

Računa ekspoziciju digitalnog modela terena na ulazu. Konačni rasterski sloj ekspozicije sadrži vrednosti od 0 do 360 koje izražavaju pravac nagiba, počev od severa (0°) i nastavljajući u smeru kazaljke na satu.



Слика24.70: Vrednosti ekspozicije

Sledeća slika prikazuje sloj ekspozicije reklasifikovan pomoću rampe boja:



Слика24.71: Reklasifikovan sloj ekspozicije

Parametri

Oznaka	Назив	Tip	Opis
Sloj visina	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Z faktor	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Vertikalno preuveličavanje. Ovaj parametar je koristan kada se Z jedinice razlikuju od X i Y jedinica, na primer stope i metri. Ovim parametrom to možete prilagoditi. Podrazumevano je 1 (bez preuveličavanja).
Ekspozicija	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj ekspozicije. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Ekspozicija	`OUTPUT`	[raster]	Izlazni rasterski sloj ekspozicije

Python kod

ID algoritma: `native:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

DMT filter (zasnovan na nagibu)

`Dodato u 3.34`

Može se koristiti za filtriranje digitalnog modela visina radi klasifikovanja njegovih ćelija na ćelije tla i objekata (koje nisu tlo).

Alat koristi koncepte kako ih je opisao Vosselman (2000) i zasniva se na pretpostavci da veliku razliku u visini između dve bliske ćelije verovatno ne izaziva strm nagib terena. Verovatnoća da viša ćelija ne pripada tlu raste kako se rastojanje između dve ćelije smanjuje. Zato filter definiše maksimalnu razliku visine (`dz_max`) između dve ćelije kao funkciju rastojanja (`d`) između ćelija (`dz_max( d ) = d`). Ćelija se klasifikuje kao teren ako unutar poluprečnika jezgra nema ćelije prema kojoj je razlika visine veća od dozvoljene maksimalne razlike visine na rastojanju između te dve ćelije.

Parametar približnog nagiba terena (`s`) koristi se za izmenu funkcije filtera radi usklađivanja sa ukupnim nagibom u oblasti proučavanja (`dz_max( d ) = d * s`). Interval poverenja od 5 % (`ci = 1.65 * sqrt( 2 * stddev )`) može se koristiti za dalju izmenu funkcije filtera, bilo ublažavanjem (`dz_max( d ) = d * s + ci`) ili pojačavanjem (`dz_max( d ) = d * s - ci`) kriterijuma filtera.

Reference: Vosselman, G. (2000): Slope based filtering of laser altimetry data. IAPRS, Vol. XXXIII, Part B3, Amsterdam, The Netherlands, 935-942

See also

Ovaj alat je port SAGA alatke DTM Filter (slope-based)

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Broj kanala	`BAND`	[rasterski kanal]	Kanal DEM-a koji se uzima u obzir
Poluprečnik jezgra (pikseli)	`RADIUS`	[numerički: ceo broj] Podrazumevano: 5	Poluprečnik jezgra filtera (u pikselima). Mora biti dovoljno velik da dosegne ćelije tla pored objekata koji nisu tlo.
Nagib terena (%, veličina piksela/vertikalne jedinice)	`TERRAIN_SLOPE`	[numerički: decimalni] Podrazumevano: 30.0	Približan nagib terena u `%`. Nagib terena mora se prilagoditi kako bi se uzeo u obzir odnos jedinica visine i dimenzija rasterskog piksela. Koristi se za ublažavanje kriterijuma filtera u strmijem terenu.

наставак на следећој страни

Табела 24.418 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izmena filtera	`FILTER_MODIFICATION`	[lista] Podrazumevano: 0	Izaberite da li se jezgro filtera primenjuje bez izmene ili se koristi interval poverenja za ublažavanje ili pojačavanje kriterijuma visine. • 0 - Nijedno • 1 - Ublaži filter • 2 - Pojačaj
Standardna devijacija	`STANDARD_DEVIATION`	[numerički: decimalni] Podrazumevano: 0.1	Standardna devijacija koja se koristi za računanje intervala poverenja od 5%, primenjenog na prag visine.
Izlazni sloj (tlo) Opciono	`OUTPUT_GROUND`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte filtriran DEM koji sadrži samo ćelije klasifikovane kao tlo. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Izlazni sloj (objekti koji nisu tlo) Opciono	`OUTPUT_NONGROUND`	[raster] Podrazumevano: `[Skip output]`	Zadajte objekte koji nisu tlo, uklonjene filterom. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREATE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj (tlo)	`OUTPUT_GROUND`	[raster]	Filtriran DEM koji sadrži samo ćelije klasifikovane kao tlo.
Izlazni sloj (objekti koji nisu tlo)	`OUTPUT_NONGROUND`	[raster]	Objekti koji nisu tlo, uklonjeni filterom.

Python kod

ID algoritma: `native:dtmslopebasedfilter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Popuni udubljenja (Wang & Liu)

`Dodato u 3.44`

Koristi metod koji su predložili Wang & Liu za prepoznavanje i popunjavanje depresija površine u digitalnim modelima visina.

Metod je unapređen da omogući pravljenje hidrološki ispravnih modela visina, tj. ne samo da popuni depresije, već i da očuva nizbrdi nagib duž toka. Ako se želi, to se postiže očuvanjem minimalnog gradijenta nagiba (a time i razlike u visini) između ćelija.

Reference: 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.

See also

Ovaj alat je port SAGA alatke Fill Sinks (Wang & Liu).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal DEM-a koji se uzima u obzir
Minimalni nagib (stepeni)	`MIN_SLOPE`	[numerički: decimalni] Podrazumevano: 0.10	Minimalni gradijent nagiba koji se očuva od ćelije do ćelije; sa vrednošću nula, udubljenja se popunjavaju do visine preliva (što daje ravne oblasti).
Izlazni sloj (popunjeni DEM) Opciono	`OUTPUT_FILLED_DEM`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni raster koji odgovara DEM-u bez depresija. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Izlazni sloj (pravci toka) Opciono	`OUTPUT_FLOW_DIRECTI`	[raster] Podrazumevano: `[Skip output]`	Zadajte izlazni raster sa izračunatim pravcima toka. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Izlazni sloj (slivovi) Opciono	`OUTPUT_WATERSHED_BA`	[raster] Podrazumevano: `[Skip output]`	Zadajte izlazni raster koji odgovara ocrtanim slivovima. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono	CREATI-ON_OPTIONS	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (|).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj (popunjeni DEM)	OUT-PUT_FILLED_DEM	[raster]	Izlazni raster koji odgovara digitalnom modelu visina bez depresija.
Izlazni sloj (pravci toka)	OUT-PUT_FLOW_DIRECTI	[raster]	Izlazni raster sa izračunatim pravcima toka; 0=S, 1=SI, 2=I, … 7=SZ.
Izlazni sloj (slivovi)	OUT-PUT_WATERSHED_BA	[raster]	Izlazni raster koji odgovara ocrtanim slivovima.

Python kod

ID algoritma: `native:fillsinkswangliu`

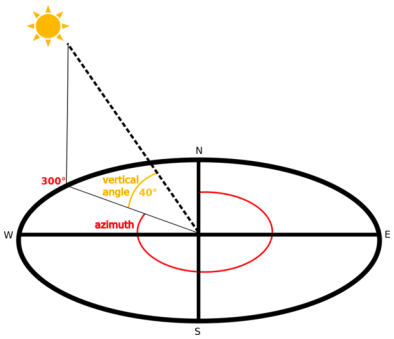
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Osenčeni reljef

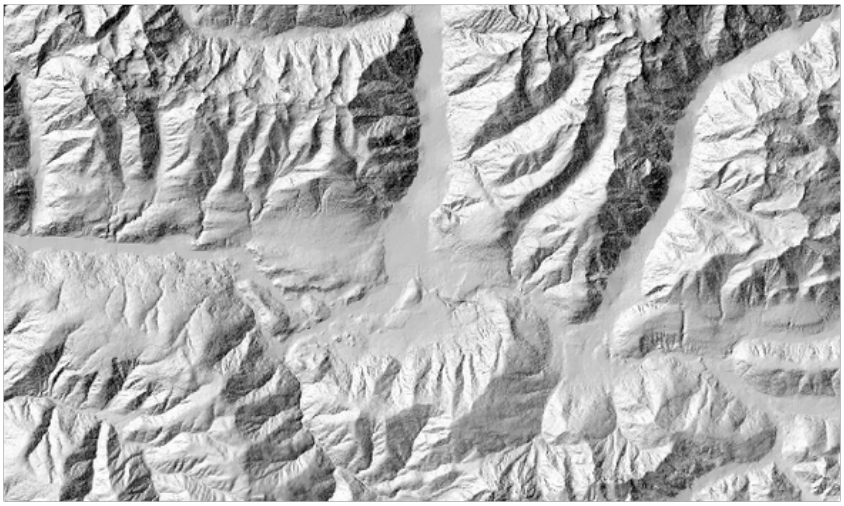
Računa rasterski sloj osenčenog reljefa za zadati ulazni digitalni model terena.

Senčenje sloja se računa prema položaju sunca: imate mogućnost da promenite i horizontalni ugao (azimut) i vertikalni ugao (visina sunca) sunca.



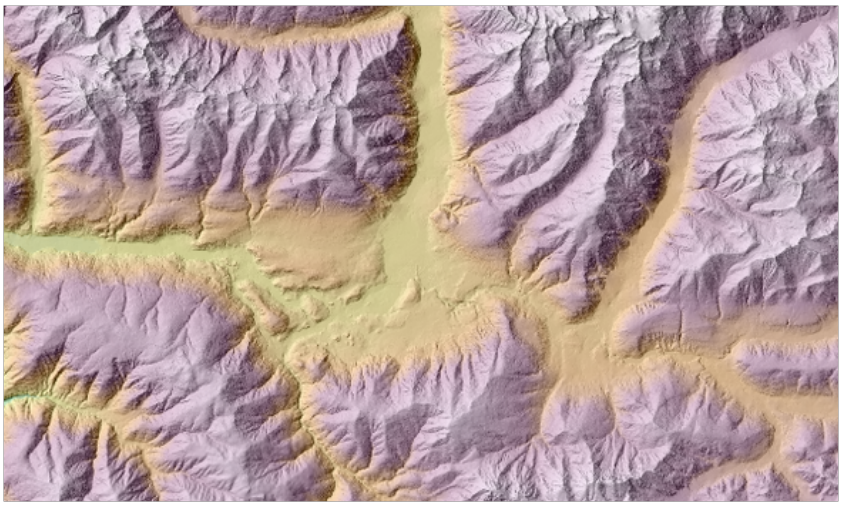
Слика24.72: Azimut i vertikalni ugao

Sloj osenčenog reljefa sadrži vrednosti od 0 (potpuna senka) do 255 (potpuno sunce). Osenčeni reljef se obično koristi za bolje razumevanje reljefa oblasti.



Слика24.73: Sloj osenčenog reljefa sa azimutom 300 i vertikalnim uglom 45

Posebno je zanimljivo dodeliti sloju osenčenog reljefa vrednost prozirnosti i preklopiti ga sa visinskim rasterom:



Слика24.74: Preklapanje osenčenog reljefa sa slojem visina

Parametri

Oznaka	Назив	Tip	Opis
Sloj visina	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Z faktor	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Vertikalno preuveličavanje. Ovaj parametar je koristan kada se Z jedinice razlikuju od X i Y jedinica, na primer stope i metri. Ovim parametrom to možete prilagoditi. Povećanje vrednosti ovog parametra preuveličaće konačni rezultat (čineći ga „brdovitijim“). Podrazumevano je 1 (bez preuveličavanja).

наставак на следећој страни

Табела 24.424 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Azimut (horizontalni ugao)	`AZIMUTH`	[numerički: decimalni] Podrazumevano: 300.0	Postavlja horizontalni ugao (u stepenima) sunca (u smeru kazaljke na satu). Opseg: 0 do 360. 0 je sever.
Vertikalni ugao	`V_ANGLE`	[numerički: decimalni] Podrazumevano: 40.0	Postavlja vertikalni ugao (u stepenima) sunca, to jest visinu sunca. Vrednosti mogu biti od 0 (minimalna visina) do 90 (maksimalna visina).
Osenčeni reljef	`OUTPUT`	[raster] Podrazumevano: `Save to temporary file`	Zadajte izlazni rasterski sloj osenčenog reljefa. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Osenčeni reljef	`OUTPUT`	[raster]	Izlazni rasterski sloj osenčenog reljefa

Python kod

ID algoritma: `native:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Hipsometrijske krive

Računa hipsometrijske krive za ulazni digitalni model visina. Krive se prave kao CSV datoteke u izlaznom folderu koji zadaje korisnik.

Hipsometrijska kriva je kumulativni histogram vrednosti visina u geografskoj oblasti.

Hipsometrijske krive možete koristiti za otkrivanje razlika u predelu usled geomorfologije teritorije.

Parametri

Oznaka	Назив	Tip	Opis
DEM za analizu	`INPUT_DEM`	[raster]	Rasterski sloj digitalnog modela terena koji se koristi za računanje nadmorskih visina
Sloj granica	`BOUNDARY_LAYER`	[vektor: poligon]	Poligonski vektorski sloj sa granicama oblasti koje se koriste za računanje hipsometrijskih krivih
Korak	`STEP`	[numerički: decimalni] Podrazumevano: 100.0	Vertikalno rastojanje između krivih

наставак на следећој страни

Табела 24.426 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Koristi % površine umesto apsolutne vrednosti	`USE_PERCENTAGE`	[logička vrednost] Podrazumevano: False	Upiši procenat površine u polje „Area“ CSV datoteke umesto apsolutne površine
Hipsometrijske krive	`OUTPUT_DIRECTORY`	[folder]	Zadajte izlazni folder za hipsometrijske krive. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Izlazi

Oznaka	Назив	Tip	Opis
Hipsometrijske krive	`OUTPUT_DIRECTORY`	[folder]	Direktorijum koji sadrži datoteke sa hipsometrijskim krivama. Za svaki objekat iz ulaznog vektorskog sloja biće napravljena CSV datoteka sa vrednostima površine i nadmorske visine. Nazivi datoteka počinju sa `histogram_`, praćeni nazivom sloja i ID-jem objekta.

	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

Python kod

ID algoritma: `qgis:hypsometriccurves`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Reljef

Pravi sloj osenčenog reljefa iz digitalnih podataka o visinama. Boju reljefa možete zadati ručno, ili možete pustiti algoritam da automatski izabere sve klase reljefa.



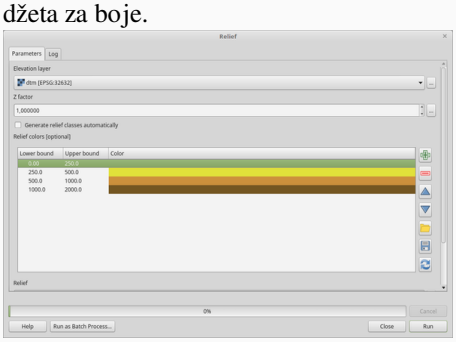
Слика24.75: Sloj reljefa

Parametri

Oznaka	Назив	Tip	Opis
Sloj visina	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Z faktor	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Vertikalno preuveličavanje. Ovaj parametar je koristan kada se Z jedinice razlikuju od X i Y jedinica, na primer stope i metri. Ovim parametrom to možete prilagoditi. Povećanje vrednosti ovog parametra preuveličaće konačni rezultat (čineći ga „brdovitijim“). Podrazumevano je 1 (bez preuveličavanja).
Generiši klase reljefa automatski	`AUTO_COLORS`	[logička vrednost] Podrazumevano: False	Ako uključite ovu opciju, algoritam će automatski napraviti sve klase boja reljefa

наставак на следећој страни

Табела 24.428 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Boje reljefa Opciono	`COLORS`	[vidžet tabele]	Koristite vidžet tabele ako želite ručno da izaberete boje reljefa. Možete dodati koliko god klasa boja želite: za svaku klasu možete izabrati donju i gornju granicu i, klikom na red boje, izabrati boju pomoću vidžeta za boje.  Слика24.76: Ručno postavljanje klasa boja reljefa Dugmad u desnom bočnom panelu daju vam mogućnost da: dodate ili uklonite klase boja, promenite redosled već definisanih klasa boja, otvorite postojeću datoteku sa klasama boja i sačuvate trenutne klase kao datoteku.
Reljef	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj reljefa. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…
Raspodela učestalosti Opciono	`FREQUENCY_DISTRIBUTION`	[vektor: tabela] Podrazumevano: `[Skip output]`	Zadajte CSV tabelu za izlaznu raspodelu učestalosti. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Reljef	`OUTPUT`	[raster]	Izlazni rasterski sloj reljefa
Raspodela učestalosti	`OUTPUT`	[vektor: tabela]	Izlazna raspodela učestalosti

Python kod

ID algoritma: `qgis:relief`

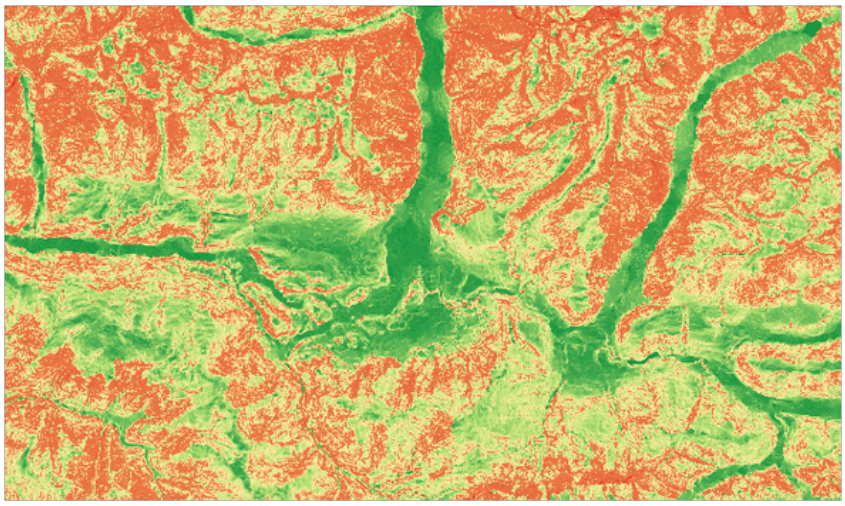
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Indeks razuđenosti

Računa kvantitativnu meru heterogenosti terena koju su opisali Riley i saradnici (1999). Računa se za svaku lokaciju sabiranjem promene visine unutar mreže piksela 3x3.

Svaki piksel sadrži razliku u visini između centralne ćelije i 8 ćelija koje je okružuju.



Слика24.77: Sloj razuđenosti od niskih (crveno) do visokih vrednosti (zeleno)

Parametri

Oznaka	Назив	Tip	Opis
Sloj visina	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Z faktor	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Vertikalno preuveličavanje. Ovaj parametar je koristan kada se Z jedinice razlikuju od X i Y jedinica, na primer stope i metri. Ovim parametrom to možete prilagoditi. Povećanje vrednosti ovog parametra preuveličaće konačni rezultat (čineći ga razuđenijim). Podrazumevano je 1 (bez preuveličavanja).
Razuđenost	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj razuđenosti. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Razuđenost	`OUTPUT`	[raster]	Izlazni rasterski sloj razuđenosti

Python kod

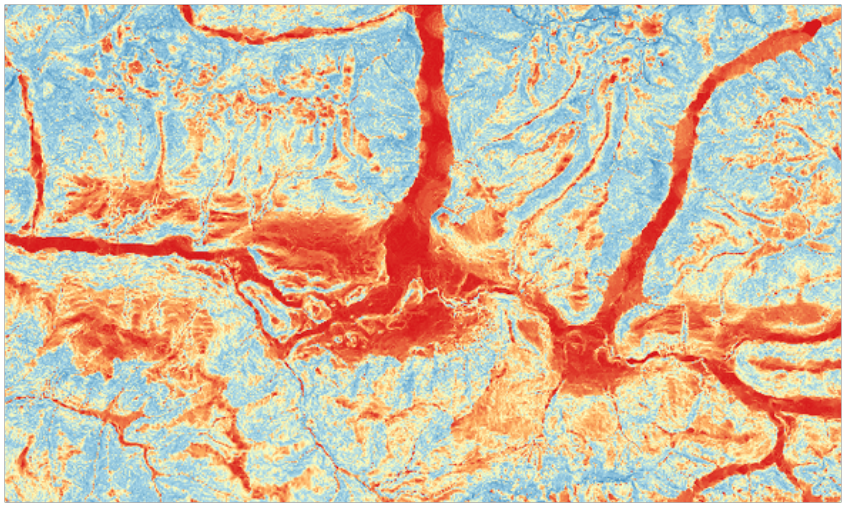
ID algoritma: `native:ruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Nagib

Računa nagib iz ulaznog rasterskog sloja. Nagib je ugao nagnutosti terena i izražava se u **stepenima**.



Слика24.78: Ravne oblasti crveno, strme oblasti plavo

Parametri

Oznaka	Назив	Tip	Opis
Sloj visina	`INPUT`	[raster]	Rasterski sloj digitalnog modela terena
Z faktor	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Vertikalno preuveličavanje. Ovaj parametar je koristan kada se Z jedinice razlikuju od X i Y jedinica, na primer stope i metri. Ovim parametrom to možete prilagoditi. Povećanje vrednosti ovog parametra preuveličaće konačni rezultat (čineći ga strmijim). Podrazumevano je 1 (bez preuveličavanja).
Nagib	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj nagiba. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Nagib	`OUTPUT`	[raster]	Izlazni rasterski sloj nagiba

Python kod

ID algoritma: `native:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.21 Rasterski alati

Poravnaj raster

Poravnava raster preuzorkovanjem na istu veličinu ćelije i reprojektovanjem u isti CRS kao referentni raster.

Упозорење

Ovaj algoritam je dostupan SAMO u kontekstu *Dizajnera modela*. Za druge kontekste koristite *Poravnaj rastere*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj koji se poravnava
Metod preuzorkovanja	`RESAMPLING_METHOD`	[nabrajanje] Podrazumevano: 0	Metod koji se koristi za preuzorkovanje ulaznog sloja. `Dostupne opcije` su: • 0 — Najbliži sused • 1 — Bilinearni (2x2 jezgro) • 2 — Kubni (4x4 jezgro) • 3 — Kubni B-splajn (4x4 jezgro) • 4 — Lancoš (6x6 jezgro) • 5 — Prosek • 6 — Mod • 7 — Maksimum • 8 — Minimum • 9 — Medijana • 10 — Prvi kvartil (Q1) • 11 — Treći kvartil (Q3)
Skaliraj vrednosti prema veličini ćelije	`RESCALE`	[logička vrednost] Podrazumevano: Ne	
Referentni sloj	`REFERENCE_LAYER`	[raster]	Rasterski sloj koji se koristi za preuzimanje obuhvata, veličine ćelije i CRS-a koji se primenjuju na ulazne slojeve.
Zameni referentni CRS Opciono	`CRS`	[crs]	CRS koji se koristi umesto CRS-a referentnog sloja

наставак на следећој страни

Табела 24.434 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zameni referentnu veličinu ćelije X Opciono	`CELL_SIZE_X`	[numerički: decimalni]	Veličina ćelije u pravcu X koja se koristi umesto veličine referentnog sloja
Zameni referentnu veličinu ćelije Y Opciono	`CELL_SIZE_Y`	[numerički: decimalni]	Veličina ćelije u pravcu Y koja se koristi umesto veličine referentnog sloja
Zameni referentni pomeraj mreže X Opciono	`GRID_OFFSET_X`	[numerički: decimalni]	Pomeraj u pravcu X koji se primenjuje na mrežu ćelija
Zameni referentni pomeraj mreže Y Opciono	`GRID_OFFSET_Y`	[numerički: decimalni]	Pomeraj u pravcu Y koji se primenjuje na mrežu ćelija
Iseci na obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Poravnati raster	`OUTPUT`	[raster]	izlazni raster sa preuzorkovanim pikselima

Izlazi

Oznaka	Назив	Tip	Opis
Poravnati raster	`OUTPUT`	[raster]	izlazni raster sa preuzorkovanim pikselima

Python kod

ID algoritma: `native:alignsingleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poravnaj rastere

Poravnava rastere preuzorkovanjem na istu veličinu ćelije i reprojektovanjem u isti CRS kao referentni raster.

Упозорење

Ovaj algoritam NIJE dostupan u kontekstu *Dizajnera modela*. Umesto njega koristite *Poravnaj raster*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Lista ulaznih rasterskih slojeva sa pridruženim opcijama preuzorkovanja (popunjava se kao stavka `QgsProcessingParameterAlignRasterLayers` — u grafičkom sučelju se to radi pritiskom na dugme *Configure Raster…* za svaki izabrani sloj): **Ulazni sloj [niska] (`inputFile`)** Puna putanja ulaznog sloja koji se poravnava **Izlazna datoteka [niska] (`outputFile`)** Puna putanja odgovarajućeg poravnatog izlaznog sloja **Metod preuzorkovanja [nabrajanje] (`resampleMethod`)** Metod koji se koristi za preuzorkovanje ulaznog sloja. `Dostupne opcije` su: • 0 — Najbliži sused • 1 — Bilinearni (2x2 jezgro) • 2 — Kubni (4x4 jezgro) • 3 — Kubni B-splajn (4x4 jezgro) • 4 — Lancoš (6x6 jezgro) • 5 — Prosek • 6 — Mod • 7 — Maksimum • 8 — Minimum • 9 — Medijana • 10 — Prvi kvartil (Q1) • 11 — Treći kvartil (Q3) **Skaliraj vrednosti prema veličini ćelije** [logička vrednost] (`rescale`)
Referentni sloj	`REFERENCE_LAYER`	[raster]	Rasterski sloj koji se koristi za preuzimanje obuhvata, veličine ćelije i CRS-a koji se primenjuju na ulazne slojeve.
Zameni referentni CRS Opciono	`CRS`	[crs]	CRS koji se koristi umesto CRS-a referentnog sloja
Zameni referentnu veličinu ćelije X Opciono	`CELL_SIZE_X`	[numerički: decimalni]	Veličina ćelije u pravcu X koja se koristi umesto veličine referentnog sloja

наставак на следећој страни

Табела 24.436 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zameni referentnu veličinu ćelije Y Opciono	`CELL_SIZE_Y`	[numerički: decimalni]	Veličina ćelije u pravcu Y koja se koristi umesto veličine referentnog sloja
Zameni referentni pomeraj mreže X Opciono	`GRID_OFFSET_X`	[numerički: decimalni]	Pomeraj u pravcu X koji se primenjuje na mrežu ćelija
Zameni referentni pomeraj mreže Y Opciono	`GRID_OFFSET_Y`	[numerički: decimalni]	Pomeraj u pravcu Y koji se primenjuje na mrežu ćelija
Iseci na obuhvat Opciono	`EXTENT`	[obuhvat]	Zadajte obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`

Izlazi

Oznaka	Назив	Tip	Opis
Poravnati rasteri	`OUTPUT_LAYERS`	[raster] [lista]	izlazni rasteri sa preuzorkovanim pikselima

Python kod

ID algoritma: `native:alignrasters`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvori kartu u raster

Pravi rastersku sliku sadržaja platna karte.

Može se izabrati *tema karte* za iscrtavanje unapred određenog skupa slojeva sa definisanim stilom za svaki sloj.

Alternativno, može se izabrati jedan sloj ako nije zadata tema karte.

Ako nisu zadati ni tema karte ni sloj, iscrtaće se trenutni sadržaj karte. Uneti minimalni obuhvat će interno biti proširen na umnožak veličine pločice.

Parametri

Oznaka	Назив	Tip	Opis
Minimalni obuhvat za iscrtavanje (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Zadajte obuhvat izlaznog rasterskog sloja. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Veličina pločice	`TILE_SIZE`	[numerički: ceo broj] Podrazumevano: 1024	Veličina pločice izlaznog rasterskog sloja. Minimalna vrednost: 64.
Jedinica karte po pikselu	`MAP_UNITS_PER_PI`	[numerički: decimalni] Podrazumevano: 100.0	Veličina piksela (u jedinicama karte). Minimalna vrednost: 0.0
Učini pozadinu prozirnom	`MAKE_BACKGROUND_TF`	[logička vrednost] Podrazumevano: False	Omogućava izvoz karte sa prozirnom pozadinom. Ako je postavljeno na `True`, daje RGBA sliku (umesto RGB).
Tema karte za iscrtavanje Opciono	`MAP_THEME`	[nabrajanje]	Koristi postojeću *temu karte* za iscrtavanje.
Jedan sloj za iscrtavanje Opciono	`LAYER`	[nabrajanje]	Izaberite jedan sloj za iscrtavanje
Izlazni sloj	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[raster]	Izlazni rasterski sloj

Python kod

ID algoritma: `native:rasterize`

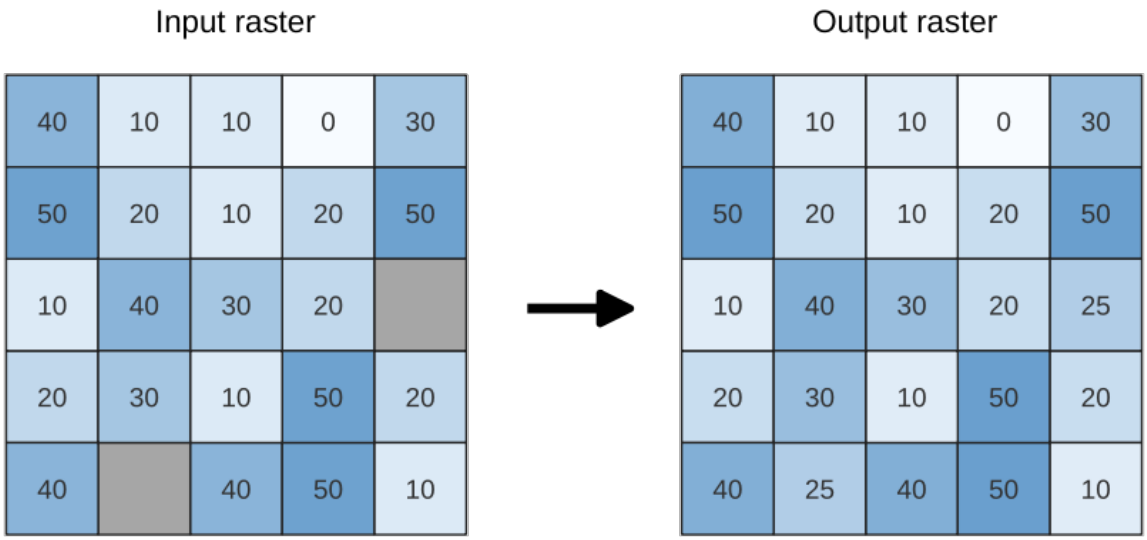
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Popuni NoData ćelije

Postavlja NoData vrednosti u ulaznom rasteru na izabranu vrednost, dajući rasterski skup podataka bez NoData piksela.

Algoritam poštuje tip podataka ulaznog rastera, npr. vrednost popune u pokretnom zarezu biće skraćena kada se primeni na celobrojni raster.



Слика24.79: Popunjavanje NoData vrednosti (sivo) u rasteru

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni raster	`INPUT`	[raster]	Raster koji se obrađuje.
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal rastera
Vrednost popune	`FILL_VALUE`	[numerički: decimalni] Podrazumevano: 1.0	Zadajte vrednost koja se koristi za NoData piksele
Izlazni raster	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Opcije kreiranja Opciono `Dodato u 3.40`	`CREATI-ON_OPTIONS` (za QGIS <= 3.42 ovo je bilo `CREA-TE_OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni raster	`OUTPUT`	[raster]	Izlazni rasterski sloj sa popunjenim ćelijama podataka.

Python kod

ID algoritma: `native:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi XYZ pločice (direktorijum)

Pravi rasterske „XYZ“ pločice koristeći trenutni QGIS projekat kao pojedinačne slike u strukturi direktorijuma.

Opciono se može napraviti izlazna Leaflet HTML datoteka koja koristi napravljene pločice kao sloj karte.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Zadajte obuhvat pločica. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Minimalno uvećanje	`ZOOM_MIN`	[numerički: ceo broj] Podrazumevano: 12	Minimum 0, maksimum 25.
Maksimalno uvećanje	`ZOOM_MAX`	[numerički: ceo broj] Podrazumevano: 12	Minimum 0, maksimum 25.
DPI	`DPI`	[numerički: ceo broj] Podrazumevano: 96	Minimum 48, maksimum 600.
Boja pozadine Opciono	`BACKGROUND_COLOR`	[boja] Podrazumevano: QColor(0, 0, 0, 0)	Izaberite boju pozadine za pločice
Omogući antialijasing	`ANTIALIAS`	[logička vrednost] Podrazumevano: True	Određuje da li se omogućava antialijasing
Format pločice	`TILE_FORMAT`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — PNG • 1 — JPG
Kvalitet (samo JPG) Opciono	`QUALITY`	[numerički: ceo broj] Podrazumevano: 75	Minimum 1, maksimum 100.
Veličina metapločice Opciono	`METATILESIZE`	[numerički: ceo broj] Podrazumevano: 4	Zadajte prilagođenu veličinu metapločice pri pravljenju XYZ pločica. Veće vrednosti mogu ubrzati iscrtavanje pločica i dati bolje natpise (manje praznina bez natpisa) uz veću potrošnju memorije. Minimum 1, maksimum 20.
Širina pločice Opciono	`TILE_WIDTH`	[numerički: ceo broj] Podrazumevano: 256	Minimum 1, maksimum 4096.
Visina pločice Opciono	`TILE_HEIGHT`	[numerički: ceo broj] Podrazumevano: 256	Minimum 1, maksimum 4096.
Koristi obrnutu Y osu pločica (TMS konvencije)	`TMS_CONVENTION`	[logička vrednost] Podrazumevano: False	

наставак на следећој страни

Табела 24.443 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT_DIRECTORY`	[folder] Podrazumevano: `[Save to temporary folder]`	Specifikacija izlaznog direktorijuma (za pločice). *Jedno od*: • Preskoči izlaz • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum
Izlazni html (Leaflet) Opciono	`OUTPUT_HTML`	[html] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne HTML datoteke. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Naslov Leaflet HTML izlaza Opciono `Dodato u 3.30`	`HTML_TITLE`	[niska] Podrazumevano: Nije postavljeno	HTML <title> oznaka koja se koristi za izlaznu Leaflet HTML datoteku.
Atribucija Leaflet HTML izlaza Opciono `Dodato u 3.30`	`HTML_ATTRIBUTION`	[niska] Podrazumevano: Nije postavljeno	Prilagođena atribucija karte koja se koristi za izlaznu Leaflet HTML datoteku. Moguće su HTML veze.
Uključi OpenStreetMap podlogu u Leaflet HTML izlaz `Dodato u 3.30`	`HTML_OSM`	[logička vrednost] Podrazumevano: False	Sloj OpenStreetMap podloge (izvor: https://tile.openstreetmap.org) uključen je u izlaznu Leaflet HTML datoteku. Odgovarajuća atribucija karte se dodaje automatski.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT_DIRECTORY`	[folder]	Izlazni direktorijum (za pločice)
Izlazni html (Leaflet)	`OUTPUT_HTML`	[html]	Izlazna HTML (Leaflet) datoteka

Python kod

ID algoritma: `native:tilesxyzdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi XYZ pločice (MBTiles)

Pravi rasterske „XYZ“ pločice koristeći trenutni QGIS projekat kao jednu datoteku u „MBTiles“ formatu.

Parametri

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Zadajte obuhvat pločica. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Minimalno uvećanje	`ZOOM_MIN`	[numerički: ceo broj] Podrazumevano: 12	Minimum 0, maksimum 25.
Maksimalno uvećanje	`ZOOM_MAX`	[numerički: ceo broj] Podrazumevano: 12	Minimum 0, maksimum 25.
DPI	`DPI`	[numerički: ceo broj] Podrazumevano: 96	Minimum 48, maksimum 600.
Boja pozadine Opciono	`BACKGROUND_COLOR`	[boja] Podrazumevano: QColor(0, 0, 0, 0)	Izaberite boju pozadine za pločice
Omogući antialijasing	`ANTIALIAS`	[logička vrednost] Podrazumevano: True	Određuje da li se omogućava antialijasing
Format pločice	`TILE_FORMAT`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — PNG • 1 — JPG
Kvalitet (samo JPG) Opciono	`QUALITY`	[numerički: ceo broj] Podrazumevano: 75	Minimum 1, maksimum 100.
Veličina metapločice Opciono	`METATILESIZE`	[numerički: ceo broj] Podrazumevano: 4	Zadajte prilagođenu veličinu metapločice pri pravljenju XYZ pločica. Veće vrednosti mogu ubrzati iscrtavanje pločica i dati bolje natpise (manje praznina bez natpisa) uz veću potrošnju memorije. Minimum 1, maksimum 20.
Izlazna datoteka (za MBTiles)	`OUTPUT_FILE`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlazna datoteka (za MBTiles)	`OUTPUT_FILE`	[datoteka]	Izlazna datoteka.

Python kod

ID algoritma: `native:tilesxyzmbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.22 Vektorska analiza

Osnovna statistika za polja

Generiše osnovnu statistiku za polje atributske tabele vektorskog sloja. Podržana su numerička polja, polja datuma, vremena i niski. Vraćena statistika zavisi od tipa polja.

Statistika se može generisati kao tabela ili HTML datoteka i dostupna je u *Obrada* ► *Pregledač rezultata*.

Podrazumevani meni: *Vektor* ► *Alatke za analizu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni vektor	`INPUT_LAYER`	[vektor: bilo koji]	Vektorski sloj nad kojim se izračunava statistika
Polje nad kojim se izračunava statistika	`FIELD_NAME`	[polje tabele: bilo koji]	Bilo koje podržano polje tabele za izračunavanje statistike
Statistika Opciono `Dodato u 3.40`	`OUTPUT`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Zadajte izlaznu tabelu za generisanu statistiku. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Izveštaj o statistici Opciono	`OUTPUT_HTML_FILE`	[html] Podrazumevano: `[Save to temporary file]`	Specifikacija datoteke za izračunatu statistiku. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Statistika Dodato u 3.40	OUTPUT	[vektor: tabela]	Tabela koja sadrži izračunatu statistiku
Izveštaj o statistici	OUTPUT_HTML_FILE	[html]	HTML datoteka sa izračunatom statistikom
Broj	COUNT	[numerički: ceo broj]	
Broj jedinstvenih vrednosti	UNIQUE	[numerički: ceo broj]	
Broj praznih (null) vrednosti	EMPTY	[numerički: ceo broj]	
Broj nepraznih vrednosti	FILLED	[numerički: ceo broj]	
Najmanja vrednost	MIN	[isto kao ulaz]	
Najveća vrednost	MAX	[isto kao ulaz]	
Najmanja dužina	MIN_LENGTH	[numerički: ceo broj]	
Najveća dužina	MAX_LENGTH	[numerički: ceo broj]	
Srednja dužina	MEAN_LENGTH	[numerički: decimalni]	
Koeficijent varijacije	CV	[numerički: decimalni]	
Zbir	SUM	[numerički: decimalni]	
Srednja vrednost	MEAN	[numerički: decimalni]	
Standardna devijacija	STD_DEV	[numerički: decimalni]	
Opseg	RANGE	[numerički: decimalni]	
Medijana	MEDIAN	[numerički: decimalni]	
Minoritet (najređa vrednost)	MINORITY	[isto kao ulaz]	
Majoritet (najčešća vrednost)	MAJORITY	[isto kao ulaz]	
Prvi kvartil	FIRSTQUARTILE	[numerički: decimalni]	
Treći kvartil	THIRDQUARTILE	[numerički: decimalni]	
Interkvartilni opseg (IQR)	IQR	[numerički: decimalni]	

Python kod

ID algoritma: `native:basicstatisticsforfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uspon duž linije

Izračunava ukupan uspon i spust duž linijskih geometrija. Ulazni sloj mora imati Z vrednosti. Ako Z vrednosti nisu dostupne, algoritam *Prekri (postavi Z vrednost iz rastera)* može se upotrebiti da se Z vrednosti dodaju iz DEM sloja.

Izlazni sloj je kopija ulaznog sloja sa dodatnim poljima koja sadrže ukupan uspon (`climb`), ukupan spust (`descent`), najmanju nadmorsku visinu (`minelev`) i najveću nadmorsku visinu (`maxelev`) za svaku linijsku geometriju. Ako ulazni sloj sadrži polja sa istim nazivima kao ova dodata polja, ona će biti preimenovana (nazivi polja biće izmenjeni u „name_2", „name_3" itd., dok se ne pronađe prvi naziv koji nije duplikat).

Parametri

Oznaka	Назив	Tip	Opis
Linijski sloj	`INPUT`	[vektor: linija]	Linijski sloj za koji se izračunava uspon. Mora imati Z vrednosti
Sloj uspona	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog (linijskog) sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Sloj uspona	`OUTPUT`	[vektor: linija]	Linijski sloj sa novim atributima koji sadrže rezultate izračunavanja uspona.
Ukupan uspon	`TOTALCLIMB`	[numerički: decimalni]	Zbir uspona za sve linijske geometrije u ulaznom sloju
Ukupan spust	`TOTALDESCENT`	[numerički: decimalni]	Zbir spusta za sve linijske geometrije u ulaznom sloju
Najmanja nadmorska visina	`MINELEVATION`	[numerički: decimalni]	Najmanja nadmorska visina geometrija u sloju
Najveća nadmorska visina	`MAXELEVATION`	[numerički: decimalni]	Najveća nadmorska visina geometrija u sloju

Python kod

ID algoritma: `native:climbalongline`

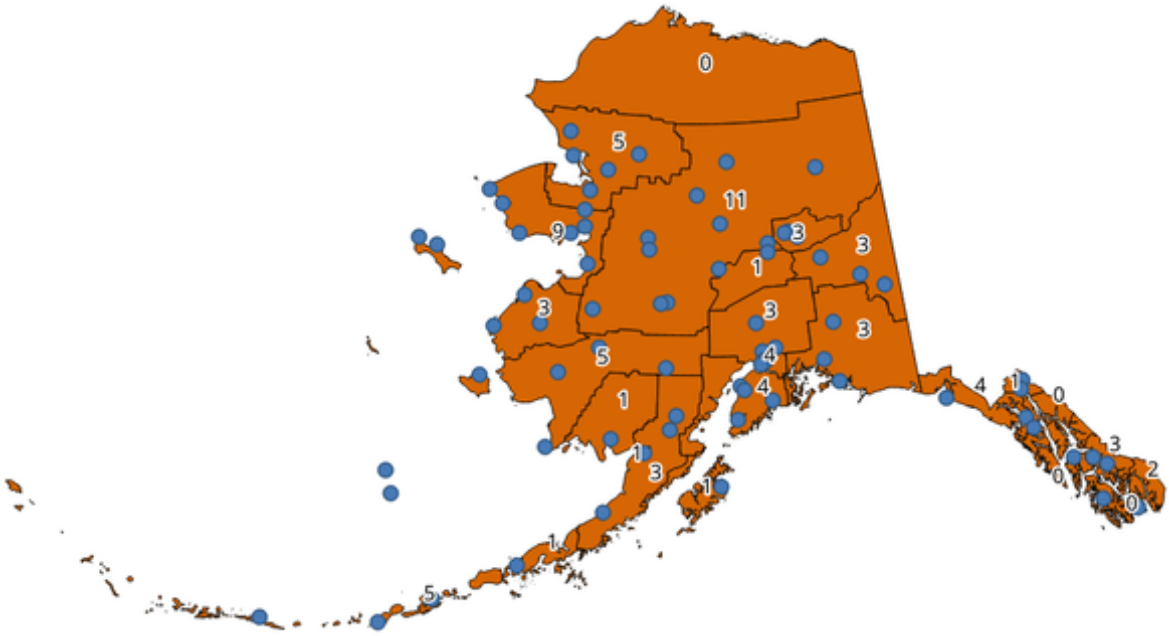
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Brojanje tačaka u poligonu

Uzima tačkasti i poligonski sloj i broji tačke iz tačkastog sloja u svakom poligonu poligonskog sloja.

Generiše se novi poligonski sloj sa potpuno istim sadržajem kao ulazni poligonski sloj, ali sa dodatnim poljem koje sadrži broj tačaka za svaki poligon.



Слика24.80: Oznake u poligonima prikazuju broj tačaka

Opciono polje težine može se koristiti za dodelu težina svakoj tački. Alternativno se može zadati polje jedinstvene klase. Ako se koriste obe opcije, polje težine ima prednost, a polje jedinstvene klase se zanemaruje.

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

`Podrazumevani meni`: *Vektor* ► *Alatke za analizu*

Parametri

Oznaka	Назив	Tip	Opis
Poligoni	`POLYGONS`	[vektor: poligon]	Poligonski sloj čijim se objektima pridružuje broj tačaka koje sadrže
Tačke	`POINTS`	[vektor: tačka]	Tačkasti sloj sa objektima koji se broje
Polje težine Opciono	`WEIGHT`	[polje tabele: bilo koji]	Polje iz tačkastog sloja. Generisani broj biće zbir polja težine tačaka sadržanih u poligonu. Ako polje težine nije numeričko, broj će biti `0`.
Polje klase Opciono	`CLASSFIELD`	[polje tabele: bilo koji]	Tačke se klasifikuju na osnovu izabranog atributa i ako se u poligonu nalazi više tačaka sa istom vrednošću atributa, broji se samo jedna od njih. Konačan broj tačaka u poligonu je, dakle, broj različitih klasa pronađenih u njemu.
Naziv polja sa brojem	`FIELD`	[niska] Podrazumevano: 'NUMPOINTS'	Naziv polja u koje se čuva broj tačaka

наставак на следећој страни

Табела 24.452 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj	`OUTPUT`	[vektor: poligon]	Rezultujući sloj čija atributska tabela sadrži novu kolonu sa brojem tačaka

Python kod

ID algoritma: `native:countpointsinpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

DBSCAN klasterovanje

Klasteruje tačkaste objekte na osnovu 2D implementacije algoritma DBSCAN (prostorno klasterovanje zasnovano na gustini, otporno na šum).

Algoritam zahteva dva parametra: najmanju veličinu klastera i najveće dozvoljeno rastojanje između tačaka u klasteru.

See also

ST-DBSCAN klasterovanje, *K-means klasterovanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Sloj za analizu
Najmanja veličina klastera	`MIN_SIZE`	[numerički: ceo broj] Podrazumevano: 5	Najmanji broj objekata potreban za formiranje klastera

наставак на следећој страни

Табела 24.454 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Najveće rastojanje između tačaka u klasteru	`EPS`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje preko koga dva objekta ne mogu pripadati istom klasteru (eps)
Klasteri	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj za rezultat klasterovanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tretiraj granične tačke kao šum (DBSCAN*)	`DBSCAN*`	[logička vrednost] Podrazumevano: False	Ako je označeno, tačke na granici klastera tretiraju se kao neklasterovane, a samo tačke u unutrašnjosti klastera označavaju se kao klasterovane.
Naziv polja klastera	`FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_ID'	Naziv polja u koje se čuva pridruženi broj klastera
Naziv polja sa veličinom klastera	`SIZE_FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_SIZE'	Naziv polja sa brojem objekata u istom klasteru

Izlazi

Oznaka	Назив	Tip	Opis
Klasteri	`OUTPUT`	[vektor: tačka]	Vektorski sloj sa originalnim objektima i poljem koje označava klaster kome pripadaju
Broj klastera	`NUM_CLUSTERS`	[numerički: ceo broj]	Broj pronađenih klastera

Python kod

ID algoritma: `native:dbscanclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Matrica rastojanja

Za tačkaste objekte izračunava rastojanja do njima najbližih objekata u istom ili drugom sloju.

Podrazumevani meni: *Vektor ► Alatke za analizu*

See also

Spajanje atributa po najbližem

Parametri

Oznaka	Назив	Tip	Opis
Ulazni tačkasti sloj	`INPUT`	[vektor: tačka]	Tačkasti sloj za koji se izračunava matrica rastojanja (**od** tačaka)
Ulazno polje sa jedinstvenim ID-om	`INPUT_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za jedinstveno identifikovanje objekata ulaznog sloja. Koristi se u izlaznoj atributskoj tabeli.
Ciljni tačkasti sloj	`TARGET`	[vektor: tačka]	Tačkasti sloj koji sadrži najbliže tačke koje se traže (**do** tačaka)
Ciljno polje sa jedinstvenim ID-om	`TARGET_FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za jedinstveno identifikovanje objekata ciljnog sloja. Koristi se u izlaznoj atributskoj tabeli.
Tip izlazne matrice	`MATRIX_TYPE`	[nabrajanje] Podrazumevano: 0	Dostupni su različiti tipovi izračunavanja: • 0 — Linearna (N * *k* x 3) matrica rastojanja: za svaku ulaznu tačku prijavljuje rastojanje do svake od *k* najbližih ciljnih tačaka. Izlazna matrica sastoji se od najviše *k* redova po ulaznoj tački, a svaki red ima tri kolone: *InputID*, *TargetID* i *Distance*. • 1 — Standardna (N x T) matrica rastojanja • 2 — Zbirna matrica rastojanja (srednja vrednost, std. dev., min, maks): za svaku ulaznu tačku prijavljuje statistiku rastojanja do njenih ciljnih tačaka.
Koristi samo najbližih (k) ciljnih tačaka	`NEAREST_POINTS`	[numerički: ceo broj] Podrazumevano: 0	Možete izabrati da se rastojanje izračuna do svih tačaka u ciljnom sloju (*0*) ili da se ograniči na određeni broj (*k*) najbližih objekata.
Matrica rastojanja	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog vektorskog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Matrica rastojanja	`OUTPUT`	[vektor: tačka]	Tačkasti (ili MultiPoint u slučaju „Linearna (N * *k* x 3)") vektorski sloj sa izračunatim rastojanjem za svaki ulazni objekat. Njegovi objekti i atributska tabela zavise od izabranog tipa izlazne matrice.

Python kod

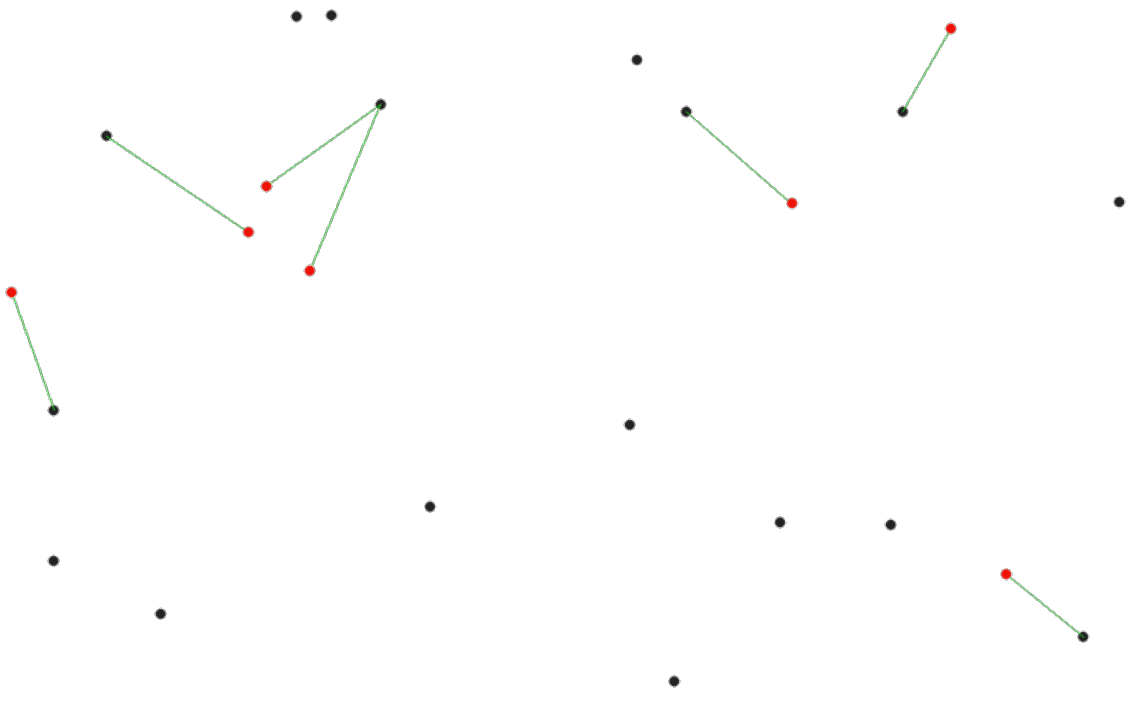
ID algoritma: `qgis:distancematrix`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rastojanje do najbližeg čvorišta (linija do čvorišta)

Pravi linije koje spajaju svaki objekat ulaznog vektorskog sloja sa najbližim objektom u odredišnom sloju. Rastojanja se izračunavaju na osnovu *centra* svakog objekta.



Слика24.81: Prikaz najbližeg čvorišta za crvene ulazne objekte

See also

Rastojanje do najbližeg čvorišta (tačke), *Spajanje atributa po najbližem*

Parametri

Oznaka	Назив	Tip	Opis
Sloj izvornih tačaka	`INPUT`	[vektor: geometrija]	Vektorski sloj za koji se traži najbliži objekat
Sloj odredišnih čvorišta	`HUBS`	[vektor: geometrija]	Vektorski sloj koji sadrži objekte koji se traže
Atribut sa nazivom u sloju čvorišta	`FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za jedinstveno identifikovanje objekata odredišnog sloja. Koristi se u izlaznoj atributskoj tabeli
Merna jedinica	`UNIT`	[nabrajanje] Podrazumevano: 0	Jedinice u kojima se prijavljuje rastojanje do najbližeg objekta: • 0 — Metri • 1 — Stope • 2 — Milje • 3 — Kilometri • 4 — Jedinice sloja
Rastojanje do čvorišta	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj koji povezuje odgovarajuće tačke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Rastojanje do čvorišta	`OUTPUT`	[vektor: linija]	Linijski vektorski sloj sa atributima ulaznih objekata, identifikatorom njima najbližeg objekta i izračunatim rastojanjem.

Python kod

ID algoritma: `qgis:distancetonearesthublinetohub`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rastojanje do najbližeg čvorišta (tačke)

Pravi tačkasti sloj koji predstavlja *centar* ulaznih objekata, uz dodatak dva polja koja sadrže identifikator najbližeg objekta (na osnovu njegove središnje tačke) i rastojanje između tačaka.

See also

Rastojanje do najbližeg čvorišta (linija do čvorišta), *Spajanje atributa po najbližem*

Parametri

Oznaka	Назив	Tip	Opis
Sloj izvornih tačaka	`INPUT`	[vektor: bilo koji]	Vektorski sloj za koji se traži najbliži objekat
Sloj odredišnih čvorišta	`HUBS`	[vektor: bilo koji]	Vektorski sloj koji sadrži objekte koji se traže
Atribut sa nazivom u sloju čvorišta	`FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za jedinstveno identifikovanje objekata odredišnog sloja. Koristi se u izlaznoj atributskoj tabeli
Merna jedinica	`UNIT`	[nabrajanje] Podrazumevano: 0	Jedinice u kojima se prijavljuje rastojanje do najbližeg objekta: • 0 — Metri • 1 — Stope • 2 — Milje • 3 — Kilometri • 4 — Jedinice sloja
Rastojanje do čvorišta	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni tačkasti vektorski sloj sa najbližim čvorištem. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Rastojanje do čvorišta	`OUTPUT`	[vektor: tačka]	Tačkasti vektorski sloj koji predstavlja centar izvornih objekata sa njihovim atributima, identifikatorom njima najbližeg objekta i izračunatim rastojanjem.

Python kod

ID algoritma: `qgis:distancetonearesthubpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje linijama (linije čvorišta)

Pravi dijagrame tipa čvorište-krak povezujući linijama tačke iz sloja krakova sa odgovarajućim tačkama u sloju čvorišta.

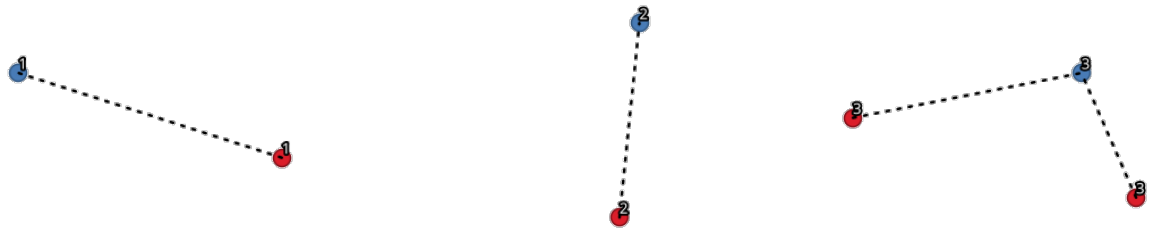
Koje čvorište pripada kojoj tački određuje se poklapanjem polja ID čvorišta na tačkama čvorišta i polja ID kraka na tačkama krakova.

Ako ulazni slojevi nisu tačkasti, kao mesto povezivanja uzima se tačka na površini geometrija.

Opciono se mogu praviti geodetske linije, koje predstavljaju najkraći put po površini elipsoida. Kada se koristi geodetski režim, napravljene linije mogu se deliti na antimeridijanu (±180 stepeni geografske dužine), što može poboljšati iscrtavanje linija. Dodatno se može zadati rastojanje između temena. Manje rastojanje daje gušću i precizniju liniju.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.



Слика24.82: Spajanje tačaka na osnovu zajedničkog polja / atributa

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Sloj čvorišta	`HUBS`	[vektor: geometrija]	Ulazni sloj
Polje sa ID-om čvorišta	`HUB_FIELD`	[polje tabele: bilo koji]	Polje sloja čvorišta sa ID-om za spajanje
Polja sloja čvorišta koja se kopiraju (ostavite prazno da biste kopirali sva polja) Opciono	`HUB_FIELDS`	[polje tabele: bilo koji] [lista]	Polja sloja čvorišta koja se kopiraju. Ako se ne izabere nijedno polje, uzimaju se sva polja.
Sloj krakova	`SPOKES`	[vektor: geometrija]	Dodatni tačkasti sloj krakova
Polje sa ID-om kraka	`SPOKE_FIELD`	[polje tabele: bilo koji]	Polje sloja krakova sa ID-om za spajanje
Polja sloja krakova koja se kopiraju (ostavite prazno da biste kopirali sva polja) Opciono	`SPOKE_FIELDS`	[polje tabele: bilo koji] [lista]	Polja sloja krakova koja se kopiraju. Ako se ne izabere nijedno polje, uzimaju se sva polja.
Napravi geodetske linije	`GEODESIC`	[logička vrednost] Podrazumevano: False	Pravi geodetske linije (najkraći put po površini elipsoida)

наставак на следећој страни

Табела 24.463 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Linije čvorišta	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj linija čvorišta. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Rastojanje između temena (samo geodetske linije)	`GEODESIC_DISTANCE`	[numerički: decimalni] Podrazumevano: 1000.0 (kilometara)	Rastojanje između uzastopnih temena (u kilometrima). Manje rastojanje daje gušću i precizniju liniju
Podeli linije na antimeridijanu (±180 stepeni geografske dužine)	`ANTIMERIDIAN_SPLIT`	[logička vrednost] Podrazumevano: False	Deli linije na ±180 stepeni geografske dužine (radi boljeg iscrtavanja linija)

Izlazi

Oznaka	Назив	Tip	Opis
Linije čvorišta	`OUTPUT`	[vektor: linija]	Rezultujući linijski sloj koji povezuje odgovarajuće tačke u ulaznim slojevima

Python kod

ID algoritma: `native:hublines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

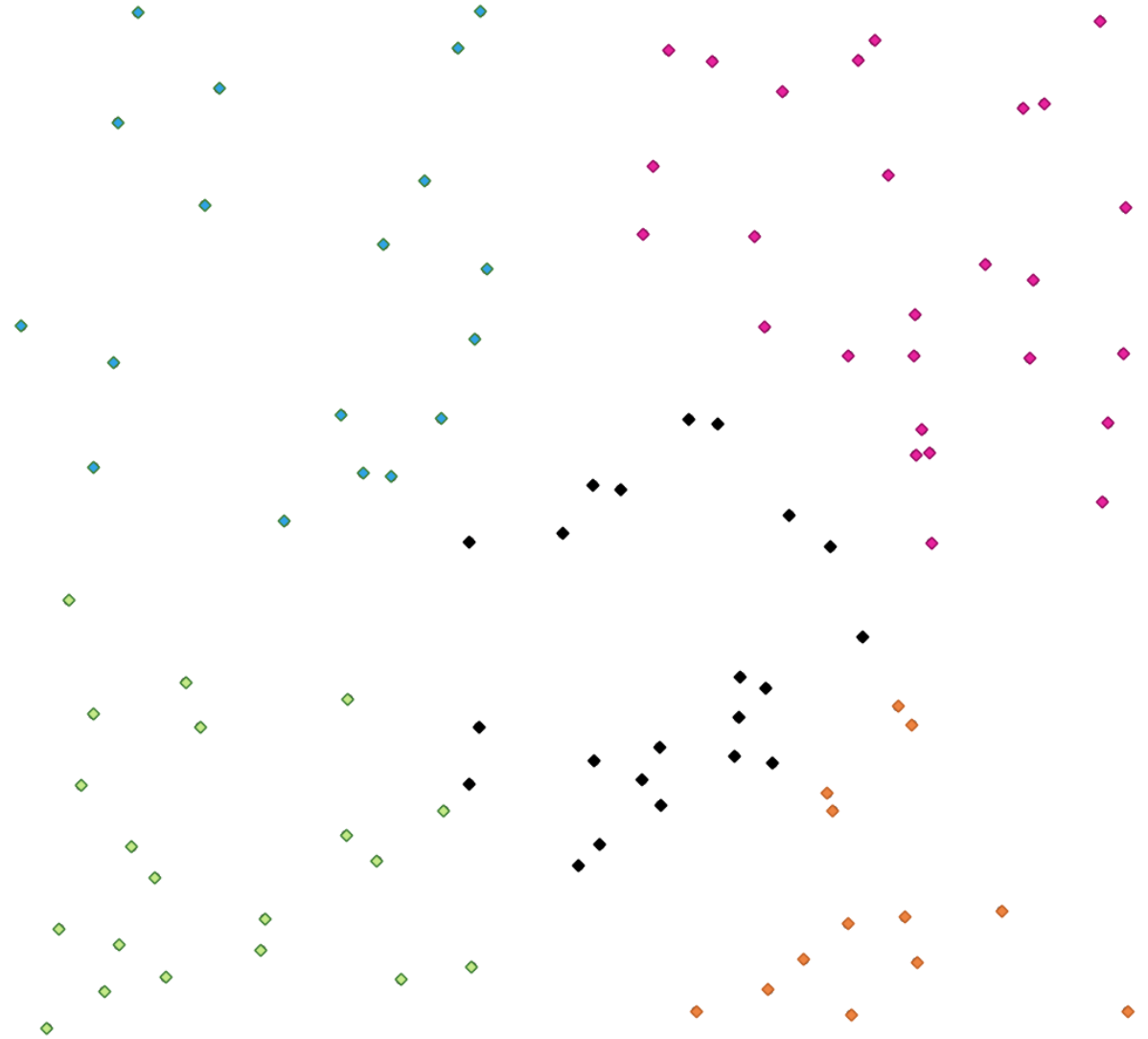
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

K-means klasterovanje

Izračunava broj k-means klastera za svaki ulazni objekat na osnovu 2D rastojanja.

K-means klasterovanje ima za cilj da podeli objekte u k klastera u kojima svaki objekat pripada klasteru sa najbližom srednjom vrednošću. Srednju tačku predstavlja baricentar klasterovanih objekata.

Ako su ulazne geometrije linije ili poligoni, klasterovanje se zasniva na centroidu objekta.



Слика24.83: Klasteri tačaka u pet klasa

наставак на следећој страни

Табела 24.466 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

See also

DBSCAN klasterovanje, *ST-DBSCAN klasterovanje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj za analizu
Broj klastera	`CLUSTERS`	[numerički: ceo broj] Podrazumevano: 5	Broj klastera koji se formira od objekata
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod izbora početnih centara. Moguće vrednosti su: • 0 — Najudaljenije tačke: za početne centre biraju se tačke koje su najudaljenije jedna od druge. • 1 — K-means++: Prvi centar bira se uniformno slučajno iz skupa ulaznih objekata. Naredni centri biraju se na osnovu težinske raspodele verovatnoće koja daje prednost objektima udaljenijim od postojećih centara. Time se obezbeđuje dobra raspoređenost početnih centara, smanjuje verovatnoća lošeg klasterovanja i poboljšavaju performanse algoritma. *Literatura: Arthur, David & Vassilvitskii, Sergei. (2007). K-Means++: The Advantages of Careful Seeding. Proc. of the Annu. ACM-SIAM Symp. on Discrete Algorithms. 8.*
Klasteri	`OUTPUT`	[vektor: isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za generisane klastere. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Naziv polja klastera	`FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_ID'	Naziv polja u koje se čuva pridruženi broj klastera
Naziv polja sa veličinom klastera	`SIZE_FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_SIZE'	Naziv polja sa brojem objekata u istom klasteru

Izlazi

Oznaka	Назив	Tip	Opis
Klasteri	`OUTPUT`	[vektor: isto kao ulaz]	Vektorski sloj sa originalnim objektima i poljima koja označavaju klaster kome pripadaju i njihov broj u njemu

Python kod

ID algoritma: `native:kmeansclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izlistavanje jedinstvenih vrednosti

Izlistava jedinstvene vrednosti polja atributske tabele i broji ih.

Podrazumevani meni: *Vektor ► Alatke za analizu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj za analizu
Ciljna polja	`FIELDS`	[polje tabele: bilo koji] [lista]	Polja koja se analiziraju
Jedinstvene vrednosti Opciono	`OUTPUT`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Zadajte zbirni tabelarni sloj sa jedinstvenim vrednostima. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.469 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
HTML izveštaj Opciono	`OUTPUT_HTML_FILE`	[html] Podrazumevano: `[Save to temporary file]`	HTML izveštaj o jedinstvenim vrednostima u *Obrada ► Pregledač rezultata*. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Jedinstvene vrednosti	`OUTPUT`	[vektor: tabela]	Zbirni tabelarni sloj sa jedinstvenim vrednostima
HTML izveštaj	`OUTPUT_HTML_FILE`	[html]	HTML izveštaj o jedinstvenim vrednostima. Može se otvoriti iz *Obrada ► Pregledač rezultata*
Ukupno jedinstvenih vrednosti	`TOTAL_VALUES`	[numerički: ceo broj]	Broj jedinstvenih vrednosti u ulaznom polju
Spojene jedinstvene vrednosti	`UNIQUE_VALUES`	[niska]	Niska sa listom jedinstvenih vrednosti iz ulaznog polja, razdvojenih zarezom

Python kod

ID algoritma: `native:listuniquevalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Srednje koordinate

Izračunava tačkasti sloj sa težištem geometrija ulaznog sloja.

Može se zadati atribut koji sadrži težine koje se primenjuju na svaki objekat pri izračunavanju težišta.

Ako se u parametru izabere atribut, objekti će biti grupisani prema vrednostima u tom polju. Umesto jedne tačke sa težištem celog sloja, izlazni sloj sadržaće težište objekata svake kategorije.

Podrazumevani meni: *Vektor ► Alatke za analizu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje težine Opciono	`WEIGHT`	[polje tabele: numerički]	Polje koje se koristi ako želite težinsku sredinu
Polje sa jedinstvenim ID-om	`UID`	[polje tabele: bilo koji]	Jedinstveno polje po kome se vrši izračunavanje sredine

наставак на следећој страни

Табела 24.471 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Srednje koordinate	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte (tačkasti vektorski) sloj za rezultat. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Srednje koordinate	`OUTPUT`	[vektor: tačka]	Rezultujući tačkasti sloj

Python kod

ID algoritma: `native:meancoordinates`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Analiza najbližeg suseda

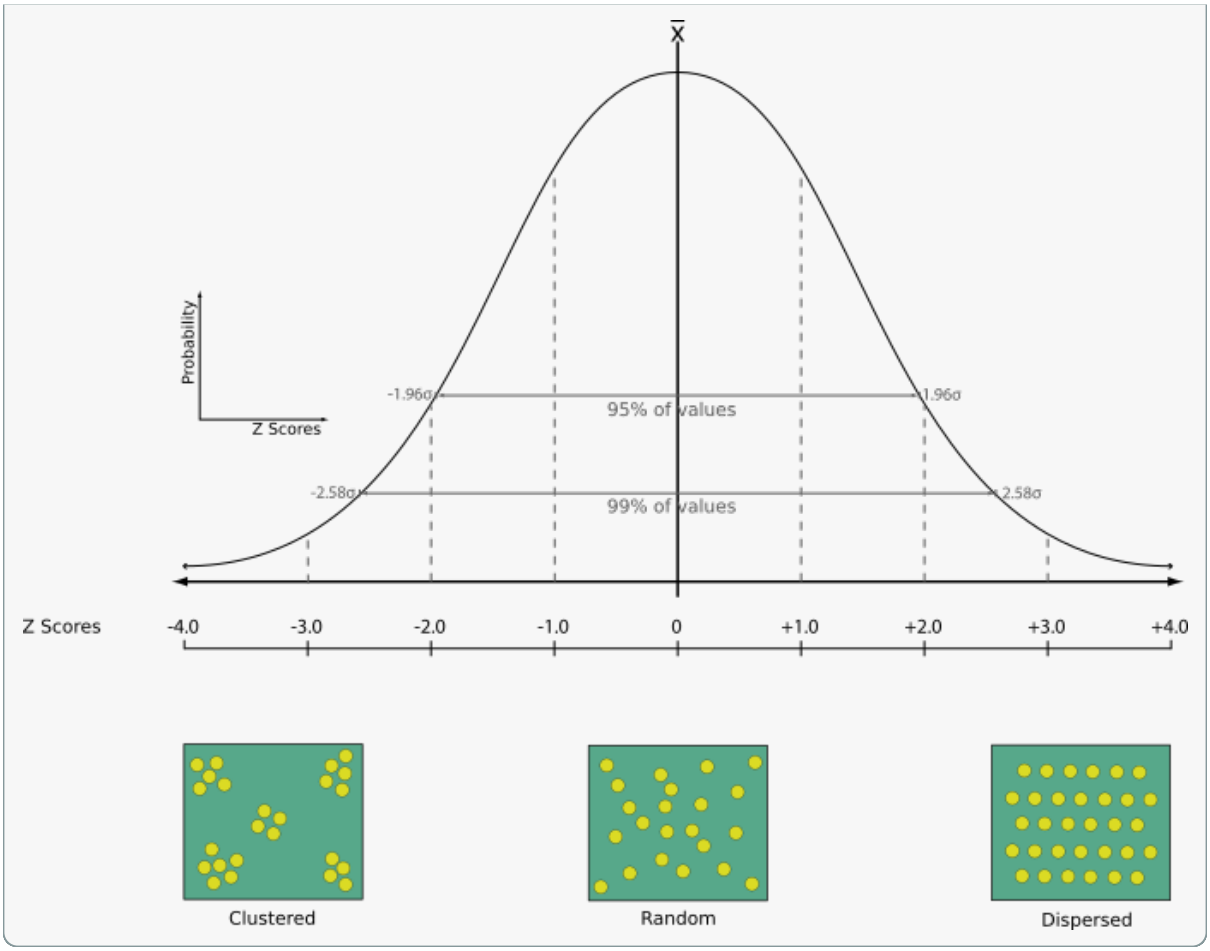
Vrši analizu najbližeg suseda za tačkasti sloj. Izlaz pokazuje kako su vaši podaci raspoređeni (klasterovano, slučajno ili ravnomerno).

Izlaz se generiše kao HTML datoteka sa izračunatim statističkim vrednostima:

- Posmatrano srednje rastojanje
- Očekivano srednje rastojanje
- Indeks najbližeg suseda
- Broj tačaka
- 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 likely to be the result of a spatially random process, while a high Z-Score means that your data A Z-Score close to 0 means that the data are likely to be the result of a spatially random process, while a Z-Score far from 0 means that the data are unlikely to be the result of a spatially random process.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.



Podrazumevani meni: *Vektor* ► *Alatke za analizu*

See also

Spajanje atributa po najbližem

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Tačkasti vektorski sloj nad kojim se izračunava statistika
Najbliži sused Opciono	`OUTPUT_HTML_FILE`	[html] Podrazumevano: `[Save to temporary file]`	Specifikacija HTML datoteke za izračunatu statistiku. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Najbliži sused	`OUTPUT_HTML_FILE`	[html]	HTML datoteka sa izračunatom statistikom
Posmatrano srednje rastojanje	`OBSERVED_MD`	[numerički: decimalni]	Posmatrano srednje rastojanje
Očekivano srednje rastojanje	`EXPECTED_MD`	[numerički: decimalni]	Očekivano srednje rastojanje
Indeks najbližeg suseda	`NN_INDEX`	[numerički: ceo broj]	Indeks najbližeg suseda
Broj tačaka	`POINT_COUNT`	[numerički: ceo broj]	Broj tačaka
Z-skor	`Z_SCORE`	[numerički: decimalni]	Z-skor

Python kod

ID algoritma: `native:nearestneighbouranalysis`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Analiza preklapanja

Izračunava površinu i procenat pokrivenosti kojom objekti iz ulaznog sloja bivaju preklopljeni objektima iz izabranih slojeva za preklapanje.

U izlazni sloj dodaju se novi atributi koji prijavljuju ukupnu površinu preklapanja i procenat ulaznog objekta preklopljen svakim od izabranih slojeva za preklapanje.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni sloj.
Slojevi za preklapanje	`LAYERS`	[vektor: poligon] [lista]	Slojevi za preklapanje.
Preklapanje	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Veličina mreže Opciono	`GRID_SIZE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se zada, ulazne geometrije se prilepljuju na mrežu date veličine, a temena rezultata izračunavaju se na toj istoj mreži.

Izlazi

Oznaka	Назив	Tip	Opis
Preklapanje	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa dodatnim poljima koja prijavljuju preklapanje (u jedinicama karte i procentima) ulaznog objekta sa svakim od izabranih slojeva.

Python kod

ID algoritma: `native:calculatevectoroverlaps`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

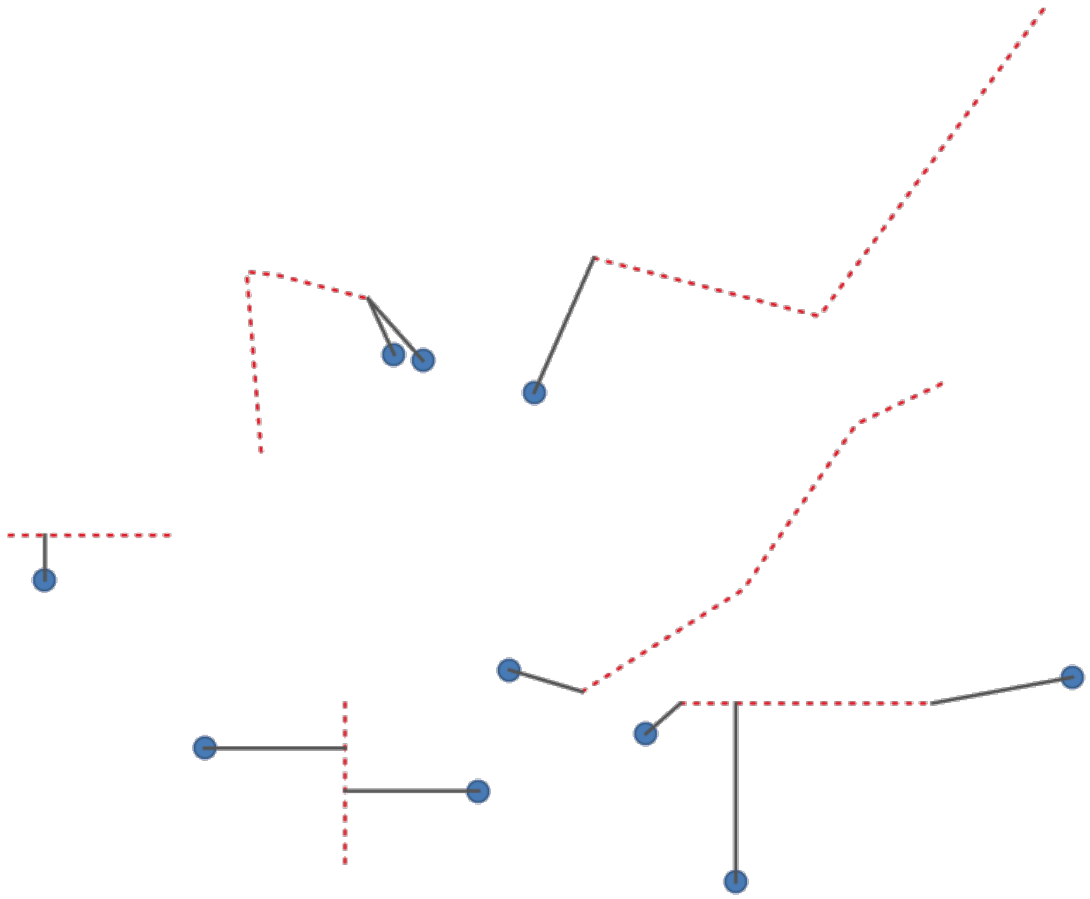
Najkraća linija između objekata

Pravi linijski sloj kao najkraću liniju između izvornog i odredišnog sloja. Podrazumevano se uzima u obzir samo prvi najbliži objekat odredišnog sloja. Može se zadati broj n najbližih susednih objekata. Ako se zada najveće rastojanje, uzimaju se u obzir samo objekti bliži od tog rastojanja.

Izlazni objekti sadržaće sve atribute izvornog sloja, sve atribute n najbližeg objekta i dodatno polje sa rastojanjem.

Важно

Ovaj algoritam koristi isključivo Dekartova izračunavanja rastojanja i ne uzima u obzir geodetska svojstva ni svojstva elipsoida pri određivanju blizine objekata. Koordinatni sistem merenja i izlaza zasnovan je na koordinatnom sistemu izvornog sloja.



Слика24.84: Najkraća linija od tačkastih objekata do linija

Parametri

Oznaka	Назив	Tip	Opis
Izvorni sloj	`SOURCE`	[vektor: geometrija]	Polazni sloj za koji se traže najbliži susedi
Odredišni sloj	`DESTINATION`	[vektor: geometrija]	Ciljni sloj u kome se traže najbliži susedi
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod izračunavanja najkraćeg rastojanja. Moguće vrednosti su: • 0 — Rastojanje do najbliže tačke na objektu • 1 — Rastojanje do centroida objekta
Najveći broj suseda	`NEIGHBORS`	[numerički: ceo broj] Podrazumevano: 1	Najveći broj suseda koji se traži

наставак на следећој страни

Табела 24.478 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Najveće rastojanje Opciono	`DISTANCE`	[numerički: decimalni]	Uzimaju se u obzir samo odredišni objekti bliži od ovog rastojanja.
Najkraće linije	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[vektor: linija]	Linijski vektorski sloj koji spaja izvorne objekte sa njima najbližim susedima u odredišnom sloju. Sadrži sve atribute i izvornih i odredišnih objekata, kao i izračunato rastojanje.

Python kod

ID algoritma: `native:shortestline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

ST-DBSCAN klasterovanje

Klasteruje tačkaste objekte na osnovu 2D implementacije algoritma ST-DBSCAN (prostorno-vremensko klasterovanje zasnovano na gustini, otporno na šum).

See also

DBSCAN klasterovanje, *K-means klasterovanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Sloj za analizu
Polje sa datumom/vremenom	`DATETIME_FIELD`	[polje tabele: datum]	Polje koje sadrži vremenske podatke

наставак на следећој страни

Табела 24.480 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Najmanja veličina klastera	`MIN_SIZE`	[numerički: ceo broj] Podrazumevano: 5	Najmanji broj objekata potreban za formiranje klastera
Najveće rastojanje između tačaka u klasteru	`EPS`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje preko koga dva objekta ne mogu pripadati istom klasteru (eps)
Najduže vreme između tačaka u klasteru	`EPS2`	[numerički: decimalni] Podrazumevano: 0.0 (dana)	Vreme preko koga dva objekta ne mogu pripadati istom klasteru (eps2). Dostupne jedinice vremena su milisekunde, sekunde, minuti, sati, dani i nedelje.
Klasteri	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj za rezultat klasterovanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Tretiraj granične tačke kao šum (DBSCAN*)	`DBSCAN*`	[logička vrednost] Podrazumevano: False	Ako je označeno, tačke na granici klastera tretiraju se kao neklasterovane, a samo tačke u unutrašnjosti klastera označavaju se kao klasterovane.
Naziv polja klastera	`FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_ID'	Naziv polja u koje se čuva pridruženi broj klastera
Naziv polja sa veličinom klastera	`SIZE_FIELD_NAME`	[niska] Podrazumevano: 'CLUSTER_SIZE'	Naziv polja sa brojem objekata u istom klasteru

Izlazi

Oznaka	Назив	Tip	Opis
Klasteri	`OUTPUT`	[vektor: tačka]	Vektorski sloj sa originalnim objektima i poljem koje označava klaster kome pripadaju
Broj klastera	`NUM_CLUSTERS`	[numerički: ceo broj]	Broj pronađenih klastera

Python kod

ID algoritma: `native:stdbscanclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Statistika po kategorijama

Izračunava statistiku polja u zavisnosti od nadređene klase. Nadređena klasa je kombinacija vrednosti iz drugih polja.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni vektorski sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj sa jedinstvenim klasama i vrednostima
Polje nad kojim se izračunava statistika (ako je prazno, izračunava se samo broj) Opciono	`VALUES_FIELD_NAME`	[polje tabele: bilo koji]	Ako je prazno, izračunava se samo broj
Polja sa kategorijama	`CATEGORIES_FIELD_NAME`	[polje tabele: bilo koji] [lista]	Polja koja (u kombinaciji) definišu kategorije
Statistika po kategoriji	`OUTPUT`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Zadajte izlaznu tabelu za generisanu statistiku. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Statistika po kategoriji	`OUTPUT`	[vektor: tabela]	Tabela koja sadrži statistiku

U zavisnosti od tipa polja koje se analizira, za svaku grupisanu vrednost vraćaju se sledeće statistike:

Statistika	Niska	Numerič-ko	Datum
Broj (`COUNT`)	☑	☑	☑
Jedinstvene vrednosti (`UNIQUE`)	☑		☑
Prazne (null) vrednosti (`EMPTY`)	☑		☑
Neprazne vrednosti (`FILLED`)	☑		☑
Najmanja vrednost (`MIN`)	☑	☑	☑
Najveća vrednost (`MAX`)	☑	☑	☑
Opseg (`RANGE`)		☑	

наставак на следећој страни

Табела 24.485 – настављено са претходне стране

Statistika	Niska	Numeričko	Datum
Zbir (SUM)		☑	
Srednja vrednost (MEAN)		☑	
Medijana (MEDIAN)		☑	
Standardna devijacija (STD_DEV)		☑	
Koeficijent varijacije (CV)		☑	
Minoritet (najređa vrednost - MINORITY)		☑	
Majoritet (najčešća vrednost - MAJORITY)		☑	
Prvi kvartil (FIRSTQUARTILE)		☑	
Treći kvartil (THIRDQUARTILE)		☑	
Interkvartilni opseg (IQR)		☑	
Najmanja dužina (MIN_LENGTH)	☑		
Srednja dužina (MEAN_LENGTH)	☑		
Najveća dužina (MAX_LENGTH)	☑		

Python kod

ID algoritma: `qgis:statisticsbycategories`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zbir dužina linija

Uzima poligonski i linijski sloj i meri ukupnu dužinu linija i ukupan broj linija koje presecaju svaki poligon.

Rezultujući sloj ima iste objekte kao ulazni poligonski sloj, ali sa dva dodatna atributa koja sadrže dužinu i broj linija kroz svaki poligon.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

☑ Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za analizu*

Parametri

Oznaka	Назив	Tip	Opis
Linije	LINES	[vektor: linija]	Ulazni vektorski linijski sloj
Poligoni	POLYGONS	[vektor: poligon]	Poligonski vektorski sloj

наставак на следећој страни

Табела 24.486 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Naziv polja sa dužinom linija	`LEN_FIELD`	[niska] Podrazumevano: 'LENGTH'	Naziv polja za dužinu linija
Naziv polja sa brojem linija	`COUNT_FIELD`	[niska] Podrazumevano: 'COUNT'	Naziv polja za broj linija
Dužina linija	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski sloj sa generisanom statistikom. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Dužina linija	`OUTPUT`	[vektor: poligon]	Izlazni poligonski sloj sa poljima dužine i broja linija

Python kod

ID algoritma: `native:sumlinelengths`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.23 Vektorska pokrivenost

Stapanje pokrivenosti

`Dodato u 3.36`

Radi nad pokrivenošću (predstavljenom kao skup poligonskih objekata sa potpuno poklopljenom geometrijom ivica) kako bi stopio (unijom objedinio) geometrije. Nudi znatno optimizovaniji pristup objedinjavanju ovih objekata u odnosu na standardne alatke za stapanje.

See also

Stapanje

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Stopljeno	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Stopljeno	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj sa stopljenim geometrijama.

Python kod

ID algoritma: `native:coverageunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pojednostavljivanje pokrivenosti

`Dodato u 3.36`

Radi nad pokrivenošću (predstavljenom kao skup poligonskih objekata sa potpuno poklopljenom geometrijom ivica) kako bi na ivice primenio Visvalingam–Whyatt pojednostavljivanje, smanjujući složenost srazmerno zadatoj toleranciji, uz zadržavanje ispravne pokrivenosti (tj. nakon pojednostavljivanja se nijedna ivica neće ukrštati ni dodirivati). Geometrije se nikada ne uklanjaju, ali mogu biti pojednostavljene sve do prostog trougla. Takođe, neke geometrije (kao što su poligoni sa premalo tačaka koje se ne ponavljaju) biće vraćene nepromenjene.

Ako ulazni skup podataka nije ispravna pokrivenost zbog preklapanja, i dalje će biti pojednostavljen, ali neispravna topologija, kao što su ukrštene ivice, ostaće neispravna.

Пажња

Pokretanje ovog algoritma zahteva QGIS instaliran sa GEOS >= 3.12 (pogledajte meni *Pomoć* ► *O programu*).

See also

Uprosti, *Zagladi*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Tolerancija	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Najveće rastojanje (u izabranoj jedinici) između dva uzastopna temena da bi se spojila.
Sačuvaj granicu	`PRESERVE_BOUNDARY`	[logička vrednost] Podrazumevano: False	Kada je omogućeno, spoljašnje ivice pokrivenosti biće sačuvane bez pojednostavljivanja
Pojednostavljeno	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pojednostavljeno	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj, sa manjim brojem temena.

Python kod

ID algoritma: `native:coveragesimplify`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Provera pokrivenosti

`Dodato u 3.36`

Analizira pokrivenost (predstavljenu kao skup poligonskih objekata sa potpuno poklopljenom geometrijom ivica) kako bi pronašla mesta na kojima pretpostavka o potpunom poklapanju ivica nije ispunjena. Neispravnost obuhvata poligone koji se preklapaju ili imaju procepe uže od zadate širine procepa.

Пажња

Pokretanje ovog algoritma zahteva QGIS instaliran sa GEOS >= 3.12 (pogledajte meni *Pomoć* ► *O programu*).

See also

Proveri validnost, *Small gaps*, *Overlaps*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Širina procepa	`GAP_WIDTH`	[numerički: decimalni] Podrazumevano: 0.0	Najveća širina procepa koji se otkriva
Neispravne ivice	`INVALID_EDGES`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Neispravne ivice	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj koji prikazuje ivice sa greškom za svaki neispravan poligon.
Rezultat provere ispravnosti	`IS_VALID`	[logička vrednost]	Vraća da li je pokrivenost ispravna ili nije.

Python kod

ID algoritma: `native:coveragevalidate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

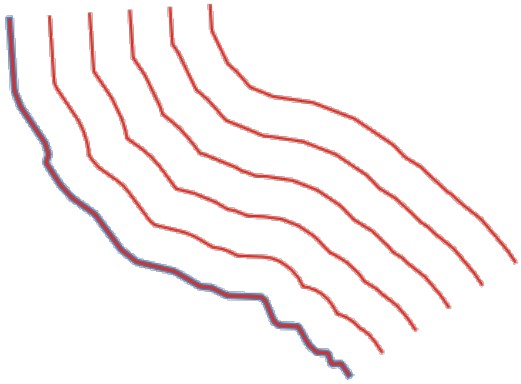
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.24 Kreiranje vektora

Niz odmaknutih (paralelnih) linija

Pravi kopije linijskih objekata u sloju tako što za svaki objekat kreira više odmaknutih verzija. Svaka nova verzija odmaknuta je za zadato rastojanje više od prethodne.

Pozitivno rastojanje odmiče linije ulevo, a negativno udesno.



Слика24.85: Plavo je izvorni sloj, crveno odmaknuti

Omogućava *izmenu objekata na licu mesta* za linijske objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

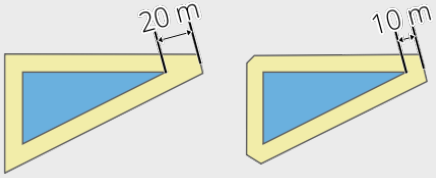
Pomeri linije, *Niz translatiranih objekata*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj koji se koristi za odmicanje.
Broj objekata koji se kreira	`COUNT`	[numerički: ceo broj] Podrazumevano: 10	Broj odmaknutih kopija koje se generišu za svaki objekat
Korak odstojanja	`OFFSET`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje između dve uzastopne odmaknute kopije
Odmaknute linije	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj sa odmaknutim objektima. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Segmenti	`SEGMENTS`	[numerički: decimalni] Podrazumevano: 8	Broj linijskih segmenata kojima se aproksimira četvrtina kruga pri kreiranju zaobljenih odmicanja
Stil spoja	`JOIN_STYLE`	[nabrajanje] Podrazumevano: 0	Zadajte da li se pri odmicanju uglova linije koriste zaobljeni, oštri ili zakošeni spojevi. Jedno od: • 0 — Zaobljeno • 1 — Oštro • 2 — Zakošeno  Слика24.86: Zaobljeni, oštri i zakošeni stilovi spoja
Granica oštrog spoja	`MITER_LIMIT`	[numerički: decimalni] Podrazumevano: 2.0	Postavlja najveće rastojanje od odmaknute geometrije koje se koristi pri kreiranju oštrog spoja, kao faktor rastojanja odmicanja (primenjuje se samo na oštre stilove spoja). Minimum: 1.0  Слика24.87: Bafer od 10 m sa granicom 2 i bafer od 10 m sa granicom 1

Izlazi

Oznaka	Назив	Tip	Opis
Odmaknute linije	`OUTPUT`	[vektor: linija]	Izlazni linijski sloj sa originalnim i odmaknutim objektima. Atributi se kopiraju u odgovarajuće izlaze. Dodaju se sledeći atributi vezani za ulazne parametre: `instance`, koji indeksira `COUNT` odmaknutih linija, i `offset`, koji označava rastojanje odmicanja od originalne linije.

Python kod

ID algoritma: `native:arrayoffsetlines`

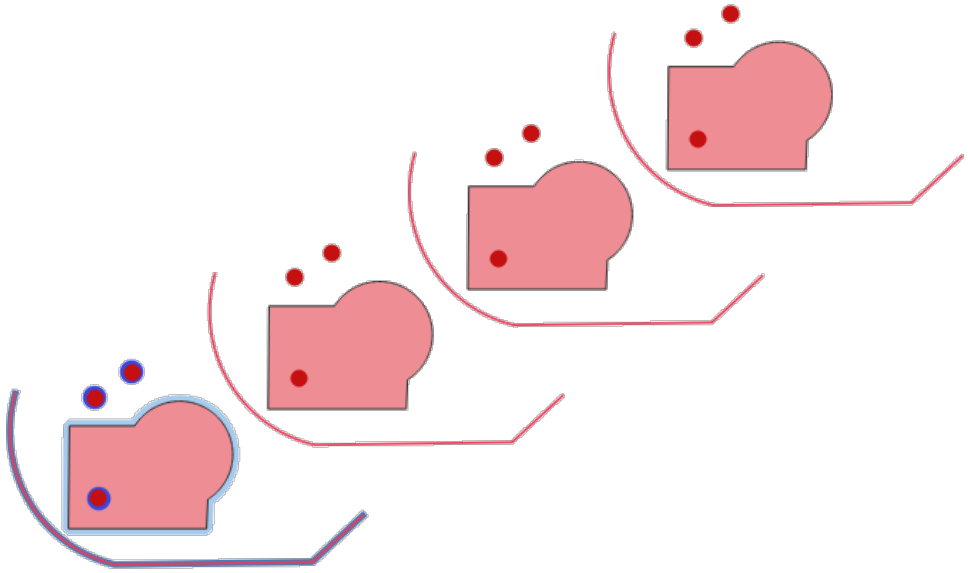
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Niz translatiranih objekata

Pravi kopije objekata u sloju tako što za svaki kreira više translatiranih verzija. Svaka kopija pomerena je za unapred zadatu vrednost po X, Y i/ili Z osi više od prethodne.

Mogu se translatirati i M vrednosti prisutne u geometriji.



Слика24.88: Ulazni slojevi u plavim tonovima, izlazni slojevi sa translatiranim objektima u crvenim tonovima

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Transliraj, *Niz odmaknutih (paralelnih) linija*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj za translaciju
Broj objekata koji se kreira	`COUNT`	[numerički: ceo broj] Podrazumevano: 10	Broj kopija koje se generišu za svaki objekat

наставак на следећој страни

Табела 24.497 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Korak pomeranja (x-osa)	`DELTA_X`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po X osi
Korak pomeranja (y-osa)	`DELTA_Y`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po Y osi
Korak pomeranja (z-osa)	`DELTA_Z`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po Z osi
Korak pomeranja (m vrednosti)	`DELTA_M`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na M
Translatirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Izlazni vektorski sloj sa translatiranim (pomerenim) kopijama objekata. Kopiraju se i originalni objekti. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Translatirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa translatiranim (pomerenim) kopijama objekata. Kopiraju se i originalni objekti.

Python kod

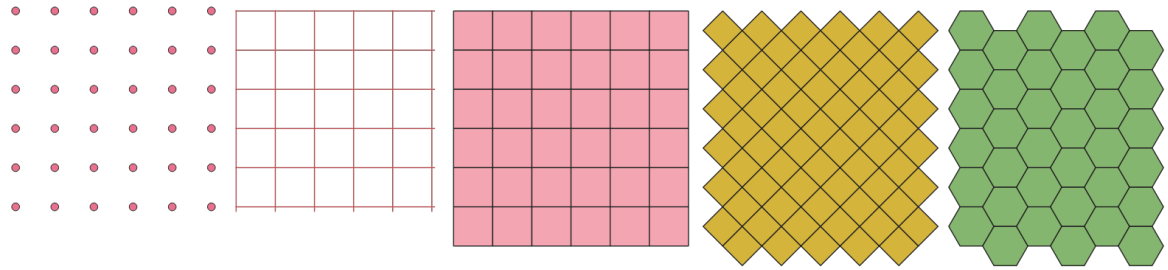
ID algoritma: `native:arraytranslatedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje mreže

Pravi vektorski sloj sa mrežom koja pokriva zadati obuhvat. Ćelije mreže mogu imati različite oblike:



Слика24.89: Različiti oblici ćelija mreže primenjeni na isti obuhvat, bez preklapanja

Veličina i/ili položaj svakog elementa mreže definiše se vodoravnim i uspravnim razmakom. CRS izlaznog sloja mora biti definisan. Obuhvat mreže i vrednosti razmaka moraju biti izraženi u koordinatama i jedinicama tog CRS-a.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

Parametri

Oznaka	Назив	Tip	Opis
Tip mreže	`TYPE`	[nabrajanje] Podrazumevano: 0	Oblik mreže. Jedno od: • 0 — Tačka • 1 — Linija • 2 — Pravougaonik (poligon) • 3 — Romb (poligon) • 4 — Šestougao (poligon)
Obuhvat mreže	`EXTENT`	[obuhvat]	Obuhvat mreže Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Vodoravni razmak	`HSPACING`	[numerički: decimalni] Podrazumevano: 1.0	Veličina ćelije mreže po X osi
Uspravni razmak	`VSPACING`	[numerički: decimalni] Podrazumevano: 1.0	Veličina ćelije mreže po Y osi
Vodoravno preklapanje	`HOVERLAY`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje preklapanja između dve uzastopne ćelije mreže po X osi

наставак на следећој страни

Табела 24.499 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Uspravno preklapanje	`VOVERLAY`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje preklapanja između dve uzastopne ćelije mreže po Y osi
CRS mreže	`CRS`	[crs] Podrazumevano: *CRS projekta*	Koordinatni referentni sistem koji se primenjuje na mrežu
Mreža	`OUTPUT`	[vektor: geometrija] Podrazumevano: `[Create temporary layer]`	Rezultujući vektorski sloj mreže. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Mreža	`OUTPUT`	[vektor: geometrija]	Rezultujući vektorski sloj mreže. Tip izlazne geometrije (tačka, linija ili poligon) zavisi od *Tip mreže*. Objekti se kreiraju odozgo nadole i sleva nadesno. Atributska tabela se popunjava sa: • `id` • koordinatama strana `left`, `right`, `top` i `bottom` • i njihovim položajem u mreži: `row_index` i `column_index` (dostupno za mreže tipa tačka, pravougaonik i šestougao)

Python kod

ID algoritma: `native:creategrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje sloja iz obuhvata

Pravi novi vektorski sloj koji sadrži jedan objekat čija geometrija odgovara obuhvatu ulaznog sloja.

Može se koristiti u modelima za pretvaranje doslovnog obuhvata (format `xmin`, `xmax`, `ymin`, `ymax`) u sloj koji se može upotrebiti u drugim algoritmima koji zahtevaju ulaz u obliku sloja.

See also

Kreiranje sloja iz tačke

Parametri

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`INPUT`	[obuhvat]	Ulazni obuhvat Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Obuhvat	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Obuhvat	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj (obuhvat)

Python kod

ID algoritma: `native:extenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje sloja iz tačke

Pravi novi vektorski sloj koji sadrži jedan objekat čija geometrija odgovara parametru tačke. Može se koristiti u modelima za pretvaranje tačke u tačkasti sloj za algoritme koji zahtevaju ulaz u obliku sloja.

See also

Kreiranje sloja iz obuhvata

Parametri

Oznaka	Назив	Tip	Opis
Tačka	`INPUT`	[koordinate]	Ulazna tačka, uključujući podatak o CRS-u (primer: `397254,6214446 [EPSG:32632]`). Ako se CRS ne zada, koristi se CRS projekta. Tačka se može zadati klikom na platno karte.
Tačka	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačka	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj koji sadrži ulaznu tačku.

Python kod

ID algoritma: `native:pointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje tačkastog sloja iz tabele

Pravi tačkasti sloj od tabele sa kolonama koje sadrže polja koordinata.

Pored X i Y koordinata možete zadati i Z i M polja.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj ili tabela.

наставак на следећој страни

Табела 24.505 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
X polje	`XFIELD`	[polje tabele: bilo koji]	Polje koje sadrži X koordinatu
Y polje	`YFIELD`	[polje tabele: bilo koji]	Polje koje sadrži Y koordinatu
Z polje Opciono	`ZFIELD`	[polje tabele: bilo koji]	Polje koje sadrži Z koordinatu
M polje Opciono	`MFIELD`	[polje tabele: bilo koji]	Polje koje sadrži M vrednost
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: `EPSG:4326`	Koordinatni referentni sistem koji se koristi za sloj. Pretpostavlja se da zadate koordinate njemu odgovaraju.
Tačke iz tabele	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte rezultujući tačkasti sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačke iz tabele	`OUTPUT`	[vektor: tačka]	Rezultujući tačkasti sloj

Python kod

ID algoritma: `native:createpointslayerfromtable`

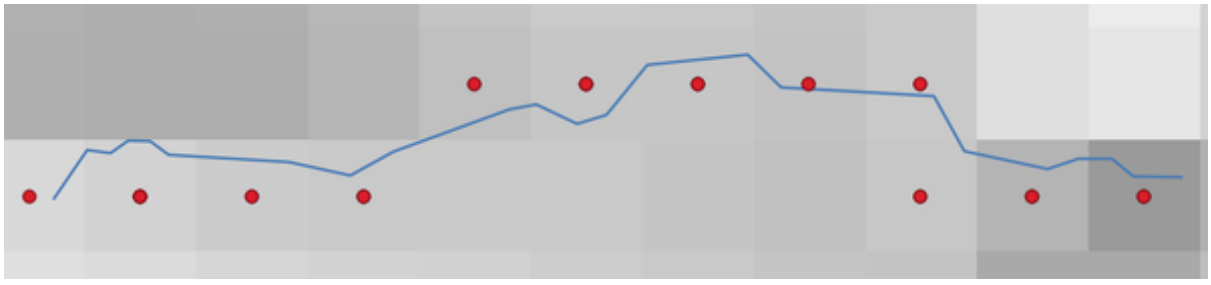
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Generisanje tačaka (centroida piksela) duž linije

Generiše tačkasti vektorski sloj iz ulaznog rasterskog i linijskog sloja.

Tačke odgovaraju centroidima piksela koji presecaju linijski sloj.



Слика24.90: Tačke centroida piksela

Parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Vektorski sloj	`INPUT_VECTOR`	[vektor: linija]	Ulazni linijski vektorski sloj
Tačke duž linije	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Rezultujući tačkasti sloj sa centroidima piksela. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačke duž linije	`OUTPUT`	[vektor: tačka]	Rezultujući tačkasti sloj sa centroidima piksela

Python kod

ID algoritma: `qgis:generatepointspixelcentroidsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

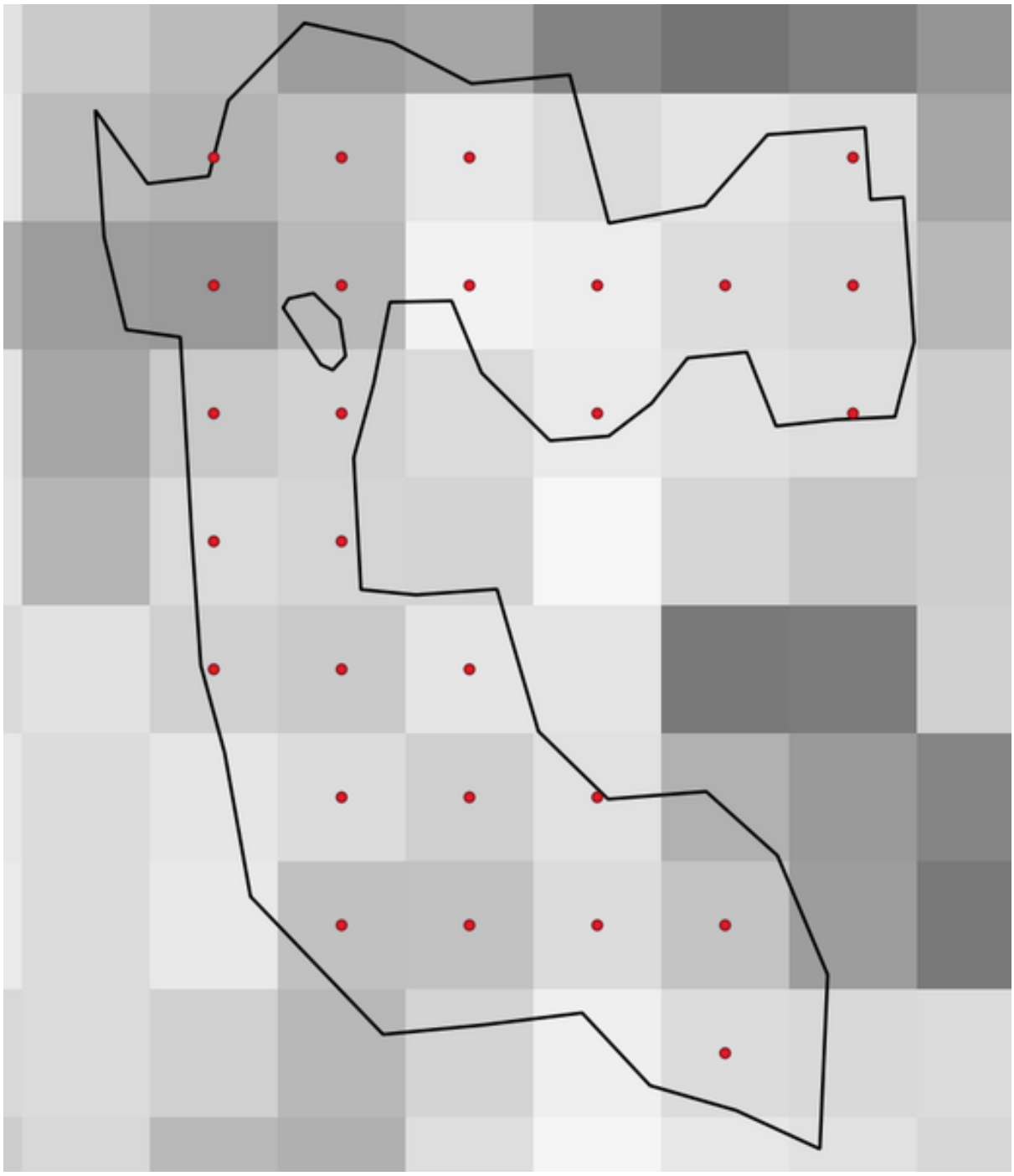
Generisanje tačaka (centroida piksela) unutar poligona

Generiše tačkasti vektorski sloj iz ulaznog rasterskog i poligonskog sloja.

Tačke odgovaraju centroidima piksela koji presecaju poligonski sloj.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.



Слика24.91: Tačke centroida piksela

Parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Vektorski sloj	`INPUT_VECTOR`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Tačke unutar poligona	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Rezultujući tačkasti sloj centroida piksela. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačke unutar poligona	`OUTPUT`	[vektor: tačka]	Rezultujući tačkasti sloj centroida piksela

Python kod

ID algoritma: `native:generatepointspixelcentroidsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uvoz geotagovanih fotografija

Pravi tačkasti sloj koji odgovara geotagovanim lokacijama iz JPEG ili HEIC/HEIF slika iz izvornog foldera.

Tačkasti sloj sadržaće po jedan PointZ objekat za svaku ulaznu datoteku iz koje su geotagovi mogli da se pročitaju. Podatak o nadmorskoj visini iz geotagova koristi se za postavljanje Z vrednosti tačke.

Pored geografske dužine i širine, tački se kao atributi dodaju i podaci o nadmorskoj visini, pravcu i vremenskoj oznaci, ako postoje u fotografiji.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni folder	`FOLDER`	[folder]	Putanja do izvornog foldera sa geotagovanim fotografijama
Skeniraj rekurzivno	`RECURSIVE`	[logička vrednost] Podrazumevano: False	Ako je označeno, skeniraće se folder i njegovi podfolderi

наставак на следећој страни

Табела 24.511 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Fotografije Opciono	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte tačkasti vektorski sloj za geotagovane fotografije. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Tabela neispravnih fotografija Opciono	`INVALID`	[vektor: tabela] Podrazumevano: `[Skip output]`	Zadajte tabelu nečitljivih ili negeotagovanih fotografija. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Fotografije	`OUTPUT`	[vektor: tačka]	Tačkasti vektorski sloj sa geotagovanim fotografijama. Formular sloja automatski se popunjava putanjama i podešavanjima pregleda fotografija.
Tabela neispravnih fotografija Opciono	`INVALID`	[vektor: tabela]	Može se napraviti i tabela nečitljivih ili negeotagovanih fotografija.

Python kod

ID algoritma: `native:importphotos`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Tačke u putanju

Pretvara tačkasti sloj u linijski spajanjem tačaka redosledom koji definiše izraz ili polje u ulaznom tačkastom sloju.

Tačke se mogu grupisati po polju ili izrazu radi razlikovanja linijskih objekata.

Pored linijskog vektorskog sloja, kao izlaz se dobija i tekstualna datoteka koja opisuje rezultujuću liniju kao početnu tačku i niz azimuta / pravaca i rastojanja.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Napravi zatvorene putanje	`CLOSE_PATH`	[logička vrednost] Podrazumevano: False	Ako je označeno, prva i poslednja tačka linije biće povezane i zatvoriće generisanu putanju
Izraz za redosled Opciono	`ORDER_EXPRESSION`	[izraz]	Polje ili izraz koji daje redosled povezivanja tačaka u putanji. Ako se ne postavi, koristi se ID objekta (`$id`).
Sortiraj tekst sa brojevima prirodno	`NATURAL_SORT`	[logička vrednost] Podrazumevano: False	Ako je označeno, prirodno sortira objekte na osnovu zadatog izraza (tj. 'a9' < 'a10').
Izraz za grupisanje putanja Opciono	`GROUP_EXPRESSION`	[izraz]	Tačkasti objekti sa istom vrednošću u polju ili izrazu biće grupisani u istu liniju. Ako se ne postavi, crta se jedna putanja sa svim ulaznim tačkama.
Putanje	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte linijski vektorski sloj putanje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Direktorijum za tekstualni izlaz Opciono	`OUTPUT_TEXT_DIR`	[folder] Podrazumevano: `[Skip output]`	Zadajte direktorijum koji će sadržati opisne datoteke tačaka i putanja. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Izlazi

Oznaka	Назив	Tip	Opis
Putanje	`OUTPUT`	[vektor: linija]	Linijski vektorski sloj putanje
Direktorijum za tekstualni izlaz	`OUTPUT_TEXT_DIR`	[folder]	Direktorijum sa opisnim datotekama tačaka i putanja

Python kod

ID algoritma: `native:pointstopath`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

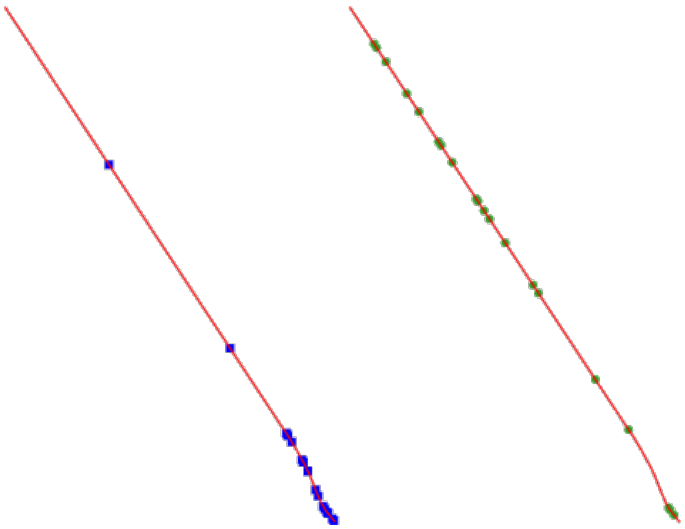
Slučajne tačke duž linije

Pravi novi tačkasti sloj sa tačkama postavljenim na linije drugog sloja.

Za svaku liniju u ulaznom sloju, zadati broj tačaka dodaje se u rezultujući sloj. Postupak dodavanja tačke je:

1. slučajno izabrati linijski objekat iz ulaznog sloja
2. ako je objekat višedelni, slučajno izabrati jedan njegov deo
3. slučajno izabrati segment te linije
4. slučajno izabrati položaj na tom segmentu.

Ovaj postupak znači da će zakrivljeni delovi linija (sa relativno kratkim segmentima) dobiti više tačaka nego pravi delovi (sa relativno dugim segmentima), kao što je prikazano na ilustraciji ispod, gde se izlaz algoritma *Slučajne tačke duž linije* može uporediti sa izlazom algoritma *Slučajne tačke na linijama* (koji daje tačke sa, u proseku, ravnomernom raspodelom duž linija).



Слика24.92: Primer izlaza algoritma. Levo: *Slučajne tačke duž linije*, desno: *Slučajne tačke na linijama*

Može se zadati najmanje rastojanje kako tačke ne bi bile preblizu jedna drugoj.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Slučajne tačke na linijama

Parametri

Oznaka	Назив	Tip	Opis
Ulazni tačkasti sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj

наставак на следећој страни

Табела 24.515 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj tačaka	`POINTS_NUMBER`	[numerički: ceo broj] Podrazumevano: 1	Broj tačaka koje se kreiraju
Najmanje rastojanje između tačaka	`MIN_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka
Slučajne tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke	`OUTPUT`	[vektor: tačka]	Izlazni sloj slučajnih tačaka.

Python kod

ID algoritma: `qgis:qgisrandompointsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajne tačke u obuhvatu

Pravi novi tačkasti sloj sa zadatim brojem slučajnih tačaka, sve unutar zadatog obuhvata.

Može se zadati faktor rastojanja kako tačke ne bi bile preblizu jedna drugoj. Ako najmanje rastojanje između tačaka onemogući kreiranje novih tačaka, može se smanjiti rastojanje ili povećati najveći broj pokušaja.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni obuhvat	`EXTENT`	[obuhvat]	Obuhvat karte za slučajne tačke Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Broj tačaka	`POINTS_NUMBER`	[numerički: ceo broj] Podrazumevano: 1	Broj tačaka koje se kreiraju
Najmanje rastojanje između tačaka	`MIN_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: *CRS projekta*	CRS sloja slučajnih tačaka
Slučajne tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Najveći broj pokušaja pretrage pri zadatom najmanjem rastojanju	`MAX_ATTEMPTS`	[numerički: ceo broj] Podrazumevano: 200	Najveći broj pokušaja postavljanja tačaka

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke	`OUTPUT`	[vektor: tačka]	Izlazni sloj slučajnih tačaka.

Python kod

ID algoritma: `native:randompointsinextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajne tačke u granicama sloja

Pravi novi tačkasti sloj sa zadatim brojem slučajnih tačaka, sve unutar obuhvata zadatog sloja.

Može se zadati najmanje rastojanje kako tačke ne bi bile preblizu jedna drugoj.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski sloj koji definiše područje
Broj tačaka	`POINTS_NUMBER`	[numerički: ceo broj] Podrazumevano: 1	Broj tačaka koje se kreiraju
Najmanje rastojanje između tačaka	`MIN_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka
Slučajne tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke	`OUTPUT`	[vektor: tačka]	Izlazni sloj slučajnih tačaka.

Python kod

ID algoritma: `qgis:randompointsinlayerbounds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

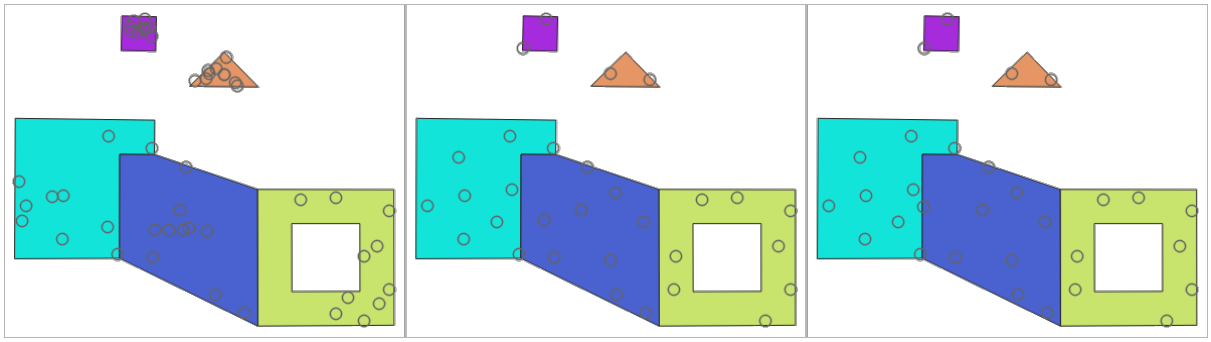
Slučajne tačke u poligonima

Pravi tačkasti sloj sa tačkama postavljenim unutar poligona drugog sloja.

Za geometriju svakog objekta (poligon / multipoligon) u ulaznom sloju, zadati broj tačaka dodaje se u rezultujući sloj.

Mogu se zadati najmanja rastojanja po objektu i globalno, kako tačke u izlaznom tačkastom sloju ne bi bile preblizu. Ako se zada najmanje rastojanje, možda neće biti moguće generisati zadati broj tačaka za svaki objekat. Ukupan broj generisanih i propuštenih tačaka dostupan je kao izlaz algoritma.

Ilustracija ispod prikazuje efekat najmanjih rastojanja po objektu i globalnih, kao i nultih/nenultih najmanjih rastojanja (generisano sa istim semenom, pa će bar prva generisana tačka biti ista).



Слика24.93: Deset tačaka po poligonskom objektu, *levo*: min. rastojanja = 0, *sredina*: min. rastojanja = 1, *desno*: min. rastojanje = 1, globalno min. rastojanje = 0

Može se zadati najveći broj pokušaja po tački. To je bitno samo za nenulto najmanje rastojanje.

Može se zadati seme za generator slučajnih brojeva, čime se omogućava dobijanje identičnih nizova slučajnih brojeva u različitim pokretanjima algoritma.

Mogu se uključiti atributi poligonskog objekta na kome je tačka generisana (*Uključi atribute poligona*).

Ako želite približno istu gustinu tačaka za sve objekte, broj tačaka možete definisati podacima koristeći površinu geometrije poligonskog objekta.

See also

Slučajne tačke unutar poligona

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni poligonski sloj	`INPUT`	[vektor: linija]	Ulazni poligonski vektorski sloj

наставак на следећој страни

Табела 24.522 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj tačaka za svaki objekat	POINTS_NUMBER	[numerički: ceo broj] Podrazumevano: 1	Broj tačaka koje se kreiraju
Najmanje rastojanje između tačaka Opciono	MIN_DISTANCE	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka unutar jednog poligonskog objekta
Slučajne tačke u poligonima	OUTPUT	[vektor: tačka] Podrazumevano: [Create temporary layer]	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Globalno najmanje rastojanje između tačaka Opciono	MIN_DISTANCE_GLC	[numerički: decimalni] Podrazumevano: 0.0	Globalno najmanje rastojanje između tačaka. Treba da bude manje od *Najmanje rastojanje između tačaka (po objektu)* da bi taj parametar imao efekta.
Najveći broj pokušaja pretrage (za min. rast. > 0) Opciono	MAX_TRIES_PER_PC	[numerički: ceo broj] Podrazumevano: 10	Najveći broj pokušaja po tački. Bitno samo ako je zadato najmanje rastojanje između tačaka (i veće je od 0).
Seme slučajnosti Opciono	SEED	[numerički: ceo broj] Podrazumevano: Nije postavljeno	Seme koje se koristi za generator slučajnih brojeva.
Uključi atribute poligona	INCLUDE_POLYGON_ATTRI	[logička vrednost] Podrazumevano: True	Ako je postavljeno, tačka dobija atribute linije na kojoj je postavljena.

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke u poligonima	OUTPUT	[vektor: tačka]	Izlazni sloj slučajnih tačaka.
Broj objekata sa praznom geometrijom ili bez nje	FEATURES_WITH_EMPTY_C	[numerički: ceo broj]	
Ukupan broj generisanih tačaka	OUTPUT_POINTS	[numerički: ceo broj]	
Broj propuštenih tačaka	POINTS_MISSED	[numerički: ceo broj]	Broj tačaka koje nisu mogle da budu generisane zbog ograničenja najmanjeg rastojanja.

наставак на следећој страни

Табела 24.524 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj objekata sa propuštenim tačkama	`POLYGONS_WITH_MISS`	[numerički: ceo broj]	Ne uključuje objekte sa praznom geometrijom ili bez nje

Python kod

ID algoritma: `native:randompointsinpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajne tačke unutar poligona

Pravi novi tačkasti sloj sa zadatim brojem slučajnih tačaka unutar svakog poligona ulaznog poligonskog sloja.

Dostupne su dve strategije uzorkovanja:

- Broj tačaka: broj tačaka za svaki objekat
- Gustina tačaka: gustina tačaka za svaki objekat

Može se zadati najmanje rastojanje kako tačke ne bi bile preblizu jedna drugoj.

Podrazumevani meni: *Vektor* ► *Istraživačke alatke*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Slučajne tačke u poligonima

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Strategija uzorkovanja	`STRATEGY`	[nabrajanje] Podrazumevano: 0	Strategija uzorkovanja koja se koristi. Jedno od: • 0 — Broj tačaka: broj tačaka za svaki objekat • 1 — Gustina tačaka: gustina tačaka za svaki objekat
Broj ili gustina tačaka	`VALUE`	[numerički: decimalni] Podrazumevano: 1.0	Broj ili gustina tačaka, u zavisnosti od izabrane *Strategija uzorkovanja*.

наставак на следећој страни

Табела 24.525 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Najmanje rastojanje između tačaka	`MIN_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka
Slučajne tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke	`OUTPUT`	[vektor: tačka]	Izlazni sloj slučajnih tačaka.

Python kod

ID algoritma: `qgis:randompointsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajne tačke na linijama

Pravi tačkasti sloj sa tačkama postavljenim na linije drugog sloja.

Za geometriju svakog objekta (linija / multilinija) u ulaznom sloju, zadati broj tačaka dodaje se u rezultujući sloj.

Mogu se zadati najmanja rastojanja po objektu i globalno, kako tačke u izlaznom tačkastom sloju ne bi bile preblizu. Ako se zada najmanje rastojanje, možda neće biti moguće generisati zadati broj tačaka za svaki objekat. Ukupan broj generisanih i propuštenih tačaka dostupan je kao izlaz algoritma.

Ilustracija ispod prikazuje efekat najmanjih rastojanja po objektu i globalnih, kao i nultih/nenultih najmanjih rastojanja (generisano sa istim semenom, pa će bar prva generisana tačka biti ista).



Слика24.94: Pet tačaka po linijskom objektu, *levo*: min. rastojanja = 0, *sredina*: min. rastojanja != 0, *desno*: min. rastojanje != 0, globalno min. rastojanje = 0

Može se zadati najveći broj pokušaja po tački. To je bitno samo za nenulto najmanje rastojanje.

Može se zadati seme za generator slučajnih brojeva, čime se omogućava dobijanje identičnih nizova slučajnih brojeva u različitim pokretanjima algoritma.

Mogu se uključiti atributi linijskog objekta na kome je tačka generisana (*Uključi atribute linije*).

Ako želite približno istu gustinu tačaka za sve linijske objekte, broj tačaka možete definisati podacima koristeći dužinu geometrije linijskog objekta.

See also

Slučajne tačke duž linije

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni linijski sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Broj tačaka za svaki objekat	`POINTS_NUMBER`	[numerički: ceo broj] Podrazumevano: 1	Broj tačaka koje se kreiraju

наставак на следећој страни

Табела 24.527 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Najmanje rastojanje između tačaka (po objektu) Opciono	MIN_DISTANCE	[numerički: decimalni] Podrazumevano: 0.0	Najmanje rastojanje između tačaka unutar jednog linijskog objekta
Slučajne tačke na linijama	OUTPUT	[vektor: tačka] Podrazumevano: [Create temporary layer]	Izlazne slučajne tačke. *Jedno od*: • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Globalno najmanje rastojanje između tačaka Opciono	MIN_DISTANCE_GLC	[numerički: decimalni] Podrazumevano: 0.0	Globalno najmanje rastojanje između tačaka. Treba da bude manje od *Najmanje rastojanje između tačaka (po objektu)* da bi taj parametar imao efekta.
Najveći broj pokušaja pretrage (za min. rast. > 0) Opciono	MAX_TRIES_PER_PC	[numerički: ceo broj] Podrazumevano: 10	Najveći broj pokušaja po tački. Bitno samo ako je zadato najmanje rastojanje između tačaka (i veće je od 0).
Seme slučajnosti Opciono	SEED	[numerički: ceo broj] Podrazumevano: Nije postavljeno	Seme koje se koristi za generator slučajnih brojeva.
Uključi atribute linije	INCLUDE_LINE_ATTRIBUT	[logička vrednost] Podrazumevano: True	Ako je postavljeno, tačka dobija atribute linije na kojoj je postavljena.

Izlazi

Oznaka	Назив	Tip	Opis
Slučajne tačke na linijama	OUTPUT	[vektor: tačka]	Izlazni sloj slučajnih tačaka.
Broj objekata sa praznom geometrijom ili bez nje	FEATURES_WITH_EMPTY_C	[numerički: ceo broj]	
Broj objekata sa propuštenim tačkama	LINES_WITH_MISSED_	[numerički: ceo broj]	Ne uključuje objekte sa praznom geometrijom ili bez nje
Ukupan broj generisanih tačaka	POINTS_GENERATED	[numerički: ceo broj]	
Broj propuštenih tačaka	POINTS_MISSED	[numerički: ceo broj]	Broj tačaka koje nisu mogle da budu generisane zbog ograničenja najmanjeg rastojanja.

Python kod

ID algoritma: `native:randompointsonlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pikseli rastera u tačke

Pravi vektorski sloj tačaka koje odgovaraju svakom pikselu rasterskog sloja.

Pretvara rasterski sloj u vektorski tako što pravi tačkaste objekte za središte svakog pojedinačnog piksela rasterskog sloja. Pikseli sa NoData preskaču se u izlazu.

Parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Broj kanala	`RASTER_BAND`	[rasterski kanal]	Rasterski kanal iz koga se izdvajaju podaci
Naziv polja	`FIELD_NAME`	[niska] Podrazumevano: 'VALUE'	Naziv polja u koje se čuva vrednost rasterskog kanala
Vektorske tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte rezultujući tačkasti sloj centroida piksela. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Vektorske tačke	`OUTPUT`	[vektor: tačka]	Rezultujući tačkasti sloj sa centroidima piksela

Python kod

ID algoritma: `native:pixelstopoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pikseli rastera u poligone

Pravi vektorski sloj poligona koji odgovaraju svakom pikselu rasterskog sloja.

Pretvara rasterski sloj u vektorski tako što pravi poligonske objekte za obuhvat svakog pojedinačnog piksela rasterskog sloja. Pikseli sa NoData preskaču se u izlazu.

Parametri

Oznaka	Назив	Tip	Opis
Rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Broj kanala	`RASTER_BAND`	[rasterski kanal]	Rasterski kanal iz koga se izdvajaju podaci
Naziv polja	`FIELD_NAME`	[niska] Podrazumevano: 'VALUE'	Naziv polja u koje se čuva vrednost rasterskog kanala
Vektorski poligoni	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte rezultujući poligonski sloj obuhvata piksela. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Vektorski poligoni	`OUTPUT`	[vektor: poligon]	Rezultujući poligonski sloj obuhvata piksela

Python kod

ID algoritma: `native:pixelstopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pravilne tačke

Pravi novi tačkasti sloj čije su tačke raspoređene u pravilnu mrežu unutar zadatog obuhvata.

Mreža se zadaje razmakom između tačaka (isti razmak za sve dimenzije) ili brojem tačaka koje se generišu. U drugom slučaju, razmak se određuje iz obuhvata. Da bi se generisala puna pravougaona mreža, u drugom slučaju generiše se najmanje onoliko tačaka koliko je korisnik zadao.

Na razmak tačaka mogu se primeniti slučajni pomeraji, što daje nepravilan raspored tačaka.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Obuhvat karte za slučajne tačke Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Razmak/broj tačaka	`SPACING`	[numerički: ceo broj] Podrazumevano: 100	Razmak između tačaka ili broj tačaka, u zavisnosti od toga da li je `Use point spacing` označeno ili nije.
Početno uvlačenje od ugla (leva strana)	`INSET`	[numerički: decimalni] Podrazumevano: 0.0	Pomera tačke u odnosu na gornji levi ugao. Vrednost se koristi i za X i za Y osu.
Primeni slučajni pomeraj na razmak tačaka	`RANDOMIZE`	[logička vrednost] Podrazumevano: False	Ako je označeno, tačke će imati slučajan razmak
Koristi razmak tačaka	`IS_SPACING`	[logička vrednost] Podrazumevano: True	Ako nije označeno, razmak tačaka se ne uzima u obzir
CRS izlaznog sloja	`CRS`	[crs] Podrazumevano: *CRS projekta*	CRS sloja slučajnih tačaka
Pravilne tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj pravilnih tačaka. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pravilne tačke	`OUTPUT`	[vektor: tačka]	Izlazni sloj pravilnih tačaka.

Python kod

ID algoritma: `qgis:regularpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.25 Vektor opšte

Dodela projekcije

Dodeljuje novu projekciju vektorskom sloju.

Pravi novi sloj sa potpuno istim objektima i geometrijama kao ulazni, ali dodeljen novom CRS-u. Geometrije se **ne** reprojektuju, samo im se dodeljuje drugi CRS.

Ovaj algoritam se može koristiti za popravku slojeva kojima je dodeljena pogrešna projekcija.

Ovaj algoritam ne menja atribute.

See also

Definisanje projekcije, *Pronalaženje projekcije*, *Reprojektovanje sloja*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Vektorski sloj sa pogrešnim CRS-om ili bez njega
Dodeljeni CRS	`CRS`	[crs] Podrazumevano: `EPSG:4326 – WGS84`	Izaberite novi CRS koji se dodeljuje vektorskom sloju
Dodeljeni CRS Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Dodeljeni CRS	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa dodeljenom projekcijom

Python kod

ID algoritma: `native:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Grupno geokodiranje preko Nominatima

Vrši grupno geokodiranje pomoću servisa Nominatim nad tekstualnim poljem ulaznog sloja. Izlazni sloj imaće geometriju tačke koja odgovara geokodiranoj lokaciji, kao i više atributa povezanih sa tom lokacijom.

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Белешка

Ovaj algoritam je usklađen sa pravilima korišćenja servisa za geokodiranje Nominatim koji obezbeđuje OpenStreetMap Foundation.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj čiji se objekti geokodiraju
Polje sa adresom	`FIELD`	[polje tabele: niska]	Polje koje sadrži adrese za geokodiranje
Geokodirano	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj koji sadrži samo geokodirane adrese. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Geokodirano	`OUTPUT`	[vektor: tačka]	Vektorski sloj sa tačkastim objektima koji odgovaraju geokodiranim adresama

Python kod

ID algoritma: `native:batchnominatimgeocoder`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvaranje sloja u prostorne obeleživače

Pravi prostorne obeleživače koji odgovaraju obuhvatu objekata sadržanih u sloju.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni vektorski sloj
Odredište obeleživača	`DESTINATION`	[nabrajanje] Podrazumevano: 0	Izaberite odredište za obeleživače. Jedno od: • 0 — Obeleživači projekta • 1 — Korisnički obeleživači
Polje sa nazivom	`NAME_EXPRESSION`	[izraz]	Polje ili izraz koji će dati nazive generisanim obeleživačima
Polje sa grupom	`GROUP_EXPRESSION`	[izraz]	Polje ili izraz koji će obezbediti grupe za generisane obeleživače

Izlazi

Oznaka	Назив	Tip	Opis
Broj dodatih obeleživača	`COUNT`	[numerički: ceo broj]	

Python kod

ID algoritma: `native:layertobookmarks`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvaranje prostornih obeleživača u sloj

Pravi novi sloj sa poligonskim objektima za sačuvane prostorne obeleživače. Izvoz se može ograničiti samo na obeleživače tekućeg projekta, na sve korisničke obeleživače ili na kombinaciju oba.

Parametri

Oznaka	Назив	Tip	Opis
Izvor obeleživača	`SOURCE`	[nabrajanje] [lista] Podrazumevano: [0,1]	Izaberite izvor(e) obeleživača. Jedno ili više od: • 0 — Obeleživači projekta • 1 — Korisnički obeleživači

наставак на следећој страни

Табела 24.542 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni CRS	`CRS`	[crs] Podrazumevano: `EPSG:4326 – WGS 84`	CRS izlaznog sloja
Izlaz	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj (obeleživači)

Python kod

ID algoritma: `native:bookmarkstolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje indeksa atributa

Pravi indeks nad poljem atributske tabele radi ubrzavanja upita. Podrška za kreiranje indeksa zavisi od provajdera podataka sloja i od tipa polja.

Ne kreiraju se izlazi: indeks se čuva u samom sloju.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Izaberite vektorski sloj za koji želite da napravite indeks atributa
Atribut za indeksiranje	`FIELD`	[polje tabele: bilo koji]	Polje vektorskog sloja

Izlazi

Oznaka	Назив	Tip	Opis
Indeksirani sloj	`OUTPUT`	[isto kao ulaz]	Kopija ulaznog vektorskog sloja sa indeksom za zadato polje

Python kod

ID algoritma: `native:createattributeindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kreiranje prostornog indeksa

Pravi indeks koji ubrzava pristup objektima u sloju na osnovu njihove prostorne lokacije. Podrška za kreiranje prostornog indeksa zavisi od provajdera podataka sloja.

Ne kreiraju se novi izlazni slojevi.

Podrazumevani meni: *Vektor ► Alatke za upravljanje podacima*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj

Izlazi

Oznaka	Назив	Tip	Opis
Indeksirani sloj	`OUTPUT`	[isto kao ulaz]	Kopija ulaznog vektorskog sloja sa prostornim indeksom

Python kod

ID algoritma: `native:createspatialindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Definisanje projekcije

Postavlja projekciju postojećeg sloja na zadati CRS bez reprojektovanja objekata.

Za razliku od algoritma *Dodela projekcije*, menja tekući sloj i ne pravi novi sloj.

Белешка

Za Shapefile skupove podataka, datoteke `.prj` i `.qpj` biće prepisane - ili napravljene ako nedostaju - da odgovaraju zadatom CRS-u.

Podrazumevani meni: *Vektor ► Alatke za upravljanje podacima*

See also

Dodela projekcije, *Pronalaženje projekcije*, *Reprojektovanje sloja*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Vektorski sloj bez podataka o projekciji
CRS	`CRS`	[crs]	Izaberite CRS koji se dodeljuje vektorskom sloju

Izlazi

Oznaka	Назив	Tip	Opis
	`INPUT`	[isto kao ulaz]	Ulazni vektorski sloj sa definisanom projekcijom

Python kod

ID algoritma: `native:definecurrentprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Brisanje duplih geometrija

Pronalazi i uklanja duplirane geometrije.

Atributi se ne proveravaju, pa ako dva objekta imaju identične geometrije a različite atribute, samo jedan od njih biće dodat u rezultujući sloj.

Белешка

Ovaj algoritam ne zahteva ispravne geometrije na ulazu.

See also

Uklanjanje geometrija, *Ukloni nulte geometrije*, *Brisanje duplikata po atributu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj sa duplim geometrijama koji želite da očistite

наставак на следећој страни

Табела 24.550 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Očišćeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj koji sadrži jedinstvene objekte. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj odbačenih dupliranih slogova	`DUPLICATE_COUNT`	[numerički: ceo broj]	Broj odbačenih dupliranih slogova
Očišćeno	`OUTPUT`	[isto kao ulaz]	Izlazni sloj bez ijedne duplirane geometrije
Broj zadržanih slogova	`RETAINED_COUNT`	[numerički: ceo broj]	Broj jedinstvenih slogova

Python kod

ID algoritma: `native:deleteduplicategeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Brisanje duplikata po atributu

Briše duplirane redove uzimajući u obzir samo zadata polja. Prvi red koji se poklapa biće zadržan, a duplikati odbačeni.

Opciono, ovi duplirani slogovi mogu se sačuvati u zaseban izlaz radi analize.

See also

Brisanje duplih geometrija

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni sloj
Polja po kojima se traže duplikati	`FIELDS`	[polje tabele: bilo koji] [lista]	Polja koja definišu duplikate. Objekti sa identičnim vrednostima u svim ovim poljima smatraju se duplikatima.

наставак на следећој страни

Табела 24.552 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Filtrirano (bez duplikata)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj koji sadrži jedinstvene objekte. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Filtrirano (duplikati) Opciono	`DUPLICATES`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte izlazni sloj koji sadrži samo duplikate. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Filtrirano (duplikati)	`DUPLICATES`	[isto kao ulaz]	Vektorski sloj koji sadrži duplirane objekte.
Broj odbačenih dupliranih slogova	`DUPLICATE_COUNT`	[numerički: ceo broj]	Broj odbačenih dupliranih slogova
Filtrirano (bez duplikata)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj koji sadrži jedinstvene objekte.
Broj zadržanih slogova	`RETAINED_COUNT`	[numerički: ceo broj]	Broj jedinstvenih slogova

Python kod

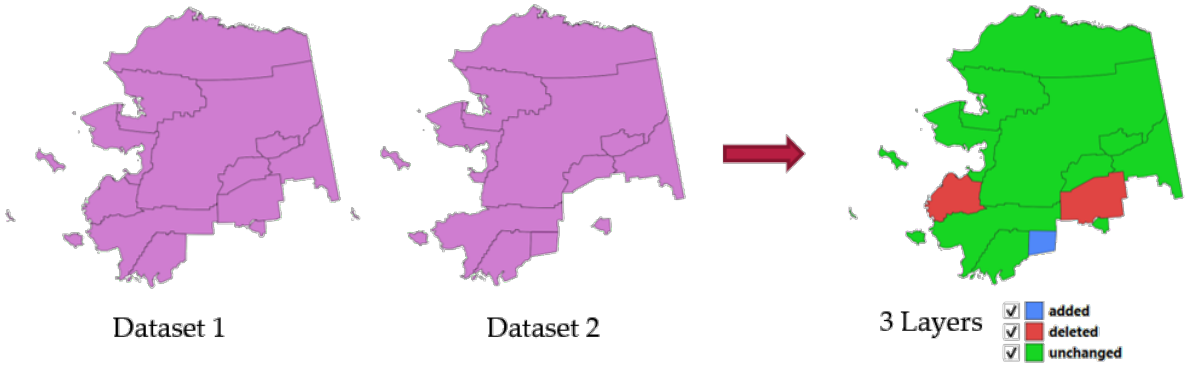
ID algoritma: `native:removeduplicatesbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Otkrivanje promena u skupu podataka

Poredi dva vektorska sloja i utvrđuje koji su objekti nepromenjeni, dodati ili obrisani između njih. Namenjen je poređenju dve različite verzije istog skupa podataka.



Слика24.95: Primer otkrivanja promene u skupu podataka

Parametri

Oznaka	Назив	Tip	Opis
Originalni sloj	`ORIGINAL`	[vektor: geometrija]	Vektorski sloj koji se smatra originalnom verzijom
Izmenjeni sloj	`REVISED`	[vektor: geometrija]	Izmenjeni ili modifikovani vektorski sloj
Atributi koji se uzimaju u obzir pri poklapanju Opciono	`COMPARE_ATTRIBUTES`	[polje tabele: bilo koji] [lista]	Atributi koji se uzimaju u obzir pri poklapanju. Podrazumevano se porede svi atributi.
Način poređenja geometrija Opciono	`MATCH_TYPE`	[nabrajanje] Podrazumevano: 1	Definiše kriterijume poređenja. Opcije: • 0 — Tačno poklapanje: uključuje redosled i broj temena geometrija • 1 — Tolerantno poklapanje (topološka jednakost): geometrije se smatraju jednakim
Nepromenjeni objekti Opciono	`UNCHANGED`	[vektor: isto kao originalni sloj]	Zadajte izlazni vektorski sloj koji sadrži nepromenjene objekte. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Dodati objekti Opciono	`ADDED`	[vektor: isto kao originalni sloj]	Zadajte izlazni vektorski sloj koji sadrži dodate objekte. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.554 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obrisani objekti Opciono	DELETED	[vektor: isto kao originalni sloj]	Zadajte izlazni vektorski sloj koji sadrži obrisane objekte. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Nepromenjeni objekti	UNCHANGED	[vektor: isto kao originalni sloj]	Vektorski sloj koji sadrži nepromenjene objekte.
Dodati objekti	ADDED	[vektor: isto kao originalni sloj]	Vektorski sloj koji sadrži dodate objekte.
Obrisani objekti	DELETED	[vektor: isto kao originalni sloj]	Vektorski sloj koji sadrži obrisane objekte.
Broj nepromenjenih objekata	UNCHANGED_COUNT	[numerički: ceo broj]	Broj nepromenjenih objekata.
Broj objekata dodatih u izmenjenom sloju	ADDED_COUNT	[numerički: ceo broj]	Broj objekata dodatih u izmenjenom sloju.
Broj objekata obrisanih iz originalnog sloja	DELETED_COUNT	[numerički: ceo broj]	Broj objekata obrisanih iz originalnog sloja.

Python kod

ID algoritma: `native:detectvectorchanges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uklanjanje geometrija

Pravi jednostavnu kopiju atributske tabele ulaznog sloja *bez geometrije*. Zadržava atributsku tabelu izvornog sloja.

Ako se datoteka čuva u lokalnom folderu, možete birati između mnogih formata datoteka.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Brisanje duplih geometrija, *Ukloni nulte geometrije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Uklonjene geometrije	`OUTPUT`	[vektor: tabela]	Zadajte izlazni sloj bez geometrije. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Uklonjene geometrije	`OUTPUT`	[vektor: tabela]	Izlazni sloj bez geometrije. Kopija originalne atributske tabele.

Python kod

ID algoritma: `native:dropgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvršavanje SQL-a

Pokreće jednostavan ili složen upit zasnovan isključivo na SELECT-u sa `SQL` sintaksom nad izvornim slojem.

Ulazni izvori podataka označavaju se sa `input1`, `input2`… `inputN`, a jednostavan upit izgleda ovako: `SELECT * FROM input1`.

Pored jednostavnog upita, možete dodati izraze ili promenljive unutar samog parametra `SQL query`. To je naročito korisno ako se algoritam izvršava unutar modela Obrade i želite da ulaz modela upotrebite kao parametar upita. Primer takvog upita je `SELECT * FROM [% @table %]`, gde je `@table` promenljiva koja označava ulaz modela.

Rezultat upita biće dodat kao novi sloj.

See also

SpatiaLite izvrši SQL, *PostgreSQL izvrši SQL*

Parametri

Oznaka	Назив	Tip	Opis
Dodatni ulazni izvori podataka (u upitu se zovu input1, .., inputN)	`INPUT_DATASOURCES`	[vektor: bilo koji] [lista]	Lista slojeva nad kojima se izvršava upit. U SQL editoru na te slojeve možete se pozvati njihovim **pravim** nazivom ili sa **input1**, **input2**, **inputN**, u zavisnosti od toga koliko je slojeva izabrano.
SQL upit	`INPUT_QUERY`	[niska]	Ukucajte tekst svog SQL upita, npr. `SELECT * FROM input1`.
Polje sa jedinstvenim identifikatorom Opciono	`INPUT_UID_FIELD`	[niska]	Zadajte kolonu sa jedinstvenim ID-om
Polje sa geometrijom Opciono	`INPUT_GEOMETRY_FIE`	[niska]	Zadajte polje sa geometrijom
Tip geometrije Opciono	`INPUT_GEOMETRY_TYP`	[nabrajanje] Podrazumevano: 0	Izaberite geometriju rezultata. Podrazumevano je algoritam sam otkriva. Jedno od: • 0 — Automatsko otkrivanje • 1 — Bez geometrije • 2 — Tačka • 3 — LineString • 4 — Poligon • 5 — MultiPoint • 6 — MultiLineString • 7 — MultiPolygon
CRS Opciono	`INPUT_GEOMETRY_CRS`	[crs]	CRS koji se dodeljuje izlaznom sloju
SQL izlaz	`OUTPUT`	[vektor: geometrija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj koji upit kreira. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
SQL izlaz	`OUTPUT`	[vektor: geometrija]	Vektorski sloj koji je upit kreirao

Python kod

ID algoritma: `qgis:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz slojeva u DXF

Izvozi slojeve u DXF datoteku. Za svaki sloj možete izabrati polje čije se vrednosti koriste za razdvajanje objekata u generisane odredišne slojeve u DXF izlazu.

See also

Creating new DXF files

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[vektor: geometrija] [lista]	Lista ulaznih vektorskih slojeva sa pripadajućim opcijama (popunjava se kao stavka `QgsProcessingParameterDxfLayers` — u grafičkom interfejsu pritiskom na dugme *Podesi sloj…* za svaki izabrani sloj): **Sloj [niska] (`layer`)** Puna putanja ulaznog sloja koji se izvozi **Atribut izlaznog sloja [polje tabele: bilo koji] (`attributeIndex`)** Indeks atributa po kome se ulazni sloj deli koristeći jedinstvene vrednosti **Naziv izlaznog sloja [niska] (`overriddenLayerName`)** Zamenski naziv sloja koji se koristi u izvezenoj DXF datoteci (originalni sloj projekta ostaje nepromenjen) **Dozvoli blokove simbola definisane podacima** [logička vrednost] (`buildDataDefinedBlocks`) **Najveći broj blokova simbola [numerički: ceo broj] (`dataDefinedBlocksMaximumNumberOfClasses`)** `-1` znači bez ograničenja.
Režim simbologije	`SYMBOLOGY_MODE`	[nabrajanje] Podrazumevano: 0	Tip simbologije koji se primenjuje na izlazne slojeve. Možete birati između: • 0 — Bez simbologije • 1 — Simbologija objekata • 2 — Simbologija slojeva simbola
Razmera simbologije	`SYMBOLOGY_SCALE`	[razmera] Podrazumevano: 1:1 000 000	Podrazumevana razmera izvoza podataka.
Tema karte Opciono	`MAP_THEME`	[tema karte]	Uskladi stil sloja sa zadatom temom karte.
Kodiranje	`ENCODING`	[nabrajanje]	Kodiranje koje se primenjuje na slojeve.
CRS	`CRS`	[crs]	Izaberite CRS za izlazni sloj.
Obuhvat Opciono	`EXTENT`	[obuhvat]	Ograniči izvezene objekte na one čije geometrije presecaju zadati obuhvat.
Koristi naslov sloja kao naziv	`USE_LAYER_TITLE`	[logička vrednost] Podrazumevano: False	Imenuj izlazni sloj naslovom sloja (kako je postavljen u metapodacima sloja ili svojstvima QGIS Servera) umesto nazivom sloja.

наставак на следећој страни

Табела 24.560 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Prinudno 2D	`FORCE_2D`	[logička vrednost] Podrazumevano: False	
Izvezi oznake kao MTEXT elemente	`MTEXT`	[logička vrednost] Podrazumevano: True	Izvozi oznake kao MTEXT ili TEXT elemente
Koristi samo izabrane objekte	`SELECTED_FEATURES_ONI`	[logička vrednost] Podrazumevano: False	Izvozi samo izabrane objekte.
DXF	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne DXF datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
DXF	`OUTPUT`	[datoteka]	`.DXF` datoteka koja sadrži ulazne slojeve

Python kod

ID algoritma: `native:dxfexport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje izabranih objekata

Čuva izabrane objekte kao novi sloj.

наставак на следећој страни

Табела 24.562 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

Белешка

Ako u izabranom sloju nema izabranih objekata, novokreirani sloj biće prazan.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj iz koga se čuva izbor
Izabrani objekti	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj za izabrane objekte. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izabrani objekti	`OUTPUT`	[isto kao ulaz]	Vektorski sloj samo sa izabranim objektima, ili bez ijednog objekta ako nijedan nije izabran.

Python kod

ID algoritma: `native:saveselectedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje kodiranja Shapefile-a

Izdvaja podatke o kodiranju atributa ugrađene u Shapefile. Uzimaju se u obzir i kodiranje zadato opcionom datotekom `.cpg` i svi detalji o kodiranju prisutni u LDID zaglavlju datoteke `.dbf`.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	ESRI Shapefile (`.SHP`) sloj iz koga se izdvajaju podaci o kodiranju.

Izlazi

Oznaka	Назив	Tip	Opis
Kodiranje Shapefile-a	`ENCODING`	[niska]	Podaci o kodiranju zadati u ulaznoj datoteci
CPG kodiranje	`CPG_ENCODING`	[niska]	Podaci o kodiranju zadati u opcionoj datoteci `.CPG`
LDID kodiranje	`LDID_ENCODING`	[niska]	Podaci o kodiranju zadati u LDID zaglavlju datoteke `.dbf`

Python kod

ID algoritma: `native:shpencodinginfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pronalaženje projekcije

Pravi uži izbor kandidata za koordinatni referentni sistem, na primer za sloj sa nepoznatom projekcijom.

Područje koje sloj treba da pokriva mora se zadati parametrom ciljnog područja. Koordinatni referentni sistem tog ciljnog područja mora biti poznat QGIS-u.

Algoritam radi tako što testira obuhvat sloja u svakom poznatom referentnom sistemu, a zatim navodi one kod kojih bi granice bile blizu ciljnog područja da je sloj u toj projekciji.

See also

Dodela projekcije, *Definisanje projekcije*, *Reprojektovanje sloja*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj sa nepoznatom projekcijom

наставак на следећој страни

Табела 24.566 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ciljno područje za sloj (xmin, xmax, ymin, ymax)	`TARGET_AREA`	[obuhvat]	Područje koje sloj pokriva. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Kandidati za CRS	`OUTPUT`	[vektor: tabela] Podrazumevano: `[Create temporary layer]`	Zadajte tabelu (sloj bez geometrije) za predloge CRS-a (EPSG kodovi). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Kandidati za CRS	`OUTPUT`	[vektor: tabela]	Tabela sa svim CRS-ovima (EPSG kodovi) koji odgovaraju kriterijumima.

Python kod

ID algoritma: `native:findprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

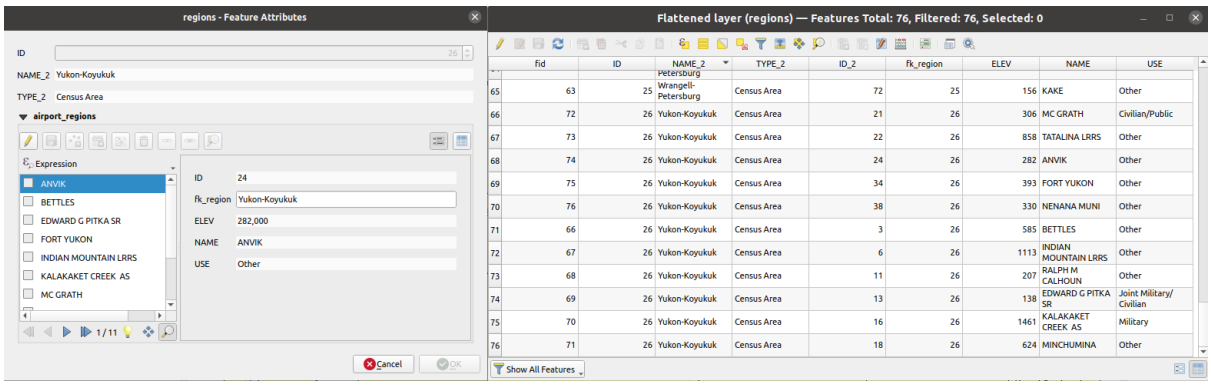
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spljoštavanje relacije

Spljoštava *relaciju* vektorskog sloja, izvozeći jedan sloj koji sadrži po jedan roditeljski objekat za svaki povezani podređeni objekat. Taj glavni objekat sadrži sve atribute povezanih objekata. Time se relacija dobija kao obična tabela koja se može npr. izvesti u CSV.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.



Слика24.96: Formular regiona sa povezanim podređenim zapisima (levo) - Duplirani objekat regiona za svaki povezani podređeni zapis, sa spojenim atributima (desno)

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj sa relacijom koja treba da se denormalizuje
Spljošteni sloj Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (spljošteni) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Spljošteni sloj	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži glavne objekte sa svim atributima povezanih objekata

Python kod

ID algoritma: `native:flattenrelationships`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje atributa po vrednosti polja

Uzima ulazni vektorski sloj i pravi novi vektorski sloj koji je proširena verzija ulaznog, sa dodatnim atributima u atributskoj tabeli.

Dodatni atributi i njihove vrednosti preuzimaju se iz drugog vektorskog sloja. U svakom od njih bira se atribut koji definiše kriterijum spajanja.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Spajanje atributa po najbližem, *Spajanje atributa po lokaciji*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj. Izlazni sloj sastojaće se od objekata ovog sloja sa atributima objekata koji im odgovaraju u drugom sloju.
Polje tabele	`FIELD`	[polje tabele: bilo koji]	Polje izvornog sloja koje se koristi za spajanje
Ulazni sloj 2	`INPUT_2`	[vektor: bilo koji]	Sloj sa atributskom tabelom koja se spaja
Polje tabele 2	`FIELD_2`	[polje tabele: bilo koji]	Polje drugog sloja (za spajanje) koje se koristi za spajanje. Tip polja mora biti jednak tipu polja ulazne tabele (ili sa njim kompatibilan).
Polja sloja 2 koja se kopiraju Opciono	`FIELDS_TO_COPY`	[polje tabele: bilo koji] [lista]	Izaberite konkretna polja koja želite da dodate. Podrazumevano se dodaju sva polja.
Tip spajanja	`METHOD`	[nabrajanje] Podrazumevano: 1	Tip konačnog spojenog sloja. Jedno od: • 0 — Napravi zaseban objekat za svaki odgovarajući objekat (jedan prema više) • 1 — Uzmi atribute samo prvog odgovarajućeg objekta (jedan prema jedan)
Odbaci slogove koji nisu mogli da se spoje	`DISCARD_NONMATCHIN`	[logička vrednost] Podrazumevano: True	Označite ako ne želite da zadržite objekte koji nisu mogli da se spoje
Prefiks spojenih polja Opciono	`PREFIX`	[niska]	Dodajte prefiks spojenim poljima kako biste ih lako prepoznali i izbegli sudar naziva polja

наставак на следећој страни

Табела 24.570 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Spojeni sloj Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za spajanje. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Objekti iz prvog sloja koji se ne mogu spojiti Opciono	`NON_MATCHING`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte izlazni vektorski sloj za objekte iz prvog sloja koji se ne mogu spojiti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj spojenih objekata iz ulazne tabele	`JOINED_COUNT`	[numerički: ceo broj]	
Objekti iz prvog sloja koji se ne mogu spojiti Opciono	`NON_MATCHING`	[isto kao ulaz]	Vektorski sloj sa objektima bez poklapanja
Spojeni sloj Opciono	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa atributima dodatim spajanjem
Broj objekata iz ulazne tabele koji se ne mogu spojiti Opciono	`UNJOINABLE_COUNT`	[numerički: ceo broj]	

Python kod

ID algoritma: `native:joinattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje atributa po lokaciji

Uzima ulazni vektorski sloj i pravi novi vektorski sloj koji je proširena verzija ulaznog, sa dodatnim atributima u atributskoj tabeli.

Dodatni atributi i njihove vrednosti preuzimaju se iz drugog vektorskog sloja. Prostorni kriterijum se primenjuje da bi se izabrale vrednosti iz drugog sloja koje se dodaju svakom objektu iz prvog sloja.

Podrazumevani meni: *Vektor ► Alatke za upravljanje podacima*

Упозорење

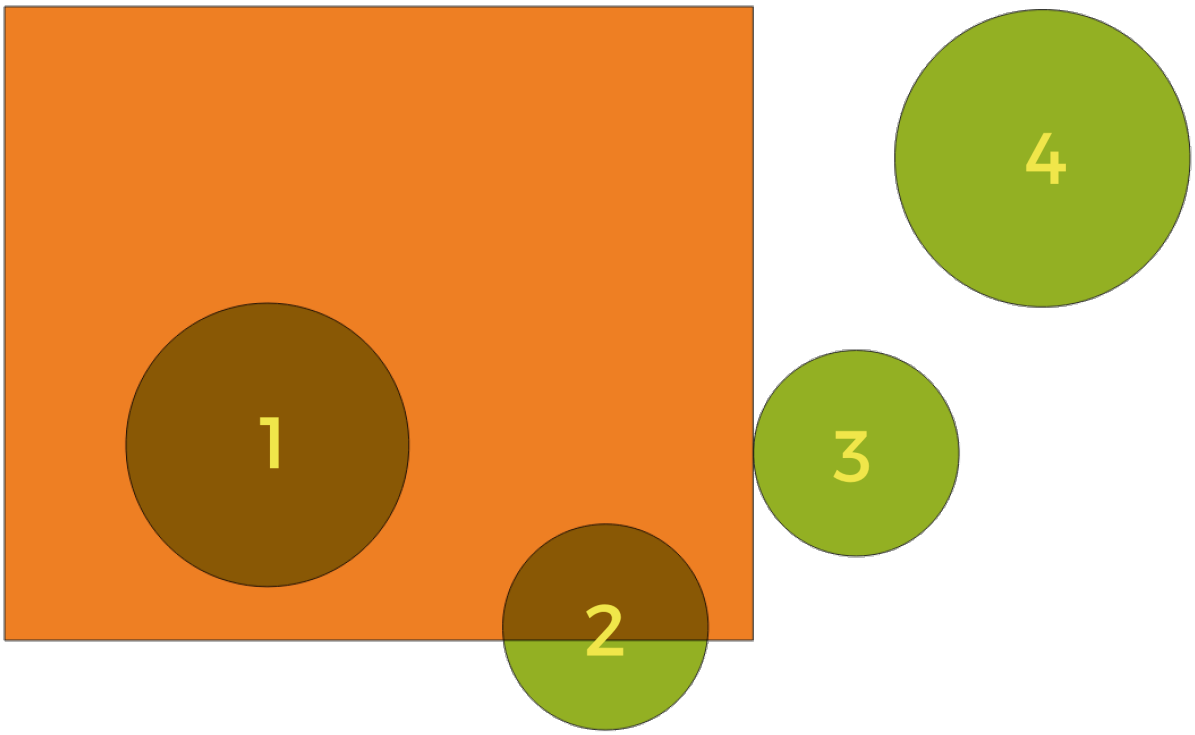
Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Spajanje atributa po najbližem, *Spajanje atributa po vrednosti polja*, *Spajanje atributa po lokaciji (rezime)*

Istraživanje prostornih odnosa

Geometrijski predikati su logičke funkcije koje se koriste za utvrđivanje prostornog odnosa jednog objekta prema drugom, poređenjem da li i kako njihove geometrije dele deo prostora.



Слика24.97: Traženje prostornih odnosa između slojeva

Na osnovu gornje slike, tražimo zelene krugove prostornim poređenjem sa narandžastim pravougaonim objektom. Dostupni geometrijski predikati su:

Preseca
: Proverava da li jedna geometrija preseca drugu. Vraća 1 (tačno) ako se geometrije prostorno presecaju (dele bilo koji deo prostora - preklapaju se ili dodiruju), a 0 ako se ne presecaju. Na gornjoj slici, ovo vraća krugove 1, 2 i 3.

Sadrži
Vraća 1 (tačno) ako i samo ako nijedna tačka geometrije b ne leži izvan geometrije a i bar jedna tačka unutrašnjosti geometrije b leži u unutrašnjosti geometrije a. Na slici se ne vraća nijedan krug, ali bi se vratio pravougaonik da tražite obrnuto, jer on u potpunosti sadrži krug 1. Ovo je suprotno od *su unutar*.

Razdvojeno
Vraća 1 (tačno) ako geometrije ne dele nijedan deo prostora (bez preklapanja, bez dodirivanja). Vraća se samo krug 4.

Jednako
Vraća 1 (tačno) ako i samo ako su geometrije potpuno iste. Neće se vratiti nijedan krug.

Dodiruje
Proverava da li jedna geometrija dodiruje drugu. Vraća 1 (tačno) ako geometrije imaju bar jednu zajedničku tačku, ali im se unutrašnjosti ne presecaju. Vraća se samo krug 3.

Preklapa
Proverava da li se jedna geometrija preklapa sa drugom. Vraća 1 (tačno) ako geometrije dele prostor, iste su dimenzije, ali nisu potpuno sadržane jedna u drugoj. Vraća se samo krug 2.

Su unutar
Proverava da li je jedna geometrija unutar druge. Vraća 1 (tačno) ako je geometrija a potpuno unutar geometrije b. Vraća se samo krug 1.

Ukršta
Vraća 1 (tačno) ako zadate geometrije imaju neke, ali ne sve, zajedničke unutrašnje tačke i ako je samo ukrštanje niže dimenzije od najviše zadate geometrije. Na primer, linija koja ukršta poligon ukršta ga kao linija (tačno). Dve linije koje se ukrštaju ukrštaju se kao tačka (tačno). Dva poligona ukrštaju se kao poligon (netačno). Na slici se neće vratiti nijedan krug.

Parametri

Oznaka	Назив	Tip	Opis
Spoji sa objektima u	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj. Izlazni sloj sastojaće se od objekata ovog sloja sa atributima objekata koji im odgovaraju u drugom sloju.
Gde objekti	`PREDICATE`	[nabrajanje] [lista] Podrazumevano: [0]	Tip prostornog odnosa koji izvorni objekat treba da ima sa ciljnim objektom da bi mogli da se spoje. Jedno ili više od: • 0 — presecaju • 1 — sadrže • 2 — jednake • 3 — dodiruju • 4 — preklapaju • 5 — su unutar • 6 — ukrštaju Ako je izabrano više od jednog uslova, bar jedan od njih (operacija OR) mora biti ispunjen da bi objekat bio izdvojen.
Poređenjem sa	`JOIN`	[vektor: geometrija]	Sloj za spajanje. Objekti ovog vektorskog sloja **dodaće** svoje atribute atributskoj tabeli izvornog sloja ako zadovolje prostorni odnos.
Polja koja se dodaju (ostavite prazno da biste koristili sva polja) Opciono	`JOIN_FIELDS`	[polje tabele: bilo koji] [lista]	Izaberite konkretna polja koja želite da dodate iz sloja za spajanje. Podrazumevano se dodaju sva polja.

наставак на следећој страни

Табела 24.572 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip spajanja	METHOD	[nabrajanje]	Tip konačnog spojenog sloja. Jedno od: • 0 — Napravi zaseban objekat za svaki odgovarajući objekat (jedan prema više) • 1 — Uzmi atribute samo prvog odgovarajućeg objekta (jedan prema jedan) • 2 — Uzmi atribute samo objekta sa najvećim preklapanjem (jedan prema jedan)
Odbaci slogove koji nisu mogli da se spoje	DISCARD_NONMATCHIN	[logička vrednost] Podrazumevano: False	Ukloni iz izlaza objekte ulaznog sloja koji nisu mogli da se spoje
Prefiks spojenih polja Opciono	PREFIX	[niska]	Dodajte prefiks spojenim poljima kako biste ih lako prepoznali i izbegli sudar naziva polja
Spojeni sloj Opciono	OUTPUT	[isto kao ulaz] Podrazumevano: [Create temporary layer]	Zadajte izlazni vektorski sloj za spajanje. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Objekti iz prvog sloja koji se ne mogu spojiti Opciono	NON_MATCHING	[isto kao ulaz] Podrazumevano: [Skip output]	Zadajte izlazni vektorski sloj za objekte iz prvog sloja koji se ne mogu spojiti. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (TEMPORARY_OUTPUT) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj spojenih objekata iz ulazne tabele	JOINED_COUNT	[numerički: ceo broj]	
Objekti iz prvog sloja koji se ne mogu spojiti Opciono	NON_MATCHING	[isto kao ulaz]	Vektorski sloj objekata bez poklapanja
Spojeni sloj	OUTPUT	[isto kao ulaz]	Izlazni vektorski sloj sa atributima dodatim spajanjem

Python kod

ID algoritma: `native:joinattributesbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje atributa po lokaciji (rezime)

Uzima ulazni vektorski sloj i pravi novi vektorski sloj koji je proširena verzija ulaznog, sa dodatnim atributima u atributskoj tabeli.

Dodatni atributi i njihove vrednosti preuzimaju se iz drugog vektorskog sloja. Prostorni kriterijum se primenjuje da bi se izabrale vrednosti iz drugog sloja koje se dodaju svakom objektu iz prvog sloja.

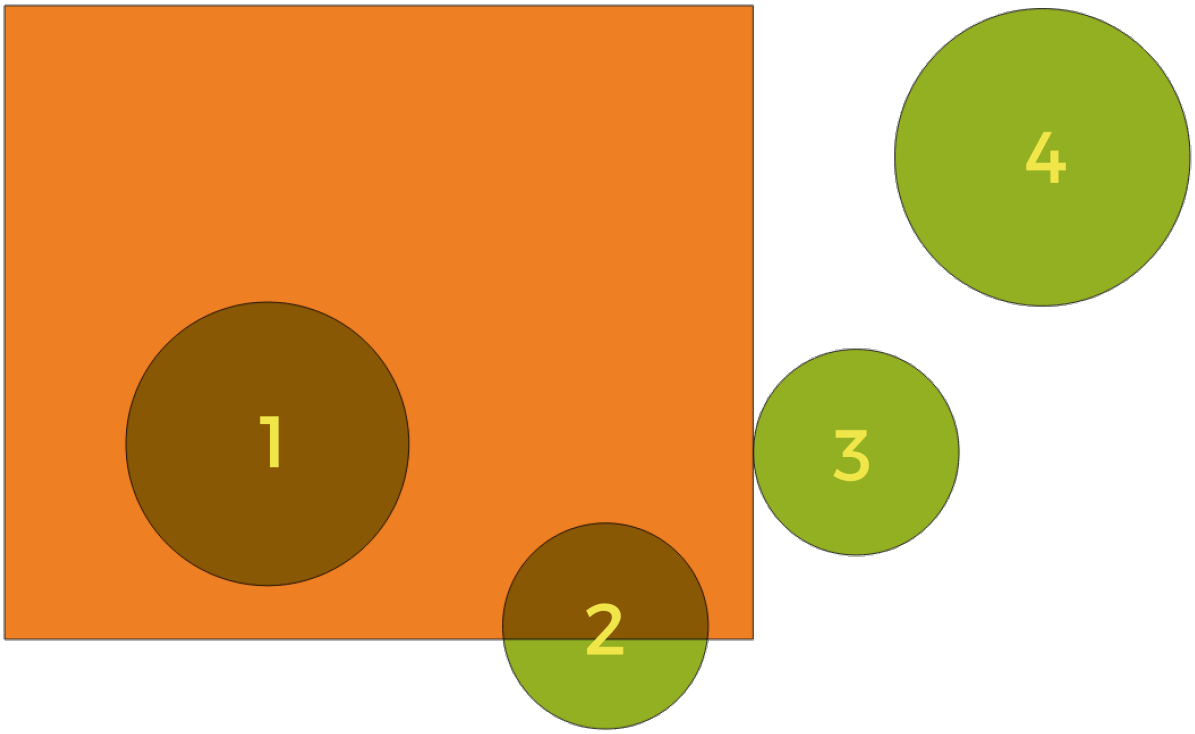
Algoritam izračunava statistički rezime vrednosti objekata koji se poklapaju u drugom sloju (npr. maksimalna vrednost, srednja vrednost itd.).

See also

Spajanje atributa po lokaciji

Istraživanje prostornih odnosa

Geometrijski predikati su logičke funkcije koje se koriste za utvrđivanje prostornog odnosa jednog objekta prema drugom, poređenjem da li i kako njihove geometrije dele deo prostora.



Слика24.98: Traženje prostornih odnosa između slojeva

Na osnovu gornje slike, tražimo zelene krugove prostornim poređenjem sa narandžastim pravougaonim objektom. Dostupni geometrijski predikati su:

Preseca
Proverava da li jedna geometrija preseca drugu. Vraća 1 (tačno) ako se geometrije prostorno presecaju (dele bilo koji deo prostora - preklapaju se ili dodiruju), a 0 ako se ne presecaju. Na gornjoj slici, ovo vraća krugove 1, 2 i 3.

Sadrži
Vraća 1 (tačno) ako i samo ako nijedna tačka geometrije b ne leži izvan geometrije a i bar jedna tačka unutrašnjosti geometrije b leži u unutrašnjosti geometrije a. Na slici se ne vraća nijedan krug, ali bi se vratio pravougaonik da tražite obrnuto, jer on u potpunosti sadrži krug 1. Ovo je suprotno od *su unutar*.

Razdvojeno
Vraća 1 (tačno) ako geometrije ne dele nijedan deo prostora (bez preklapanja, bez dodirivanja). Vraća se samo krug 4.

Jednako
Vraća 1 (tačno) ako i samo ako su geometrije potpuno iste. Neće se vratiti nijedan krug.

Dodiruje
Proverava da li jedna geometrija dodiruje drugu. Vraća 1 (tačno) ako geometrije imaju bar jednu zajedničku tačku, ali im se unutrašnjosti ne presecaju. Vraća se samo krug 3.

Preklapa
Proverava da li se jedna geometrija preklapa sa drugom. Vraća 1 (tačno) ako geometrije dele prostor, iste su dimenzije, ali nisu potpuno sadržane jedna u drugoj. Vraća se samo krug 2.

Su unutar
Proverava da li je jedna geometrija unutar druge. Vraća 1 (tačno) ako je geometrija a potpuno unutar geometrije b. Vraća se samo krug 1.

Ukršta
Vraća 1 (tačno) ako zadate geometrije imaju neke, ali ne sve, zajedničke unutrašnje tačke i ako je samo ukrštanje niže dimenzije od najviše zadate geometrije. Na primer, linija koja ukršta poligon ukršta ga kao linija (tačno). Dve linije koje se ukrštaju ukrštaju se kao tačka (tačno). Dva poligona ukrštaju se kao poligon (netačno). Na slici se neće vratiti nijedan krug.

Parametri

Oznaka	Назив	Tip	Opis
Spoji sa objektima u	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj. Izlazni sloj sastojaće se od objekata ovog sloja sa atributima objekata koji im odgovaraju u drugom sloju.
Gde objekti	`PREDICATE`	[nabrajanje] [lista] Podrazumevano: [0]	Tip prostornog odnosa koji izvorni objekat treba da ima sa ciljnim objektom da bi mogli da se spoje. Jedno ili više od: • 0 — presecaju • 1 — sadrže • 2 — jednake • 3 — dodiruju • 4 — preklapaju • 5 — su unutar • 6 — ukrštaju Ako je izabrano više od jednog uslova, bar jedan od njih (operacija OR) mora biti ispunjen da bi objekat bio izdvojen.

наставак на следећој страни

Табела 24.574 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Poređenjem sa	`JOIN`	[vektor: geometrija]	Sloj za spajanje. Objekti ovog vektorskog sloja **dodaće** rezimee svojih atributa atributskoj tabeli izvornog sloja ako zadovolje prostorni odnos.
Polja za rezimiranje (ostavite prazno da biste koristili sva polja) Opciono	`JOIN_FIELDS`	[polje tabele: bilo koji] [lista]	Izaberite konkretna polja koja želite da dodate iz sloja za spajanje. Podrazumevano se dodaju sva polja.
Rezimei koji se izračunavaju (ostavite prazno da biste koristili sva polja) Opciono	`SUMMARIES`	[nabrajanje] [lista] Podrazumevano: []	Za svaki ulazni objekat, statistike se izračunavaju nad spojenim poljima objekata koji im odgovaraju. Jedno ili više od: • 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
Odbaci slogove koji nisu mogli da se spoje	`DISCARD_NONMATCHIN`	[logička vrednost] Podrazumevano: False	Ukloni iz izlaza objekte ulaznog sloja koji nisu mogli da se spoje
Spojeni sloj	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za spajanje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Spojeni sloj	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa rezimiranim atributima iz spajanja

Python kod

ID algoritma: `native:joinbylocationsummary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje atributa po najbližem

Uzima ulazni vektorski sloj i pravi novi vektorski sloj sa dodatnim poljima u atributskoj tabeli. Dodatni atributi i njihove vrednosti preuzimaju se iz drugog vektorskog sloja. Objekti se spajaju pronalaženjem najbližih objekata iz svakog sloja.

Podrazumevano se spaja samo najbliži objekat, ali spajanje može obuhvatiti i k najbližih susednih objekata.

Ako se zada najveće rastojanje, poklopiće se samo objekti koji su bliži od tog rastojanja.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Analiza najbližeg suseda, *Spajanje atributa po vrednosti polja*, *Spajanje atributa po lokaciji*, *Matrica rastojanja*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni sloj.
Ulazni sloj 2	`INPUT_2`	[vektor: geometrija]	Sloj za spajanje.
Polja sloja 2 koja se kopiraju (ostavite prazno da biste kopirali sva polja)	`FIELDS_TO_COPY`	[polje tabele: bilo koji] [lista]	Polja sloja za spajanje koja se kopiraju (ako je prazno, kopiraju se sva polja).
Odbaci slogove koji nisu mogli da se spoje	`DISCARD_NONMATCHIN`	[logička vrednost] Podrazumevano: False	Ukloni iz izlaza slogove ulaznog sloja koji nisu mogli da se spoje
Prefiks spojenih polja	`PREFIX`	[niska]	Prefiks spojenih polja
Najveći broj najbližih suseda	`NEIGHBORS`	[numerički: ceo broj] Podrazumevano: 1	Najveći broj najbližih suseda
Najveće rastojanje	`MAX_DISTANCE`	[numerički: decimalni]	Najveće rastojanje pretrage

наставак на следећој страни

Табела 24.576 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Spojeni sloj Opciono	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj koji sadrži spojene objekte. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Objekti iz prvog sloja koji se ne mogu spojiti	`NON_MATCHING`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte vektorski sloj koji sadrži objekte koji nisu mogli da se spoje. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Spojeni sloj	`OUTPUT`	[isto kao ulaz]	Izlazni spojeni sloj.
Objekti iz prvog sloja koji se ne mogu spojiti	`NON_MATCHING`	[isto kao ulaz]	Sloj koji sadrži objekte iz prvog sloja koji nisu mogli da se spoje ni sa jednim objektom u sloju za spajanje.
Broj spojenih objekata iz ulazne tabele	`JOINED_COUNT`	[numerički: ceo broj]	Broj objekata iz ulazne tabele koji su spojeni.
Broj objekata iz ulazne tabele koji se ne mogu spojiti	`UNJOINABLE_COUNT`	[numerički: ceo broj]	Broj objekata iz ulazne tabele koji nisu mogli da se spoje.

Python kod

ID algoritma: `native:joinbynearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

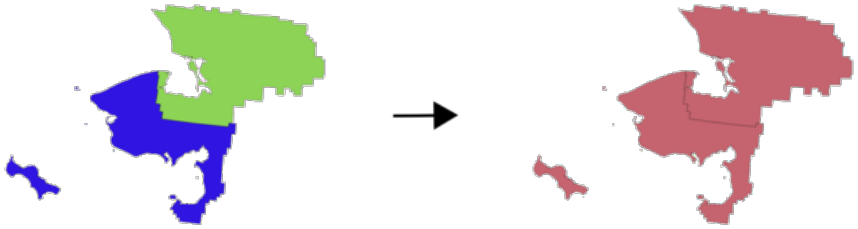
Spajanje vektorskih slojeva

Objedinjuje više vektorskih slojeva **iste geometrije** u jedan.

Atributska tabela rezultujućeg sloja sadržaće polja iz svih ulaznih slojeva. Ako se pronađu polja istog naziva a različitog tipa, izvezeno polje biće automatski pretvoreno u polje tipa niska. Opciono se mogu dodati nova polja koja čuvaju naziv i izvor originalnog sloja.

Ako bilo koji ulazni sloj sadrži Z ili M vrednosti, izlazni sloj takođe će ih sadržati. Slično, ako je bilo koji od ulaznih slojeva višedelni, i izlazni sloj biće višedelni.

Opciono se može postaviti odredišni koordinatni referentni sistem (CRS) spojenog sloja. Ako se ne postavi, CRS se preuzima iz prvog ulaznog sloja. Svi slojevi biće reprojektovani da odgovaraju tom CRS-u.



Podrazumevani meni: *Vektor ► Alatke za upravljanje podacima*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Deljenje vektorskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`LAYERS`	[vektor: bilo koji] [lista]	Slojevi koji treba da se spoje u jedan sloj. Slojevi treba da budu iste geometrije.
Odredišni CRS Opciono	`CRS`	[crs]	Izaberite CRS za izlazni sloj. Ako se ne zada, koristi se CRS prvog ulaznog sloja.
Dodaj podatke o izvornom sloju (naziv i putanja sloja) `Dodato u 3.42`	`ADD_SOURCE_FIELD`	[logička vrednost] Podrazumevano: True	Dodaje polja koja čuvaju naziv i putanju originalnog sloja
Spojeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Spojeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj koji sadrži sve objekte i atribute iz ulaznih slojeva.

Python kod

ID algoritma: `native:mergevectorlayers`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Sortiranje po izrazu

Sortira vektorski sloj prema izrazu: menja indeks objekata prema izrazu.

Budite oprezni, kod nekih provajdera možda neće raditi kako se očekuje i redosled se možda neće uvek zadržati.

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj za sortiranje
Izraz	`EXPRESSION`	[izraz]	Izraz koji se koristi za sortiranje
Sortiraj rastuće	`ASCENDING`	[logička vrednost] Podrazumevano: True	Ako je označeno, vektorski sloj biće sortiran od manjih ka većim vrednostima.
Prvo sortiraj Null vrednosti	`NULLS_FIRST`	[logička vrednost] Podrazumevano: False	Ako je označeno, Null vrednosti se postavljaju na početak
Sortirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Sortirano	`OUTPUT`	[isto kao ulaz]	Izlazni (sortirani) vektorski sloj

Python kod

ID algoritma: `native:orderbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Popravka Shapefile-a

Popravlja oštećen ESRI Shapefile skup podataka tako što (ponovo) kreira SHX datoteku.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni Shapefile	`INPUT`	[datoteka]	Puna putanja do ESRI Shapefile skupa podataka sa nedostajućom ili oštećenom SHX datotekom

Izlazi

Oznaka	Назив	Tip	Opis
Popravljeni sloj	`OUTPUT`	[vektor: geometrija]	Ulazni vektorski sloj sa popravljenom SHX datotekom

Python kod

ID algoritma: `native:repairshapefile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Reprojektovanje sloja

Reprojektuje vektorski sloj u drugi CRS. Reprojektovani sloj imaće iste objekte i atribute kao ulazni sloj.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Dodela projekcije, *Definisanje projekcije*, *Pronalaženje projekcije*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj za reprojektovanje

наставак на следећој страни

Табела 24.584 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ciljni CRS	`TARGET_CRS`	[crs] Podrazumevano: `EPSG:4326 – WGS 84`	Odredišni koordinatni referentni sistem
Pretvori krive geometrije u prave segmente `Dodato u 3.32`	`CONVERT_CURVED_GEOM`	[logička vrednost] Podrazumevano: False	Ako je označeno, krive geometrije biće u toku obrade pretvorene u prave segmente, čime se izbegavaju mogući problemi sa izobličenjem.
Reprojektovano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Koordinatna operacija Opciono	`OPERATION`	[niska]	Konkretna operacija koja se koristi za određeni zadatak reprojektovanja, umesto da se uvek nameću podešavanja transformacije tekućeg projekta. Korisno kada se reprojektuje određeni sloj i potrebna je kontrola nad tačnim tokom transformacije. Zahteva proj verziju >= 6. Više o tome pročitajte na *Datum Transformations*.

Izlazi

Oznaka	Назив	Tip	Opis
Reprojektovano	`OUTPUT`	[isto kao ulaz]	Izlazni (reprojektovani) vektorski sloj

Python kod

ID algoritma: `native:reprojectlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Čuvanje vektorskih objekata u datoteku

Čuva vektorske objekte u zadati skup podataka u datoteci.

Za formate skupova podataka koji podržavaju slojeve, opcioni parametar naziva sloja može se koristiti za zadavanje proizvoljnog teksta. Mogu se zadati i opcione GDAL opcije skupa podataka i sloja. Za više informacija o tome pročitajte GDAL dokumentaciju o formatu na mreži.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Vektorski objekti	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj.
Sačuvani objekti	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Save to temporary file]`	Zadajte datoteku u koju se čuvaju objekti. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Naziv sloja Opciono	`LAYER_NAME`	[niska]	Naziv koji se koristi za izlazni sloj
GDAL opcije skupa podataka Opciono	`DATASOURCE_OPTIONS`	[niska]	GDAL opcije kreiranja skupa podataka izlaznog formata. Pojedinačne opcije razdvojite tačkom i zarezom.
GDAL opcije sloja Opciono	`LAYER_OPTIONS`	[niska]	GDAL opcije kreiranja sloja izlaznog formata. Pojedinačne opcije razdvojite tačkom i zarezom.
Radnja nad već postojećom datotekom	`ACTION_ON_EXISTING_F`	[nabrajanje] Podrazumevano: 0	Kako postupati sa postojećim objektima. Dozvoljeni metodi su: • 0 — Napravi ili prepiši datoteku • 1 — Napravi ili prepiši sloj • 2 — Dodaj objekte u postojeći sloj, ali ne pravi nova polja • 3 — Dodaj objekte u postojeći sloj i po potrebi napravi nova polja

Izlazi

Oznaka	Назив	Tip	Opis
Sačuvani objekti	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa sačuvanim objektima.
Naziv i putanja datoteke	`FILE_PATH`	[niska]	Naziv i putanja izlazne datoteke.
Naziv sloja	`LAYER_NAME`	[niska]	Naziv sloja, ako postoji.

Python kod

ID algoritma: `native:savefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavljanje kodiranja sloja

Postavlja kodiranje koje se koristi za čitanje atributa sloja. Na sloju se ne prave trajne izmene, već ovo utiče samo na to kako se sloj čita u tekućoj sesiji.

Белешка

Promena kodiranja je podržana samo za neke izvore podataka vektorskih slojeva.

Parametri

Oznaka	Назив	Tip	Opis
Sačuvani objekti	`INPUT`	[vektor: geometrija]	Vektorski sloj kome se postavlja kodiranje.
Kodiranje	`ENCODING`	[niska]	Kodiranje teksta koje se dodeljuje sloju u tekućoj QGIS sesiji.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni sloj	`OUTPUT`	[isto kao ulaz]	Ulazni vektorski sloj sa postavljenim kodiranjem.

Python kod

ID algoritma: `native:setlayerencoding`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Deljenje objekata po znaku

Objekti se dele na više izlaznih objekata deljenjem vrednosti polja na zadatom znaku. Na primer, ako sloj sadrži objekte sa više vrednosti razdvojenih zarezom u jednom polju, ovaj algoritam se može koristiti da se te vrednosti razdvoje na više izlaznih objekata. Geometrije i ostali atributi ostaju nepromenjeni u izlazu. Opciono, tekst razdvajača može biti regularni izraz radi veće fleksibilnosti.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Podeli koristeći vrednosti u polju	`FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za deljenje
Podeli vrednost pomoću znaka	`CHAR`	[niska]	Znak koji se koristi za deljenje
Koristi regularni izraz kao razdvajač	`REGEX`	[logička vrednost] Podrazumevano: False	
Podeljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `Create temporary layer`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podeljeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj.

Python kod

ID algoritma: `native:splitfeaturesbycharacter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Deljenje vektorskog sloja

Pravi skup vektorskih slojeva u izlaznom folderu na osnovu ulaznog sloja i atributa. Izlazni folder sadržaće onoliko slojeva koliko ima jedinstvenih vrednosti u željenom polju.

Broj generisanih datoteka jednak je broju različitih vrednosti pronađenih za zadati atribut.

To je suprotna operacija od *spajanja*.

Podrazumevani meni: *Vektor ► Alatke za upravljanje podacima*

See also

Spajanje vektorskih slojeva

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Polje sa jedinstvenim ID-om	`FIELD`	[polje tabele: bilo koji]	Polje koje se koristi za deljenje
Izlazni direktorijum	`OUTPUT`	[folder] Podrazumevano: `[Save to temporary folder]`	Zadajte direktorijum za izlazne slojeve. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Napredni parametri

Oznaka	Назив	Tip	Opis
Tip izlazne datoteke Opciono	`FILE_TYPE`	[nabrajanje] Podrazumevano: `gpkg` u prozoru dijaloga	Izaberite ekstenziju izlaznih datoteka. Ako nije zadata ili nije ispravna, format izlaznih datoteka biće onaj postavljen u podešavanju Obrade „Podrazumevana ekstenzija izlaznog vektorskog sloja“.

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT`	[folder]	Direktorijum za izlazne slojeve
Izlazni slojevi	`OUTPUT_LAYERS`	[isto kao ulaz] [lista]	Izlazni vektorski slojevi nastali deljenjem.

Python kod

ID algoritma: `native:splitvectorlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pražnjenje tabele

Prazni sloj brisanjem svih objekata iz njega.

Упозорење

Ovaj algoritam menja sloj na licu mesta i obrisani objekti se ne mogu vratiti!

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj

Izlazi

Oznaka	Назив	Tip	Opis
Ispražnjeni sloj	`OUTPUT`	[folder]	Ulazni sloj, svi objekti obrisani

Python kod

ID algoritma: `native:truncatetable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.26 Vektorska geometrija

Dodaj atribute geometrije

Računa geometrijska svojstva objekata u vektorskom sloju i uključuje ih u izlazni sloj.

Generiše novi vektorski sloj sa istim sadržajem kao ulazni, ali sa dodatnim atributima koji sadrže geometrijska merenja zasnovana na izabranom CRS-u.

Atributi dodati u tabelu zavise od tipa geometrije i dimenzije ulaznog sloja:

- za **tačkaste** slojeve: X (`xcoord`), Y (`ycoord`), Z (`zcoord`) koordinate i/ili M vrednost (`mvalue`)
- za **linijske** slojeve: `length` i, za LineString i CompoundCurve tipove geometrije, `sinuosity` objekta i pravo rastojanje (`straightdis`)
- za **poligonske** slojeve: `perimeter` i `area`

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Računaj pomoću	`CALC_METHOD`	[nabrajanje] Podrazumevano: 0	Parametri izračunavanja koji se koriste za geometrijska svojstva. Jedno od: • 0 — CRS sloja • 1 — CRS projekta • 2 — Elipsoidno
Dodate info o geometriji	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (kopija ulaza sa geometrijom) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Dodate info o geometriji	`OUTPUT`	[isto kao ulaz]	Kopija ulaznog vektorskog sloja uz dodatak polja geometrije

Python kod

ID algoritma: `native:exportaddgeometrycolumns`

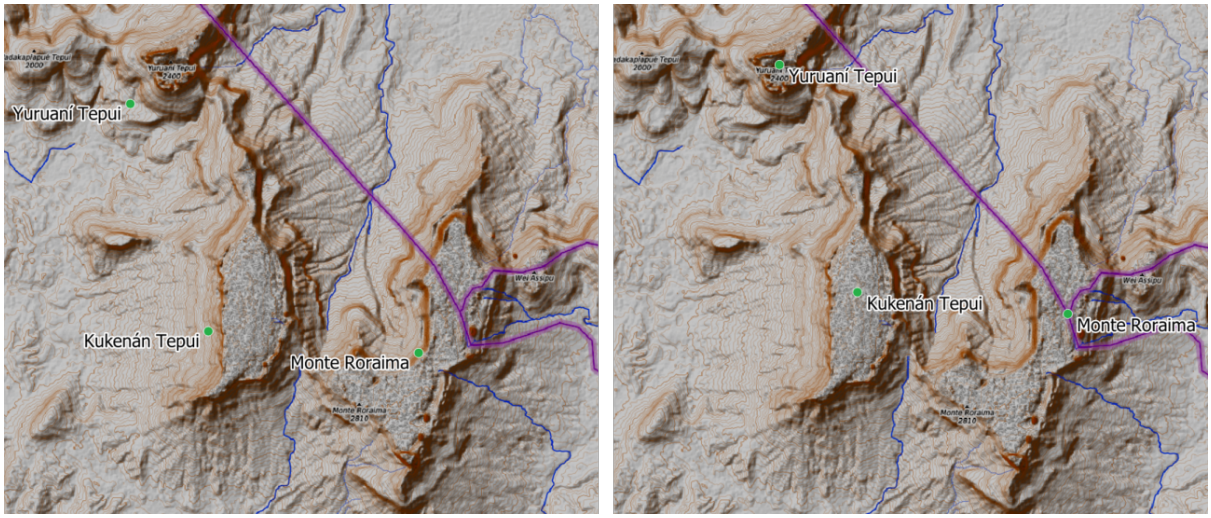
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Afina transformacija

Primenjuje afinu transformaciju na geometrije sloja. Afine transformacije mogu uključivati translaciju, skaliranje i rotaciju. Operacije se izvršavaju sledećim redosledom: skaliranje, rotacija i translacija.

Z i M vrednosti (ako su prisutne) mogu se translirati i skalirati.



Слика24.99: Vektorski tačkasti sloj (zelene tačke) pre (levo) i posle (desno) afine transformacije (translacije).

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Transliraj

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Translacija (x--osa)	`DELTA_X`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na X osi.
Translacija (y--osa)	`DELTA_Y`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na Y osi.
Translacija (z--osa)	`DELTA_Z`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na Z osi.
Translacija (m--vrednosti)	`DELTA_M`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na m vrednosti.
Faktor skaliranja (x-osa)	`SCALE_X`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja (proširenje ili sabijanje) koja se primenjuje na X osi.

наставак на следећој страни

Табела 24.601 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Faktor skaliranja (y-osa)	`SCALE_Y`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja (proširenje ili sabijanje) koja se primenjuje na Y osi.
Faktor skaliranja (z-osa)	`SCALE_Z`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja (proširenje ili sabijanje) koja se primenjuje na Z osi.
Faktor skaliranja (m-vrednosti)	`SCALE_M`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja (proširenje ili sabijanje) koja se primenjuje na m vrednosti.
Rotacija oko z-ose (stepeni suprotno od kazaljke na satu)	`ROTATION_Z`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije u stepenima.
Transformisano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Transformisano	`OUTPUT`	[isto kao ulaz]	Izlazni (transformisani) vektorski sloj.

Python kod

ID algoritma: `native:affinetransform`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Agregiraj

Uzima vektorski ili tabelarni sloj i pravi novi sloj agregiranjem objekata na osnovu `group by` izraza.

Objekti za koje `group by` izraz vraća istu vrednost grupišu se zajedno.

Moguće je grupisati sve izvorne objekte zajedno pomoću konstantne vrednosti u `group by` parametru, primer: NULL.

Takođe je moguće grupisati objekte po više polja pomoću funkcije Array, primer: Array(„Field1”, „Field2”).

Geometrije (ako su prisutne) kombinuju se u jednu višedelnu geometriju za svaku grupu. Izlazni atributi se računaju u zavisnosti od svake zadate definicije agregacije.

Ovaj algoritam omogućava korišćenje podrazumevanih *agregatnih funkcija* mehanizma za izraze QGIS-a.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

See also

Prikupi geometrije, *Stapanje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Izraz za grupisanje	`GROUP_BY`	[polje tabele: bilo koji] Podrazumevano: 'NULL'	Izaberite polje za grupisanje. Ako je *NULL*, svi objekti će biti grupisani.

наставак на следећој страни

Табела 24.603 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Agregati	AGGREGATES	[lista]	Lista definicija polja izlaznog sloja. Primer definicije polja: *{'aggregate': 'sum', 'delimiter': ',', 'input': ' $area', 'length': 10, 'name': 'totarea', 'precision': 0, 'type': 6}* Podrazumevano lista sadrži sva polja ulaznog sloja. U grafičkom okruženju možete uređivati ova polja i njihove definicije, a takođe možete: • Kliknite na dugme da biste dodali novo polje. • Kliknite na da biste obrisali izabrano polje. • Koristite i da biste promenili redosled polja. • Kliknite na da biste vratili na podrazumevano (polja ulaznog sloja). Za svako polje iz kojeg biste želeli da preuzmete informacije, potrebno je da definišete sledeće: ***Ulazni izraz* (`input`) [izraz]** Polje ili izraz iz ulaznog sloja. ***Agregatna funkcija* (`aggregate`) [nabrajanje]** Funkcija koja se koristi na ulaznom izrazu za vraćanje agregirane vrednosti. To su uglavnom *Agregatne funkcije*, plus posebne funkcije „first_value“ i „last_value“ za uređene stavke liste. Podrazumevano: *concatenate* (za niskovni tip podataka), *sum* (za numerički tip podataka) ***Razdvajač* (`delimiter`) [niska]** Tekstualna niska za razdvajanje agregiranih vrednosti, na primer u slučaju spajanja. Podrazumevano: , ***Naziv izlaznog polja* (`name`) [niska]** Naziv agregiranog polja u izlaznom sloju. Podrazumevano se zadržava naziv ulaznog polja. ***Tip* (`type`) [nabrajanje]** Tip podataka izlaznog polja. Dostupni tipovi možda nisu kompatibilni sa provajderom izlaznog sloja. Jedno od: **Пажња** Za određene tipove polja, npr. liste, dodatni parametar `sub_type` pomaže u preciziranju konkretnog tipa podataka. Automatski se postavlja u grafičkom okruženju, ali može biti potreban ako algoritam pokrećete u Pythonu ili iz komandne linije.

Табела 24.603 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Učitaj polja iz sloja	Samo grafičko okruženje	[vektor: bilo koji]	Možete učitati polja iz drugog sloja i koristiti ih za agregaciju
Agregirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (agregatni) sloj *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Agregirano	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa višestrukom geometrijom sa agregiranim vrednostima

Python kod

ID algoritma: `native:aggregate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

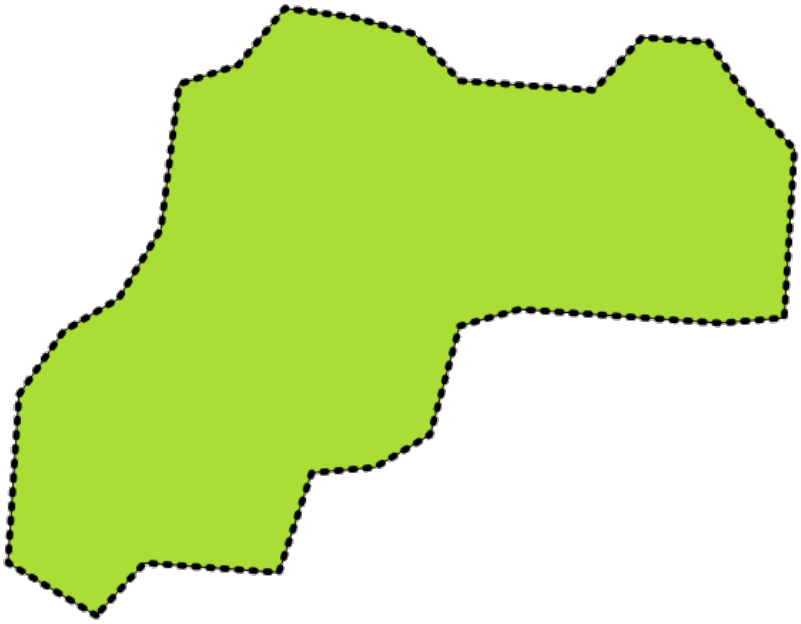
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Granica

Vraća zatvaranje kombinatorne granice ulaznih geometrija (tj. topološku granicu geometrije).

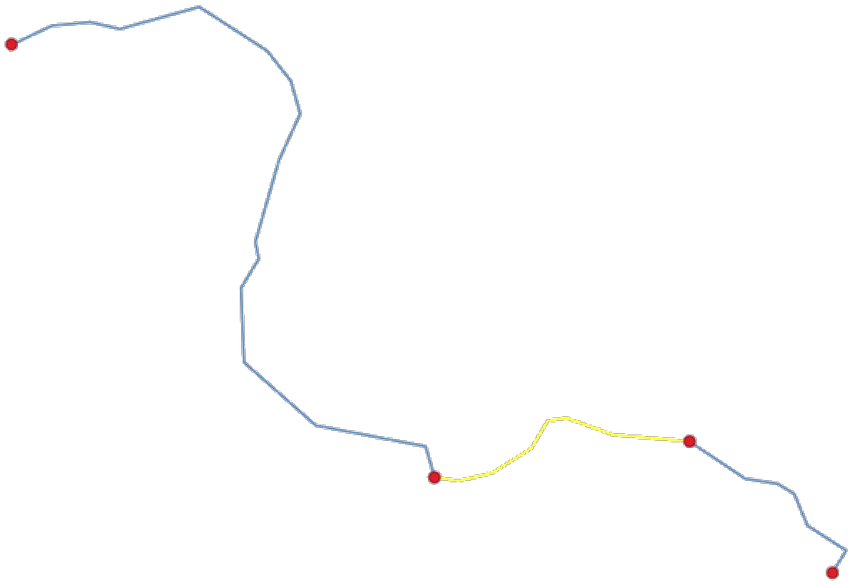
Samo za poligonske i linijske slojeve.

Za **geometrije poligona**, granica se sastoji od svih linija koje čine prstenove poligona.



Слика24.100: Granice (crna isprekidana linija) izvornog poligonskog sloja

Za **geometrije linija**, granice su njihove krajnje tačke.



Слика24.101: Sloj granica (crvene tačke) za linije. Žutom bojom izabrani objekat.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj

наставак на следећој страни

Табела 24.605 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Granica	`OUTPUT`	[vektor: tačka, linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (granični) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Granica	`OUTPUT`	[vektor: tačka, linija]	Granice iz ulaznog sloja (tačka za liniju, i linija za poligon)

Python kod

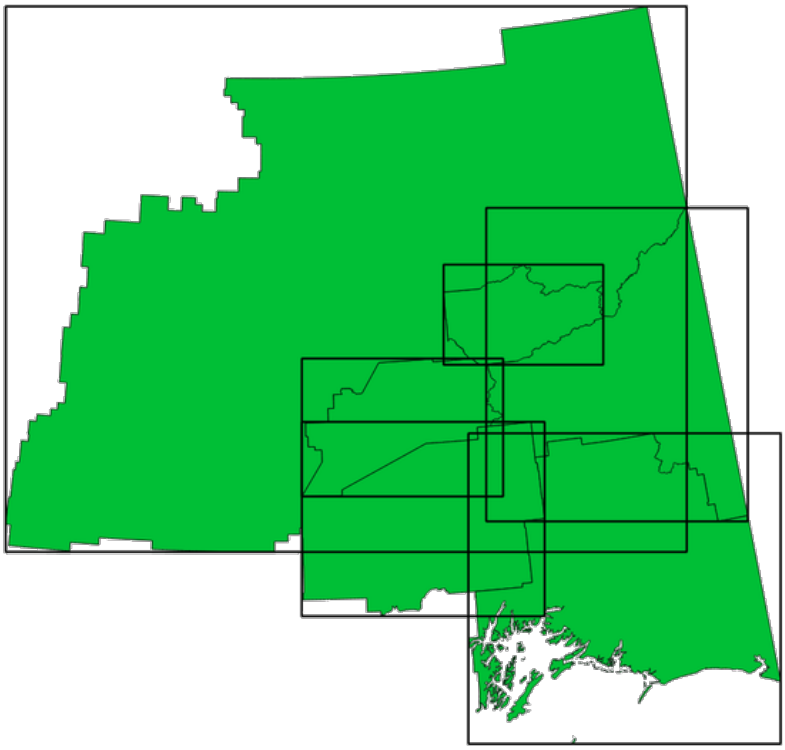
ID algoritma: `native:boundary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Granični pravougaonici

Računa granični pravougaonik (omotnicu) svakog objekta u ulaznom sloju. Podržane su geometrije poligona i linija.



Слика24.102: Crne linije predstavljaju granične pravougaonike svakog poligonskog objekta

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

See also

Minimalna granična geometrija

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Granice	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (granični pravougaonik) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Granice	`OUTPUT`	[vektor: poligon]	Granični pravougaonici ulaznog sloja. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: `width`, `height`, `area` i `perimeter` generisanog poligona.

Python kod

ID algoritma: `native:boundingboxes`

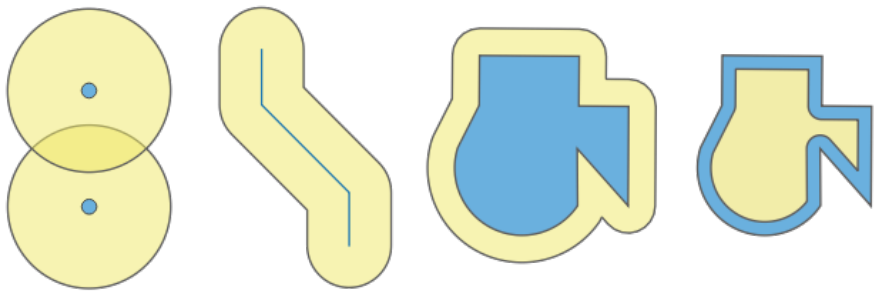
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Bafer

Računa oblast bafera za sve objekte u ulaznom sloju, koristeći fiksno ili podacima definisano rastojanje.

Moguće je koristiti negativno rastojanje za ulazne poligonske slojeve. U tom slučaju bafer će rezultirati manjim poligonom (uvlačenje).



Слика24.103: Bafer (žuto) tačaka, linije, poligona sa pozitivnim baferom i poligona sa negativnim baferom

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za geoobradu*

Упозорење

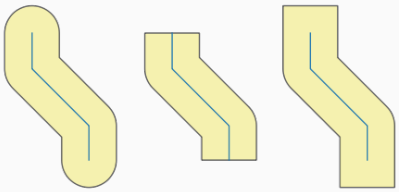
Ovaj algoritam može odbaciti postojeće primarne ključeve ili FID vrednosti i ponovo ih generisati u izlaznim slojevima, u zavisnosti od ulaznih parametara.

See also

Bafer promenljivog rastojanja, *Višeprstenasti bafer (konstantno rastojanje)*, *Bafer promenljive širine (po M vrednosti)*

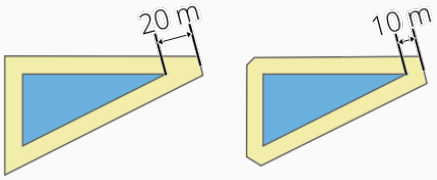
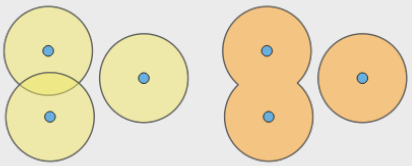
Parametri

Osnovni parametri

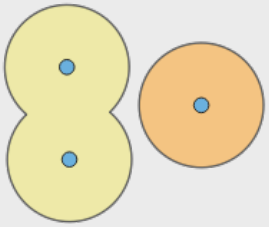
Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rastojanje	`DISTANCE`	[numerički: decimalni] Podrazumevano: 10.0	Rastojanje bafera (od granice svakog objekta). Možete koristiti dugme Zamena definisana podacima sa desne strane da biste izabrali polje iz kojeg će se izračunati poluprečnik. Na ovaj način možete imati različit poluprečnik za svaki objekat.
Segmenti	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 5	Kontroliše broj segmenata linije koji se koriste za aproksimaciju četvrtine kruga pri pravljenju zaobljenih pomeraja.
Stil završetka	`END_CAP_STYLE`	[nabrajanje] Podrazumevano: 0	Kontroliše kako se rukuje završecima linija u baferu. Jedno od: • 0 — Zaobljeno • 1 — Ravno • 2 — Kvadratno  Слика24.104: Zaobljen, ravan i kvadratni stil završetka
Stil spoja	`JOIN_STYLE`	[nabrajanje] Podrazumevano: 0	Zadaje da li se koriste zaobljeni, oštri ili zasečeni spojevi pri pomeranju uglova u liniji. Opcije su: • 0 — Zaobljeno • 1 — Oštro • 2 — Zakošeno  Слика24.105: Zaobljeni, oštri i zakošeni stilovi spoja

наставак на следећој страни

Табела 24.609 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Granica oštrog spoja	`MITER_LIMIT`	[numerički: decimalni] Podrazumevano: 2.0	Postavlja najveće rastojanje od odmaknute geometrije koje se koristi pri kreiranju oštrog spoja, kao faktor rastojanja odmicanja (primenjuje se samo na oštre stilove spoja). Minimum: 1.0  Слика24.106: Bafer od 10 m sa granicom 2 i bafer od 10 m sa granicom 1
Stopi rezultat	`DISSOLVE`	[logička vrednost] Podrazumevano: False	Stopi konačni bafer. Ako je `True` (označeno), preklapajući baferi biće stopljeni (kombinovani) u jedan višedelni objekat.  Слика24.107: Standardno (tri jednodelna objekta - levo), stopljeno (1 višedelni objekat sa 2 dela - desno)
Baferovano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (bafer) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Zadrži razdvojene objekte odvojenima `Dodato u 3.32`	`SEPARATE_DISJOINT`	[logička vrednost] Podrazumevano: False	Ako je `True` (označeno) i stopljeno je označeno, objekti koji se ne preklapaju niti dodiruju biće izvezeni kao odvojeni objekti (umesto delova višedelnog objekta).  Слика24.108: Rezultira sa 2 jednodelna objekta

Izlazi

Oznaka	Назив	Tip	Opis
Baferovano	`OUTPUT`	[vektor: poligon]	Izlazni (bafer) poligonski sloj

Python kod

ID algoritma: `native:buffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

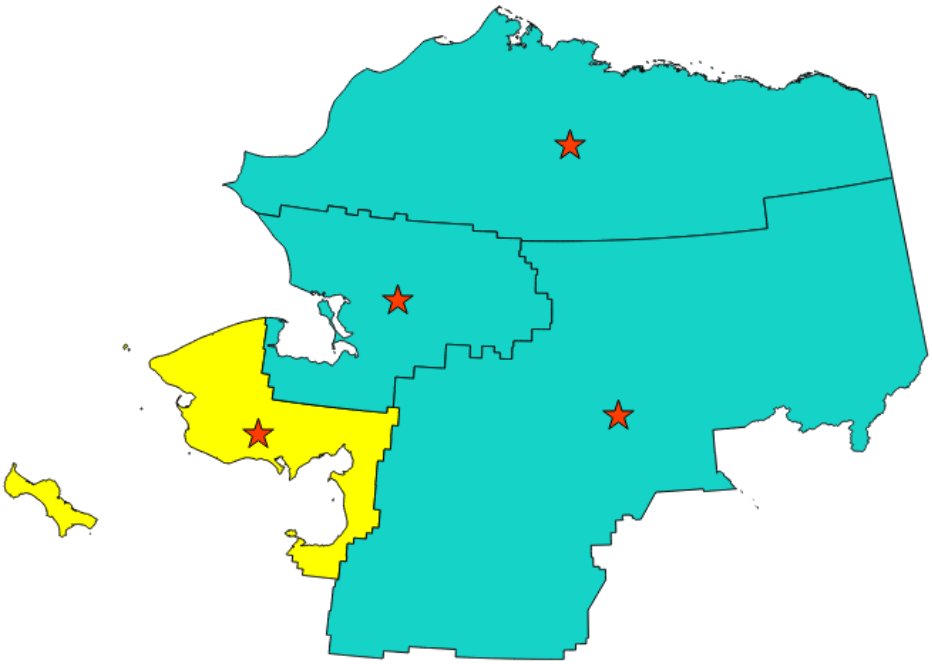
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Centroidi

Pravi novi tačkasti sloj, sa tačkama koje predstavljaju centroide geometrija ulaznog sloja.

Centroid je jedna tačka koja predstavlja baricentar (svih delova) objekta, pa može biti izvan granica objekta. Ali može biti i tačka na svakom delu objekta.

Atributi tačaka u izlaznom sloju isti su kao za originalne objekte.



Слика24.109: Crvene zvezde predstavljaju centroide objekata ulaznog sloja.

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

Упозорење

Ovaj algoritam može odbaciti postojeće primarne ključeve ili FID vrednosti i ponovo ih generisati u izlaznim slojevima, u zavisnosti od ulaznih parametara.

наставак на следећој страни

Табела 24.612 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

See also

Tačka na površini

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Napravi centroid za svaki deo	`ALL_PARTS`	[logička vrednost] Podrazumevano: False	Ako je True (označeno), centroid će biti napravljen za svaki deo geometrije
Centroidi	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (centroid) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Centroidi	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj (centroidi)

Python kod

ID algoritma: `native:centroids`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Proveri validnost

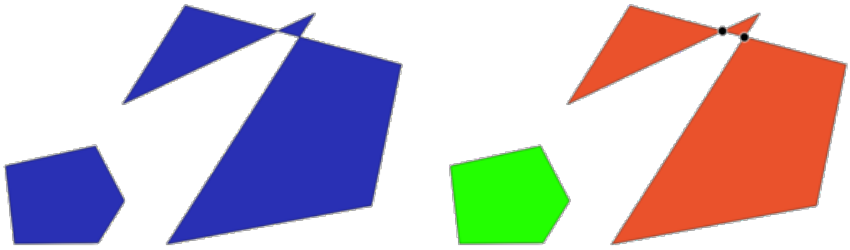
Vrši proveru validnosti na geometrijama vektorskog sloja.

Geometrije se klasifikuju u tri grupe (validne, nevažeće i greška) i za svaku grupu se generiše vektorski sloj sa njenim objektima:

- Sloj **Validni izlaz** sadrži samo validne objekte (bez topoloških grešaka).
- Sloj **Nevažeći izlaz** sadrži sve nevažeće objekte koje je algoritam pronašao.
- Sloj **Izlaz greške** je tačkasti sloj koji pokazuje na mesta gde su nevažeći objekti pronađeni.

Atributske tabele generisanih slojeva sadržaće neke dodatne informacije („message“ za sloj **greške**, „FID“ i „_errors“ za **nevažeći** sloj i samo „FID“ za **validni** sloj):

Atributska tabela svakog generisanog vektorskog sloja sadržaće neke dodatne informacije (broj pronađenih grešaka i tipovi grešaka):



Слика24.110: Levo: ulazni sloj. Desno: validni sloj (zeleno), nevažeći sloj (narandžasto)

Podrazumevani meni: *Vektor ► Alatke za geometriju*

See also

Check Geometry, *Popravi geometrije*, *Provera pokrivenosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT_LAYER`	[vektor: geometrija]	Ulazni vektorski sloj
Metod	`METHOD`	[nabrajanje] Podrazumevano: 2	Metod koji se koristi za proveru validnosti. Opcije: • 0: Onaj izabran u podešavanjima digitalizovanja • 1: QGIS • 2: GEOS
Zanemari samopresek prstena	`IGNORE_RING_SELF_INT`	[logička vrednost] Podrazumevano: False	Zanemari prstenove koji se samopresecaju pri proveri validnosti.
Validni izlaz	`VALID_OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj koji sadrži kopiju validnih objekata izvornog sloja. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.614 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Nevažeći izlaz	`INVALID_OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Vektorski sloj koji sadrži kopiju nevažećih objekata izvornog sloja sa poljem `_errors` koje navodi rezime pronađene(ih) greške(a). *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Izlaz greške	`ERROR_OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Tačkasti sloj tačne pozicije otkrivenih problema validnosti sa poljem `message` koje opisuje pronađenu(e) grešku(e). *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj grešaka	`ERROR_COUNT`	[numerički: ceo broj]	Broj geometrija koje su izazvale greške.
Izlaz greške	`ERROR_OUTPUT`	[vektor: tačka]	Tačkasti sloj tačne pozicije otkrivenih problema validnosti sa poljem `message` koje opisuje pronađenu(e) grešku(e).
Broj nevažećih objekata	`INVALID_COUNT`	[numerički: ceo broj]	Broj nevažećih geometrija.
Nevažeći izlaz	`INVALID_OUTPUT`	[isto kao ulaz]	Vektorski sloj koji sadrži kopiju nevažećih objekata izvornog sloja sa poljem `_errors` koje navodi rezime pronađene(ih) greške(a).
Broj validnih objekata	`VALID_COUNT`	[numerički: ceo broj]	Broj validnih geometrija.
Validni izlaz	`VALID_OUTPUT`	[isto kao ulaz]	Vektorski sloj koji sadrži kopiju validnih objekata izvornog sloja.

Python kod

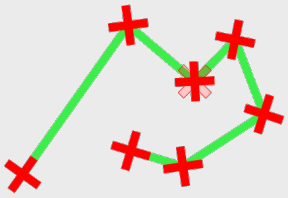
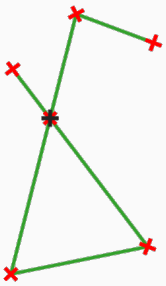
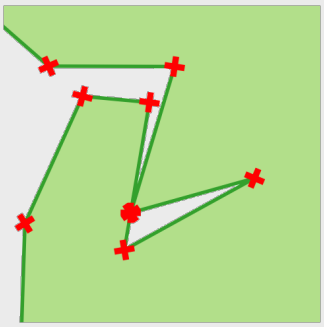
ID algoritma: `native:checkvalidity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

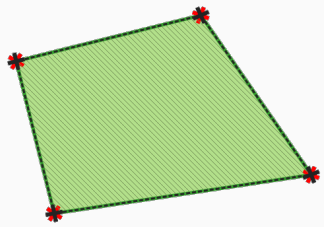
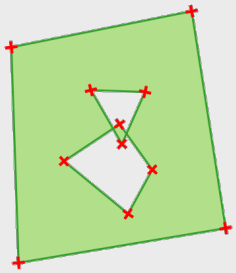
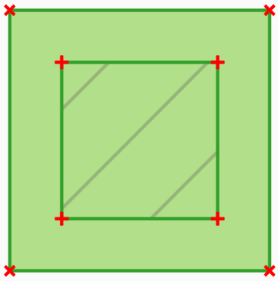
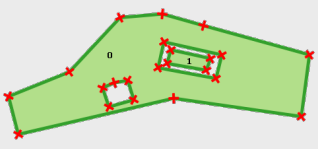
Tipovi poruka o grešci i njihova značenja

Табела24.616: Neki primeri neuspeha provere geometrije u QGIS-u

Kontekst geometrije	GEOS provera i poruka o grešci	QGIS provera i poruka o grešci
Uzastopne tačke na liniji imaju iste koordinate 	✗ *Ponovljena tačka*	✗ *Linija a sadrži x dupliran(ih) čvor(ova) na b*
Segmenti linije se seku međusobno 		✗ *Segmenti a i b linije c seku se u d*
Geometrija poligona dodiruje samu sebe i generiše prsten 	✗ *Samopresek prstena*	✗ *Samopresek prstena*

наставак на следећој страни

Табела 24.616 – настављено са претходне стране

Kontekst geometrije	GEOS provera i poruka o grešci	QGIS provera i poruka o grešci
Dva prstena (spoljašnji ili unutrašnji) geometrije poligona su identična 	✗ *Duplirani prstenovi*	
Geometrija dodiruje samu sebe 	✗ *Samopresek*	
Geometrija poligona je iznad druge geometrije poligona 	✗ *Ugnežđena ljuska*	
Deo MultiPolygon geometrije je unutar rupe MultiPolygon geometrije 		✗ *Poligon a leži unutar poligona b*

наставак на следећој страни

Табела 24.616 – настављено са претходне стране

Kontekst geometrije	GEOS provera i poruka o grešci	QGIS provera i poruka o grešci
Geometrija tačke nema ispravan par koordinata. Par koordinata ne sadrži vrednost geografske širine i vrednost geografske dužine tim redosledom.	*Nevažeća koordinata*	

Prikupi geometrije

Uzima vektorski sloj i prikuplja njegove geometrije u nove višedelne geometrije.

Može se zadati jedan ili više atributa da bi se prikupile samo geometrije koje pripadaju istoj klasi (imaju istu vrednost za zadate atribute), ili se alternativno mogu prikupiti sve geometrije.

Sve izlazne geometrije biće pretvorene u višedelne geometrije, čak i one sa samo jednim delom. Ovaj algoritam ne stapa preklapajuće geometrije - biće prikupljene zajedno bez izmene oblika svakog dela geometrije.

Za alternativne opcije pogledajte algoritme „Unapredi u višedelno“ ili „Agregiraj“.

Podrazumevani meni: *Vektor ► Alatke za geometriju*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Agregiraj, *Unapredi u višedelno*, *Stapanje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polja jedinstvenog ID-ja	`FIELD`	[polje tabele: bilo koji] [lista]	Izaberite jedan ili više atributa za prikupljanje geometrija
Prikupljeno	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa prikupljenim geometrijama

Izlazi

Oznaka	Назив	Tip	Opis
Prikupljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za prikupljene geometrije. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Python kod

ID algoritma: `native:collect`

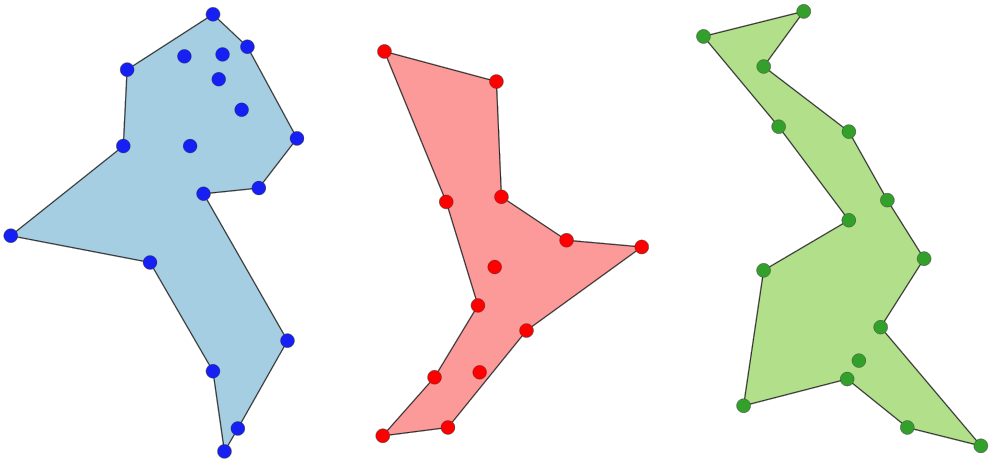
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Konkavni omotač (po objektu)

`Dodato u 3.44`

Računa konkavni omotač za svaki višetačkasti objekat u ulaznom sloju. Za računanje konkavnog omotača koji pokriva ceo sloj pogledajte algoritam *Konkavni omotač (po sloju)*.

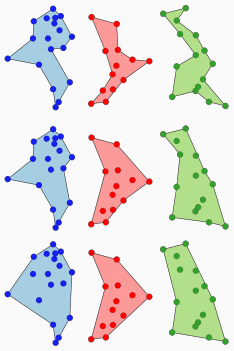


Слика24.111: Konkavni omotači po objektu

See also

Konveksni omotač

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni vektorski sloj sa višetačkastim objektima
Prag	`ALPHA`	[numerički: decimalni] Podrazumevano: 0.3	Broj od 0 (maksimalni konkavni omotač) do 1 (konveksni omotač).  Слика24.112: Konkavni omotači sa različitim pragovima (0.3, 0.6, 0.9)
Dozvoli rupe	`HOLES`	[logička vrednost] Podrazumevano: True	Izaberite da li da dozvolite rupe u konačnom konkavnom omotaču
Konkavni omotači	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Konkavni omotači	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj sa poligonima koji pokrivaju pojedinačne višetačkaste objekte. Dodaju se polja površine i obima.

Python kod

ID algoritma: `native:concavehullbyfeature`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

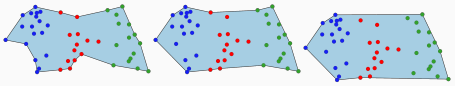
Konkavni omotač (po sloju)

Računa konkavni omotač koji pokriva sve objekte iz ulaznog tačkastog sloja. Za računanje konkavnog omotača koji pokriva pojedinačne objekte iz sloja pogledajte algoritam *Konkavni omotač (po objektu)*.

See also

Konveksni omotač

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Prag	`ALPHA`	[numerički: decimalni] Podrazumevano: 0.3	Broj od 0 (maksimalni konkavni omotač) do 1 (konveksni omotač).  Слика24.113: Konkavni omotači sa različitim pragovima (0.3, 0.6, 0.9)
Dozvoli rupe	`HOLES`	[logička vrednost] Podrazumevano: True	Izaberite da li da dozvolite rupe u konačnom konkavnom omotaču
Podeli višedelnu geometriju na jednodelne geometrije	`NO_MULTIGEOMETRY`	[logička vrednost] Podrazumevano: False	Označite ako želite da imate jednodelne geometrije umesto višedelnih.
Konkavni omotač	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Konkavni omotač	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj koji pokriva sve objekte iz ulaznog tačkastog sloja.

Python kod

ID algoritma: `native:concavehull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvori tip geometrije

Generiše novi sloj na osnovu postojećeg, sa drugačijim tipom geometrije.

Atributska tabela izlaznog sloja ista je kao ona ulaznog sloja.

Nisu sva pretvaranja moguća. Na primer, linijski sloj se može pretvoriti u tačkasti sloj, ali se tačkasti sloj ne može pretvoriti u linijski sloj.

See also

Poligonizuj, *Linije u poligone*, *Poligoni u linije*, *Tačke u putanju*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Novi tip geometrije	`TYPE`	[nabrajanje] Podrazumevano: 0	Tip geometrije koji se primenjuje na izlazne objekte. Jedno od: • 0 — Centroidi • 1 — Čvorovi • 2 — Linijski nizovi • 3 — Višestruki linijski nizovi • 4 — Poligoni
Pretvoreno	`OUTPUT`	[vektor: geometrija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[vektor: geometrija]	Izlazni vektorski sloj - tip zavisi od parametara

Python kod

ID algoritma: `native:convertgeometrytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pretvori u zakrivljene geometrije

Pretvara geometriju u njen zakrivljeni geometrijski ekvivalent.

Već zakrivljene geometrije biće zadržane bez izmene.

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija ili poligon]	Ulazni vektorski sloj
Tolerancija maksimalnog rastojanja	`DISTANCE`	[numerički: decimalni] Podrazumevano: 0.000001	Maksimalno rastojanje dozvoljeno između originalne lokacije temena i mesta gde bi pala na pretvorenim zakrivljenim geometrijama
Tolerancija maksimalnog ugla	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.000001	Segmenti se smatraju pogodnim za zamenu lukom ako su tačke sve pravilno raspoređene na kandidat luku. Ovaj parametar zadaje maksimalno ugaono odstupanje (u stepenima) dozvoljeno pri proveri pravilnog razmaka tačaka. Između 0 i 45°.
Krive	`OUTPUT`	[vektor: složena kriva ili kriva poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Krive	`OUTPUT`	[vektor: složena kriva ili kriva poligon]	Izlazni vektorski sloj sa zakrivljenim geometrijama

Python kod

ID algoritma: `native:converttocurves`

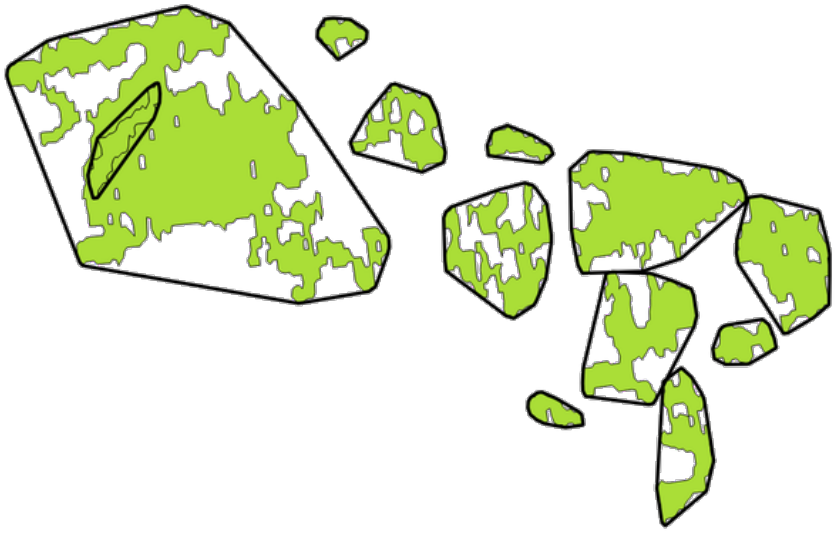
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Konveksni omotač

Računa konveksni omotač za svaki objekat u ulaznom sloju.

Za računanje konveksnog omotača koji pokriva ceo sloj ili grupisane podskupove objekata pogledajte algoritam „Minimalna granična geometrija“.



Слика24.114: Crne linije označavaju konveksni omotač za svaki objekat sloja

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za geoobradu*

See also

Minimalna granična geometrija, *Konkavni omotač (po sloju)*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Konveksni omotač	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Konveksni omotač	`OUTPUT`	[vektor: poligon]	Izlazni (konveksni omotač) vektorski sloj. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: `area` i `perimeter` generisanog poligona.

Python kod

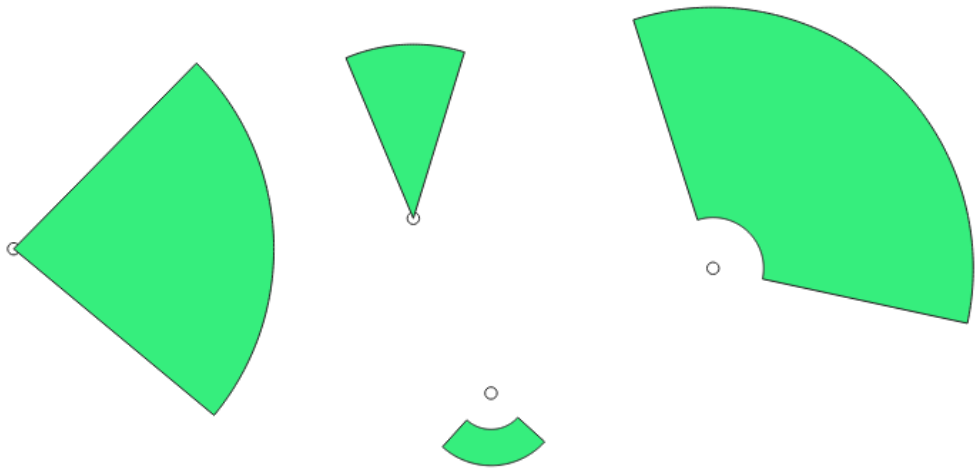
ID algoritma: `native:convexhull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napravi klinaste bafere

Pravi bafere klinastog oblika iz ulaznih tačaka.



Слика24.115: Klinasti baferi

Nativni izlaz ovog algoritma su CurvePolygon geometrije, ali one mogu biti automatski segmentizovane u poligone u zavisnosti od izlaznog formata.

See also

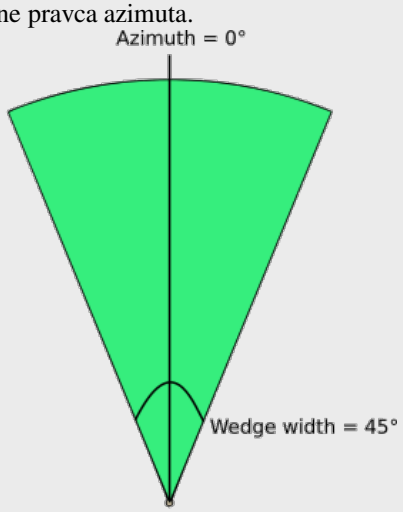
Bafer, *Bafer promenljive širine (po M vrednosti)*, *Suženi baferi*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Azimut (stepeni od severa)	`AZIMUTH`	[numerički: decimalni] Podrazumevano: 0.0	Ugao (u stepenima) kao srednja vrednost klina

наставак на следећој страни

Табела 24.629 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Širina klina (u stepenima)	`WIDTH`	[numerički: decimalni] Podrazumevano: 45.0	Širina (u stepenima) bafera. Klin će se protezati do polovine ugaone širine sa svake strane pravca azimuta.  Слика24.116: Vrednosti azimuta i širine klinastog bafera
Spoljašnji poluprečnik	`OUTER_RADIUS`	[numerički: decimalni] Podrazumevano: 1.0	Spoljašnja *veličina* (dužina) klina: veličina se meri od izvorne tačke do ivice klinastog oblika.
Unutrašnji poluprečnik Opciono	`INNER_RADIUS`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost unutrašnjeg poluprečnika. Ako je 0, klin će početi od izvorne tačke.
Baferi	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Baferi	`OUTPUT`	[vektor: poligon]	Izlazni (klinasti bafer) vektorski sloj

Python kod

ID algoritma: `native:wedgebuffers`

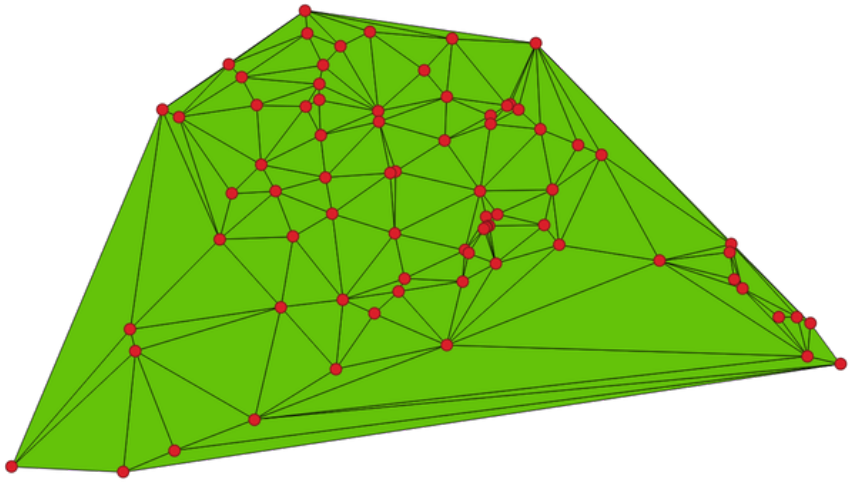
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Delonijeva triangulacija

Pravi poligonski sloj sa Delonijevom triangulacijom koja odgovara ulaznom tačkastom sloju.



Слика24.117: Delonijeva triangulacija nad tačkama

Podrazumevani meni: *Vektor ► Alatke za geometriju*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Tolerancija Opciono `Dodato u 3.34`	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Zadaje opcionu toleranciju lepljenja koja se može koristiti za poboljšanje robusnosti triangulacije.
Dodaj ID-jeve tačaka u izlaz `Dodato u 3.34`	`ADD_ATTRIBUTES`	[logička vrednost] Podrazumevano: True	Zadaje da li treba dodati polja koja čuvaju ID uključenih tačkastih objekata u izlaz. Ako je False, koristi se `id` polje za identifikaciju poligona.
Delonijeva triangulacija	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Delonijeva triangulacija	`OUTPUT`	[vektor: poligon]	Izlazni (Delonijeva triangulacija) vektorski sloj

Python kod

ID algoritma: `native:delaunaytriangulation`

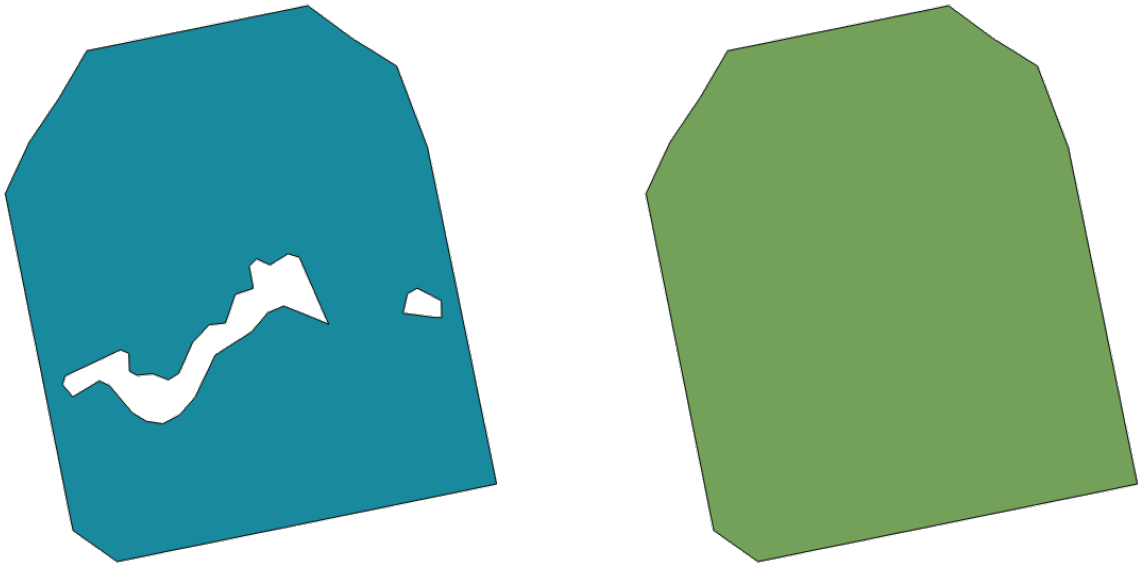
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Obriši rupe

Uzima poligonski sloj i uklanja rupe u poligonima. Pravi novi vektorski sloj u kojem su poligoni sa rupama zamenjeni poligonima sa samo njihovim spoljašnjim prstenom. Atributi se ne menjaju.

Opcioni parametar minimalne površine omogućava uklanjanje samo rupa koje su manje od zadatog praga površine. Ostavljanje ovog parametra na `0.0` daje uklanjanje svih rupa.



Слика24.118: Pre i posle čišćenja

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

See also

qgisfixgeometryhole

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Ukloni rupe sa površinom manjom od Opciono	`MIN_AREA`	[numerički: decimalni] Podrazumevano: 0.0	Biće obrisane samo rupe sa površinom manjom od ovog praga. Sa vrednošću `0.0`, biće obrisane **sve** rupe.
Očišćeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Očišćeno	`OUTPUT`	[isto kao ulaz]	Izlazni (očišćeni) vektorski sloj

Python kod

ID algoritma: `native:deleteholes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

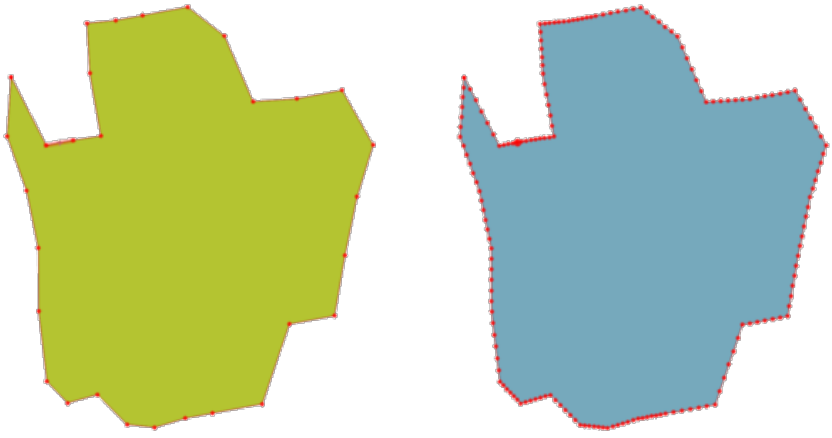
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zgusni po broju

Uzima poligonski ili linijski sloj i generiše novi u kojem geometrije imaju veći broj temena od originalnog.

Ako su Z ili M vrednosti prisutne u geometrijama, one će biti linearno interpolirane na dodatim temenima.

Broj novih temena koja se dodaju svakom segmentu zadaje se kao ulazni parametar.



Слика24.119: Crvene tačke prikazuju temena pre i posle zgušnjavanja

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

Podrazumevani meni: *Vektor ► Alatke za geometriju*

See also

Zgusni po intervalu

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Temena za dodavanje	`VERTICES`	[numerički: ceo broj] Podrazumevano: 1	Broj temena koja se dodaju svakom segmentu
Zgusnuto	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zgusnuto	`OUTPUT`	[isto kao ulaz]	Izlazni (zgusnuti) vektorski sloj

Python kod

ID algoritma: `native:densifygeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zgusni po intervalu

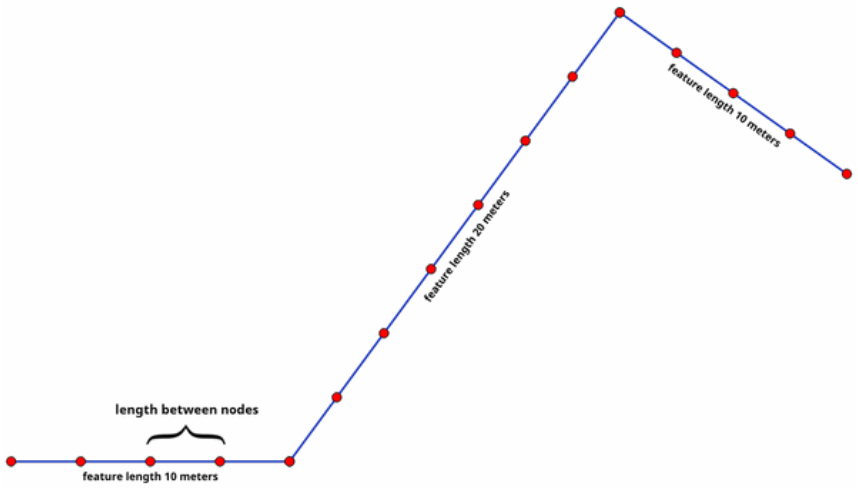
Uzima poligonski ili linijski sloj i generiše novi u kojem geometrije imaju veći broj temena od originalnog.

Geometrije se zgušnjavaju dodavanjem pravilno raspoređenih dodatnih temena unutar svakog segmenta tako da maksimalno rastojanje između bilo koja dva temena ne prelazi zadato rastojanje.

Ako su Z ili M vrednosti prisutne u geometrijama, one će biti linearno interpolirane na dodatim temenima.

Primer

Zadavanje rastojanja od 3 uzrokovalo bi da se segment `[0 0] -> [10 0]` pretvori u `[0 0] -> [2.5 0] -> [5 0] -> [7.5 0] -> [10 0]`, pošto su na segmentu potrebna 3 dodatna temena, a njihovo raspoređivanje u koracima od 2.5 omogućava im da budu ravnomerno raspoređena preko segmenta.



Слика24.120: Zgušnjavanje geometrije na zadatom intervalu

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

See also

Zgusni po broju

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Interval između temena za dodavanje	`INTERVAL`	[numerički: decimalni] Podrazumevano: 1.0	Maksimalno rastojanje između dva uzastopna temena
Zgusnuto	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zgusnuto	`OUTPUT`	[isto kao ulaz]	Izlazni (zgusnuti) vektorski sloj

Python kod

ID algoritma: `native:densifygeometriesgivenaninterval`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

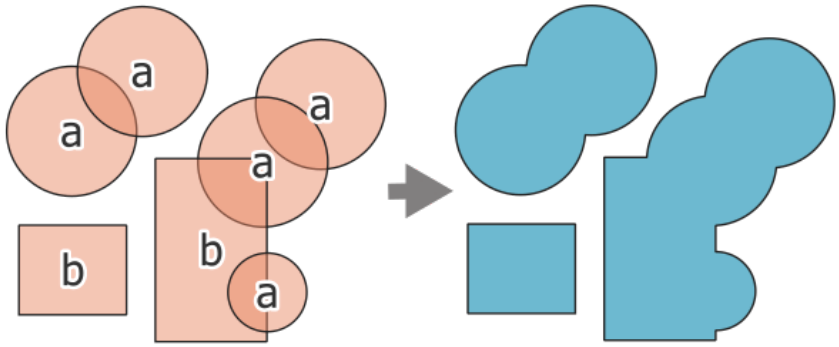
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Stapanje

Uzima vektorski sloj i kombinuje njegove objekte u nove objekte. Može se zadati jedan ili više atributa da bi se stopili objekti koji pripadaju istoj klasi (imaju istu vrednost za zadate atribute), ili se alternativno svi objekti mogu stopiti u jedan objekat.

Sve izlazne geometrije biće pretvorene u višedelne geometrije. U slučaju da je ulaz poligonski sloj, zajedničke granice susednih poligona koji se stapaju biće izbrisane. Ako je omogućeno, opciono podešavanje „Zadrži razdvojene objekte odvojenima“ uzrokovaće da objekti i delovi koji se ne preklapaju niti dodiruju budu izvezeni kao odvojeni objekti (umesto delova jednog višedelnog objekta).

Rezultujuća atributska tabela imaće ista polja kao ulazni sloj. Vrednosti u poljima izlaznog sloja su one prvog ulaznog objekta koji se obradi.



Слика24.121: Stapanje sloja u jedan višedelni objekat

Podrazumevani meni: *Vektor* ► *Alatke za geoobradu*

Упозорење

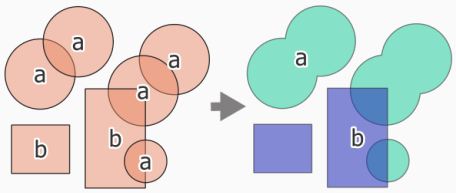
Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

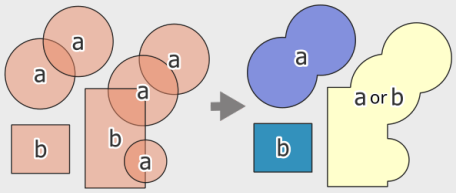
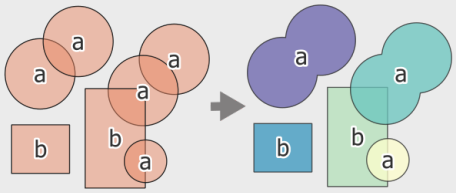
Stapanje pokrivenosti, *Agregiraj*, *Prikupi geometrije*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polja za stapanje Opciono	`FIELD`	[polje tabele: bilo koji] [lista] Podrazumevano: []	Objekti koji imaju istu vrednost za izabrano polje(a) biće zamenjeni jednim, a njihove geometrije se spajaju. Ako se ne zada nijedno polje, tada se svi objekti stapaju, dajući jedan (višedelni) objekat.  Слика24.122: Stapanje poligonskog sloja po zajedničkom atributu (2 višedelna objekta)
Stopljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Zadrži razdvojene objekte odvojenima	`SEPARATE_DISJOINT`	[logička vrednost] Podrazumevano: False	Delovi stopljenih objekata izvoze se kao odvojeni objekti (umesto delova višedelnog objekta).  Слика24.123: izvor (levo), stopi sve (3 različita objekta - desno)  Слика24.124: izvor (levo), stopi po polju (5 različitih objekata - desno)

Izlazi

Oznaka	Назив	Tip	Opis
Stopljeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa stopljenim geometrijama

Python kod

ID algoritma: `native:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Prekri (postavi Z vrednost iz rastera)

Koristi vrednosti uzorkovane iz kanala unutar rasterskog sloja za postavljanje Z vrednosti za svako preklapajuće teme u geometriji objekta. Vrednosti rastera se opciono mogu skalirati unapred zadatim iznosom.

Ako Z vrednosti već postoje u sloju, biće prepisane novom vrednošću. Ako Z vrednosti ne postoje, geometrija će biti unapređena da uključi Z dimenziju.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim Z

See also

Postavi M vrednost iz rastera, *Postavi Z vrednost*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rasterski sloj	`RASTER`	[raster]	Rasterski sloj sa Z vrednostima
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Rasterski kanal iz kojeg se uzimaju Z vrednosti
Vrednost za NoData ili teme koja se ne seku	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost koja se koristi u slučaju da teme ne preseca (validan piksel) rastera
Faktor skaliranja	`SCALE`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja: vrednosti kanala se množe ovom vrednošću.
Pomeraj	`OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost pomeraja: algebarski se dodaje vrednostima kanala nakon primene „Faktora skaliranja“.
Ažurirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa Z vrednostima iz rasterskog sloja). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Ažurirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa Z vrednostima iz rasterskog sloja

Python kod

ID algoritma: `native:setzfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Odbaci M/Z vrednosti

Uklanja M (mera) ili Z (nadmorska visina) vrednosti iz ulaznih geometrija.

See also

Postavi M vrednost, *Postavi Z vrednost*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj sa M ili Z vrednostima
Odbaci M vrednosti	`DROP_M_VALUES`	[logička vrednost] Podrazumevano: False	Uklanja M vrednosti iz geometrija
Odbaci Z vrednosti	`DROP_Z_VALUES`	[logička vrednost] Podrazumevano: False	Uklanja Z vrednosti iz geometrija
Z/M odbačeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Z/M odbačeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (istovetan ulaznom sloju, osim što su M i/ili Z dimenzije uklonjene iz geometrija).

Python kod

ID algoritma: `native:dropmzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Eliminiši izabrane poligone

Kombinuje izabrane poligone ulaznog sloja sa određenim susednim poligonima brisanjem njihove zajedničke granice. Susedni poligon može biti ili onaj sa najvećom ili najmanjom površinom ili onaj koji deli najveću zajedničku granicu sa poligonom koji se eliminiše.

Eliminacija se obično koristi da se otarasite tankih poligona, tj. sitnih poligona koji su rezultat procesa preseka poligona gde su granice ulaza slične, ali ne identične.

Podrazumevani meni: *Vektor ► Alatke za geoobradu*

See also

Popravi geometrije

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Stopi izbor sa susednim poligonom sa	`MODE`	[nabrajanje] Podrazumevano: Nije postavljeno	Izaberite parametar koji se koristi da biste se otarasili izabranih poligona: • 0 — Najveća površina • 1 — Najmanja površina • 2 — Najveća zajednička granica
Eliminisano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Eliminisano	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj.

Python kod

ID algoritma: `qgis:eliminateselectedpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasturi linije

Uzima linijski sloj i pravi novi u kojem je svaki linijski sloj zamenjen skupom linija koje predstavljaju segmente originalne linije.

Svaka linija u rezultujućem sloju sadrži samo početnu i krajnju tačku, bez međutemena između njih.

Ako se ulazni sloj sastoji od CircularStrings ili CompoundCurves, izlazni sloj biće istog tipa i sadržati samo pojedinačne segmente krivih.

Белешка

- Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.
- Ovaj algoritam ne zahteva ispravne geometrije na ulazu.



Слика24.125: Originalni linijski sloj i rasturen

Omogućava *izmenu objekata na licu mesta* za linijske objekte

See also

Podpodeli, *Podniska linije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Raščlanjeno	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Raščlanjeno	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj sa objektima koji predstavljaju svaki segment ulaznog sloja.

Python kod

ID algoritma: `native:explodelines`

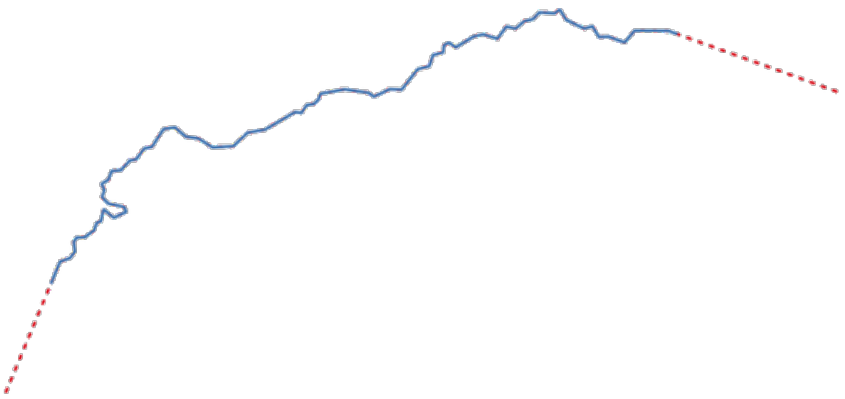
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Produži linije

Produžava geometriju linije za zadati iznos na početku i kraju linije.

Linije se produžavaju koristeći smer prvog i poslednjeg segmenta u liniji.



Слика24.126: Crvene crte predstavljaju početno i krajnje proširenje originalnog sloja

Omogućava *izmenu objekata na licu mesta* za linijske objekte

See also

Podniska linije

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj

наставак на следећој страни

Табела 24.650 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Početno rastojanje	`START_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje za koje se produžava prvi segment linije (početna tačka)
Krajnje rastojanje	`END_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje za koje se produžava poslednji segment linije (krajnja tačka)
Produženo	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Produženo	`OUTPUT`	[vektor: linija]	Izlazni (produženi) linijski vektorski sloj.

Python kod

ID algoritma: `native:extendlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvoj M vrednosti

Izdvaja M vrednosti iz geometrija u atribute objekata.

Podrazumevano se izdvaja samo M vrednost iz prvog temena svakog objekta, međutim algoritam može opciono izračunati statistiku na svim M vrednostima geometrije, uključujući zbir, srednju vrednost, minimum i maksimum.

See also

Izdvoj Z vrednosti, *Postavi M vrednost*, *Odbaci M/Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rezimei za računanje	`SUMMARIES`	[nabrajanje] Podrazumevano: [0]	Statistika na M vrednostima geometrije. Jedno ili više od: • 0 — Prvi • 1 — Poslednji • 2 — Broj • 3 — Zbir • 4 — Srednja vrednost • 5 — Medijana • 6 — St.dev (pop) • 7 — Minimum • 8 — Maksimum • 9 — Raspon • 10 — Manjina • 11 — Većina • 12 — Raznolikost • 13 — Q1 • 14 — Q3 • 15 — IQR
Prefiks izlazne kolone	`COLUMN_PREFIX`	[niska] Podrazumevano: 'm_'	Prefiks za izlaznu (M) kolonu
Izdvojeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izdvojeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (sa M vrednostima)

Python kod

ID algoritma: `native:extractmvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvoj određena temena

Uzima vektorski sloj i generiše tačkasti sloj sa tačkama koje predstavljaju određena temena u ulaznim geometrijama.

Na primer, ovaj algoritam se može koristiti za izdvajanje prvog ili poslednjeg temena u geometriji. Atributi povezani sa svakom tačkom isti su kao oni povezani sa objektom kojem teme pripada.

Parametar indeksa temena prihvata nisku razdvojenu zarezima koja zadaje indekse temena za izdvajanje. Prvom temenu odgovara indeks 0, drugo teme ima indeks 1, itd. Negativni indeksi se mogu koristiti za pronalaženje temena na kraju geometrije, npr. indeks -1 odgovara poslednjem temenu, -2 odgovara pretposlednjem temenu, itd.

Temenima se dodaju dodatna polja koja označavaju određeni položaj temena (npr. 0, -1, itd.), originalni indeks temena, deo temena i njegov indeks unutar dela (kao i njegov prsten za poligone), rastojanje duž originalne geometrije i ugao simetrale temena za originalnu geometriju.

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Izdvoj temena, *Filtriraj temena po M vrednosti*, *Filtriraj temena po Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Indeksi temena	`VERTICES`	[niska] Podrazumevano: '0'	Niska razdvojena zarezima indeksa temena za izdvajanje.
Temena	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Temena	`OUTPUT`	[vektor: tačka]	Izlazni (tačkasti) vektorski sloj koji sadrži zadata temena iz geometrija ulaznog sloja. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: • `vertex_pos`: indeks temena u geometriji, kako je zatražio korisnik. • `vertex_index`: indeks temena unutar geometrije. Prvo teme je indeksirano kao 0. • `vertex_part`: broj dela geometrije kojem teme pripada. Za jednodelne geometrije, ovo je uvek 0. Za višedelne geometrije (npr. MultiPolygon ili MultiLine), ovo označava iz kog dela teme potiče. • `vertex_part_ring`: za geometrije poligona, ovo označava da li teme pripada spoljašnjem prstenu (0) ili jednom od unutrašnjih prstenova (1, 2, …). • `vertex_part_index`: indeks temena u njegovom određenom prstenu (spoljašnjem ili unutrašnjem). • `distance`: kumulativna dužina segmenata linije od prvog temena geometrije. • `angle`: ugao (u stepenima) između prethodnog i trenutnog temena.

Python kod

ID algoritma: `native:extractspecificvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

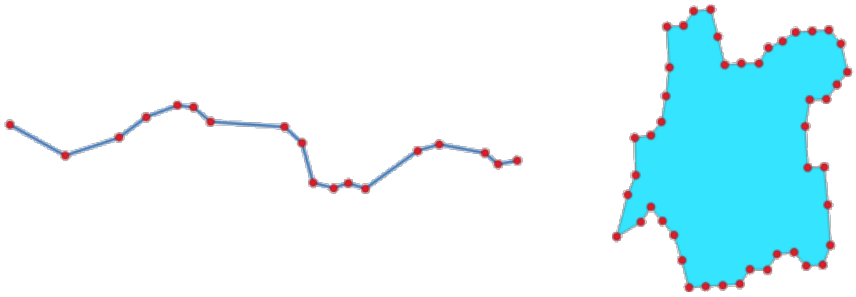
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvoj temena

Uzima vektorski sloj i generiše tačkasti sloj sa tačkama koje predstavljaju temena u ulaznim geometrijama.

Atributi povezani sa svakom tačkom isti su kao oni povezani sa objektom kojem teme pripada.

Temenima se dodaju dodatna polja koja označavaju indeks temena (počev od 0), deo objekta i njegov indeks unutar dela (kao i njegov prsten za poligone), rastojanje duž originalne geometrije i ugao simetrale temena za originalnu geometriju.



Слика24.127: Temena izdvojena za linijski i poligonski sloj

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Izdvoj određena temena, *Filtriraj temena po M vrednosti*, *Filtriraj temena po Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Temena	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Temena	`OUTPUT`	[vektor: tačka]	Izlazni (tačkasti) vektorski sloj koji sadrži temena iz geometrija ulaznog sloja. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: • `vertex_index`: indeks temena unutar geometrije. Prvo teme je indeksirano kao 0. • `vertex_part`: broj dela geometrije kojem teme pripada. Za jednodelne geometrije, ovo je uvek 0. Za višedelne geometrije (npr. MultiPolygon ili MultiLine), ovo označava iz kog dela teme potiče. • `vertex_part_ring`: za geometrije poligona, ovo označava da li teme pripada spoljašnjem prstenu (0) ili jednom od unutrašnjih prstenova (1, 2, …). • `vertex_part_index`: indeks temena u njegovom određenom prstenu (spoljašnjem ili unutrašnjem). • `distance`: rastojanje od prethodnog temena. • `angle`: vraća približan ugao na temenu. Obično prosečan ugao između susednih segmenata, i može se zamisliti kao orijentacija linije koja prati zakrivljenost geometrije na zadatom temenu.

Python kod

ID algoritma: `native:extractvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvoj Z vrednosti

Izdvaja Z vrednosti iz geometrija u atribute objekata.

Podrazumevano se izdvaja samo Z vrednost iz prvog temena svakog objekta, međutim algoritam može opciono izračunati statistiku na svim Z vrednostima geometrije, uključujući zbir, srednju vrednost, minimum i maksimum.

See also

Izdvoj M vrednosti, *Postavi Z vrednost*, *Odbaci M/Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rezimei za računanje	`SUMMARIES`	[nabrajanje] Podrazumevano: [0]	Statistika na Z vrednostima geometrije. Jedno ili više od: • 0 — Prvi • 1 — Poslednji • 2 — Broj • 3 — Zbir • 4 — Srednja vrednost • 5 — Medijana • 6 — St.dev (pop) • 7 — Minimum • 8 — Maksimum • 9 — Raspon • 10 — Manjina • 11 — Većina • 12 — Raznolikost • 13 — Q1 • 14 — Q3 • 15 — IQR
Prefiks izlazne kolone	`COLUMN_PREFIX`	[niska] Podrazumevano: 'z_'	Prefiks za izlaznu (Z) kolonu
Izdvojeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izdvojeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (sa Z vrednostima)

Python kod

ID algoritma: `native:extractzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

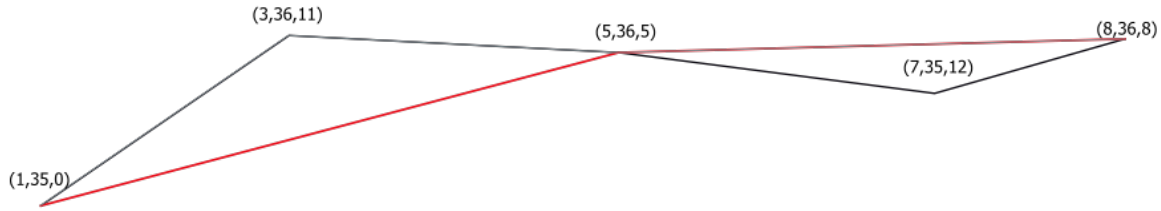
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filtriraj temena po M vrednosti

Filtrira temena na osnovu njihove M vrednosti, vraćajući geometrije sa samo onim temenima koja imaju M vrednost veću od ili jednaku zadatoj minimalnoj vrednosti i/ili manju od ili jednaku maksimalnoj vrednosti.

Ako minimalna vrednost nije zadata, tada se proverava samo maksimalna vrednost, i slično, ako maksimalna vrednost nije zadata, tada se proverava samo minimalna vrednost.



Слика24.128: Crvena linija predstavlja crnu liniju sa samo onim temenima čija je M vrednost <=10.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim M

Упозорење

U zavisnosti od atributa ulazne geometrije i korišćenih filtera, geometrije napravljene ovim algoritmom možda više neće biti validne.

See also

Filtriraj temena po Z vrednosti, *Izdvoj temena*, *Izdvoj određena temena*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj iz kojeg se uklanjaju temena
Minimum Opciono	`MIN`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Minimum dozvoljenih M vrednosti
Maksimum Opciono	`MAX`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Maksimum dozvoljenih M vrednosti

наставак на следећој страни

Табела 24.660 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Filtrirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Filtrirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj objekata sa samo filtriranim temenima.

Python kod

ID algoritma: `native:filterverticesbym`

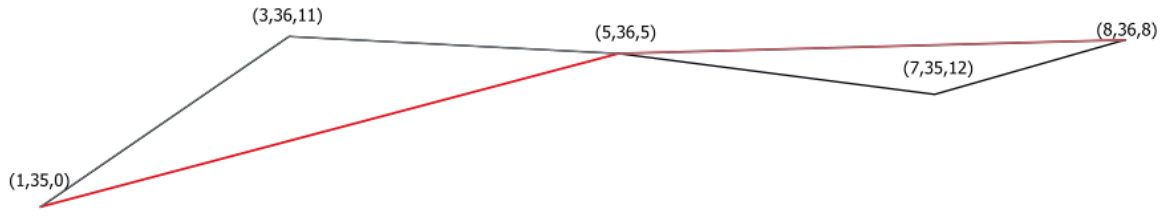
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filtriraj temena po Z vrednosti

Filtrira temena na osnovu njihove Z vrednosti, vraćajući geometrije sa samo onim temenima koja imaju Z vrednost veću od ili jednaku zadatoj minimalnoj vrednosti i/ili manju od ili jednaku maksimalnoj vrednosti.

Ako minimalna vrednost nije zadata, tada se proverava samo maksimalna vrednost, i slično, ako maksimalna vrednost nije zadata, tada se proverava samo minimalna vrednost.



Слика24.129: Crvena linija predstavlja crnu liniju sa samo onim temenima čija je Z vrednost <=10.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim Z

Упозорење

U zavisnosti od atributa ulazne geometrije i korišćenih filtera, geometrije napravljene ovim algoritmom možda više neće biti validne. Možda ćete morati da pokrenete algoritam *Popravi geometrije* da biste obezbedili njihovu validnost.

See also

Filtriraj temena po M vrednosti, *Izdvoj temena*, *Izdvoj određena temena*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj iz kojeg se uklanjaju temena
Minimum Opciono	`MIN`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Minimum dozvoljenih Z vrednosti
Maksimum Opciono	`MAX`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Maksimum dozvoljenih Z vrednosti
Filtrirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Filtrirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj objekata sa samo filtriranim temenima.

Python kod

ID algoritma: `native:filterverticesbyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Popravi geometrije

Pokušava da napravi validnu reprezentaciju zadate nevažeće geometrije bez gubljenja bilo kog ulaznog temena. Već validne geometrije se vraćaju bez daljeg mešanja. Uvek daje sloj sa višestrukom geometrijom.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte bez omogućenog M

Упозорење

M vrednosti će biti odbačene iz izlaza.

See also

Proveri validnost, *Fix Geometry*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Metod popravke	`METHOD`	[nabrajanje] Podrazumevano: 1	Metod koji se koristi za popravku geometrija. Jedno od: • 0 — `Linework`: kombinuje sve prstenove u skup povezanih linija, a zatim iz te linijske mreže izdvaja validne poligone • 1 — `Structure`: prvo čini sve prstenove validnim, a zatim spaja ljuske i oduzima rupe od ljuski da bi generisao validan rezultat. Pretpostavlja da su rupe i ljuske ispravno kategorizovane. **Пажња** Korišćenje ovog metoda zahteva QGIS instaliran sa GEOS >= 3.10 (pogledajte meni *Pomoć* ► *O programu*).
Popravljene geometrije	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Popravljene geometrije	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa popravljenim geometrijama.

Python kod

ID algoritma: `native:fixgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Nametni pravilo desne ruke

Prisiljava geometrije poligona da poštuju pravilo desne ruke, u kojem je površina omeđena poligonom sa desne strane granice. Naročito, spoljašnji prsten je orijentisan u smeru kazaljke na satu, a bilo koji unutrašnji prstenovi suprotno od kazaljke na satu.

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni vektorski sloj
Preorijentisano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Preorijentisano	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj sa preorijentisanim geometrijama.

Python kod

ID algoritma: `native:forcerhr`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Geodetsko deljenje linije na antimeridijanu

Deli liniju na više geodetskih segmenata, svaki put kada linija pređe antimeridijan (±180 stepeni geografske dužine).

Deljenje na antimeridijanu pomaže vizuelnom prikazu linija u nekim projekcijama. Vraćena geometrija će uvek biti višedelna geometrija.

Svaki put kada segmenti linije u ulaznoj geometriji pređu antimeridijan, biće podeljeni na dva segmenta, sa geografskom širinom tačke preloma određenom pomoću geodetske linije koja povezuje tačke sa obe strane tog segmenta. Pri računanju ove tačke preloma koristiće se trenutno podešavanje elipsoida projekta.

Ako ulazna geometrija sadrži M ili Z vrednosti, one će biti linearno interpolirane za nova temena napravljena na antimeridijanu.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Omogućava *izmenu objekata na licu mesta* za linijske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Podeljeno	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podeljeno	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj podeljen na antimeridijanu.

Python kod

ID algoritma: `native:antimeridiansplit`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Geometrija pomoću izraza

Ažurira postojeće geometrije (ili pravi nove geometrije) za ulazne objekte korišćenjem QGIS izraza.

Ovo omogućava složene izmene geometrije koje mogu iskoristiti svu fleksibilnost mehanizma za izraze QGIS-a za manipulaciju i pravljenje geometrija za izlazne objekte.

Za pomoć sa funkcijama QGIS izraza, pogledajte ugrađenu pomoć dostupnu u *graditelju izraza*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Izlazni tip geometrije	`OUTPUT_GEOMETRY`	[nabrajanje] Podrazumevano: 0	Izlazna geometrija u velikoj meri zavisi od izraza: na primer, ako napravite bafer, tip geometrije mora biti poligon. Jedno od: • 0 — Poligon • 1 — Linija • 2 — Tačka
Izlazna geometrija ima z vrednosti	`WITH_Z`	[logička vrednost] Podrazumevano: False	Izaberite da li izlazna geometrija treba da uključuje Z dimenziju
Izlazna geometrija ima m vrednosti	`WITH_M`	[logička vrednost] Podrazumevano: False	Izaberite da li izlazna geometrija treba da uključuje M dimenziju
Izraz geometrije	`EXPRESSION`	[izraz] Podrazumevano: '$geometry'	Dodajte izraz geometrije koji želite da koristite. Možete koristiti dugme za otvaranje dijaloga izraza. Dijalog navodi sve relevantne izraze, zajedno sa njihovom pomoći i vodičem.
Izmenjena geometrija	`OUTPUT`	[vektor: geometrija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izmenjena geometrija	`OUTPUT`	[vektor: geometrija]	Izlazni vektorski sloj

Python kod

ID algoritma: `native:geometrybyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

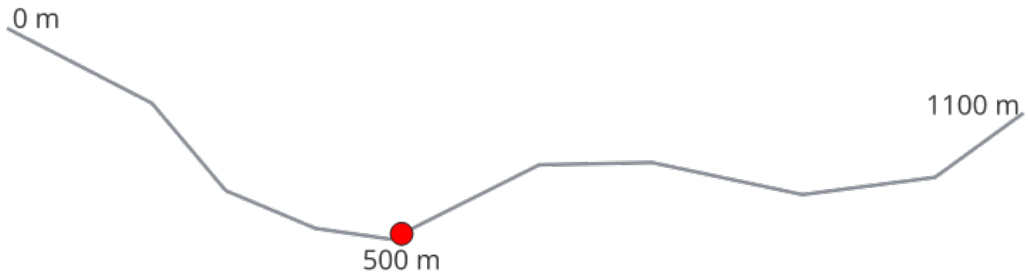
Interpoliraj tačku na liniji

Pravi geometriju tačke interpoliranu na zadatom rastojanju duž geometrija linije ili krive.

Z i M vrednosti se linearno interpoliraju iz postojećih vrednosti.

Ako se naiđe na višedelnu geometriju, samo se prvi deo uzima u obzir pri računanju podniske.

Ako je zadato rastojanje veće od dužine ulaznog objekta, rezultujući objekat imaće nultu geometriju.



Слика24.130: Interpolirana tačka na 500 m od početka linije

See also

Tačke duž geometrije

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Rastojanje	`DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje od početka linije
Interpolirane tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirane tačke	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj sa objektima na zadatom rastojanju duž linije ili granice poligona

Python kod

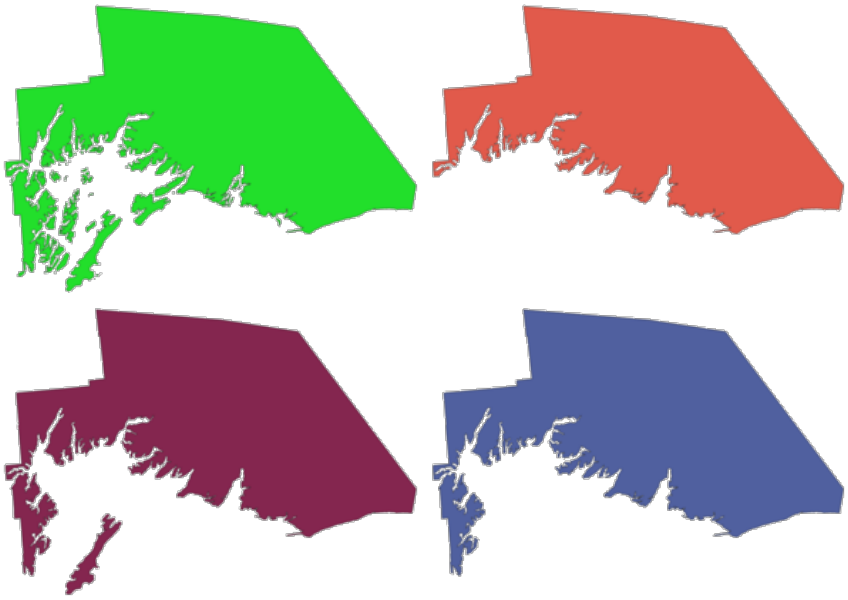
ID algoritma: `native:interpolatepoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zadrži N najvećih delova

Uzima sloj sa poligonima ili multipoligonima i vraća novi sloj u kojem se zadržava samo *n* najvećih poligona svakog multipoligonskog objekta. Ako objekat ima *n* ili manje delova, objekat će jednostavno biti kopiran.



Слика24.131: U smeru kazaljke na satu od gornjeg levog: originalni višedelni objekat, jedan, dva i tri najveća dela zadržana

Parametri

Oznaka	Назив	Tip	Opis
Poligoni	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Delovi za zadržavanje	`PARTS`	[numerički: ceo broj] Podrazumevano: 1	Broj delova za zadržavanje. Ako je 1, zadržaće se samo najveći deo objekta.
Delovi	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Delovi	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj sa N najvećih delova svakog objekta

Python kod

ID algoritma: `native:keepnbiggestparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Podniska linije

Vraća deo linije (ili krive) koji pada između zadatih početnog i krajnjeg rastojanja (mereno od početka linije).

Z i M vrednosti se linearno interpoliraju iz postojećih vrednosti.

Ako se naiđe na višedelnu geometriju, samo se prvi deo uzima u obzir pri računanju podniske.



Слика24.132: Podniska linije sa početnim rastojanjem postavljenim na 0 metara i krajnjim rastojanjem na 250 metara.

Omogućava *izmenu objekata na licu mesta* za linijske objekte

See also

Produži linije

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Početno rastojanje	`START_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje duž ulazne linije do početne tačke izlaznog objekta

наставак на следећој страни

Табела 24.676 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Krajnje rastojanje	`END_DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje duž ulazne linije do krajnje tačke izlaznog objekta
Podniska	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podniska	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj.

Python kod

ID algoritma: `native:linesubstring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Linije u poligone

Generiše poligonski sloj koristeći kao prstenove poligona linije iz ulaznog linijskog sloja.

Atributska tabela izlaznog sloja ista je kao ona ulaznog sloja.

Podrazumevani meni: *Vektor ► Alatke za geometriju*

See also

Poligoni u linije, *Poligonizuj*, *Pretvori tip geometrije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj

наставак на следећој страни

Табела 24.678 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Poligoni	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Poligoni	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj.

Python kod

ID algoritma: `qgis:linestopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spoji linije

Spaja sve povezane delove MultiLineString geometrija u pojedinačne LineString geometrije.

Ako bilo koji delovi ulaznih MultiLineString geometrija nisu povezani, rezultujuća geometrija biće MultiLineString koji sadrži sve linije koje su mogle biti spojene i sve nepovezane delove linija.

Omogućava *izmenu objekata na licu mesta* za linijske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Spojeno	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Spojeno	`OUTPUT`	[vektor: linija]	Izlazni (spojeni) linijski vektorski sloj.

Python kod

ID algoritma: `native:mergelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

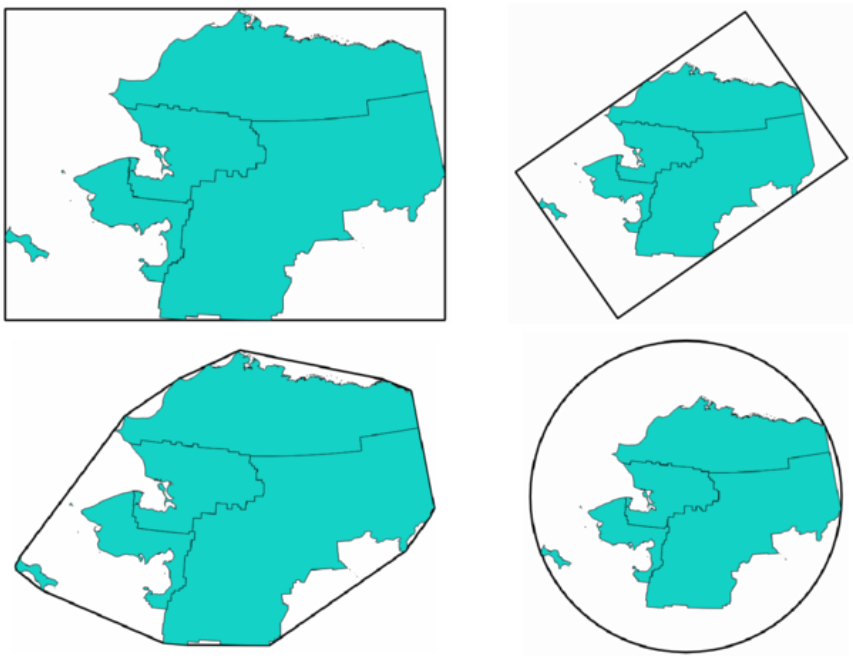
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Minimalna granična geometrija

Pravi geometrije koje obuhvataju objekte iz ulaznog sloja. Objekti se mogu grupisati po polju. Izlazni sloj će tada sadržati po jedan objekat po vrednosti grupe sa geometrijom (MBB) koja pokriva geometrije objekata sa odgovarajućom vrednošću.

Podržani su sledeći tipovi obuhvatajuće geometrije:

- granični pravougaonik (omotnica)
- orijentisani pravougaonik
- krug
- konveksni omotač



Слика24.133: U smeru kazaljke na satu od gornjeg levog: omotnica, orijentisani pravougaonik, krug, konveksni omotač

See also

Minimalni obuhvatajući krugovi

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje Opciono	`FIELD`	[polje tabele: bilo koji]	Objekti se mogu grupisati po polju. Ako je postavljeno, ovo uzrokuje da izlazni sloj sadrži po jedan objekat po grupisanoj vrednosti sa minimalnom geometrijom koja pokriva samo objekte sa odgovarajućim vrednostima.
Tip geometrije	`TYPE`	[nabrajanje] Podrazumevano: 0	Tipovi obuhvatajuće geometrije. Jedno od: • 0 — Omotnica (granični pravougaonik) • 1 — Minimalni orijentisani pravougaonik • 2 — Minimalni obuhvatajući krug • 3 — Konveksni omotač
Granična geometrija	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Granična geometrija	`OUTPUT`	[vektor: poligon]	Izlazni (granični) poligonski vektorski sloj. Ima polja iz ulaznog sloja i, u zavisnosti od izabranog parametra `TYPE`, izlaz takođe ima atribute: • `width`, `height`, `area` i `perimeter` omotnice (graničnog pravougaonika) • `width`, `height`, `area`, `angle` i `perimeter` minimalnog orijentisanog pravougaonika • `radius` i `area` minimalnog obuhvatajućeg kruga • `area` i `perimeter` konveksnog omotača

Python kod

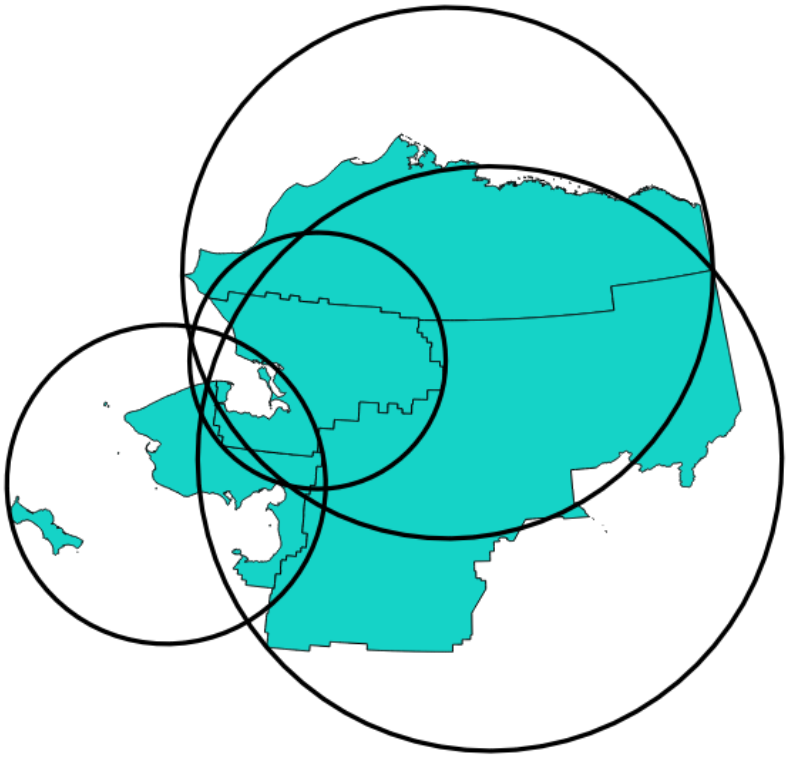
ID algoritma: `qgis:minimumboundinggeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Minimalni obuhvatajući krugovi

Računa minimalne obuhvatajuće krugove objekata u ulaznom sloju.



Слика24.134: Obuhvatajući krugovi za svaki objekat

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

See also

Minimalna granična geometrija

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometri-ja]	Ulazni vektorski sloj

наставак на следећој страни

Табела 24.684 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj segmenata u krugovima	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 72	Broj segmenata koji se koristi za aproksimaciju kruga. Minimum 8, maksimum 100000.
Minimalni obuhvatajući krugovi	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Minimalni obuhvatajući krugovi	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: `area` i `radius` generisanog poligona.

Python kod

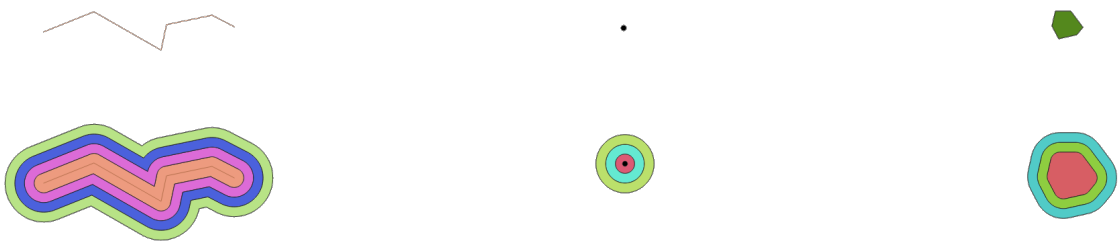
ID algoritma: `native:minimumenclosingcircle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Višeprstenasti bafer (konstantno rastojanje)

Računa višeprstenasti (*krofna*) bafer za objekte ulaznog sloja, koristeći fiksno ili dinamičko rastojanje i broj prstenova.



Слика24.135: Višeprstenasti bafer za linijski, tačkasti i poligonski sloj

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Bafer, *Bafer promenljivog rastojanja*, *Pravougaonici, ovali, rombovi*, *Jednostrani bafer*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Broj prstenova	`RINGS`	[numerički: ceo broj] Podrazumevano: 1	Broj prstenova. Može biti jedinstvena vrednost (isti broj prstenova za sve objekte) ili se može uzeti iz podataka objekata (broj prstenova zavisi od vrednosti objekata).
Rastojanje između prstenova	`DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje između prstenova. Može biti jedinstvena vrednost (isto rastojanje za sve objekte) ili se može uzeti iz podataka objekata (rastojanje zavisi od vrednosti objekata).
Višeprstenasti bafer (konstantno rastojanje)	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Višeprstenasti bafer (konstantno rastojanje)	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: `ringId` i `distance` generisanog bafer poligona.

Python kod

ID algoritma: `native:multiringconstantbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Višedelno u jednodelno

Deli višedelne objekte u ulaznom sloju na jednodelne objekte.

Atributi izlaznog sloja su isti kao originalni, ali podeljeni na pojedinačne objekte.



Слика24.136: Levo višedelni izvorni sloj, a desno jednodelni izlazni rezultat

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Prikupi geometrije, *Unapredi u višedelno*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Jednodelni	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Jednodelni	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj.

Python kod

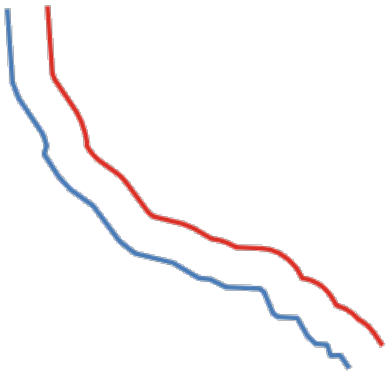
ID algoritma: `native:multiparttosingleparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pomeri linije

Odmiče linije za zadato rastojanje. Pozitivna rastojanja odmiču linije ulevo, a negativna udesno.



Слика24.137: Plavo je izvorni sloj, crveno odmaknuti

Omogućava *izmenu objekata na licu mesta* za linijske objekte

See also

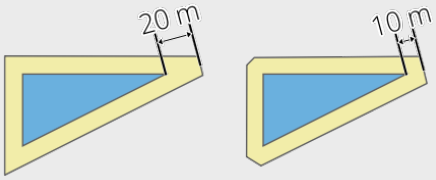
Niz odmaknutih (paralelnih) linija, *Transliraj*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Rastojanje	`DISTANCE`	[numerički: decimalni] Podrazumevano: 10.0	Rastojanje pomeraja. Možete koristiti dugme Zamena definisana podacima sa desne strane da biste izabrali polje iz kojeg će se izračunati poluprečnik. Na ovaj način možete imati različit poluprečnik za svaki objekat (pogledajte *Bafer promenljivog rastojanja*).
Segmenti	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 8	Kontroliše broj segmenata linije koji se koriste za aproksimaciju četvrtine kruga pri pravljenju zaobljenih pomeraja.

наставак на следећој страни

Табела 24.690 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Stil spoja	`JOIN_STYLE`	[nabrajanje] Podrazumevano: 0	Zadaje da li se koriste zaobljeni, oštri ili zasečeni spojevi pri pomeranju uglova u liniji. Opcije su: • 0 — Zaobljeno • 1 — Oštro • 2 — Zakošeno  Слика24.138: Zaobljeni, oštri i zakošeni stilovi spoja
Granica oštrog spoja	`MITER_LIMIT`	[numerički: decimalni] Podrazumevano: 2.0	Postavlja najveće rastojanje od odmaknute geometrije koje se koristi pri kreiranju oštrog spoja, kao faktor rastojanja odmicanja (primenjuje se samo na oštre stilove spoja). Minimum: 1.0  Слика24.139: Bafer od 10 m sa granicom 2 i bafer od 10 m sa granicom 1
Pomeraj	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (pomereni) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pomeraj	`OUTPUT`	[vektor: linija]	Izlazni (pomereni) linijski sloj

Python kod

ID algoritma: `native:offsetline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

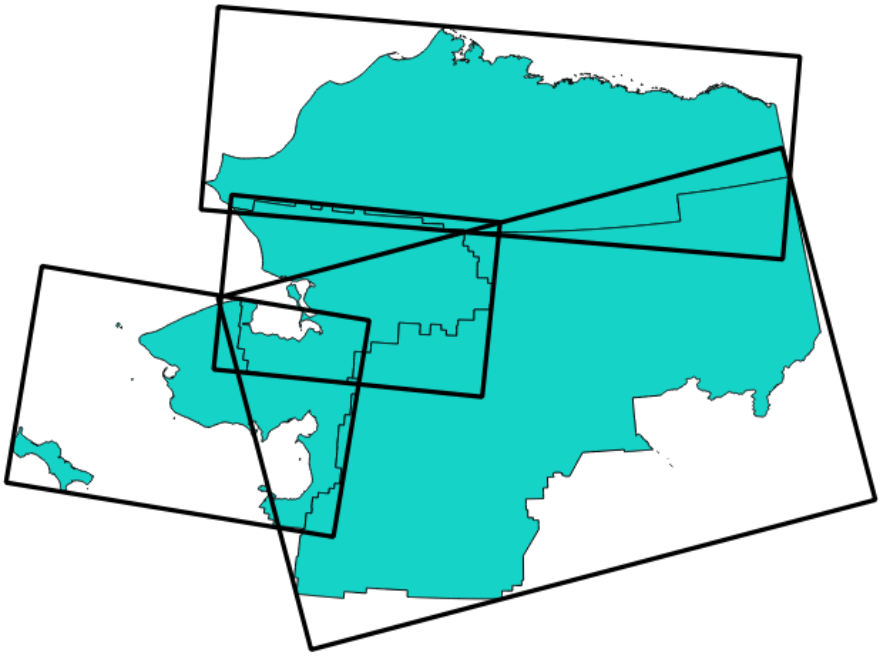
ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Orijentisani minimalni granični pravougaonik

Računa rotirani pravougaonik minimalne površine za svaki objekat u ulaznom sloju.

Белешка

Za jednodelne tačkaste objekte, izlaz odgovara graničnom pravougaoniku geometrije.



Слика24.140: Orijentisani minimalni granični pravougaonik

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

наставак на следећој страни

Табела 24.692 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

See also

Minimalna granična geometrija

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Granični pravougaonici	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Granični pravougaonici	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj. Pored ulaznih atributa, izlazni sloj takođe sadrži sledeća polja: `width`, `height`, `angle`, `area` i `perimeter` generisanog poligona.

Python kod

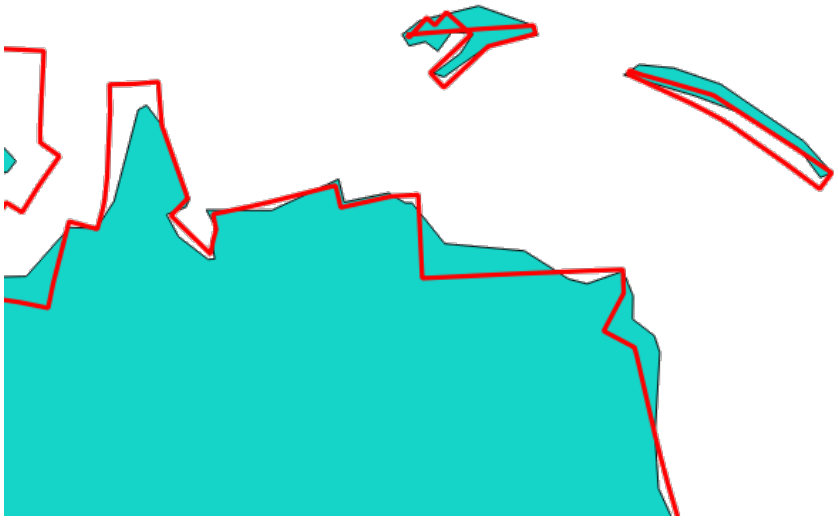
ID algoritma: `native:orientedminimumboundingbox`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Ortogonalizuj

Pokušava da ortogonalizuje geometrije ulaznog linijskog ili poligonskog sloja. Ovaj proces pomera temena u geometrijama da bi pokušao da svaki ugao u geometriji učini ili pravim uglom ili pravom linijom.



Слика24.141: Plavom bojom izvorni sloj, a crvenom ortogonalizovani rezultat

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Tolerancija maksimalnog ugla (stepeni)	`ANGLE_TOLERANCE`	[numerički: decimalni] Podrazumevano: 15.0	Zadajte maksimalno odstupanje od pravog ugla ili prave linije koje teme može imati da bi bilo prilagođeno. Manje tolerancije znače da će biti prilagođena samo temena koja su već bliža pravim uglovima, a veće tolerancije znače da će biti prilagođena i temena koja više odstupaju od pravih uglova.
Maksimalan broj iteracija algoritma	`MAX_ITERATIONS`	[numerički: ceo broj] Podrazumevano: 1000	Postavljanje većeg broja za maksimalan broj iteracija daće ortogonalniju geometriju uz cenu dodatnog vremena obrade.
Ortogonalizovano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Ortogonalizovano	`OUTPUT`	[isto kao ulaz]	Izlazni poligonski vektorski sloj sa prilagođenim uglovima.

Python kod

ID algoritma: `native:orthogonalize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Tačka na površini

Za svaki objekat ulaznog sloja, vraća tačku za koju je zagarantovano da leži na površini geometrije objekta.

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Упозорење

Ovaj algoritam može odbaciti postojeće primarne ključeve ili FID vrednosti i ponovo ih generisati u izlaznim slojevima, u zavisnosti od ulaznih parametara.

See also

Centroidi

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Napravi tačku na površini za svaki deo	`ANGLE_TOLERANCE`	[logička vrednost]	Ako je označeno, tačka će biti napravljena za svaki deo geometrije.
Tačka	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni tačkasti vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačka	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj.

Python kod

ID algoritma: `native:pointonsurface`

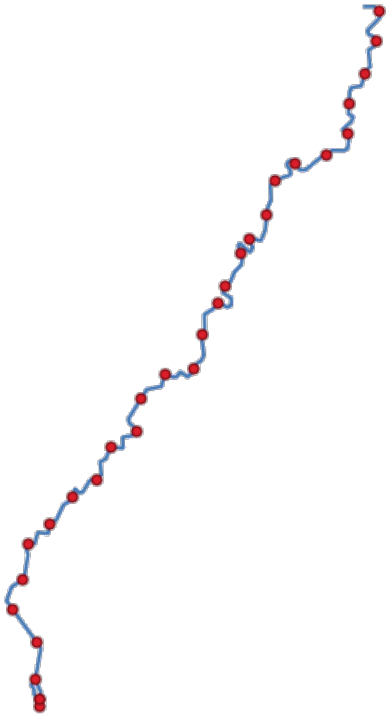
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Tačke duž geometrije

Pravi tačke u pravilnim intervalima duž geometrija linije ili poligona. Napravljene tačke imaće dodata nova polja za rastojanje duž geometrije i ugao linije na tački.

Može se zadati opcioni početni i krajnji pomeraj, koji kontroliše koliko daleko od početka i kraja geometrije tačke treba da budu napravljene.



Слика24.142: Tačke napravljene duž izvornog linijskog sloja

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Interpoliraj tačku na liniji

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Rastojanje	`DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje između dve uzastopne tačke duž linije
Početni pomeraj	`START_OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje od početka ulazne linije, koje predstavlja položaj prve tačke.
Krajnji pomeraj	`END_OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	Rastojanje od kraja ulazne linije, koje predstavlja položaj iza kojeg ne treba praviti nijedan tačkasti objekat.
Interpolirane tačke	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirane tačke	`OUTPUT`	[vektor: tačka]	Tačkasti vektorski sloj sa objektima postavljenim duž linija ili granica poligona ulaznog sloja. Pored ulaznih atributa na svakoj povezanoj tački, izlazni sloj takođe sadrži sledeća polja: `distance` i `angle` generisane tačke.

Python kod

ID algoritma: `native:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pomeranje tačaka

Za zadato rastojanje blizine, prepoznaje bliske tačkaste objekte i radijalno ih raspoređuje po krugu čiji centar predstavlja njihov baricentar. Zgodna alatka za rasipanje preklopljenih objekata.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Minimalno rastojanje do drugih tačaka	`PROXIMITY`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje ispod kojeg se tačkasti objekti smatraju bliskim. Bliski objekti se raspoređuju svi zajedno.
Rastojanje pomeranja	`DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Poluprečnik kruga na koji se postavljaju bliski objekti
Horizontalna raspodela za slučaj dve tačke	`HORIZONTAL`	[logička vrednost] Podrazumevano: False	Kada su samo dve tačke prepoznate kao bliske, poravnava ih horizontalno na krugu umesto vertikalno.
Pomereno	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pomereno	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj

Python kod

ID algoritma: `qgis:pointsdisplacement`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pol nepristupačnosti

Računa pol nepristupačnosti za poligonski sloj, što je najudaljenija unutrašnja tačka od granice površine.

Ovaj algoritam koristi algoritam „polylabel“ (Vladimir Agafonkin, 2016), koji je iterativni pristup zagarantovan da pronađe pravi pol nepristupačnosti unutar zadate tolerancije. Preciznija tolerancija (niža vrednost) zahteva više iteracija i trajaće duže da se izračuna.



Слика24.143: Pol nepristupačnosti

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni vektorski sloj
Tolerancija	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 1.0	Postavite toleranciju za izračunavanje
Tačka	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Tačka	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti vektorski sloj. Sadržaće novi atribut sa rastojanjem `dist_pole` od izračunatog pola do granice poligona.

Python kod

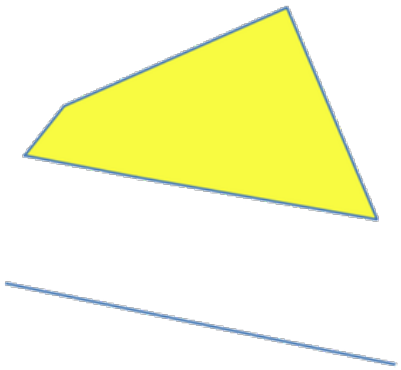
ID algoritma: `native:poleofinaccessibility`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poligonizuj

Pravi poligonski sloj čije se granice objekata generišu iz linijskog sloja **zatvorenih** objekata.



Слика24.144: Žuti poligoni generisani iz zatvorenih linija

Белешка

Linijski sloj mora imati zatvorene oblike da bi bio transformisan u poligon.

See also

Poligoni u linije, *Linije u poligone*, *Pretvori tip geometrije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Zadrži polja iz ulaznog sloja	`KEEP_FIELDS`	[logička vrednost] Podrazumevano: False	Označite da biste zadržali polja (samo strukturu tabele, ne vrednosti) ulaznog sloja

наставак на следећој страни

Табела 24.704 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Poligoni iz linija	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni poligonski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Poligoni iz linija	`OUTPUT`	[vektor: poligon]	Izlazni poligonski vektorski sloj iz linija

Python kod

ID algoritma: `native:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poligoni u linije

Uzima poligonski sloj i pravi linijski sloj, sa linijama koje predstavljaju granice poligona u ulaznom sloju.

Atributska tabela izlaznog sloja ista je kao ona ulaznog sloja.



Слика24.145: Crne linije kao rezultat algoritma

Podrazumevani meni: *Vektor ► Alatke za geometriju*

See also

Linije u poligone, *Poligonizuj*, *Pretvori tip geometrije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Linije	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Linije	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj iz poligona

Python kod

ID algoritma: `native:polygonstolines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Projektuj tačke (kartezijanski)

Projektuje geometrije tačaka za zadato rastojanje i smer (azimut).

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Smer (stepeni od severa)	`BEARING`	[numerički: decimalni] Podrazumevano: 0.0	Ugao u smeru kazaljke na satu počev od severa, u jedinici stepeni (°)
Rastojanje	`DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Rastojanje za pomeraj geometrija, u jedinicama sloja

наставак на следећој страни

Табела 24.708 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Projektovano	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni tačkasti vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Projektovano	`OUTPUT`	[vektor: tačka]	Izlazni (projektovani) tačkasti vektorski sloj

Python kod

ID algoritma: `native:projectpointcartesian`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Unapredi u višedelno

Uzima vektorski sloj sa jednodelnim geometrijama i generiše novi u kojem su sve geometrije višedelne.

Ulazni objekti koji su već višedelni objekti ostaće nepromenjeni.

Ovaj algoritam se može koristiti za prisiljavanje geometrija na višedelne tipove radi kompatibilnosti sa provajderima podataka koji zahtevaju višedelne objekte.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Agregiraj, *Prikupi geometrije*, *Convert to strictly multipart*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj

наставак на следећој страни

Табела 24.710 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Višedelni	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni višedelni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Višedelni	`OUTPUT`	[isto kao ulaz]	Izlazni višedelni vektorski sloj

Python kod

ID algoritma: `native:promotetomulti`

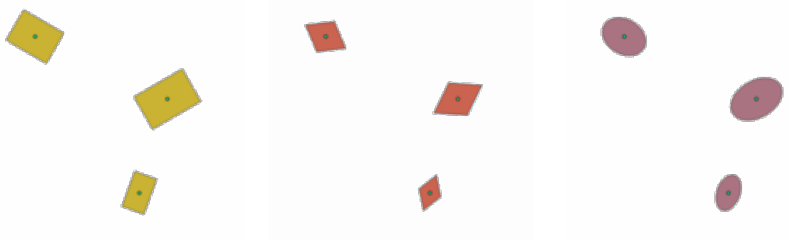
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pravougaonici, ovali, rombovi

Pravi oblast bafera oblika pravougaonika, ovala ili romba za svaki objekat ulaznog tačkastog sloja.

Parametri oblika mogu biti fiksni za sve objekte ili dinamički pomoću polja ili izraza.



Слика24.146: Različiti oblici bafera sa dinamičkim parametrima

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj

наставак на следећој страни

Табела 24.712 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Oblik bafera	`SHAPE`	[nabrajanje]	Oblik koji se koristi. Jedno od: • 0 — Pravougaonici • 1 — Ovali • 2 — Rombovi
Širina	`WIDTH`	[numerički: decimalni] Podrazumevano: 1.0	Širina oblika bafera
Visina	`HEIGHT`	[numerički: decimalni] Podrazumevano: 1.0	Visina oblika bafera
Rotacija Opciono	`ROTATION`	[numerički: decimalni] Podrazumevano: 0.0	Rotacija oblika bafera
Broj segmenata	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 36	Broj segmenata za pun krug (oblik *Ovali*)
Izlaz	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj (sa oblicima bafera)

Python kod

ID algoritma: `native:rectanglesovalsdiamonds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Ukloni duplirana temena

Uklanja duplirana temena iz objekata, svuda gde uklanjanje temena ne rezultira degenerisanom geometrijom.

Parametar tolerancije zadaje toleranciju za koordinate pri određivanju da li su temena identična.

Podrazumevano se Z vrednosti ne uzimaju u obzir pri otkrivanju dupliranih temena. Npr. dva temena sa istom X i Y koordinatom, ali različitim Z vrednostima i dalje će se smatrati dupliranim i jedno će biti uklonjeno. Ako je

parametar *Koristi Z vrednost* tačan, tada se testiraju i Z vrednosti, a temena sa istim X i Y, ali različitim Z biće zadržana.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Белешка

Duplirana temena se ne testiraju između različitih delova višedelne geometrije, npr. višetačkasta geometrija sa preklapajućim tačkama neće biti promenjena ovim metodom.

See also

Izdvoj temena, *Izdvoj određena temena*, *Brisanje duplih geometrija*, *Delete duplicated vertices*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Tolerancija	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.000001	Temena bliža od zadatog rastojanja smatraju se duplikatima
Koristi Z vrednost	`USE_Z_VALUE`	[logička vrednost] Podrazumevano: False	Ako je parametar *Koristi Z vrednost* tačan, tada se testiraju i Z vrednosti, a temena sa istim X i Y, ali različitim Z biće zadržana.
Očišćeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Očišćeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (bez dupliranih temena)

Python kod

ID algoritma: `native:removeduplicatevertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Ukloni nulte geometrije

Uklanja sve objekte koji nemaju geometriju iz vektorskog sloja. Svi ostali objekti biće kopirani nepromenjeni.

Objekti sa nultim geometrijama mogu se sačuvati u zaseban sloj.

Ako je označeno *Ukloni i prazne geometrije*, algoritam uklanja objekte čije geometrije nemaju koordinate, tj. geometrije koje su prazne. U tom slučaju, i nulti izlaz će odražavati ovu opciju, sadržeći i nulte i prazne geometrije.

See also

Brisanje duplih geometrija

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj (sa ne-NULL geometrijama)
Ukloni i prazne geometrije	`REMOVE_EMPTY`	[logička vrednost]	
Ne-nulte geometrije	`OUTPUT` Opciono	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za ne-NULL (i neprazne) geometrije. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Nulte geometrije Opciono	`NULL_OUTPUT`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte izlazni vektorski sloj za NULL (i prazne) geometrije. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Nulte geometrije	`NULL_OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (za NULL i, ako je izabrano, prazne geometrije)
Ne-nulte geometrije	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (bez NULL i, ako je izabrano, praznih geometrija)

Python kod

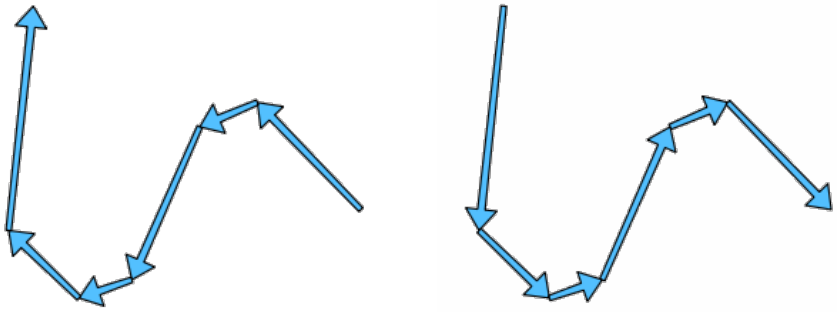
ID algoritma: `native:removenullgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Obrni smer linije

Invertuje smer linijskog sloja.



Слика24.147: Pre i posle inverzije smera

Omogućava *izmenu objekata na licu mesta* za linijske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Obrnuto	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Obrnuto	`OUTPUT`	[vektor: linija]	Izlazni linijski vektorski sloj (sa obrnutim linijama)

Python kod

ID algoritma: `native:reverselinedirection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rotiraj

Rotira geometrije objekata za zadati ugao u smeru kazaljke na satu. Rotacija se dešava oko centroida svakog objekta, ili opciono oko jedinstvene unapred zadate tačke.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Transliraj, *Zameni X i Y koordinate*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rotacija (stepeni u smeru kazaljke na satu)	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije u stepenima
Tačka sidra rotacije (x, y) Opciono	`ANCHOR`	[koordinate] Podrazumevano: Nije postavljeno	X,Y koordinate tačke oko koje se rotiraju objekti. Ako nije postavljeno, rotacija se dešava oko centroida svakog objekta.
Rotirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa rotiranim geometrijama). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Rotirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa rotiranim geometrijama

Python kod

ID algoritma: `native:rotatefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zaobljenost

Računa zaobljenost svakog objekta i čuva je kao novo polje `roundness`. Ulazni vektorski sloj mora sadržati poligone.

Zaobljenost poligona definiše se kao $4\pi \times$ površina poligona / obim^2. Vrednost zaobljenosti varira između 0 i 1. Savršen krug ima zaobljenost 1, dok potpuno ravan poligon ima zaobljenost 0.

Белешка

Algoritam vraća NULL za višedelne poligonske objekte.

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni vektorski sloj
Zaobljenost	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa poljem zaobljenosti). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zaobljenost	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa `roundness` vrednošću u polju

Python kod

ID algoritma: `native:roundness`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Segmentizuj po maksimalnom uglu

Segmentizuje geometriju pretvaranjem zakrivljenih delova u linearne delove.

Segmentizacija se vrši zadavanjem maksimalnog dozvoljenog ugla poluprečnika između temena na ispravljenoj geometriji (npr. ugla luka napravljenog od centra originalnog luka do uzastopnih izlaznih temena na linearizovanoj geometriji). Nezakrivljene geometrije biće zadržane bez izmene.

See also

Segmentizuj po maksimalnom rastojanju, *Uprosti*, *Zagladi*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Maksimalni ugao između temena (stepeni)	`ANGLE`	[numerički: decimalni] Podrazumevano: 5.0	Maksimalni dozvoljeni ugao poluprečnika između temena na ispravljenoj geometriji
Segmentizovano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa segmentizovanim geometrijama). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Segmentizovano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa segmentizovanim geometrijama

Python kod

ID algoritma: `native:segmentizebymaxangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Segmentizuj po maksimalnom rastojanju

Segmentizuje geometriju pretvaranjem zakrivljenih delova u linearne delove.

Segmentizacija se vrši zadavanjem maksimalnog dozvoljenog rastojanja pomeraja između originalne krive i segmentizovane reprezentacije. Nezakrivljene geometrije biće zadržane bez izmene.

See also

Segmentizuj po maksimalnom uglu, *Uprosti*, *Zagladi*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Maksimalno rastojanje pomeraja	`DISTANCE`	[numerički: decimalni] Podrazumevano: 1.0	Maksimalno dozvoljeno rastojanje pomeraja između originalne krive i segmentizovane reprezentacije, u jedinicama sloja.
Segmentizovano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa segmentizovanim geometrijama). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Segmentizovano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa segmentizovanim geometrijama

Python kod

ID algoritma: `native:segmentizebymaxdistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi M vrednost

Postavlja M vrednost za geometrije u sloju.

Ako M vrednosti već postoje u sloju, biće prepisane novom vrednošću. Ako M vrednosti ne postoje, geometrija će biti unapređena da uključi M vrednosti, a zadata vrednost se koristi kao početna M vrednost za sve geometrije.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim M

Савет

Koristite dugme 

da biste proverili dodatu M vrednost: rezultati su dostupni u dijalogu *Rezultati identifikacije*.

See also

Postavi M vrednost iz rastera, *Postavi Z vrednost*, *Odbaci M/Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
M vrednost	`M_VALUE`	[numerički: decimalni] Podrazumevano: 0.0	M vrednost koja se dodeljuje geometrijama objekata
M dodato	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
M dodato	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (sa M vrednostima dodeljenim geometrijama)

Python kod

ID algoritma: `native:setmvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi M vrednost iz rastera

Koristi vrednosti uzorkovane iz kanala unutar rasterskog sloja za postavljanje M vrednosti za svako preklapajuće teme u geometriji objekta. Vrednosti rastera se opciono mogu skalirati unapred zadatim iznosom.

Ako M vrednosti već postoje u sloju, biće prepisane novom vrednošću. Ako M vrednosti ne postoje, geometrija će biti unapređena da uključi M vrednosti.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim M

See also

Prekri (postavi Z vrednost iz rastera), *Postavi M vrednost*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rasterski sloj	`RASTER`	[raster]	Rasterski sloj sa M vrednostima
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Rasterski kanal iz kojeg se uzimaju M vrednosti
Vrednost za NoData ili teme koja se ne seku	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost koja se koristi u slučaju da teme ne preseca (validan piksel) rastera
Faktor skaliranja	`SCALE`	[numerički: decimalni] Podrazumevano: 1.0	Vrednost skaliranja: vrednosti kanala se množe ovom vrednošću.
Pomeraj	`OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost pomeraja: algebarski se dodaje vrednostima kanala nakon primene „Faktora skaliranja“.
Ažurirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj (sa ažuriranim M vrednostima). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Ažurirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (sa ažuriranim M vrednostima)

Python kod

ID algoritma: `native:setmfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Postavi Z vrednost

Postavlja Z vrednost za geometrije u sloju.

Ako Z vrednosti već postoje u sloju, biće prepisane novom vrednošću. Ako Z vrednosti ne postoje, geometrija će biti unapređena da uključi Z vrednosti, a zadata vrednost se koristi kao početna Z vrednost za sve geometrije.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte sa omogućenim Z

Савет

Koristite dugme Identifikuj objekte da biste proverili dodatu Z vrednost: rezultati su dostupni u dijalogu *Rezultati identifikacije*.

See also

Prekri (postavi Z vrednost iz rastera), *Postavi M vrednost*, *Odbaci M/Z vrednosti*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Z vrednost	`Z_VALUE`	[numerički: decimalni] Podrazumevano: 0.0	Z vrednost koja se dodeljuje geometrijama objekata
Z dodato	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Z dodato	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj (sa dodeljenim Z vrednostima)

Python kod

ID algoritma: `native:setzvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Uprosti

Uprošćava geometrije u linijskom ili poligonskom sloju. Pravi novi sloj sa istim objektima kao oni u ulaznom sloju, ali sa geometrijama koje sadrže manji broj temena.

Algoritam nudi izbor metoda uprošćavanja, uključujući zasnovan na rastojanju (algoritam „Douglas-Peucker“), zasnovan na površini (algoritam „Visvalingam“) i lepljenje geometrija za mrežu.



Слика24.148: Sleva nadesno, izvorni sloj i rastuće tolerancije uprošćavanja

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

See also

Pojednostavljivanje pokrivenosti, *Zagladi*, *Zgusni po broju*, *Zgusni po intervalu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Metod uprošćavanja	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod uprošćavanja. Jedno od: • 0 — Rastojanje (Douglas-Peucker) • 1 — Zalepi za mrežu • 2 — Površina (Visvalingam)
Tolerancija	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 1.0	Tolerancija praga (u jedinicama sloja): ako je rastojanje između dva čvora manje od vrednosti tolerancije, segment će biti uprošćen, a temena uklonjena.
Pojednostavljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (uprošćeni) vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Pojednostavljeno	`OUTPUT`	[isto kao ulaz]	Izlazni (uprošćeni) vektorski sloj

Python kod

ID algoritma: `native:simplifygeometries`

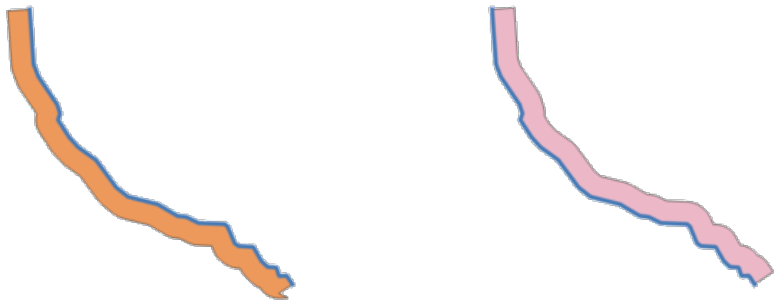
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Jednostrani bafer

Računa bafer na linijama za zadato rastojanje samo sa jedne strane linije.

Bafer uvek rezultira poligonskim slojem.



Слика24.149: Bafer sa leve i sa desne strane na istom vektorskom linijskom sloju

See also

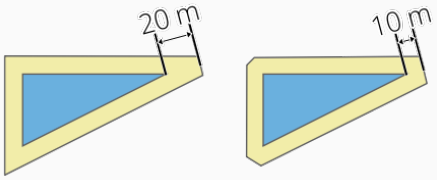
Bafer

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	INPUT	[vektor: linija]	Ulazni linijski vektorski sloj
Rastojanje	DISTANCE	[numerički: decimalni] Podrazumevano: 10.0	Rastojanje bafera.
Strana	SIDE	[nabrajanje] Podrazumevano: 0	Na kojoj strani se pravi bafer. Jedno od: • 0 – Levo • 1 – Desno
Segmenti	SEGMENTS	[numerički: ceo broj] Podrazumevano: 8	Kontroliše broj segmenata linije koji se koriste za aproksimaciju četvrtine kruga pri pravljenju zaobljenih pomeraja.
Stil spoja	JOIN_STYLE	[nabrajanje] Podrazumevano: 0	Zadaje da li se koriste zaobljeni, oštri ili zasečeni spojevi pri pomeranju uglova u liniji. Opcije su: • 0 — Zaobljeno • 1 — Oštro • 2 — Zakošeno  Слика24.150: Zaobljeni, oštri i zakošeni stilovi spoja

наставак на следећој страни

Табела 24.736 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Granica oštrog spoja	`MITER_LIMIT`	[numerički: decimalni] Podrazumevano: 2.0	Postavlja najveće rastojanje od odmaknute geometrije koje se koristi pri kreiranju oštrog spoja, kao faktor rastojanja odmicanja (primenjuje se samo na oštre stilove spoja). Minimum: 1.0  Слика24.151: Bafer od 10 m sa granicom 2 i bafer od 10 m sa granicom 1
Bafer	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (bafer) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Bafer	`OUTPUT`	[vektor: poligon]	Izlazni (bafer) poligonski sloj

Python kod

ID algoritma: `native:singlesidedbuffer`

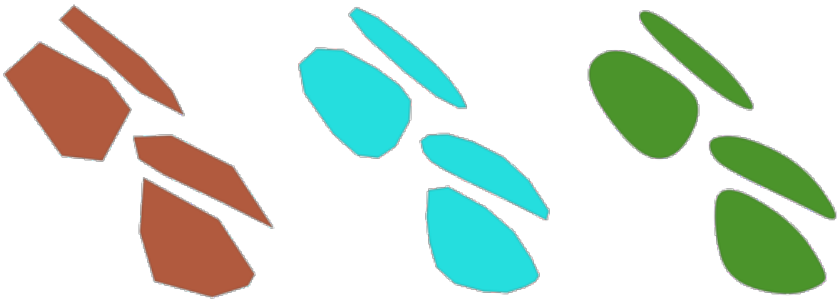
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zagladi

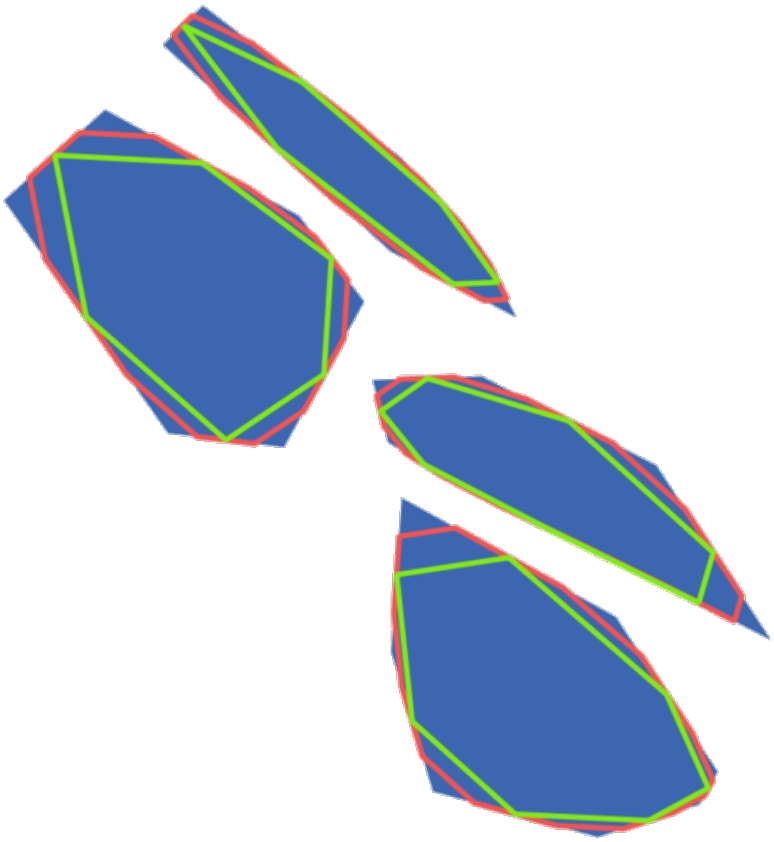
Zaglađuje geometrije u linijskom ili poligonskom sloju dodavanjem više **temena i uglova** geometrijama objekata.

Parametar iteracija određuje koliko će se iteracija zaglađivanja primeniti na svaku geometriju. Veći broj iteracija rezultira glađim geometrijama uz cenu većeg broja čvorova u geometrijama.



Слика24.152: Povećanje broja iteracija daje glatkije geometrije

Parametar pomeraja kontroliše koliko „tesno“ zaglađene geometrije prate originalne geometrije. Manje vrednosti rezultiraju tešnjim uklapanjem, a veće vrednosti će napraviti labavije uklapanje.



Слика24.153: Plavo: ulazni sloj. Pomeraj 0.25 daje crvenu liniju, dok pomeraj 0.50 daje zelenu liniju.

Parametar maksimalnog ugla može se koristiti za sprečavanje zaglađivanja čvorova sa velikim uglovima. Bilo koji čvor gde je ugao segmenata sa obe strane veći od ovoga neće biti zaglađen. Na primer, postavljanje maksimalnog ugla na 90 stepeni ili niže očuvalo bi prave uglove u geometriji.

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

See also

Uprosti, *Pojednostavljivanje pokrivenosti*, *Zgusni po broju*, *Zgusni po intervalu*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Ulazni linijski ili poligonski vektorski sloj
Iteracije	`ITERATIONS`	[numerički: ceo broj] Podrazumevano: 1	Povećanje broja iteracija daće glađe geometrije (i više temena).
Pomeraj	`OFFSET`	[numerički: decimalni] Podrazumevano: 0.25	Povećanje vrednosti će *pomeriti* zaglađene linije / granice dalje od ulaznih linija / granica.
Maksimalni ugao čvora za zaglađivanje	`MAX_ANGLE`	[numerički: decimalni] Podrazumevano: 180.0	Svaki čvor ispod ove vrednosti biće zaglađen
Zaglađeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (zaglađeni) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zaglađeno	`OUTPUT`	[isto kao ulaz]	Izlazni (zaglađeni) vektorski sloj

Python kod

ID algoritma: `native:smoothgeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zalepi geometrije za sloj

Lepi geometrije u sloju ili za geometrije iz drugog sloja, ili za geometrije unutar istog sloja.

Podudaranje se vrši na osnovu rastojanja tolerancije, a temena će biti umetnuta ili uklonjena po potrebi da bi se geometrije podudarile sa referentnim geometrijama.

 Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Zalepi tačke za mrežu

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Referentni sloj	`REFERENCE_LAYER`	[vektor: geometrija]	Vektorski sloj za koji se lepi
Tolerancija	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 10.0	Kontrolišite koliko blizu ulazna temena treba da budu geometrijama referentnog sloja pre nego što se zalepe.

наставак на следећој страни

Табела 24.740 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ponašanje	BEHAVIOR	[nabrajanje] Podrazumevano: 0	Lepljenje se može vršiti za postojeći čvor ili segment (njegovu najbližu tačku temenu koje se pomera). Dostupne opcije lepljenja: • 0 — Preferiraj poravnavanje čvorova, umetni dodatna temena gde je potrebno Preferiraj lepljenje za čvorove, čak i kada segment može biti bliži od čvora. Novi čvorovi biće umetnuti da bi se geometrije tačno pratile kada su unutar dozvoljene tolerancije. • 1 — Preferiraj najbližu tačku, umetni dodatna temena gde je potrebno Zalepi za najbližu tačku, bez obzira na to da li je čvor ili segment. Novi čvorovi biće umetnuti da bi se geometrije tačno pratile kada su unutar dozvoljene tolerancije. • 2 — Preferiraj poravnavanje čvorova, ne umeći nova temena Preferiraj lepljenje za čvorove, čak i kada segment može biti bliži od čvora. Neće biti umetnuti novi čvorovi. • 3 — Preferiraj najbližu tačku, ne umeći nova temena Zalepi za najbližu tačku, bez obzira na to da li je čvor ili segment. Neće biti umetnuti novi čvorovi. • 4 — Pomeri samo krajnje tačke, preferiraj poravnavanje čvorova Zalepi samo početne/krajnje tačke linija (tačkasti objekti će takođe biti zalepljeni, poligonski objekti neće biti izmenjeni), preferiraj lepljenje za čvorove. • 5 — Pomeri samo krajnje tačke, preferiraj najbližu tačku Zalepi samo početne/krajnje tačke linija (tačkasti objekti će takođe biti zalepljeni, poligonski objekti neće biti izmenjeni), zalepi za najbližu tačku • 6 — Zalepi krajnje tačke samo za krajnje tačke Zalepi samo početne/krajnje tačke linija za druge početne/krajnje tačke linija • 7 — Zalepi za sidra čvorove (samo jedan sloj)

наставак на следећој страни

Табела 24.740 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zalepljena geometrija	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (zalepljeni) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zalepljena geometrija	`OUTPUT`	[isto kao ulaz]	Izlazni (zalepljeni) vektorski sloj

Python kod

ID algoritma: `native:snapgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zalepi tačke za mrežu

Menja koordinate geometrija u vektorskom sloju, tako da su sve tačke ili temena zalepljeni za najbližu tačku mreže.

Ako se zalepljena geometrija ne može izračunati (ili je potpuno kolabirana), geometrija objekta biće očišćena.

Lepljenje se može vršiti na X, Y, Z ili M osi. Razmak mreže od 0 za bilo koju osu onemogućiće lepljenje za tu osu.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Упозорење

Lepljenje za mrežu može generisati nevažeću geometriju u nekim graničnim slučajevima.

See also

Zalepi geometrije za sloj

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj

наставак на следећој страни

Табела 24.742 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Razmak X mreže	`HSPACING`	[numerički: decimalni] Podrazumevano: 1.0	Razmak mreže na X osi
Razmak Y mreže	`VSPACING`	[numerički: decimalni] Podrazumevano: 1.0	Razmak mreže na Y osi
Razmak Z mreže	`ZSPACING`	[numerički: decimalni] Podrazumevano: 0.0	Razmak mreže na Z osi
Razmak M mreže	`MSPACING`	[numerički: decimalni] Podrazumevano: 0.0	Razmak mreže na M osi
Zalepljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (zalepljeni) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zalepljeno	`OUTPUT`	[isto kao ulaz]	Izlazni (zalepljeni) vektorski sloj

Python kod

ID algoritma: `native:snappointstogrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Podeli linije po maksimalnoj dužini

Uzima linijski (ili kriva) sloj i deli svaki objekat na više delova, gde je svaki deo zadate maksimalne dužine. Z i M vrednosti na početku i kraju novih podniski linija linearno se interpoliraju iz postojećih vrednosti.

Omogućava *izmenu objekata na licu mesta* za linijske objekte

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Maksimalna dužina linije	`LENGTH`	[numerički: decimalni] Podrazumevano: 10.0	Maksimalna dužina linije u izlazu.
Podeljeno	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podeljeno	`OUTPUT`	[vektor: linija]	Novi linijski vektorski sloj. Dužina geometrija objekata manja je od ili jednaka dužini zadatoj u LENGTH parametru. Biće dodato dodatno polje koje sadrži `order`.

Python kod

ID algoritma: `native:splitlinesbylength`

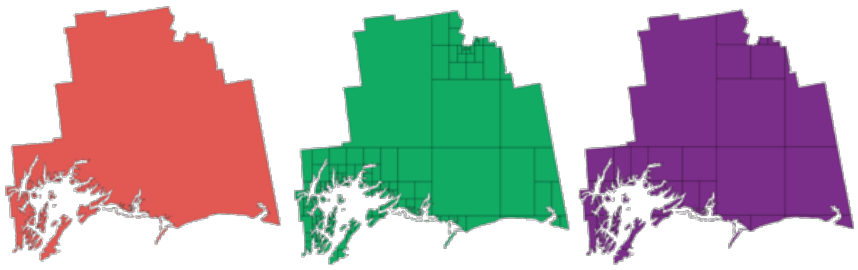
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Podpodeli

Podpodeljuje geometriju. Vraćena geometrija biće zbirka koja sadrži podpodeljene delove iz originalne geometrije, gde nijedan deo nema više od zadatog maksimalnog broja čvorova.

Ovo je korisno za deljenje složene geometrije na manje složene delove, lakše za prostorno indeksiranje i brže za izvršavanje prostornih operacija. Zakrivljene geometrije biće segmentizovane pre podpodele.



Слика24.154: Levo ulazni sloj, u sredini maksimalna vrednost čvorova je 100, a desno maksimalna vrednost 200

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Белешка

Podpodela geometrije može generisati delove geometrije koji možda nisu validni i mogu sadržati samopreseke.

See also

Rasturi linije, *Podniska linije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Maksimalno čvorova u delovima	`MAX_NODES`	[numerički: ceo broj] Podrazumevano: 256	Maksimalan broj temena koji svaki novi deo geometrije sme da ima. Manje *poddelova* za veće vrednosti.
Podpodeljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (podpodeljeni) vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podpodeljeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj

Python kod

ID algoritma: `native:subdivide`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zameni X i Y koordinate

Menja X i Y vrednosti koordinata u ulaznim geometrijama.

Može se koristiti za popravku geometrija koje su slučajno imale obrnute vrednosti geografske širine i dužine.

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Transliraj, *Rotiraj*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Zamenjeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zamenjeno	`OUTPUT`	[isto kao ulaz]	Izlazni (zamenjeni) vektorski sloj

Python kod

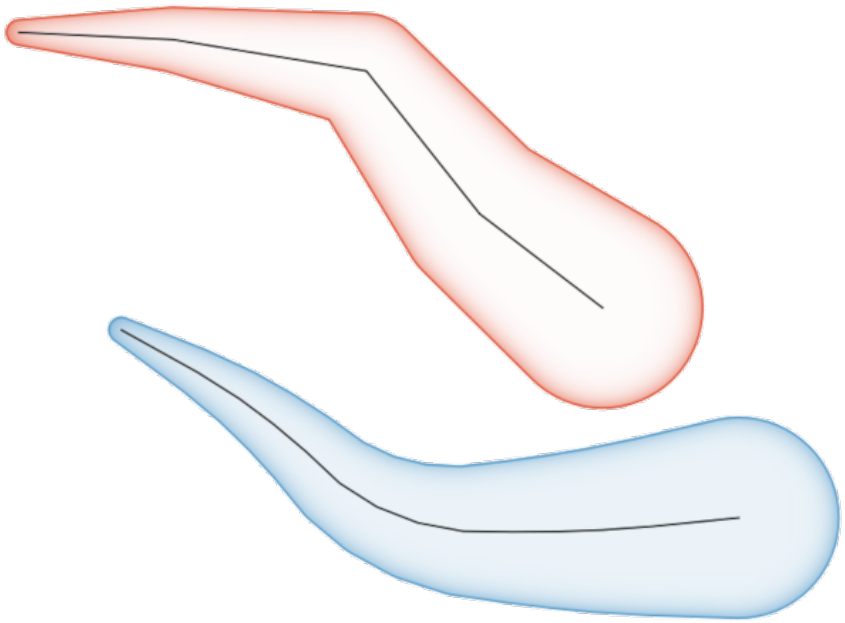
ID algoritma: `native:swapxy`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Suženi baferi

Pravi suženi bafer duž geometrija linija, koristeći zadati početni i krajnji prečnik bafera.



Слика24.155: Primer suženog bafera

See also

Bafer promenljive širine (po M vrednosti), *Bafer*, *Napravi klinaste bafere*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Početna širina	`START_WIDTH`	[numerički: decimalni] Podrazumevano: 0.0	Predstavlja poluprečnik bafera primenjenog na početnoj tački linijskog objekta
Krajnja širina	`END_WIDTH`	[numerički: decimalni] Podrazumevano: 0.0	Predstavlja poluprečnik bafera primenjenog na krajnjoj tački linijskog objekta.
Segmenti	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 16	Kontroliše broj segmenata linije koji se koriste za aproksimaciju četvrtine kruga pri pravljenju zaobljenih pomeraja.

наставак на следећој страни

Табела 24.750 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Baferovano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (bafer) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Baferovano	`OUTPUT`	[vektor: poligon]	Izlazni (bafer) poligonski sloj

Python kod

ID algoritma: `native:taperedbuffer`

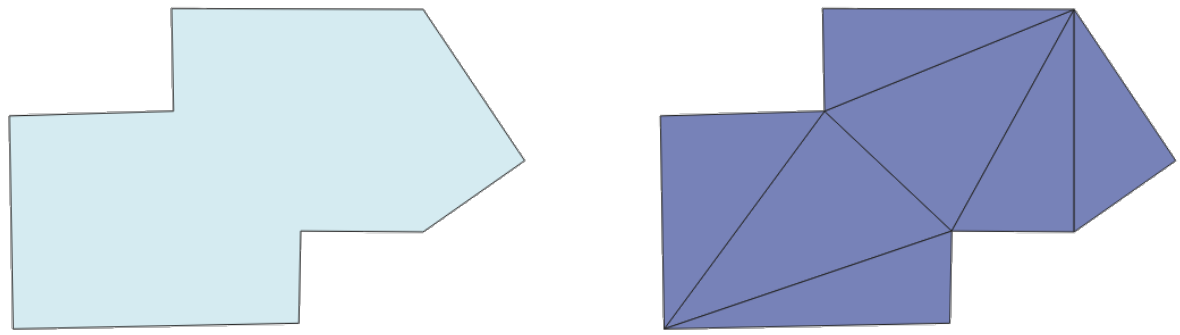
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Teselacija

Teseliše sloj geometrije poligona, deleći geometrije na trougaone komponente.

Izlazni sloj se sastoji od multipoligonskih geometrija za svaki ulazni objekat, gde se svaki multipoligon sastoji od više trougaonih komponentnih poligona.



Слика24.156: Teselirani poligon (desno)

Omogućava *izmenu objekata na licu mesta* za poligonske objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: poligon]	Ulazni poligonski vektorski sloj
Teselirano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Teselirano	`OUTPUT`	[vektor: poligon]	Izlazni multipoligonZ sloj

Python kod

ID algoritma: `3d:tessellate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poprečni presek

Pravi poprečne preseke na temenima za (multi)linijski niz.

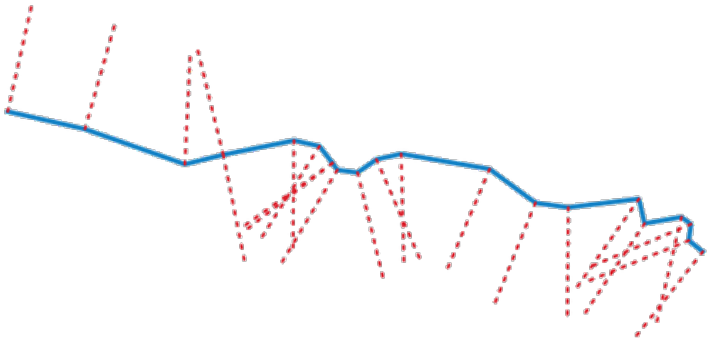
Poprečni presek je linija orijentisana pod uglom (podrazumevano upravno) na ulazne polilinije (na temenima).

Polje(a) iz objekta(ata) vraćaju se u poprečnom preseku sa ovim novim poljima:

- TR_FID: ID originalnog objekta
- TR_ID: ID poprečnog preseka. Svaki poprečni presek ima jedinstveni ID
- TR_SEGMENT: ID segmenta linijskog niza
- TR_ANGLE: Ugao u stepenima od originalne linije na temenu
- TR_LENGTH: Ukupna dužina vraćenog poprečnog preseka
- TR_ORIENT: Strana poprečnog preseka (samo levo ili desno od linije, ili obe strane)

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.



Слика24.157: Isprekidane crvene linije predstavljaju poprečni presek ulaznog linijskog sloja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Dužina poprečnog preseka	`LENGTH`	[numerički: decimalni] Podrazumevano: 5.0	Dužina u jedinici karte poprečnog preseka
Ugao u stepenima od originalne linije na temenima	`ANGLE`	[numerički: decimalni] Podrazumevano: 90.0	Promeni ugao poprečnog preseka
Strana za pravljenje poprečnog preseka	`SIDE`	[nabrajanje]	Izaberite stranu poprečnog preseka. Dostupne opcije su: • 0 — Levo • 1 — Desno • 2 — Obe
Poprečni presek	`OUTPUT`	[vektor: linija] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni linijski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Poprečni presek	`OUTPUT`	[vektor: linija]	Izlazni linijski sloj

Python kod

ID algoritma: `native:transect`

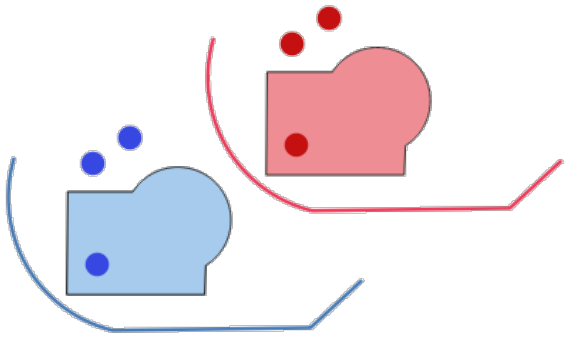
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Transliraj

Pomera geometrije unutar sloja, pomerajući ih za unapred definisani X i Y pomeraj.

Z i M vrednosti prisutne u geometriji takođe se mogu translirati.



Слика24.158: Objekti crvenom bojom predstavljaju transliranu geometriju ulaznog (plavog) sloja

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

See also

Niz translatiranih objekata, *Pomeri linije*, *Rotiraj*, *Zameni X i Y koordinate*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Rastojanje pomeraja (x-osa)	`DELTA_X`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po X osi

наставак на следећој страни

Табела 24.756 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Rastojanje pomeraja (y-osa)	`DELTA_Y`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po Y osi
Rastojanje pomeraja (z-osa)	`DELTA_Z`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje po Z osi
Rastojanje pomeraja (m vrednosti)	`DELTA_M`	[numerički: decimalni] Podrazumevano: 0.0	Pomeraj koji se primenjuje na M osi
Translatirano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Translatirano	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj

Python kod

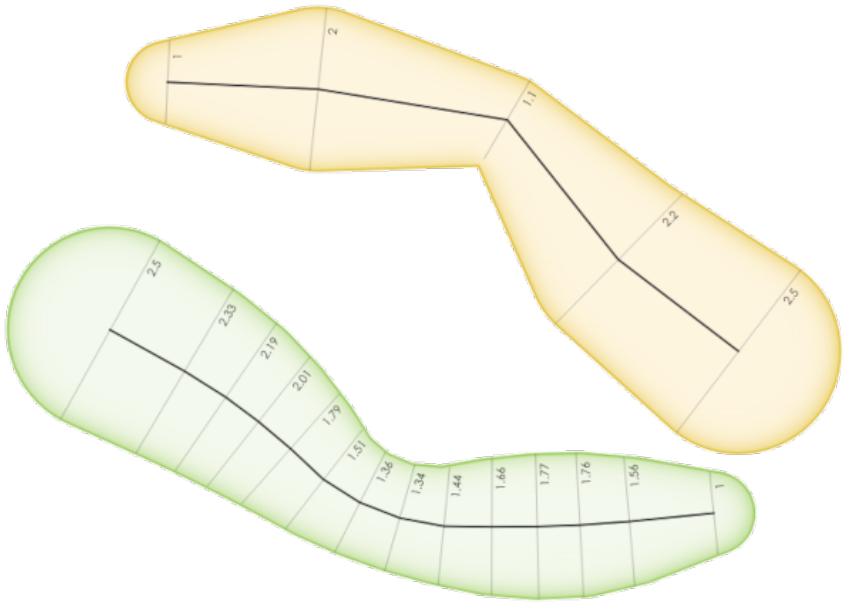
ID algoritma: `native:translategeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Bafer promenljive širine (po M vrednosti)

Pravi bafere promenljive širine duž linija, koristeći M vrednost geometrija linija kao prečnik bafera na svakom temenu.



Слика24.159: Primer promenljivog bafera

See also

Suženi baferi, *Bafer*, *Postavi M vrednost*, *Bafer promenljivog rastojanja*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski vektorski sloj
Segmenti	`SEGMENTS`	[numerički: ceo broj] Podrazumevano: 16	Broj segmenata bafera po četvrtini kruga. Može biti jedinstvena vrednost (ista vrednost za sve objekte), ili se može uzeti iz podataka objekata (vrednost može zavisiti od atributa objekata).
Baferovano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni (bafer) sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Baferovano	`OUTPUT`	[vektor: poligon]	Poligonski sloj promenljivog bafera

Python kod

ID algoritma: `native:bufferbym`

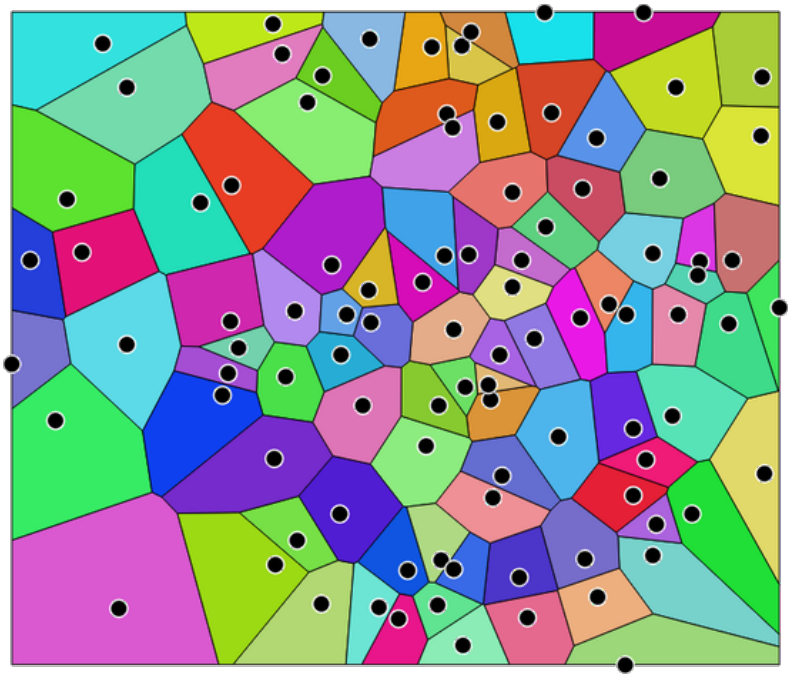
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Voronoi poligoni

Uzima tačkasti sloj i generiše poligonski sloj koji sadrži Voronoi poligone (poznate i kao Thiessen poligoni) koji odgovaraju tim ulaznim tačkama.

Bilo koja lokacija unutar Voronoi poligona bliža je pridruženoj tački nego bilo kojoj drugoj tački.



Слика24.160: Voronoi poligoni

Podrazumevani meni: *Vektor* ► *Alatke za geometriju*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Region bafera (% obuhvata)	`BUFFER`	[numerički: decimalni] Podrazumevano: 0.0	Obuhvat izlaznog sloja biće toliko veći od obuhvata ulaznog sloja

наставак на следећој страни

Табела 24.760 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tolerancija Opciono `Dodato u 3.34`	`TOLERANCE`	[numerički: decimalni] Podrazumevano: 0.0	Zadaje opcionu toleranciju lepljenja koja se može koristiti za poboljšanje robusnosti voronoi.
Kopiraj atribute iz ulaznih objekata `Dodato u 3.34`	`COPY_ATTRIBUTES`	[logička vrednost] Podrazumevano: True	Zadaje da li treba dodati polja koja čuvaju ID uključenih tačkastih objekata u izlaz. Ako je False, pravi se `id` polje za identifikaciju poligona.
Voronoi poligoni	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj (sa Voronoi poligonima). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Voronoi poligoni	`OUTPUT`	[vektor: poligon]	Voronoi poligoni ulaznog tačkastog vektorskog sloja

Python kod

ID algoritma: `native:voronoipolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.27 Vektorsko preklapanje

Isecanje

Iseca vektorski sloj pomoću objekata dodatnog poligonskog sloja.

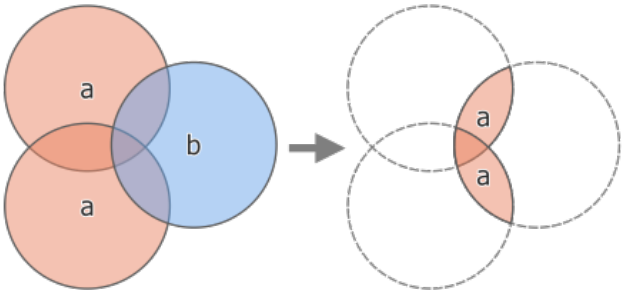
U rezultujući sloj biće dodati samo oni delovi objekata iz ulaznog sloja koji padaju unutar poligona sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.

Ovaj algoritam koristi prostorne indekse na provajderima, pripremljene geometrije i primenjuje operaciju isecanja ako geometrija nije potpuno sadržana u maskirnoj geometriji.



Слика24.161: Operacija isecanja između ulaznog sloja „a“ sa dva objekta i sloja preklapanja „b“ sa jednim objektom (levo) — daje novi sloj sa izmenjenim objektima „a“ (desno)

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Podrazumevani meni: *Vektor ► Alatke za geoobradu*

See also

Presek, *Razlika*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj koji sadrži objekte koji se isecaju
Sloj preklapanja	`OVERLAY`	[vektor: poligon]	Sloj koji sadrži objekte za isecanje
Isečeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži objekte iz ulaznog sloja koji su unutar sloja preklapanja (za isecanje). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži objekte iz ulaznog sloja podeljene slojem preklapanja.

Python kod

ID algoritma: `native:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Razlika

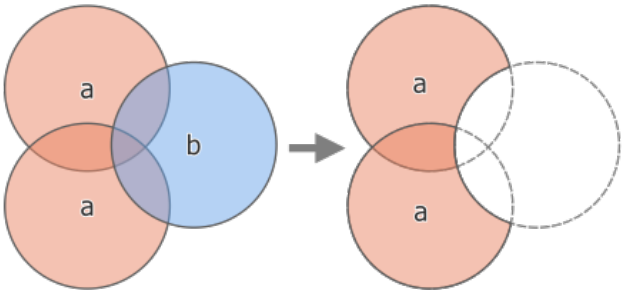
Izdvaja objekte iz ulaznog sloja koji ne padaju unutar granica sloja preklapanja.

Objekti ulaznog sloja koji se delimično preklapaju sa objektom(ima) sloja preklapanja dele se duž granice tih objekata i zadržavaju se samo delovi izvan objekata sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.



Слика24.162: Operacija razlike između ulaznog sloja „a“ sa dva objekta i sloja preklapanja „b“ sa jednim objektom (levo) — daje novi sloj sa izmenjenim objektima „a“ (desno)

Omogućava *izmenu objekata na licu mesta* za tačkaste, linijske i poligonske objekte

Podrazumevani meni: *Vektor* ► *Alatke za geoobradu*

See also

Razlika (višestruka), *Simetrična razlika*, *Isecanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj iz kojeg se izdvajaju (delovi) objekata.
Sloj preklapanja	`OVERLAY`	[vektor: geometrija]	Sloj koji sadrži geometrije koje se oduzimaju od geometrija ulaznog sloja. Očekuje se da ima bar onoliko dimenzija (tačka: 0D, linija: 1D, poligon: 2D, zapremina: 3D) koliko i geometrije ulaznog sloja.

наставак на следећој страни

Табела 24.764 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Razlika	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (delove) objekata iz ulaznog sloja koji nisu unutar sloja preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Veličina mreže Opciono	`GRID_SIZE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se zada, ulazne geometrije se prilepljuju na mrežu date veličine, a temena rezultata izračunavaju se na toj istoj mreži.

Izlazi

Oznaka	Назив	Tip	Opis
Razlika	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži (delove) objekata iz ulaznog sloja koji se ne preklapaju sa slojem preklapanja.

Python kod

ID algoritma: `native:difference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Razlika (višestruka)

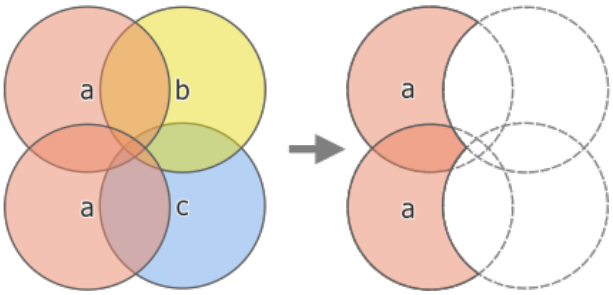
Izdvaja objekte iz ulaznog sloja koji padaju potpuno izvan ili se samo delimično preklapaju sa objektima iz bilo kog sloja preklapanja.

Za svaki sloj preklapanja razlika se računa između rezultata svih prethodnih operacija razlike i tog sloja preklapanja. Objekti ulaznog sloja koji se delimično preklapaju sa objektom(ima) u sloju preklapanja dele se duž granice tih objekata i zadržavaju se samo delovi izvan objekata sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.



Слика24.163: Operacija razlike između ulaznog sloja „a“ sa dva objekta i slojeva preklapanja „b“ i „c“ sa jednim objektom (levo) — daje novi sloj sa izmenjenim objektima „a“ (desno)

See also

Razlika, *Simetrična razlika*, *Isecanje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj iz kojeg se izdvajaju (delovi) objekata.
Slojevi za preklapanje	`OVERLAYS`	[vektor: geometrija] [lista]	Lista slojeva koji sadrže geometrije koje se oduzimaju od geometrija ulaznog sloja. Očekuje se da imaju bar onoliko dimenzija (tačka: 0D, linija: 1D, poligon: 2D, zapremina: 3D) koliko i geometrije ulaznog sloja.
Razlika	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (delove) objekata iz ulaznog sloja koji se ne preklapaju sa objektima slojeva preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Razlika	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži (delove) objekata iz ulaznog sloja koji se ne preklapaju sa objektima iz slojeva preklapanja.

Python kod

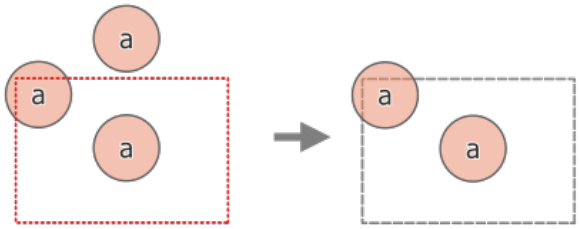
ID algoritma: `native:multidifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

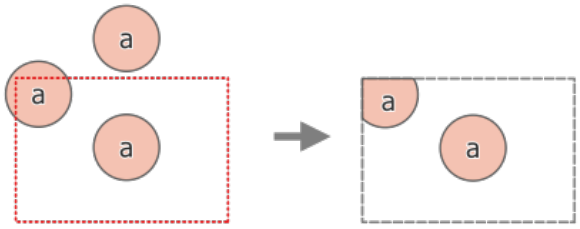
Izdvoj/iseci po obuhvatu

Pravi novi vektorski sloj koji sadrži samo objekte koji padaju unutar zadatog obuhvata. Svi objekti koji se seku sa obuhvatom biće podrazumevano uključeni.



Слика24.164: Operacija izdvajanja između ulaznog sloja „a“ sa tri objekta i isprekidanog obuhvata (levo) — rezultujući objekti sa isprekidanim obuhvatom radi poređenja (desno)

Opciono, geometrije objekata mogu se i isecati na obuhvat.



Слика24.165: Operacija izdvajanja između ulaznog sloja „a“ sa tri objekta i isprekidanog obuhvata (levo) — rezultujući objekti sa isprekidanim obuhvatom radi poređenja (desno)

See also

Isecanje

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj iz kojeg se izdvajaju (delovi) objekata.

наставак на следећој страни

Табела 24.769 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Obuhvat (xmin, xmax, ymin, ymax)	`EXTENT`	[obuhvat]	Obuhvat za isecanje. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Iseci objekte na obuhvat	`CLIP`	[logička vrednost] Podrazumevano: False	Ako je uključeno, geometrije će biti isečene na izabrani obuhvat umesto da se cela geometrija uzme kao izlaz. Osim toga, izlazne geometrije biće automatski pretvorene u višedelne geometrije radi obezbeđivanja jednoobraznih izlaznih tipova.
Izdvojeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži objekte iz ulaznog sloja koji su unutar obuhvata isecanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izdvojeno	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži isečene objekte.

Python kod

ID algoritma: `native:extractbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Presek

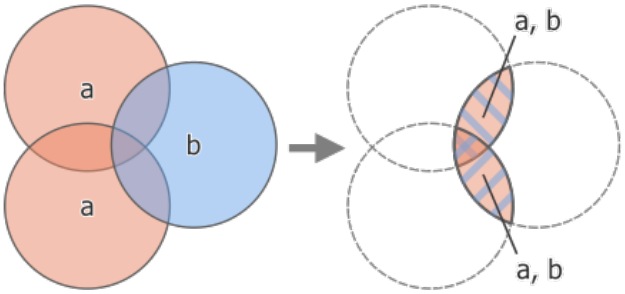
Izdvaja delove objekata iz ulaznog sloja koji se preklapaju sa objektima u sloju preklapanja.

Objektima u sloju preseka dodeljuju se atributi preklapajućih objekata iz ulaznog sloja i sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.



Слика24.166: Operacija preseka između ulaznog sloja „a“ sa dva objekta i sloja preklapanja „b“ sa jednim objektom (levo) — preklapajuće oblasti postaju novi sloj sa dva objekta i atributima oba sloja (desno)

Podrazumevani meni: *Vektor ► Alatke za geoobradu*

Упозорење

Ovaj algoritam uklanja postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Presek (višestruki), *Isecanje*, *Razlika*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj iz kojeg se izdvajaju (delovi) objekata.
Sloj preklapanja	`OVERLAY`	[vektor: geometrija]	Sloj koji sadrži objekte za proveru preklapanja. Očekuje se da geometrija njegovih objekata ima bar onoliko dimenzija (tačka: 0D, linija: 1D, poligon: 2D, zapremina: 3D) koliko i geometrija ulaznog sloja.

наставак на следећој страни

Табела 24.771 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ulazna polja koja se zadržavaju (ostavite prazno za zadržavanje svih polja) Opciono	`INPUT_FIELDS`	[polje tabele: bilo koji] [lista] Podrazumevano: Nije postavljeno	Polje(a) ulaznog sloja koja se zadržavaju u izlazu. Ako se ne izabere nijedno polje, uzimaju se sva polja.
Polja preklapanja koja se zadržavaju (ostavite prazno za zadržavanje svih polja) Opciono	`OVERLAY_FIELDS`	[polje tabele: bilo koji] [lista] Podrazumevano: Nije postavljeno	Polje(a) sloja preklapanja koja se zadržavaju u izlazu. Ako se ne izabere nijedno polje, uzimaju se sva polja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.
Presek	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (delove) objekata iz ulaznog sloja koji se preklapaju sa jednim ili više objekata iz sloja preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja preklapanja Opciono	`OVERLAY_FIELDS_PREFI`	[niska]	Dodajte prefiks za prepoznavanje polja sloja preklapanja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.
Veličina mreže Opciono	`GRID_SIZE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se zada, ulazne geometrije se prilepljuju na mrežu date veličine, a temena rezultata izračunavaju se na toj istoj mreži.

Izlazi

Oznaka	Назив	Tip	Opis
Presek	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži (delove) objekata iz ulaznog sloja koji se preklapaju sa slojem preklapanja.

Python kod

ID algoritma: `native:intersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Presek (višestruki)

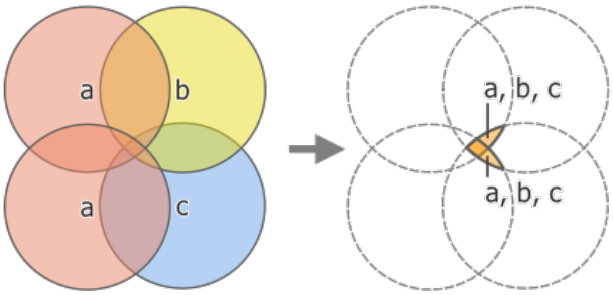
Izdvaja preklapajuće delove objekata u ulaznom sloju i svim slojevima preklapanja.

Objektima u izlaznom sloju dodeljuju se atributi preklapajućih objekata iz ulaznog sloja i sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.



Слика24.167: Operacija preseka između ulaznog sloja „a“ sa dva objekta i slojeva preklapanja „b“ i „c“ sa jednim objektom (levo) — preklapajuće oblasti postaju novi sloj sa dva objekta i atributima svih slojeva (desno)

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Presek, *Isecanje*, *Razlika*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Sloj iz kojeg se izdvajaju (delovi) objekata.
Slojevi za preklapanje	`OVERLAYS`	[vektor: geometrija] [lista]	Slojevi koji sadrže objekte za proveru preklapanja. Očekuje se da geometrija objekata ima bar onoliko dimenzija (tačka: 0D, linija: 1D, poligon: 2D, zapremina: 3D) koliko i geometrija ulaznog sloja.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja preklapanja Opciono	`OVERLAY_FIELDS_PREFI`	[niska]	Dodajte prefiks za prepoznavanje polja slojeva preklapanja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.

Izlazi

Oznaka	Назив	Tip	Opis
Presek	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži (delove) objekata iz ulaznog sloja koji se preklapaju sa svim slojevima preklapanja.

Python kod

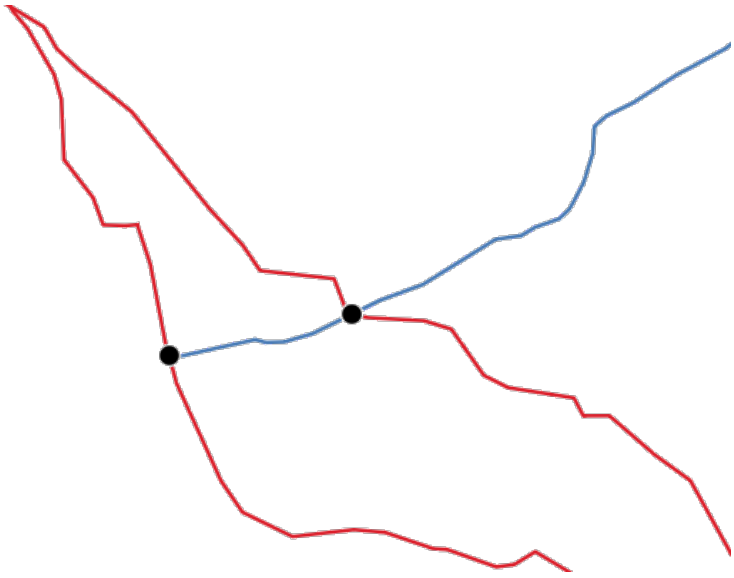
ID algoritma: `native:multiintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Presek linija

Pravi tačkaste objekte tamo gde se linije iz dva sloja seku.



Слика24.168: Tačke preseka

Podrazumevani meni: *Vektor ► Alatke za analizu*

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski sloj.
Sloj preseka	`INTERSECT`	[vektor: linija]	Sloj koji se koristi za pronalaženje preseka linija.
Ulazna polja koja se zadržavaju (ostavite prazno za zadržavanje svih polja) Opciono	`INPUT_FIELDS`	[polje tabele: bilo koji] [lista] Podrazumevano: Nije postavljeno	Polje(a) ulaznog sloja koja se zadržavaju u izlazu. Ako se ne izabere nijedno polje, uzimaju se sva polja.
Polja sloja preseka koja se zadržavaju (ostavite prazno za zadržavanje svih polja) Opciono	`INTERSECT_FIELDS`	[polje tabele: bilo koji] [lista] Podrazumevano: Nije postavljeno	Polje(a) sloja preseka koja se zadržavaju u izlazu. Ako se ne izabere nijedno polje, uzimaju se sva polja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.
Presek	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži tačke preseka linija iz ulaznog sloja i sloja preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja sloja preseka Opciono	`INTERSECT_FIELDS_PREF`	[niska]	Dodajte prefiks za prepoznavanje polja sloja preseka.

Izlazi

Oznaka	Назив	Tip	Opis
Preseci	`OUTPUT`	[vektor: tačka]	Tačkasti vektorski sloj preseka linija, sa atributima oba sloja.

Python kod

ID algoritma: `native:lineintersections`

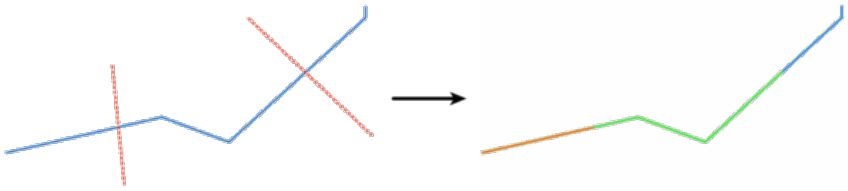
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Podeli linijama

Deli linije ili poligone u jednom sloju pomoću linija ili prstenova poligona u drugom sloju za definisanje tačaka preloma. Preseci između geometrija u oba sloja smatraju se tačkama deljenja.

Izlaz će sadržati višedelne geometrije za podeljene objekte.



Слика24.169: Podeli linije

Omogućava *izmenu objekata na licu mesta* za linijske i poligonske objekte

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija, poligon]	Sloj koji sadrži linije ili poligone za deljenje.
Sloj za deljenje	`LINES`	[vektor: linija, poligon]	Sloj čije se linije ili prstenovi koriste za definisanje tačaka preloma.
Podeljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži podeljene linijske/poligonske objekte iz ulaznog sloja (u slučaju da ih preseca linija iz sloja za deljenje). *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Podeljeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa podeljenim linijama ili poligonima iz ulaznog sloja.

Python kod

ID algoritma: `native:splitwithlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Simetrična razlika

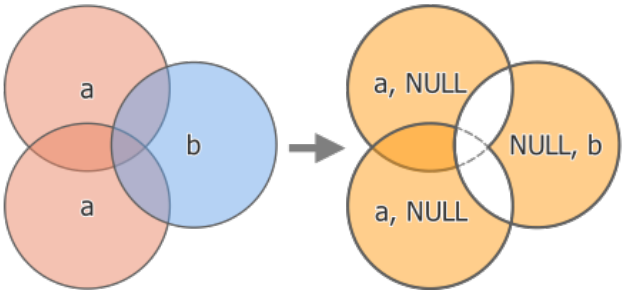
Pravi sloj koji sadrži objekte iz ulaznog sloja i sloja preklapanja, ali sa uklonjenim preklapajućim oblastima između dva sloja.

Atributska tabela sloja simetrične razlike sadrži atribute i polja iz ulaznog sloja i sloja preklapanja.

Упозорење

Samo izmena geometrije

Ova operacija menja samo geometriju objekata. Vrednosti atributa objekata se **ne menjaju**, iako će svojstva kao što su površina ili dužina objekata biti izmenjena operacijom preklapanja. Ako se takva svojstva čuvaju kao atributi, ti atributi će morati ručno da se ažuriraju.



Слика24.170: Operacija simetrične razlike između ulaznog sloja „a“ sa dva objekta i sloja preklapanja „b“ sa jednim objektom (levo) — rezultujući sloj sa tri objekta i atributima oba sloja (desno)

Podrazumevani meni: *Vektor ► Alatke za geoobradu*

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Razlika, *Isecanje*, *Presek*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Prvi sloj iz kojeg se izdvajaju (delovi) objekata.
Sloj preklapanja	`OVERLAY`	[vektor: geometrija]	Drugi sloj iz kojeg se izdvajaju (delovi) objekata. U idealnom slučaju tip geometrije treba da bude isti kao kod ulaznog sloja.
Simetrična razlika	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (delove) objekata iz ulaznog sloja i sloja preklapanja koji se ne preklapaju sa objektima iz drugog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja preklapanja Opciono	`OVERLAY_FIELDS_PREFI`	[niska]	Dodajte prefiks za prepoznavanje polja sloja preklapanja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.
Veličina mreže Opciono	`GRID_SIZE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se zada, ulazne geometrije se prilepljuju na mrežu date veličine, a temena rezultata izračunavaju se na toj istoj mreži.

Izlazi

Oznaka	Назив	Tip	Opis
Simetrična razlika	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži (delove) objekata iz svakog sloja koji se ne preklapaju sa drugim slojem, sa atributima oba sloja.

Python kod

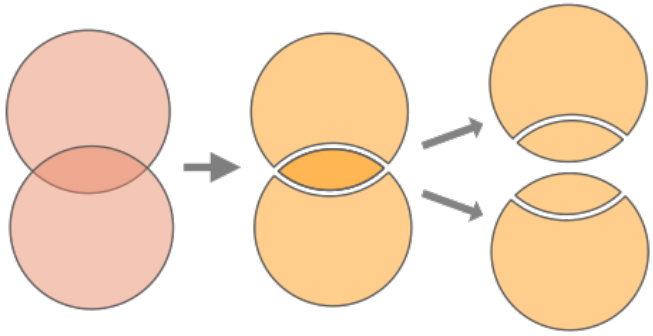
ID algoritma: `native:symmetricaldifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

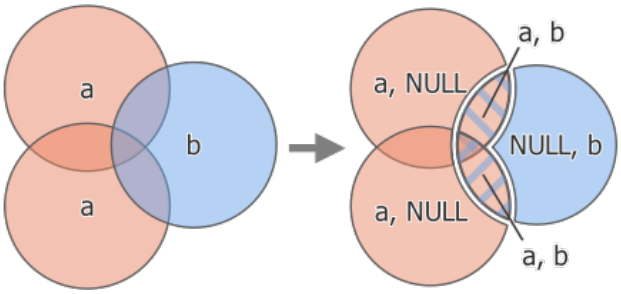
Unija

Proverava preklapanja između objekata unutar ulaznog sloja i pravi zasebne objekte za preklapajuće i neprekla-pajuće delove. Oblast preklapanja napraviće onoliko istovetnih preklapajućih objekata koliko ima objekata koji učestvuju u tom preklapanju.



Слика24.171: Operacija unije sa jednim ulaznim slojem sa dva preklapajuća objekta (levo) — daje četiri objekta (sredina) — objekti pomereni radi jasnoće (desno)

Može se koristiti i sloj preklapanja, u kom slučaju se objekti iz svakog sloja dele na svom preklapanju sa objektima iz drugog, praveći sloj koji sadrži sve delove iz ulaznog sloja i sloja preklapanja. Objekti na istom sloju neće deliti jedni druge. Atributska tabela sloja unije popunjava se vrednostima atributa odgovarajućeg originalnog sloja za nepreklapajuće objekte, i vrednostima atributa oba sloja za preklapajuće objekte.



Слика24.172: Operacija unije između ulaznog sloja „a“ sa dva objekta i sloja preklapanja „b“ sa jednim objektom (levo) — rezultujući sloj sa pet objekata i atributima oba sloja (desno)

Белешка

Uz sloj preklapanja, objekti na istom sloju neće deliti jedni druge. Ako želite da podelite preklapanja i na istom sloju i sa drugim slojevima, prvo pokrenite algoritam sa više slojeva, a zatim ga ponovo pokrenite samo sa prethodnim izlazom.

Podrazumevani meni: *Vektor* ► *Alatke za geoobradu*

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Unija (višestruka), *Isecanje*, *Razlika*, *Presek*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj koji se deli na svim presecima.
Sloj preklapanja Opciono	`OVERLAY`	[vektor: geometrija]	Sloj koji će se kombinovati sa prvim. U idealnom slučaju tip geometrije treba da bude isti kao kod ulaznog sloja.
Unija	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (podeljene i duplirane) objekte iz ulaznog sloja i sloja preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja preklapanja Opciono	`OVERLAY_FIELDS_PREFI`	[niska]	Dodajte prefiks za prepoznavanje polja sloja preklapanja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.
Veličina mreže Opciono	`GRID_SIZE`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Ako se zada, ulazne geometrije se prilepljuju na mrežu date veličine, a temena rezultata izračunavaju se na toj istoj mreži.

Izlazi

Oznaka	Назив	Tip	Opis
Unija	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži sve preklapajuće i nepreklapajuće delove iz obrađenog sloja(eva).

Python kod

ID algoritma: `native:union`

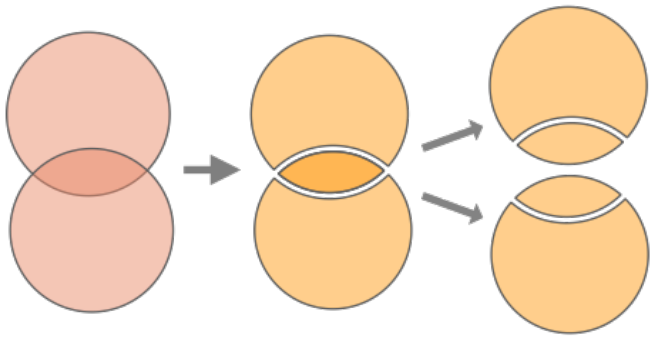
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

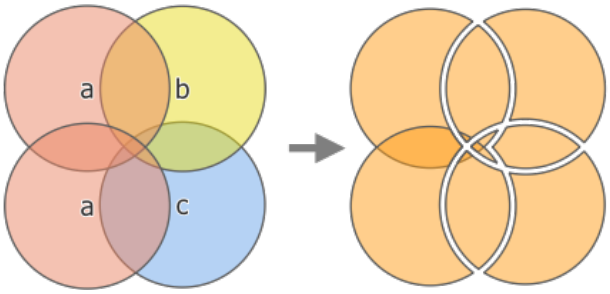
Unija (višestruka)

Proverava preklapanja između objekata unutar ulaznog sloja i pravi zasebne objekte za preklapajuće i neprekla-pajuće delove. Oblast preklapanja napraviće onoliko istovetnih preklapajućih objekata koliko ima objekata koji učestvuju u tom preklapanju.



Слика24.173: Operacija unije sa jednim ulaznim slojem sa dva preklapajuća objekta (levo) — daje četiri objekta (sredina) — objekti pomereni radi jasnoće (desno)

Može se koristiti i više slojeva preklapanja, u kom slučaju se objekti iz svakog sloja dele na svom preklapanju sa objektima iz svih ostalih slojeva, praveći sloj koji sadrži sve delove iz ulaznog sloja i sloja preklapanja. Objekti na istom sloju neće deliti jedni druge. Atributska tabela sloja unije popunjava se vrednostima atributa odgovarajućeg originalnog sloja za nepreklapajuće objekte, i vrednostima atributa iz slojeva preklapanja za preklapajuće objekte.



Слика24.174: Operacija unije između ulaznog sloja „a“ sa dva objekta i slojeva preklapanja „b“ i „c“ sa jednim objektom (levo) — rezultujući sloj sa jedanaest objekata i atributima svih slojeva (desno)

Белешка

Uz sloj preklapanja, objekti na istom sloju neće deliti jedni druge. Ako želite da podelite preklapanja i na istom sloju i sa drugim slojevima, prvo pokrenite algoritam sa više slojeva, a zatim ga ponovo pokrenite samo sa prethodnim izlazom.

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim sloje-vima.

See also

Unija, *Isecanje*, *Razlika*, *Presek*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj koji se deli na svim presecima.
Slojevi za preklapanje Opciono	`OVERLAYS`	[vektor: geometrija] [lista]	Slojevi koji će se kombinovati sa prvim. U idealnom slučaju tip geometrije treba da bude isti kao kod ulaznog sloja.
Unija	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte sloj koji sadrži (podeljene i duplirane) objekte iz ulaznog sloja i slojeva preklapanja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Napredni parametri

Oznaka	Назив	Tip	Opis
Prefiks polja preklapanja Opciono	`OVERLAY_FIELDS_PREFI`	[niska]	Dodajte prefiks za prepoznavanje polja slojeva preklapanja. Duplikatima naziva polja biće dodat brojčani sufiks radi izbegavanja sudara.

Izlazi

Oznaka	Назив	Tip	Opis
Unija	`OUTPUT`	[isto kao ulaz]	Sloj koji sadrži sve preklapajuće i nepreklapajuće delove iz obrađenog sloja(eva), sa atributima svih slojeva.

Python kod

ID algoritma: `native:multiunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.28 Izbor vektora

Izdvajanje po atributu

Pravi dva vektorska sloja od ulaznog sloja: jedan će sadržati samo objekte koji se podudaraju, dok će drugi sadržati sve objekte koji se ne podudaraju.

Kriterijum za dodavanje objekata u rezultujući sloj zasniva se na vrednostima atributa iz ulaznog sloja.

See also

Izbor po atributu

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj iz kojeg se izdvajaju objekti.
Atribut za izbor	`FIELD`	[polje tabele: bilo koji]	Polje sloja po kojem se filtrira
Operator	`OPERATOR`	[nabrajanje] Podrazumevano: 0	Dostupno je mnogo različitih operatora: • 0 — = • 1 — ≠ • 2 — >>• 3 — >= • 4 — < • 5 — <= • 6 — počinje sa • 7 — sadrži • 8 — je null • 9 — nije null • 10 — ne sadrži
Vrednost Opciono	`VALUE`	[niska]	Vrednost koja se izračunava
Izdvojeno (atribut)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create Temporary Layer]`	Zadajte izlazni vektorski sloj za objekte koji se podudaraju. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.
Izdvojeno (nepodudarno)	`FAIL_OUTPUT`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte izlazni vektorski sloj za objekte koji se ne podudaraju. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Izdvojeno (atribut)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji se podudaraju
Izdvojeno (nepodudarno)	`FAIL_OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji se ne podudaraju

Python kod

ID algoritma: `native:extractbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje po izrazu

Pravi dva vektorska sloja od ulaznog sloja: jedan će sadržati samo objekte koji se podudaraju, dok će drugi sadržati sve objekte koji se ne podudaraju.

Kriterijum za dodavanje objekata u rezultujući sloj zasniva se na QGIS izrazu. Više informacija o izrazima potražite u *Expressions*.

See also

Izbor po izrazu

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Izraz	`EXPRESSION`	[izraz]	Izraz za filtriranje vektorskog sloja
Objekti koji se podudaraju	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create Temporary Layer]`	Zadajte izlazni vektorski sloj za objekte koji se podudaraju. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.793 – настављено са претходне стране

Oznaka	Naziv	Tip	Opis
Nepodudarno	`FAIL_OUTPUT`	[isto kao ulaz] Podrazumevano: `[Skip output]`	Zadajte izlazni vektorski sloj za objekte koji se ne podudaraju. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Objekti koji se podudaraju	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji se podudaraju
Nepodudarno	`FAIL_OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji se ne podudaraju

Python kod

ID algoritma: `native:extractbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje po lokaciji

Pravi novi vektorski sloj koji sadrži samo objekte iz ulaznog sloja koji se podudaraju.

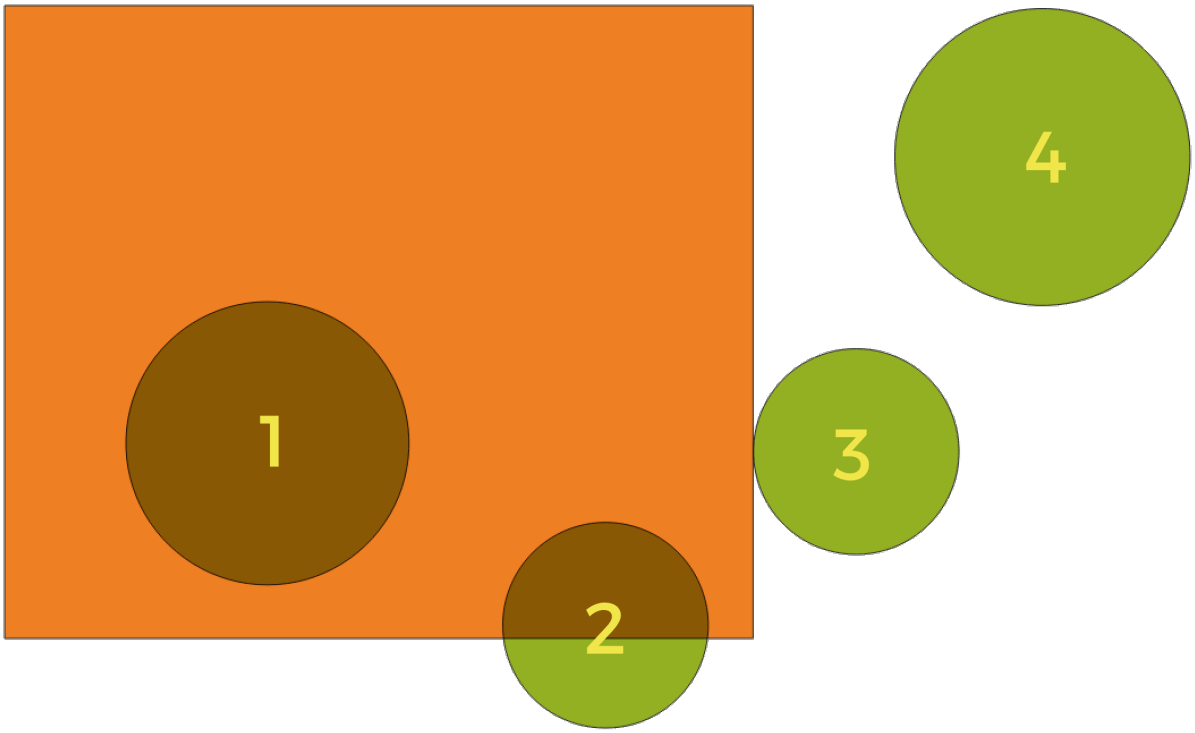
Kriterijum za dodavanje objekata u rezultujući sloj zasniva se na prostornom odnosu između svakog objekta i objekata na dodatnom sloju.

See also

Izbor po lokaciji, *Izdvajanje unutar rastojanja*

Istraživanje prostornih odnosa

Geometrijski predikati su logičke funkcije koje se koriste za utvrđivanje prostornog odnosa jednog objekta prema drugom, poređenjem da li i kako njihove geometrije dele deo prostora.



Слика24.175: Traženje prostornih odnosa između slojeva

Na osnovu gornje slike, tražimo zelene krugove prostornim poređenjem sa narandžastim pravougaonim objektom. Dostupni geometrijski predikati su:

Preseca
: Proverava da li jedna geometrija preseca drugu. Vraća 1 (tačno) ako se geometrije prostorno presecaju (dele bilo koji deo prostora - preklapaju se ili dodiruju), a 0 ako se ne presecaju. Na gornjoj slici, ovo vraća krugove 1, 2 i 3.

Sadrži
: Vraća 1 (tačno) ako i samo ako nijedna tačka geometrije b ne leži izvan geometrije a i bar jedna tačka unutrašnjosti geometrije b leži u unutrašnjosti geometrije a. Na slici se ne vraća nijedan krug, ali bi se vratio pravougaonik da tražite obrnuto, jer on u potpunosti sadrži krug 1. Ovo je suprotno od *su unutar*.

Razdvojeno
: Vraća 1 (tačno) ako geometrije ne dele nijedan deo prostora (bez preklapanja, bez dodirivanja). Vraća se samo krug 4.

Jednako
: Vraća 1 (tačno) ako i samo ako su geometrije potpuno iste. Neće se vratiti nijedan krug.

Dodiruje
: Proverava da li jedna geometrija dodiruje drugu. Vraća 1 (tačno) ako geometrije imaju bar jednu zajedničku tačku, ali im se unutrašnjosti ne presecaju. Vraća se samo krug 3.

Preklapa
Proverava da li se jedna geometrija preklapa sa drugom. Vraća 1 (tačno) ako geometrije dele prostor, iste su dimenzije, ali nisu potpuno sadržane jedna u drugoj. Vraća se samo krug 2.

Su unutar
Proverava da li je jedna geometrija unutar druge. Vraća 1 (tačno) ako je geometrija a potpuno unutar geometrije b. Vraća se samo krug 1.

Ukršta
Vraća 1 (tačno) ako zadate geometrije imaju neke, ali ne sve, zajedničke unutrašnje tačke i ako je samo ukrštanje niže dimenzije od najviše zadate geometrije. Na primer, linija koja ukršta poligon ukršta ga kao linija (tačno). Dve linije koje se ukrštaju ukrštaju se kao tačka (tačno). Dva poligona ukrštaju se kao poligon (netačno). Na slici se neće vratiti nijedan krug.

Parametri

Oznaka	Naziv	Tip	Opis
Izdvoj objekte iz	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Kod kojih objekti (geometrijski predikat)	`PREDICATE`	[nabrajanje] [lista] Podrazumevano: [0]	Vrsta prostornog odnosa koju ulazni objekat treba da ima sa objektom preseka da bi mogao biti izabran. Jedno ili više od: • 0 — presecaju • 1 — sadrže • 2 — razdvojen • 3 — jednak • 4 — dodiruje • 5 — preklapa • 6 — je unutar • 7 — prelazi preko Ako je izabrano više od jednog uslova, bar jedan od njih (operacija OR) mora biti ispunjen da bi objekat bio izdvojen.
Poređenjem sa objektima iz	`INTERSECT`	[vektor: geometrija]	Vektorski sloj preseka
Izdvojeno (lokacija)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za objekte koji imaju izabrane prostorne odnose sa jednim ili više objekata na sloju za poređenje. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Izdvojeno (lokacija)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji imaju izabrane prostorne odnose sa jednim ili više objekata na sloju za poređenje.

Python kod

ID algoritma: `native:extractbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje unutar rastojanja

Pravi novi vektorski sloj koji sadrži samo objekte iz ulaznog sloja koji se podudaraju. Objekti se kopiraju svuda gde se nalaze unutar zadatog maksimalnog rastojanja od objekata na dodatnom referentnom sloju.

See also

Izbor unutar rastojanja, *Izdvajanje po lokaciji*

Parametri

Oznaka	Naziv	Tip	Opis
Izdvoj objekte iz	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj iz kojeg se kopiraju objekti
Poređenjem sa objektima iz	`REFERENCE`	[vektor: geometrija]	Vektorski sloj čija se blizina objekata koristi
Kod kojih su objekti unutar	`DISTANCE`	[numerički: decimalni] Podrazumevano: 100.0	Maksimalno rastojanje oko referentnih objekata unutar kojeg se biraju ulazni objekti
Izmeni trenutni izbor tako što će se	`METHOD`	[nabrajanje] Podrazumevano: 0	Način na koji se upravlja izborom algoritma. Jedno od: • 0 — praviti novi izbor • 1 — dodati trenutnom izboru • 2 — birati unutar trenutnog izbora • 3 — uklanjati iz trenutnog izbora
Izdvojeno (lokacija)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za objekte koji su unutar zadatog rastojanja od referentnih objekata. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Izdvojeno (lokacija)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa objektima iz ulaznog sloja koji ispunjavaju uslov rastojanja od referentnih objekata

Python kod

ID algoritma: `native:extractwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Filtriranje po tipu geometrije

Filtrira objekte po tipu geometrije. Ulazni objekti biće usmereni na različite izlaze u zavisnosti od toga da li imaju tačkastu, linijsku ili poligonsku geometriju.

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj koji se izračunava

Izlazi

Oznaka	Naziv	Tip	Opis
Tačkasti objekti Opciono	`POINTS`	[vektor: tačka]	Sloj sa tačkama
Linijski objekti Opciono	`LINES`	[vektor: linija]	Sloj sa linijama
Poligonski objekti Opciono	`POLYGONS`	[vektor: poligon]	Sloj sa poligonima
Objekti bez geometrije Opciono	`NO_GEOMETRY`	[vektor: tabela]	Vektorski sloj bez geometrije

Python kod

ID algoritma: `native:filterbygeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajno izdvajanje

Uzima vektorski sloj i generiše novi koji sadrži samo podskup objekata iz ulaznog sloja.

Podskup se definiše nasumično, na osnovu ID-jeva objekata, koristeći procenat ili brojčanu vrednost za određivanje ukupnog broja objekata u podskupu.

Упозорење

Ovaj algoritam odbacuje postojeće primarne ključeve ili FID vrednosti i ponovo ih generiše u izlaznim slojevima.

See also

Slučajan izbor

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Izvorni vektorski sloj iz kojeg se biraju objekti
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metodi slučajnog izbora. Jedno od: • 0 — Broj izabranih objekata • 1 — Procenat izabranih objekata
Broj/procenat izabranih objekata	`NUMBER`	[numerički: ceo broj] Podrazumevano: 10	Broj ili procenat objekata koji se biraju
Izdvojeno (nasumično)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za nasumično izabrane objekte. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Izdvojeno (nasumično)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj koji sadrži nasumično izabrane objekte iz ulaznog sloja

Python kod

ID algoritma: `native:randomextract`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajno izdvajanje unutar podskupova

Uzima vektorski sloj i generiše novi koji sadrži samo podskup objekata iz ulaznog sloja.

Podskup se definiše nasumično, na osnovu ID-jeva objekata, koristeći procenat ili brojčanu vrednost za određivanje ukupnog broja objekata u podskupu. Vrednost procenta/broja ne primenjuje se na ceo sloj, već na svaku kategoriju. Kategorije se definišu prema zadatom atributu.

See also

Slučajan izbor unutar podskupova

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj iz kojeg se biraju objekti
Polje ID-ja	`FIELD`	[polje tabele: bilo koji]	Kategorija izvornog vektorskog sloja iz koje se biraju objekti
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod slučajnog izbora. Jedno od: • 0 — Broj izabranih objekata • 1 — Procenat izabranih objekata
Broj/procenat izabranih objekata	`NUMBER`	[numerički: ceo broj] Podrazumevano: 10	Broj ili procenat objekata koji se biraju
Izdvojeno (nasumično stratifikovano)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj za nasumično izabrane objekte. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Naziv	Tip	Opis
Izdvojeno (nasumično stratifikovano)	`OUTPUT`	[isto kao ulaz]	Vektorski sloj koji sadrži nasumično izabrane objekte iz ulaznog sloja

Python kod

ID algoritma: `qgis:randomextractwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajan izbor

Uzima vektorski sloj i bira podskup njegovih objekata. Ovaj algoritam ne generiše novi sloj.

Podskup se definiše nasumično, na osnovu ID-jeva objekata, koristeći procenat ili brojčanu vrednost za određivanje ukupnog broja objekata u podskupu.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

See also

Slučajno izdvajanje

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj za izbor
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod slučajnog izbora. Jedno od: • 0 — Broj izabranih objekata • 1 — Procenat izabranih objekata
Broj/procenat izabranih objekata	`NUMBER`	[numerički: ceo broj] Podrazumevano: 10	Broj ili procenat objekata koji se biraju

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `qgis:randomselection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Slučajan izbor unutar podskupova

Uzima vektorski sloj i bira podskup njegovih objekata. Ovaj algoritam ne generiše novi sloj.

Podskup se definiše nasumično, na osnovu ID-jeva objekata, koristeći procenat ili brojčanu vrednost za određivanje ukupnog broja objekata u podskupu.

Vrednost procenta/broja ne primenjuje se na ceo sloj, već na svaku kategoriju.

Kategorije se definišu prema zadatom atributu, koji se takođe zadaje kao ulazni parametar algoritma.

Ne pravi se nijedan novi izlaz.

Podrazumevani meni: *Vektor ► Istraživačke alatke*

See also

Slučajno izdvajanje unutar podskupova

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj u kojem se biraju objekti
Polje ID-ja	`FIELD`	[polje tabele: bilo koji]	Kategorija ulaznog sloja iz koje se biraju objekti
Metod	`METHOD`	[nabrajanje] Podrazumevano: 0	Metod slučajnog izbora. Jedno od: • 0 — Broj izabranih objekata • 1 — Procenat izabranih objekata
Broj/procenat izabranih objekata	`NUMBER`	[numerički: ceo broj] Podrazumevano: 10	Broj ili procenat objekata koji se biraju

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `qgis:randomselectionwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbor po atributu

Pravi izbor u vektorskom sloju.

Kriterijum za izbor objekata zasniva se na vrednostima atributa iz ulaznog sloja.

See also

Izdvajanje po atributu

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj u kojem se biraju objekti
Atribut za izbor	`FIELD`	[polje tabele: bilo koji]	Polje sloja po kojem se filtrira

наставак на следећој страни

Табела 24.809 – настављено са претходне стране

Oznaka	Naziv	Tip	Opis
Operator	`OPERATOR`	[nabrajanje] Podrazumevano: 0	Dostupno je mnogo različitih operatora: • 0 — = • 1 — ≠ • 2 — >= • 3 — >= • 4 — < • 5 — <= • 6 — počinje sa • 7 — sadrži • 8 — je null • 9 — nije null • 10 — ne sadrži
Vrednost Opciono	`VALUE`	[niska]	Vrednost koja se izračunava
Izmeni trenutni izbor tako što će se	`METHOD`	[nabrajanje] Podrazumevano: 0	Način na koji se upravlja izborom algoritma. Jedno od: • 0 — praviti novi izbor • 1 — dodati trenutnom izboru • 2 — uklanjati iz trenutnog izbora • 3 — birati unutar trenutnog izbora

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `qgis:selectbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbor po izrazu

Pravi izbor u vektorskom sloju.

Kriterijum za izbor objekata zasniva se na QGIS izrazu. Više informacija o izrazima potražite u *Expressions*.

See also

Izdvajanje po izrazu

Parametri

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Izraz	`EXPRESSION`	[izraz]	Izraz za filtriranje ulaznog sloja
Izmeni trenutni izbor tako što će se	`METHOD`	[nabrajanje] Podrazumevano: 0	Način na koji se upravlja izborom algoritma. Jedno od: • 0 — praviti novi izbor • 1 — dodati trenutnom izboru • 2 — uklanjati iz trenutnog izbora • 3 — birati unutar trenutnog izbora

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `qgis:selectbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbor po lokaciji

Pravi izbor u vektorskom sloju.

Kriterijum za izbor objekata zasniva se na prostornom odnosu između svakog objekta i objekata na dodatnom sloju.

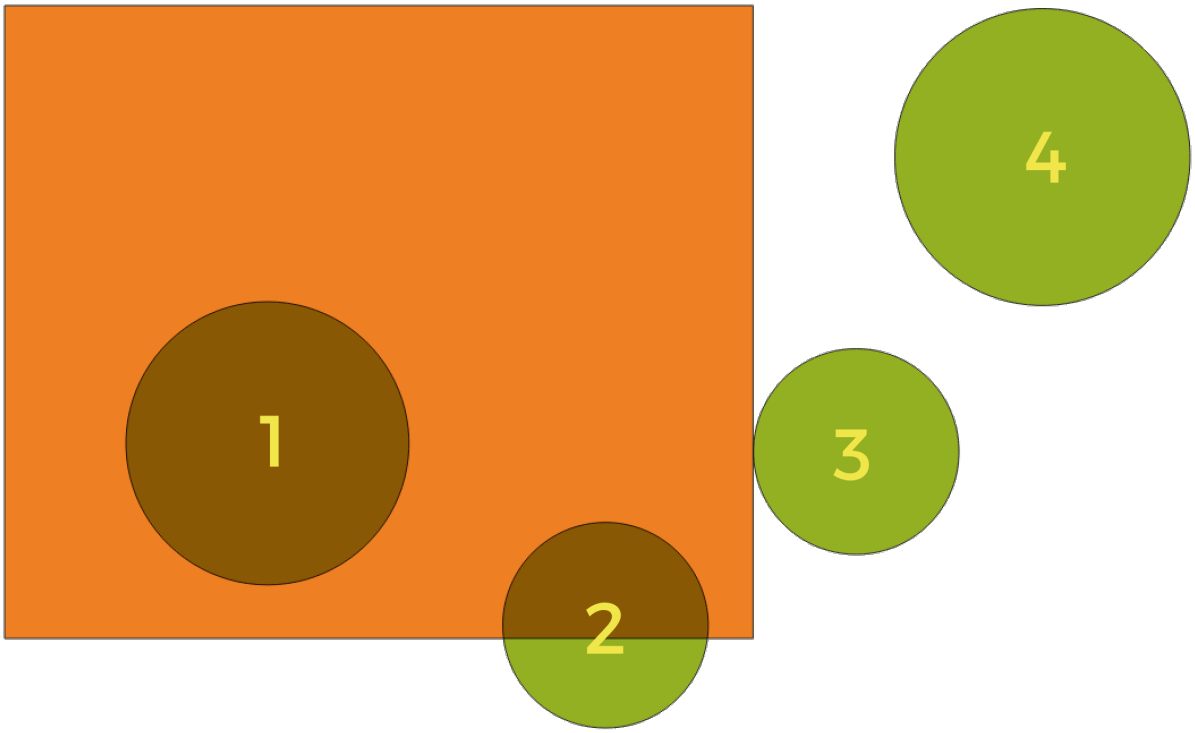
Podrazumevani meni: *Vektor ► Istraživačke alatke*

See also

Izdvajanje po lokaciji, *Izbor unutar rastojanja*

Istraživanje prostornih odnosa

Geometrijski predikati su logičke funkcije koje se koriste za utvrđivanje prostornog odnosa jednog objekta prema drugom, poređenjem da li i kako njihove geometrije dele deo prostora.



Слика24.176: Traženje prostornih odnosa između slojeva

Na osnovu gornje slike, tražimo zelene krugove prostornim poređenjem sa narandžastim pravougaonim objektom. Dostupni geometrijski predikati su:

Preseca
: Proverava da li jedna geometrija preseca drugu. Vraća 1 (tačno) ako se geometrije prostorno presecaju (dele bilo koji deo prostora - preklapaju se ili dodiruju), a 0 ako se ne presecaju. Na gornjoj slici, ovo vraća krugove 1, 2 i 3.

Sadrži
: Vraća 1 (tačno) ako i samo ako nijedna tačka geometrije b ne leži izvan geometrije a i bar jedna tačka unutrašnjosti geometrije b leži u unutrašnjosti geometrije a. Na slici se ne vraća nijedan krug, ali bi se vratio pravougaonik da tražite obrnuto, jer on u potpunosti sadrži krug 1. Ovo je suprotno od *su unutar*.

Razdvojeno
: Vraća 1 (tačno) ako geometrije ne dele nijedan deo prostora (bez preklapanja, bez dodirivanja). Vraća se samo krug 4.

Jednako
: Vraća 1 (tačno) ako i samo ako su geometrije potpuno iste. Neće se vratiti nijedan krug.

Dodiruje
: Proverava da li jedna geometrija dodiruje drugu. Vraća 1 (tačno) ako geometrije imaju bar jednu zajedničku tačku, ali im se unutrašnjosti ne presecaju. Vraća se samo krug 3.

Preklapa
Proverava da li se jedna geometrija preklapa sa drugom. Vraća 1 (tačno) ako geometrije dele prostor, iste su dimenzije, ali nisu potpuno sadržane jedna u drugoj. Vraća se samo krug 2.

Su unutar
Proverava da li je jedna geometrija unutar druge. Vraća 1 (tačno) ako je geometrija a potpuno unutar geometrije b. Vraća se samo krug 1.

Ukršta
Vraća 1 (tačno) ako zadate geometrije imaju neke, ali ne sve, zajedničke unutrašnje tačke i ako je samo ukrštanje niže dimenzije od najviše zadate geometrije. Na primer, linija koja ukršta poligon ukršta ga kao linija (tačno). Dve linije koje se ukrštaju ukrštaju se kao tačka (tačno). Dva poligona ukrštaju se kao poligon (netačno). Na slici se neće vratiti nijedan krug.

Parametri

Oznaka	Naziv	Tip	Opis
Izaberi objekte iz	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Kod kojih objekti (geometrijski predikat)	`PREDICATE`	[nabrajanje] [lista] Podrazumevano: [0]	Vrsta prostornog odnosa koju ulazni objekat treba da ima sa objektom preseka da bi mogao biti izabran. Jedno ili više od: • 0 — presecaju • 1 — sadrže • 2 — razdvojen • 3 — jednak • 4 — dodiruje • 5 — preklapa • 6 — je unutar • 7 — prelazi preko Ako je izabrano više od jednog uslova, bar jedan od njih (operacija OR) mora biti ispunjen da bi objekat bio izdvojen.
Poređenjem sa objektima iz	`INTERSECT`	[vektor: geometrija]	Vektorski sloj preseka
Izmeni trenutni izbor tako što će se	`METHOD`	[nabrajanje] Podrazumevano: 0	Način na koji se upravlja izborom algoritma. Jedno od: • 0 — praviti novi izbor • 1 — dodati trenutnom izboru • 2 — birati unutar trenutnog izbora • 3 — uklanjati iz trenutnog izbora

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `native:selectbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izbor unutar rastojanja

pravi izbor u vektorskom sloju. Objekti se biraju svuda gde se nalaze unutar zadatog maksimalnog rastojanja od objekata na dodatnom referentnom sloju.

See also

Izdvajanje unutar rastojanja, *Izbor po lokaciji*

Parametri

Oznaka	Naziv	Tip	Opis
Izaberi objekte iz	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj iz kojeg se biraju objekti
Poređenjem sa objektima iz	`REFERENCE`	[vektor: geometrija]	Vektorski sloj čija se blizina objekata koristi
Kod kojih su objekti unutar	`DISTANCE`	[numerički: decimalni] Podrazumevano: 100.0	Maksimalno rastojanje oko referentnih objekata za izbor ulaznih objekata
Izmeni trenutni izbor tako što će se	`METHOD`	[nabrajanje] Podrazumevano: 0	Način na koji se upravlja izborom algoritma. Jedno od: • 0 — praviti novi izbor • 1 — dodati trenutnom izboru • 2 — birati unutar trenutnog izbora • 3 — uklanjati iz trenutnog izbora

Izlazi

Oznaka	Naziv	Tip	Opis
Ulazni sloj	`INPUT`	[isto kao ulaz]	Ulazni sloj sa izabranim objektima

Python kod

ID algoritma: `native:selectwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.29 Vektorska tabela

Dodaj samouvećavajuće polje

Dodaje novo celobrojno polje vektorskom sloju, sa sekvencijalnom vrednošću za svaki objekat.

Ovo polje se može koristiti kao jedinstveni ID za objekte u sloju. Novi atribut se ne dodaje u ulazni sloj, već se umesto toga generiše novi sloj.

Može se zadati početna vrednost inkrementalne serije. Opciono, inkrementalna serija može se zasnivati na poljima za grupisanje, a može se zadati i redosled sortiranja objekata.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj.
Naziv polja	`FIELD_NAME`	[niska] Podrazumevano: 'AUTO'	Naziv polja sa samouvećavajućim vrednostima
Počni vrednosti od Opciono	`START`	[numerički: ceo broj] Podrazumevano: 0	Izaberite početni broj inkrementalnog brojanja
Vrednost modula Opciono	`MODULUS`	[numerički: ceo broj] Podrazumevano: 0	Zadavanje opcione vrednosti modula ponovo pokreće brojanje na START kad god vrednost polja dostigne vrednost modula. `0` znači bez ponovnog pokretanja.
Grupiši vrednosti po Opciono	`GROUP_FIELDS`	[polje tabele: bilo koji] [lista]	Izaberite polje(a) za grupisanje: umesto jednog brojanja za ceo sloj, zasebno brojanje se obrađuje za svaku vrednost koju vraća kombinacija ovih polja.
Izraz za sortiranje Opciono	`SORT_EXPRESSION`	[izraz]	Koristite izraz za sortiranje objekata u sloju, bilo globalno ili, ako je postavljeno, na osnovu polja za grupisanje.
Sortiraj rastuće	`SORT_ASCENDING`	[logička vrednost] Podrazumevano: True	Kada je postavljen `sort expression`, koristite ovu opciju za kontrolu redosleda kojim se objektima dodeljuju vrednosti.
Prvo sortiraj Null vrednosti	`SORT_NULLS_FIRST`	[logička vrednost] Podrazumevano: False	Kada je postavljen `sort expression`, koristite ovu opciju da biste postavili da li se *Null* vrednosti broje prve ili poslednje.
Uvećano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj sa samouvećavajućim poljem. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Uvećano	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa samouvećavajućim poljem

Python kod

ID algoritma: `native:addautoincrementalfield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Dodaj polje u atributsku tabelu

Dodaje novo polje vektorskom sloju.

Naziv i karakteristike atributa definišu se kao parametri.

Novi atribut se ne dodaje u ulazni sloj, već se umesto toga generiše novi sloj.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni sloj
Naziv polja	`FIELD_NAME`	[niska]	Naziv novog polja
Tip polja	`FIELD_TYPE`	[nabrajanje] Podrazumevano: 0	Tip novog polja. Možete birati između: • 0 — Ceo broj (32 bita) • 1 — Decimalni (double) • 2 — Tekst (string) • 3 — Logička vrednost • 4 — Datum • 5 — Vreme • 6 — Datum i vreme • 7 — Binarni objekat (BLOB) • 8 — Lista niski • 9 — Lista celih brojeva • 10 — Lista decimalnih (double)
Dužina polja	`FIELD_LENGTH`	[numerički: ceo broj] Podrazumevano: 10	Dužina polja
Preciznost polja	`FIELD_PRECISION`	[numerički: ceo broj] Podrazumevano: 0	Preciznost polja. Korisno kod tipa polja Float.
Alijas polja Opciono `Dodato u 3.32`	`FIELD_ALIAS`	[niska]	Postavite naziv koji se koristi kao alijas polja. Ne podržavaju ga svi tipovi formata.
Komentar polja Opciono `Dodato u 3.32`	`FIELD_COMMENT`	[niska]	Sačuvajte komentar koji opisuje polje. Ne podržavaju ga svi tipovi formata.
Dodato	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Dodato	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa dodatim novim poljem

Python kod

ID algoritma: `native:addfieldtoattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Dodaj polje indeksa jedinstvenih vrednosti

Uzima vektorski sloj i atribut i dodaje novo numeričko polje.

Vrednosti u ovom polju odgovaraju vrednostima u navedenom atributu, pa će objekti sa istom vrednošću atributa imati istu vrednost u novom numeričkom polju.

Time se pravi numerički ekvivalent navedenog atributa, koji definiše iste klase.

Novi atribut se ne dodaje u ulazni sloj, već se umesto toga generiše novi sloj.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni sloj.
Polje klase	`FIELD`	[polje tabele: bilo koji]	Objekti koji imaju istu vrednost za ovo polje dobiće isti indeks.
Naziv izlaznog polja	`FIELD_NAME`	[niska] Podrazumevano: 'NUM_FIELD'	Naziv novog polja koje sadrži indekse.
Sloj sa poljem indeksa	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Vektorski sloj sa numeričkim poljem koje sadrži indekse. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

наставак на следећој страни

Табела 24.821 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Rezime klasa	`SUMMARY_OUTPUT`	[vektor: tabela] Podrazumevano: `[Skip output]`	Zadajte tabelu koja sadrži rezime polja klase mapiranog na odgovarajuću jedinstvenu vrednost. *Jedno od*: • Preskoči izlaz • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Sloj sa poljem indeksa	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa numeričkim poljem koje sadrži indekse.
Rezime klasa	`SUMMARY_OUTPUT`	[vektor: tabela]	Tabela sa rezimeom polja klase mapiranog na odgovarajuću jedinstvenu vrednost.

Python kod

ID algoritma: `native:adduniquevalueindexfield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Dodaj X/Y polja u sloj

Dodaje X i Y (ili geografsku širinu/dužinu) polja tačkastom sloju. X/Y polja mogu se izračunati u CRS-u različitom od sloja (npr. pravljenje polja geografske širine/dužine za sloj u projektovanom CRS-u).

Omogućava *izmenu objekata na licu mesta* za tačkaste objekte

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: tačka]	Ulazni sloj.
Koordinatni sistem	`CRS`	[crs] Podrazumevano: „EPSG:4326”	Koordinatni referentni sistem koji se koristi za generisana x i y polja.
Prefiks polja Opciono	`PREFIX`	[niska]	Prefiks koji se dodaje nazivima novih polja radi izbegavanja sudara naziva sa poljima u ulaznom sloju.

наставак на следећој страни

Табела 24.823 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodata polja	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Dodata polja	`OUTPUT`	[vektor: tačka]	Izlazni sloj - istovetan ulaznom sloju, ali sa dva nova double polja, `x` i `y`.

Python kod

ID algoritma: `native:addxyfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Napredni Python kalkulator polja

Dodaje novi atribut vektorskom sloju, sa vrednostima koje su rezultat primene izraza na svaki objekat.

Izraz se definiše kao Python funkcija.

Упозорење

Ovaj algoritam je potencijalni bezbednosni rizik ako se izvrši sa neproverenim ulazima i može dovesti do oštećenja sistema ili curenja podataka.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Naziv polja rezultata	`FIELD_NAME`	[niska] Podrazumevano: 'NewField'	Naziv novog polja

наставак на следећој страни

Табела 24.825 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip polja	`FIELD_TYPE`	[nabrajanje] Podrazumevano: 0	Tip novog polja. Jedno od: • 0 — Ceo broj (32 bita) • 1 — Decimalni (double) • 2 — Tekst (string) • 3 — Logička vrednost • 4 — Datum • 5 — Vreme • 6 — Datum i vreme • 7 — Binarni objekat (BLOB) • 8 — Lista niski • 9 — Lista celih brojeva • 10 — Lista decimalnih (double)
Dužina polja	`FIELD_LENGTH`	[numerički: ceo broj] Podrazumevano: 10	Dužina polja
Preciznost polja	`FIELD_PRECISION`	[numerički: ceo broj] Podrazumevano: 3	Preciznost polja. Korisno kod tipa polja Float.
Globalni izraz Opciono	`GLOBAL`	[niska]	Kod u odeljku globalnog izraza izvršiće se samo jednom pre nego što kalkulator počne da prolazi kroz sve objekte ulaznog sloja. Zato je ovo pravo mesto za uvoz potrebnih modula ili za računanje promenljivih koje će se koristiti u kasnijim izračunavanjima.

Formula	`FORMULA`	[niska]	Python formula koja se izračunava. Primer: Za računanje površine ulaznog poligonskog sloja možete dodati:
			`value = $geom.area()`
			Da biste pristupili postojećem polju, koristite njegov naziv unutar uglastih zagrada. Na primer, za pretvaranje vrednosti polja „address“ u velika slova:
			`value = <address>.upper()`

Izračunato	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte vektorski sloj sa novim izračunatim poljem. Jedno od: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izračunato	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa novim izračunatim poljem

Python kod

ID algoritma: `qgis:advancedpythonfieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Odbaci polje(a)

Uzima vektorski sloj i generiše novi koji ima iste objekte, ali bez izabranih kolona.

See also

Zadrži polja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj iz kojeg se odbacuju polja
Polja za odbacivanje	`COLUMN`	[polje tabele: bilo koji] [lista]	Polje(a) za odbacivanje
Preostala polja	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj sa preostalim poljima. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Preostala polja	`OUTPUT`	[isto kao ulaz]	Vektorski sloj sa preostalim poljima

Python kod

ID algoritma: `native:deletecolumn`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Raščlani HStore polje

Pravi kopiju ulaznog sloja i dodaje novo polje za svaki jedinstveni ključ u HStore polju.

Očekivana lista polja je opciona lista razdvojena zarezima. Ako je ova lista zadata, dodaju se samo ta polja, a HStore polje se ažurira. Podrazumevano se dodaju svi jedinstveni ključevi.

PostgreSQL HStore je jednostavno skladište ključ-vrednost koje se koristi u PostgreSQL-u i GDAL-u (pri čitanju OSM datoteke sa poljem `other_tags`.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
HStore polje	`FIELD`	[polje tabele: bilo koji]	Polje(a) za odbacivanje
Očekivana lista polja razdvojenih zarezom Opciono	`EXPECTED_FIELDS`	[niska] Podrazumevano: ”	Lista polja za izdvajanje, razdvojenih zarezima. HStore polje će biti ažurirano uklanjanjem ovih ključeva.
Raščlanjeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Zadajte izlazni vektorski sloj. Jedno od: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Raščlanjeno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj

Python kod

ID algoritma: `native:explodehstorefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvoj binarno polje

Izdvaja sadržaj iz binarnog polja, čuvajući ga u pojedinačne datoteke. Nazivi datoteka mogu se generisati pomoću vrednosti uzetih iz atributa u izvornoj tabeli ili na osnovu složenijeg izraza.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj koji sadrži binarne podatke
Binarno polje	`FIELD`	[polje tabele: bilo koji]	Polje koje sadrži binarne podatke
Naziv datoteke	`FILENAME`	[izraz]	Tekst zasnovan na polju ili izrazu za imenovanje svake izlazne datoteke
Odredišni folder	`FOLDER`	[folder] Podrazumevano: `[Save to temporary folder]`	Folder u kojem se čuvaju izlazne datoteke. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Izlazi

Oznaka	Назив	Tip	Opis
Folder	`FOLDER`	[folder]	Folder koji sadrži izlazne datoteke.

Python kod

ID algoritma: `native:extractbinary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Kalkulator polja

Otvara kalkulator polja (pogledajte *Expressions*). Možete koristiti sve podržane izraze i funkcije.

Pravi se novi sloj sa rezultatom izraza.

Kalkulator polja je veoma koristan kada se koristi u *Dizajner modela*.

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj nad kojim se računa
Naziv izlaznog polja	`FIELD_NAME`	[niska]	Naziv polja za rezultate

наставак на следећој страни

Табела 24.833 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznog polja	`FIELD_TYPE`	[nabrajanje] Podrazumevano: 0	Tip polja. Jedno od: • 0 — Decimalni (double) • 1 — Ceo broj (32 bita) • 2 — Tekst (string) • 3 — Datum • 4 — Vreme • 5 — Datum i vreme • 6 — Logička vrednost • 7 — Binarni objekat (BLOB) • 8 — Lista niski • 9 — Lista celih brojeva • 10 — Lista decimalnih (double)
Širina izlaznog polja	`FIELD_LENGTH`	[numerički: ceo broj] Podrazumevano: 0	Dužina polja rezultata (minimum 0)
Preciznost polja	`FIELD_PRECISION`	[numerički: ceo broj] Podrazumevano: 0	Preciznost polja rezultata (minimum 0, maksimum 15)
Napravi novo polje	`NEW_FIELD`	[logička vrednost] Podrazumevano: True	Da li polje rezultata treba da bude novo polje
Formula	`FORMULA`	[izraz]	Formula koja se koristi za računanje rezultata
Izračunato	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Izračunato	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa izračunatim vrednostima polja

Python kod

ID algoritma: `native:fieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preuredi polja

Omogućava uređivanje strukture atributske tabele vektorskog sloja.

Poljima se mogu izmeniti tip i naziv, pomoću mapiranja polja.

Originalni sloj se ne menja. Generiše se novi sloj koji sadrži izmenjenu atributsku tabelu, u skladu sa zadatim mapiranjem polja.

Белешка

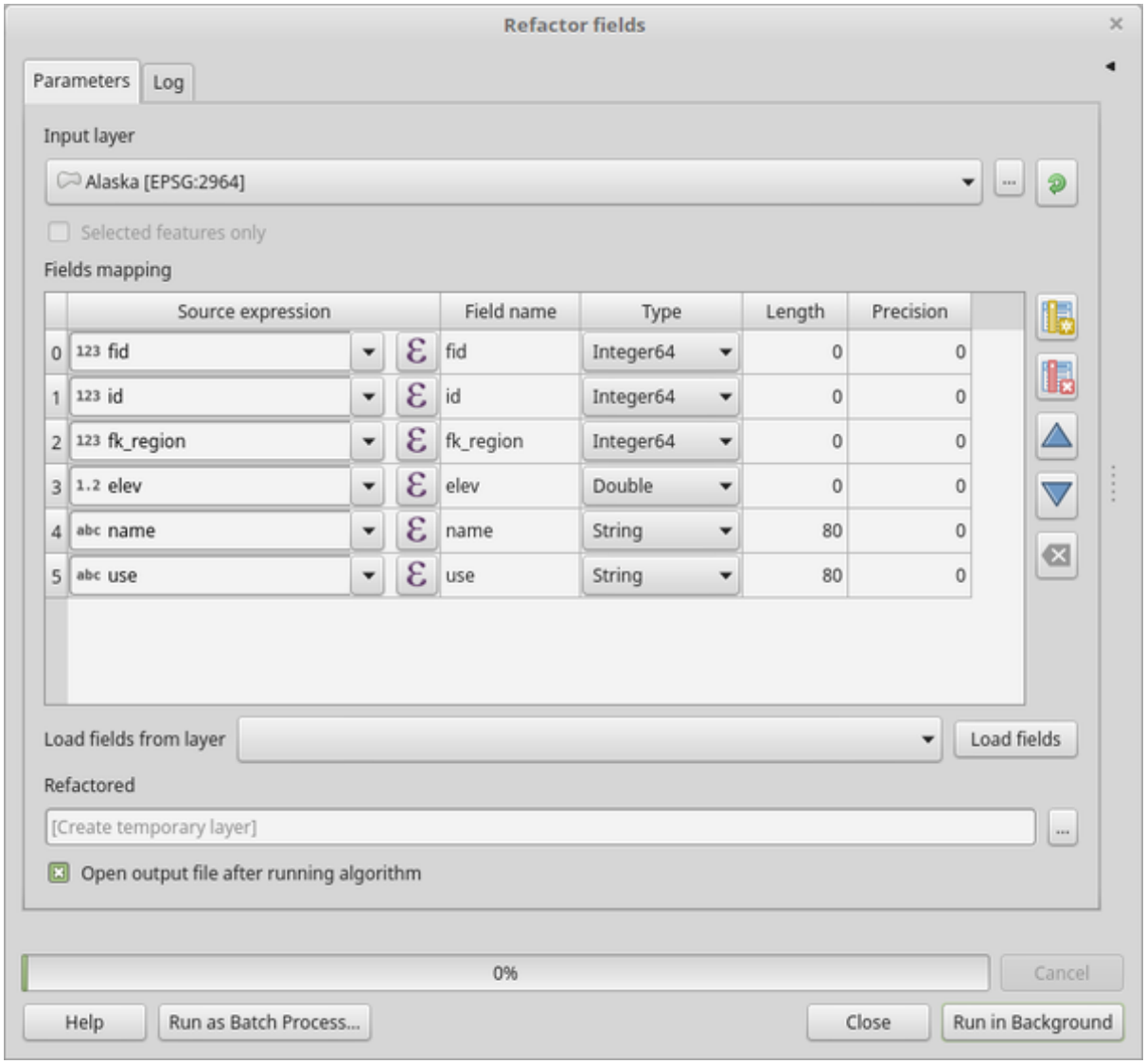
Pri korišćenju šablonskog sloja sa *ograničenjima* na poljima, informacija se prikazuje u vidžetu obojenom pozadinom i iskačućom pomoći. Tretirajte ovu informaciju kao savet tokom konfiguracije. Nikakva ograničenja neće biti dodata na izlazni sloj, niti će ih algoritam proveravati ili sprovoditi.

Algoritam Preuredi polja omogućava da se:

- Promene nazivi i tipovi polja
- Dodaju i uklone polja
- Preurede polja
- Izračunaju nova polja na osnovu izraza
- Učita lista polja sa drugog sloja

Белешка

Ovaj algoritam koristi merenja zasnovana na elipsoidu i poštuje tekuća *podešavanja elipsoida*.



Слика24.177: Dijalog Preuredi polja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Sloj koji se menja

наставак на следећој страни

Табела 24.835 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Mapiranje polja	`FIELDS_MAPPING`	[lista]	Lista izlaznih polja sa njihovim definicijama. Ugrađena tabela navodi sva polja izvornog sloja i omogućava vam da ih uređujete: • Kliknite na  da biste napravili novo polje. • Kliknite na  da biste uklonili polje. • Koristite  i  da biste promenili redosled izabranog polja. • Kliknite na  da biste vratili na podrazumevani prikaz. • Kliknite na  da biste obrnuli izbor u listi polja. Za svako polje koje želite ponovo da iskoristite, potrebno je da popunite sledeće opcije: ***Izvorni izraz* (`expression`) [izraz]** Polje ili izraz iz ulaznog sloja. ***Naziv* (`name`) [niska]** Naziv polja u izlaznom sloju. Podrazumevano se zadržava naziv ulaznog polja. ***Tip* (`type`) [nabrajanje]** Tip podataka izlaznog polja. Dostupni tipovi možda nisu kompatibilni sa provajderom izlaznog sloja. Jedno od: **Пажња** Za određene tipove polja, npr. liste, dodatni parametar `sub_type` pomaže u preciziranju konkretnog tipa podataka. Automatski se postavlja u grafičkom okruženju, ali može biti potreban ako algoritam pokrećete u Pythonu ili iz komandne linije. • 1 — Logička vrednost • 2 — Ceo broj (32-bitni) • 4 — Ceo broj (64-bitni) • 6 — Decimalni broj (double) • 9 — Lista celih brojeva • 9 — Lista celih brojeva (64--bitnih) • 9 — Lista decimalnih brojeva (double) • 10 — Tekst (niska) • 11 — Lista niski • 12 — Binarni objekat (BLOB) • 14 — Datum • 15 — Vreme • 16 — Datum i vreme ***Podtip* (`sub_type`) [nabrajanje]** Za određene tipove polja, npr. liste, ovaj parametar pomaže u preciziranju konkretnog `type` podata-

Табела 24.835 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Preuređeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Preuređeno	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa preuređenim poljima

Python kod

ID algoritma: `native:refactorfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preimenuj polje

Preimenuje postojeće polje vektorskog sloja.

Originalni sloj se ne menja. Generiše se novi sloj u kojem atributska tabela sadrži preimenovano polje.

See also

Preuredi polja

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Polje za preimenovanje	`FIELD`	[polje tabele: bilo koji]	Polje koje se menja
Novi naziv polja	`NEW_NAME`	[niska]	Novi naziv polja

наставак на следећој страни

Табела 24.837 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Preimenovano	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Preimenovano	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa preimenovanim poljem

Python kod

ID algoritma: `native:renametablefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Zadrži polja

Uzima vektorski sloj i generiše novi koji zadržava samo izabrana polja. Sva ostala polja biće odbačena.

See also

Odbaci polje(a)

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Polja za zadržavanje	`FIELDS`	[polje tabele: bilo koji] [lista]	Lista polja koja se zadržavaju u sloju
Zadržana polja	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create temporary layer]`	Specifikacija izlaznog sloja. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Zadržana polja	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa zadržanim poljima

Python kod

ID algoritma: `native:retainfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Tekst u broj sa pokretnim zarezom

Menja tip zadatog atributa u vektorskom sloju, pretvarajući tekstualni atribut koji sadrži numeričke niske u numerički atribut (npr. '1' u `1.0`).

Algoritam pravi novi vektorski sloj, tako da se izvorni ne menja.

Ako pretvaranje nije moguće, izabrana kolona imaće `NULL` vrednosti.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj.
Tekstualni atribut za pretvaranje u broj sa pokretnim zarezom	`FIELD`	[polje tabele: niska]	Niskovno polje ulaznog sloja koje se pretvara u polje sa pokretnim zarezom.
Broj sa pokretnim zarezom iz teksta	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Create Temporary Layer]`	Zadajte izlazni sloj. *Jedno od*: • Napravi privremeni sloj (`TEMPORARY_OUTPUT`) • Sačuvaj u datoteku… • Sačuvaj u Geopackage… • Sačuvaj u tabelu baze podataka… • Dodaj u sloj… Ovde se može promeniti i kodiranje datoteke.

Izlazi

Oznaka	Назив	Tip	Opis
Broj sa pokretnim zarezom iz teksta	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj sa niskovnim poljem pretvorenim u polje sa pokretnim zarezom

Python kod

ID algoritma: `qgis:texttofloat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.1.30 Vektorske pločice

Preuzimanje vektorskih pločica

`Dodato u 3.32`

Preuzima vektorske pločice ulaznog sloja vektorskih pločica i čuva ih u lokalnu datoteku vektorskih pločica.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektorske pločice]	Sloj vektorskih pločica iz koga se izdvajaju pločice
Obuhvat	`EXTENT`	[obuhvat]	Zadajte prostorni obuhvat područja koje se preuzima. Interno će biti proširen na umnožak veličine pločice. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Najviši nivo uvećanja koji se preuzima	`MAX_ZOOM`	[numerički: ceo broj] Podrazumevano: 10	Definiše koliko se uvećava i do kog nivoa se preuzimaju podaci iz pločica
Ograničenje broja pločica	`TILE_LIMIT`	[numerički: ceo broj] Podrazumevano: 100	Najveći broj pločica koje se preuzimaju, uzimajući u obzir nivoe uvećanja i obuhvat.
Izlaz	`OUTPUT`	[vektorske pločice] Podrazumevano: [Save to temporary file]	Specifikacija izlazne datoteke vektorskih pločica. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[vektorske pločice]	Lokalna datoteka vektorskih pločica u kojoj se čuvaju preuzete pločice.

Python kod

ID algoritma: `native:downloadvectortiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Upis vektorskih pločica (MBTiles)

Izvozi jedan ili više vektorskih slojeva u vektorske pločice, format podataka optimizovan za brzo iscrtavanje karte i malu veličinu podataka.

MBTiles je specifikacija za čuvanje podataka karte u pločicama unutar SQLite baza podataka, radi neposrednog korišćenja i prenosa. MBTiles datoteke poznate su kao skupovi pločica.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[vektor: bilo koji] [lista]	Lista slojeva koji se objedinjuju radi generisanja vektorskih pločica
Najniži nivo uvećanja	`MIN_ZOOM`	[numerički: ceo broj] Podrazumevano: 0	Najniži nivo uvećanja za koji skup pločica pruža podatke. Postavlja se između 0 i 24.
Najviši nivo uvećanja	`MAX_ZOOM`	[numerički: ceo broj] Podrazumevano: 3	Najviši nivo uvećanja za koji skup pločica pruža podatke. Postavlja se između 0 i 24.
Obuhvat Opciono	`EXTENT`	[obuhvat] Podrazumevano: Nije postavljeno	Najveći obuhvat iscrtanog područja karte. Granice moraju definisati područje pokriveno svim nivoima uvećanja.
Metapodaci: Naziv Opciono	`META_NAME`	[niska]	Naziv skupa pločica
Metapodaci: Opis Opciono	`METADATA_DESCRIPTION`	[niska]	Opis sadržaja skupa pločica
Metapodaci: Pripisivanje Opciono	`METADATA_ATTRIBUTION`	[niska]	Niska sa pripisivanjem, koja objašnjava izvore podataka i/ili stila za kartu.
Metapodaci: Verzija Opciono	`META_VERSION`	[niska]	Verzija skupa pločica. Odnosi se na reviziju samog skupa pločica, ne na MBTiles specifikaciju.
Metapodaci: Tip Opciono	`META_TYPE`	[niska]	Tip skupa pločica. Moguće vrednosti su `overlay` ili `baselayer`.

наставак на следећој страни

Табела 24.845 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Metapodaci: Centar Opciono	`META_CENTER`	[niska]	Centar (niska brojeva razdvojenih zarezom: geografska dužina, geografska širina i nivo uvećanja) podrazumevanog prikaza karte. Primer: `-122.1906,37.7599,11`
Odredišni MBTiles	`OUTPUT`	[vektorske pločice] Podrazumevano: [Save to temporary file]	Specifikacija izlazne MBTiles datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Odredišni MBTiles	`OUTPUT`	[datoteka]	Izlazna `.mbtiles` datoteka vektorskih pločica.

Python kod

ID algoritma: `native:writevectortiles_mbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Upis vektorskih pločica (XYZ)

Izvozi jedan ili više vektorskih slojeva u vektorske pločice, format podataka optimizovan za brzo iscrtavanje karte i malu veličinu podataka.

Parametri

Oznaka	Назив	Tip	Opis
Šablon datoteke	`XYZ_TEMPLATE`	[niska] Podrazumevano: '{z}/{x}/{y}.pbf'	Šablon za generisanje url-a vektorskih pločica
Ulazni slojevi	`INPUT`	[vektor: bilo koji] [lista]	Lista slojeva koji se objedinjuju radi generisanja vektorskih pločica
Najniži nivo uvećanja	`MIN_ZOOM`	[numerički: ceo broj] Podrazumevano: 0	Najniži nivo uvećanja za koji skup pločica pruža podatke. Postavlja se između 0 i 24.
Najviši nivo uvećanja	`MAX_ZOOM`	[numerički: ceo broj] Podrazumevano: 3	Najviši nivo uvećanja za koji skup pločica pruža podatke. Postavlja se između 0 i 24.
Obuhvat Opciono	`EXTENT`	[obuhvat] Podrazumevano: Nije postavljeno	Najveći obuhvat iscrtanog područja karte. Granice moraju definisati područje pokriveno svim nivoima uvećanja.

наставак на следећој страни

Табела 24.847 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT_DIRECTORY`	[folder] Podrazumevano: [Save to temporary folder]	Specifikacija izlaznog foldera vektorskih pločica. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT_DIRECTORY`	[folder]	Folder koji sadrži različite podskupove datoteka vektorskih pločica (`.pbf`) smeštene u podfoldere koji odgovaraju nivoima uvećanja.

Python kod

ID algoritma: `native:writevectortiles_xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2 Dobavljač GDAL algoritma

GDAL (Biblioteka za apstrakciju geoprostornih podataka) je biblioteka za prevod rasterskih i vektorskih geoprostornih formata podataka. Algoritmi u okviru za obradu podataka su izvedeni iz GDAL rasterskih programa i GDAL vektorskih programa.

24.2.1 Analiza rastera

Ekspozicija

Generiše kartu ekspozicije iz bilo kog visinskog rastera koji GDAL podržava. Ekspozicija je strana sveta ka kojoj je nagib okrenut. Pikseli će imati vrednost od 0-360° mereno u stepenima od severa, što označava azimut. Na severnoj polulopti, severna strana nagiba je često u senci (mali azimut od 0°-90°), dok južna strana prima više sunčevog zračenja (veći azimut od 180°-270°).

Ovaj algoritam je izveden iz GDAL DEM alatke.

Podrazumevani meni: *Raster ► Analiza*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina

наставак на следећој страни

Табела 24.849 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi kao nadmorska visina
Vrati trigonometrijski ugao umesto azimuta	`TRIG_ANGLE`	[logička vrednost] Podrazumevano: False	Aktiviranje trigonometrijskog ugla daje drugačije kategorije: 0° (istok), 90° (sever), 180° (zapad), 270° (jug).
Vrati 0 za ravno umesto -9999	`ZERO_FLAT`	[logička vrednost] Podrazumevano: False	Aktiviranje ove opcije umetnuće vrednost 0 umesto vrednosti -9999 na ravnim oblastima.
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Koristi Zevenbergen&Thorne formulu umesto Hornove	`ZEVENBERGEN`	[logička vrednost] Podrazumevano: False	Aktivira Zevenbergen&Thorne formulu za glatke predele
Ekspozicija	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Ekspozicija	`OUTPUT`	[raster]	Izlazni raster sa vrednostima ugla u stepenima

Python kod

ID algoritma: `gdal:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

Reljef u boji

Generiše kartu reljefa u boji iz bilo kog visinskog rastera koji GDAL podržava. Reljefi u boji mogu se posebno koristiti za prikaz nadmorskih visina. Algoritam daje raster sa 4 kanala, sa vrednostima izračunatim iz visine i tekstualne konfiguracione datoteke boja. Podrazumevano se boje između zadatih vrednosti visine glatko stapaju, a rezultat je lep obojeni visinski raster.

Ovaj algoritam je izveden iz GDAL DEM alatke.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi kao nadmorska visina
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Konfiguraciona datoteka boja	`COLOR_TABLE`	[datoteka]	Tekstualna konfiguraciona datoteka boja
Režim podudaranja	`MATCH_MODE`	[nabrajanje] Podrazumevano: 2	Jedno od: • 0 — Koristi strogo podudaranje boja • 1 — Koristi najbliže RGBA četvorke • 2 — Koristi glatko stopljene boje
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Reljef u boji	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Reljef u boji	`OUTPUT`	[raster]	Izlazni raster sa 4 kanala

Python kod

ID algoritma: `gdal:colorrelief`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Popuni NoData

Popunjava rasterske oblasti sa NoData vrednostima interpolacijom sa ivica. Vrednosti za NoData oblasti računaju se iz okolnih vrednosti piksela pomoću inverznog ponderisanja rastojanja. Nakon interpolacije vrši se zaglađivanje rezultata. Ulaz može biti bilo koji rasterski sloj koji GDAL podržava. Ovaj algoritam je uopšteno pogodan za interpolaciju oblasti koje nedostaju kod rastera koji se prilično neprekidno menjaju (kao što su npr. visinski modeli). Pogodan je i za popunjavanje malih rupa i pukotina kod nepravilnije promenljivih slika (kao što su aerofotografije). Uopšteno nije naročito dobar za interpolaciju rastera iz retkih tačkastih podataka.

Ovaj algoritam je izveden iz GDAL fillnodata alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal nad kojim se radi. NoData vrednosti moraju biti predstavljene vrednošću 0.
Maksimalno rastojanje (u pikselima) za traženje vrednosti za interpolaciju	`DISTANCE`	[numerički: ceo broj] Podrazumevano: 10	Broj piksela za pretragu u svim pravcima radi pronalaženja vrednosti iz kojih se interpolira
Broj iteracija zaglađivanja koje se pokreću nakon interpolacije	`ITERATIONS`	[numerički: ceo broj] Podrazumevano: 0	Broj prolaza filtera 3x3 koji se pokreću (0 ili više) radi zaglađivanja rezultata interpolacije.
Maska ispravnosti	`MASK_LAYER`	[raster]	Rasterski sloj koji definiše oblasti koje se popunjavaju.
Popunjeno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.855 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komандne linije

Izlazi

Oznaka	Назив	Tip	Opis
Popunjeno	`OUTPUT`	[raster]	Izlazni raster

Python kod

ID algoritma: `gdal:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (metrike podataka)

Računa neke metrike podataka pomoću zadatog prozora i geometrije izlazne mreže.

Ovaj algoritam je izveden iz GDAL grid alatke.

Podrazumevani meni: *Raster ► Analiza*

наставак на следећој страни

Табела 24.857 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

See also

GDAL grid tutorijal

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Metrika podataka koja se koristi	`METRIC`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — Minimum, minimalna vrednost pronađena u pretražnoj elipsi čvora mreže • 1 — Maksimum, maksimalna vrednost pronađena u pretražnoj elipsi čvora mreže • 2 — Raspon, razlika između minimalne i maksimalne vrednosti pronađene u pretražnoj elipsi čvora mreže • 3 — Broj, broj tačaka podataka pronađenih u pretražnoj elipsi čvora mreže • 4 — Prosečno rastojanje, prosečno rastojanje između čvora mreže (centra pretražne elipse) i svih tačaka podataka pronađenih u pretražnoj elipsi čvora mreže • 5 — Prosečno rastojanje između tačaka, prosečno rastojanje između tačaka podataka pronađenih u pretražnoj elipsi čvora mreže. Rastojanje između svakog para tačaka unutar elipse se računa, a prosek svih rastojanja postavlja se kao vrednost čvora mreže
Prvi poluprečnik pretražne elipse	`RADIUS_1`	[numerički: decimalni] Podrazumevano: 0.0	Prvi poluprečnik (X osa ako je ugao rotacije 0) pretražne elipse
Drugi poluprečnik pretražne elipse	`RADIUS_2`	[numerički: decimalni] Podrazumevano: 0.0	Drugi poluprečnik (Y osa ako je ugao rotacije 0) pretražne elipse
Ugao rotacije pretražne elipse u stepenima (suprotno od kazaljke na satu)	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije elipse u stepenima. Elipsa rotirana suprotno od kazaljke na satu.

наставак на следећој страни

Табела 24.857 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Minimalni broj tačaka podataka koje se koriste	`MIN_POINTS`	[numerički: decimalni] Podrazumevano: 0.0	Minimalni broj tačaka podataka za usrednjavanje. Ako se pronađe manji broj tačaka, čvor mreže se smatra praznim i biće popunjen NoData oznakom.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka
Interpolirano (metrike podataka)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

наставак на следећој страни

Табела 24.858 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (metrike podataka)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:griddatametrics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (IDW sa pretragom najbližeg suseda)

Računa mrežu inverznog rastojanja na stepen kombinovanu sa metodom najbližeg suseda. Idealno kada je potreban maksimalni broj tačaka podataka koje se koriste.

Ovaj algoritam je izveden iz GDAL grid alatke.

See also

GDAL grid tutorijal

Parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Stepen ponderisanja	`POWER`	[numerički: decimalni] Podrazumevano: 2.0	Stepen ponderisanja
Zaglađivanje	`SMOOTHING`	[numerički: decimalni] Podrazumevano: 0.0	Parametar zaglađivanja
Poluprečnik pretražnog kruga	`RADIUS`	[numerički: decimalni] Podrazumevano: 1.0	Poluprečnik pretražnog kruga
Maksimalni broj tačaka podataka koje se koriste	`MAX_POINTS`	[numerički: ceo broj] Podrazumevano: 12	Ne traži više tačaka od ovog broja.
Minimalni broj tačaka podataka koje se koriste	`MIN_POINTS`	[numerički: ceo broj] Podrazumevano: 0	Minimalni broj tačaka podataka za usrednjavanje. Ako se pronađe manji broj tačaka, čvor mreže se smatra praznim i biće popunjen NoData oznakom.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka
Interpolirano (IDW sa NN pretragom)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.861 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (IDW sa NN pretragom)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:gridinversedistancenearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (inverzno rastojanje na stepen)

Metod mreže inverznog rastojanja na stepen je interpolator ponderisanog proseka.

Treba da zadate ulazne nizove sa rasejanim vrednostima podataka, uključujući koordinate svake tačke podataka, i geometriju izlazne mreže. Funkcija će izračunati interpoliranu vrednost za zadatu poziciju u izlaznoj mreži.

Ovaj algoritam je izveden iz GDAL grid alatke.

Podrazumevani meni: *Raster ► Analiza*

See also

GDAL grid tutorijal

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Stepen ponderisanja	`POWER`	[numerički: decimalni] Podrazumevano: 2.0	Stepen ponderisanja
Zaglađivanje	`SMOOTHING`	[numerički: decimalni] Podrazumevano: 0.0	Parametar zaglađivanja
Prvi poluprečnik pretražne elipse	`RADIUS_1`	[numerički: decimalni] Podrazumevano: 0.0	Prvi poluprečnik (X osa ako je ugao rotacije 0) pretražne elipse
Drugi poluprečnik pretražne elipse	`RADIUS_2`	[numerički: decimalni] Podrazumevano: 0.0	Drugi poluprečnik (Y osa ako je ugao rotacije 0) pretražne elipse
Ugao rotacije pretražne elipse u stepenima (suprotno od kazaljke na satu)	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije elipse u stepenima. Elipsa rotirana suprotno od kazaljke na satu.
Maksimalni broj tačaka podataka koje se koriste	`MAX_POINTS`	[numerički: ceo broj] Podrazumevano: 0	Ne traži više tačaka od ovog broja.
Minimalni broj tačaka podataka koje se koriste	`MIN_POINTS`	[numerički: ceo broj] Podrazumevano: 0	Minimalni broj tačaka podataka za usrednjavanje. Ako se pronađe manji broj tačaka, čvor mreže se smatra praznim i biće popunjen NoData oznakom.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka

наставак на следећој страни

Табела 24.863 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Interpolirano (IDW)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (IDW)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:gridinversedistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (linearna)

Linearni metod vrši linearnu interpolaciju tako što računa Delonijevu triangulaciju oblaka tačaka, pronalazi u kom trouglu triangulacije se tačka nalazi i vrši linearnu interpolaciju iz njenih baricentričnih koordinata unutar trougla. Ako tačka nije ni u jednom trouglu, u zavisnosti od poluprečnika, algoritam će koristiti vrednost najbliže tačke ili NoData vrednost.

Ovaj algoritam je izveden iz GDAL grid alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Rastojanje pretrage	`RADIUS`	[numerički: decimalni] Podrazumevano: -1.0	U slučaju da tačka koja se interpolira ne staje u trougao Delonijeve triangulacije, koristi to maksimalno rastojanje za pretragu najbližeg suseda, ili u suprotnom koristi NoData. Ako je postavljeno na `-1`, rastojanje pretrage je beskonačno. Ako je postavljeno na `0`, koristiće se NoData vrednost.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka
Interpolirano (linearno)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju

наставак на следећој страни

Табела 24.867 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (linearno)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:gridlinear`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (pokretni prosek)

Pokretni prosek je jednostavan algoritam usrednjavanja podataka. Koristi pokretni prozor eliptičnog oblika za pretragu vrednosti i usrednjava sve tačke podataka unutar prozora. Pretražna elipsa se može rotirati za zadati ugao, centar elipse nalazi se u čvoru mreže. Takođe se može postaviti minimalni broj tačaka podataka za usrednjavanje; ako u prozoru nema dovoljno tačaka, čvor mreže se smatra praznim i biće popunjen zadatom NoData vrednošću.

Ovaj algoritam je izveden iz GDAL grid alatke.

Podrazumevani meni: *Raster* ► *Analiza*

See also

GDAL grid tutorijal

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Prvi poluprečnik pretražne elipse	`RADIUS_1`	[numerički: decimalni] Podrazumevano: 0.0	Prvi poluprečnik (X osa ako je ugao rotacije 0) pretražne elipse
Drugi poluprečnik pretražne elipse	`RADIUS_2`	[numerički: decimalni] Podrazumevano: 0.0	Drugi poluprečnik (Y osa ako je ugao rotacije 0) pretražne elipse
Ugao rotacije pretražne elipse u stepenima (suprotno od kazaljke na satu)	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije elipse u stepenima. Elipsa rotirana suprotno od kazaljke na satu.
Minimalni broj tačaka podataka koje se koriste	`MIN_POINTS`	[numerički: ceo broj] Podrazumevano: 0	Minimalni broj tačaka podataka za usrednjavanje. Ako se pronađe manji broj tačaka, čvor mreže se smatra praznim i biće popunjen NoData oznakom.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka
Interpolirano (pokretni prosek)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (pokretni prosek)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:gridaverage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Mreža (najbliži sused)

Metod najbližeg suseda ne vrši nikakvu interpolaciju ili zaglađivanje, već samo uzima vrednost najbliže tačke pronađene u pretražnoj elipsi čvora mreže i vraća je kao rezultat. Ako se ne pronađe nijedna tačka, vratiće se zadata NoData vrednost.

Ovaj algoritam je izveden iz GDAL grid alatke.

Podrazumevani meni: *Raster* ► *Analiza*

See also

GDAL grid tutorijal

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Tačkasti sloj	`INPUT`	[vektor: tačka]	Ulazni tačkasti vektorski sloj
Prvi poluprečnik pretražne elipse	`RADIUS_1`	[numerički: decimalni] Podrazumevano: 0.0	Prvi poluprečnik (X osa ako je ugao rotacije 0) pretražne elipse
Drugi poluprečnik pretražne elipse	`RADIUS_2`	[numerički: decimalni] Podrazumevano: 0.0	Drugi poluprečnik (Y osa ako je ugao rotacije 0) pretražne elipse
Ugao rotacije pretražne elipse u stepenima (suprotno od kazaljke na satu)	`ANGLE`	[numerički: decimalni] Podrazumevano: 0.0	Ugao rotacije elipse u stepenima. Elipsa rotirana suprotno od kazaljke na satu.
NoData	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	NoData oznaka za popunjavanje praznih tačaka
Interpolirano (najbliži sused)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Z vrednost iz polja Opciono	`Z_FIELD`	[polje tabele: numerički]	Polje za interpolaciju

наставак на следећој страни

Табела 24.873 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Interpolirano (najbliži sused)	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:gridnearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje

NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Osenčeni reljef

Daje raster sa lepim efektom osenčenog reljefa. Veoma je koristan za vizuelizaciju terena. Opciono možete zadati azimut i visinu izvora svetlosti, faktor vertikalnog preuveličavanja i faktor skaliranja radi uzimanja u obzir razlika između vertikalnih i horizontalnih jedinica.

Ovaj algoritam je izveden iz GDAL DEM alatke .

Podrazumevani meni: *Raster ► Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal koji sadrži informaciju o visini
Z faktor (vertikalno preuveličavanje)	`Z_FACTOR`	[numerički: decimalni] Podrazumevano: 1.0	Faktor preuveličava visinu izlaznog visinskog rastera
Razmera (odnos vertikalnih i horizontalnih jedinica)	`SCALE`	[numerički: decimalni] Podrazumevano: 1.0	Odnos vertikalnih i horizontalnih jedinica
Azimut svetlosti	`AZIMUTH`	[numerički: decimalni] Podrazumevano: 315.0	Definiše azimut svetlosti koja obasjava visinski raster u stepenima. Ako dolazi sa vrha rastera vrednost je 0, ako dolazi sa istoka 90 itd.
Visina svetlosti	`ALTITUDE`	[numerički: decimalni] Podrazumevano: 45.0	Definiše visinu svetlosti u stepenima. 90 ako svetlost dolazi iznad visinskog rastera, 0 ako je svetlost pod oštrim uglom.
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Koristi Zevenbergen&Thorne formulu (umesto Hornove)	`ZEVENBERGEN`	[logička vrednost] Podrazumevano: False	Aktivira Zevenbergen&Thorne formulu za glatke predele
Kombinovano senčenje	`COMBINED`	[logička vrednost] Podrazumevano: False	
Višesmerno senčenje	`MULTIDIRECTIONAL`	[logička vrednost] Podrazumevano: False	
Osenčeni reljef	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj sa interpoliranim vrednostima. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Osenčeni reljef	`OUTPUT`	[raster]	Izlazni raster sa interpoliranim vrednostima

Python kod

ID algoritma: `gdal:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Skoro crno

Pretvara skoro crne/bele granice u crne.

Ovaj algoritam će skenirati sliku i pokušati da sve piksele koji su skoro ili tačno crni, beli ili jedne ili više prilagođenih boja oko oboda postavi na crne ili bele. Ovo se često koristi za „popravku“ aerofotografija sa gubitkom kompresije, kako bi se obojeni pikseli mogli tretirati kao prozirni pri pravljenju mozaika.

Ovaj algoritam je izveden iz GDAL nearblack alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina

наставак на следећој страни

Табела 24.878 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Koliko daleko od crne (bele)	`NEAR`	[numerički: ceo broj] Podrazumevano: 15	Izaberite koliko daleko od crne, bele ili prilagođenih boja vrednosti piksela mogu biti a da se i dalje smatraju skoro crnim, belim ili prilagođene boje.
Traži skoro bele piksele umesto skoro crnih	`WHITE`	[logička vrednost] Podrazumevano: False	Traži skoro bele (255) piksele umesto skoro crnih piksela
Skoro crno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Skoro crno	`OUTPUT`	[raster]	Izlazni raster

Python kod

ID algoritma: `gdal:nearblack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Blizina (rastojanje u rasteru)

Generiše rastersku kartu blizine koja označava rastojanje od centra svakog piksela do centra najbližeg piksela označenog kao ciljni piksel. Ciljni pikseli su oni u izvornom rasteru čija je vrednost piksela u skupu vrednosti ciljnih piksela.

Ovaj algoritam je izveden iz GDAL proximity alatke.

Podrazumevani meni: *Raster ► Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal koji sadrži informaciju o visini
Lista ciljnih piksela Opciono	`VALUES`	[niska] Podrazumevano: ”	Lista vrednosti piksela u izvornoj slici, razdvojenih zarezima, koji se smatraju ciljnim pikselima. Ako nije zadata, svi pikseli različiti od nule smatraće se ciljnim pikselima.
Jedinice rastojanja	`UNITS`	[nabrajanje] Podrazumevano: 1	Označava da li generisana rastojanja treba da budu u pikselskim ili georeferenciranim koordinatama. Jedno od: • 0 — Georeferencirane koordinate • 1 — Pikselske koordinate
Maksimalno rastojanje koje se generiše Opciono	`MAX_DISTANCE`	[numerički: decimalni] Podrazumevano: 0.0	Maksimalno rastojanje koje se generiše. NoData vrednost će se koristiti za piksele izvan ovog rastojanja. Rastojanje se tumači prema vrednosti parametra *Jedinice rastojanja*.
Vrednost koja se primenjuje na piksele unutar maksimalnog rastojanja Opciono	`REPLACE`	[numerički: decimalni] Podrazumevano: 0.0	Zadajte vrednost koja se primenjuje na sve piksele koji su unutar maksimalnog rastojanja od ciljnih piksela (uključujući ciljne piksele) umesto vrednosti rastojanja.
NoData vrednost koja se koristi za odredišni raster blizine Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	Zadajte NoData vrednost koja se koristi za piksele izvan maksimalnog rastojanja. Ako NoData vrednost nije zadata, izlazni kanal će biti upitan za svoju NoData vrednost. Ako izlazni kanal nema NoData vrednost, koristiće se vrednost `65535`.
Karta blizine	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.882 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Karta blizine	`OUTPUT`	[raster]	Izlazni raster

Python kod

ID algoritma: `gdal:proximity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Hrapavost

Daje jednokanalni raster sa vrednostima izračunatim iz visine. Hrapavost je stepen nepravilnosti površine. Računa se kao najveća razlika između centralnog piksela i njegovih okolnih ćelija. Određivanje hrapavosti igra ulogu u analizi visinskih podataka terena, korisno je za proračune morfologije reka, u klimatologiji i fizičkoj geografiji uopšte.

Ovaj algoritam je izveden iz GDAL DEM alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi kao nadmorska visina
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Hrapavost	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Hrapavost	`OUTPUT`	[raster]	Jednokanalni izlazni raster hrapavosti. Vrednost -9999 se koristi kao NoData vrednost.

Python kod

ID algoritma: `gdal:roughness`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Prosej

Uklanja rasterske poligone manje od zadate granične veličine (u pikselima) i zamenjuje ih vrednošću piksela najvećeg susednog poligona. Korisno je ako na rasterskoj karti imate veliki broj malih oblasti.

Ovaj algoritam je izveden iz GDAL sieve alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Prag	`THRESHOLD`	[numerički: ceo broj] Podrazumevano: 10	Uklanjaju se samo rasterski poligoni manji od ove veličine
Koristi 8--povezanost	`EIGHT_CONNECTEDN`	[logička vrednost] Podrazumevano: False	Ako je False, kao susedi se smatraju samo (4) piksela koji povezuju ivice horizontalno ili vertikalno. Ako je True, svaki piksel koji dodiruje ivicu ili ugao smatra se susedom.
Ne koristi podrazumevanu masku ispravnosti za ulazni kanal	`NO_MASK`	[logička vrednost] Podrazumevano: False	
Maska ispravnosti Opciono	`MASK_LAYER`	[raster]	Maska ispravnosti koja se koristi umesto podrazumevane
Prosejano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Prosejano	`OUTPUT`	[raster]	Izlazni rasterski sloj.

Python kod

ID algoritma: `gdal:sieve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Nagib

Generiše kartu nagiba iz bilo kog visinskog rastera koji GDAL podržava. Nagib je ugao nagnutosti u odnosu na horizontalu. Imate mogućnost da zadate tip vrednosti nagiba koji želite: nagib u stepenima ili u procentima.

Ovaj algoritam je izveden iz GDAL DEM alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Kanal koji sadrži informaciju o visini
Odnos vertikalnih i horizontalnih jedinica	`SCALE`	[numerički: decimalni] Podrazumevano: 1.0	Odnos vertikalnih i horizontalnih jedinica
Izrazi nagib u procentima (umesto u stepenima)	`AS_PERCENT`	[logička vrednost] Podrazumevano: False	Izrazi nagib u procentima umesto u stepenima
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Koristi Zevenbergen&Thorne formulu (umesto Hornove)	`ZEVENBERGEN`	[logička vrednost] Podrazumevano: False	Aktivira Zevenbergen&Thorne formulu za glatke predele
Nagib	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

наставак на следећој страни

Табела 24.891 – настављено са претходне стране

Oznaka	Назив	Tip	Opis

Izlazi

Oznaka	Назив	Tip	Opis
Nagib	`OUTPUT`	[raster]	Izlazni raster

Python kod

ID algoritma: `gdal:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Indeks razuđenosti terena (TRI)

Daje jednokanalni raster sa vrednostima izračunatim iz visine. TRI je skraćenica za Indeks razuđenosti terena, koji se definiše kao srednja razlika između centralnog piksela i njegovih okolnih ćelija.

Ovaj algoritam je izveden iz GDAL DEM alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi kao nadmorska visina
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Indeks razuđenosti terena	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ''	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Indeks razuđenosti terena	`OUTPUT`	[raster]	Izlazni raster razuđenosti. Vrednost -9999 se koristi kao NoData vrednost.

Python kod

ID algoritma: `gdal:triterrainruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Indeks topografskog položaja (TPI)

Daje jednokanalni raster sa vrednostima izračunatim iz visine. TPI je skraćenica za Indeks topografskog položaja, koji se definiše kao razlika između centralnog piksela i srednje vrednosti njegovih okolnih ćelija.

Ovaj algoritam je izveden iz GDAL DEM alatke.

Podrazumevani meni: *Raster* ► *Analiza*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi za vrednosti visine
Izračunaj ivice	`COMPUTE_EDGES`	[logička vrednost] Podrazumevano: False	Generiše ivice iz visinskog rastera
Indeks razuđenosti terena	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATI-ON_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

Izlazi

Oznaka	Назив	Tip	Opis
Indeks razuđenosti terena	`OUTPUT`	[raster]	Izlazni raster.

Python kod

ID algoritma: `gdal:tpitopographicpositionindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2.2 Pretvaranje rastera

gdal2xyz

Pretvara rasterske podatke u XYZ ASCII format datoteke.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Rasterski sloj koji se pretvara
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal koji želite da pretvorite

наставак на следећој страни

Табела 24.899 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Ulazna vrednost piksela koja se tretira kao NoData Opciono	`NODATA_INPUT`	[numerički: ceo broj] Podrazumevano: Nije postavljeno	Ulazna vrednost piksela koja se tretira kao „NoData” **Пажња** Korišćenje ovog parametra zahteva QGIS instaliran sa GDAL >= 3.7 (pogledajte meni *Pomoć* ► *O programu*).
Odredišna NoData vrednost Opciono	`NODATA_OUTPUT`	[numerički: ceo broj] Podrazumevano: Nije postavljeno	Dodeli zadatu „NoData” vrednost izlazu. **Пажња** Korišćenje ovog parametra zahteva QGIS instaliran sa GDAL >= 3.7 (pogledajte meni *Pomoć* ► *O programu*).
Ne upisuj NoData vrednosti u izlaz	`SKIP_NODATA`	[logička vrednost] Podrazumevano: False	Ne upisuj „NoData” vrednosti u izlaz. **Пажња** Korišćenje ovog parametra zahteva QGIS instaliran sa GDAL >= 3.3 (pogledajte meni *Pomoć* ► *O programu*).
Izlaz kao vrednosti razdvojene zarezom	`CSV`	[logička vrednost] Podrazumevano: False	Određuje da li izlazna datoteka treba da bude tipa vrednosti razdvojenih zarezom (csv).
XYZ ASCII datoteka	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
XYZ ASCII datoteka	`INPUT`	[tabela]	Datoteka tabele koja sadrži vrednosti izvezene iz rasterskog kanala.

Python kod

ID algoritma: `gdal:gdal2xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

PCT u RGB

Pretvara 8-bitnu paletiranu sliku u 24-bitni RGB. Pseudobojeni kanal iz ulazne datoteke pretvara u RGB datoteku željenog formata.

Ovaj algoritam je izveden iz GDAL pct2rgb alatke.

Podrazumevani meni: *Raster ► Pretvaranje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazna 8-bitna rasterska slika
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal koji želite da pretvorite
Napravi RGBA datoteku	`RGBA`	[logička vrednost] Podrazumevano: False	Određuje da li izlazna datoteka treba da bude tipa RGBA.
PCT u RGB	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Specifikacija izlazne datoteke. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
PCT u RGB	`OUTPUT`	[raster]	24-bitna RGB rasterska slika

Python kod

ID algoritma: `gdal:pcttorgb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poligonizuj (raster u vektor)

Pravi vektorske poligone za sve povezane oblasti piksela u rasteru koji dele istu vrednost piksela. Svaki poligon se pravi sa atributom koji označava vrednost piksela tog poligona.

Ovaj algoritam je izveden iz GDAL polygonize alatke.

Podrazumevani meni: *Raster ► Pretvaranje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: Prvi kanal ulaznog sloja	Ako je raster višekanalni, izaberite kanal koji želite da koristite
Naziv polja koje se pravi	`FIELD`	[niska] Podrazumevano: 'DN'	Zadajte naziv polja za atribute povezanih oblasti.
Koristi 8- -povezanost	`EIGHT_CONNECTEDN`	[logička vrednost] Podrazumevano: False	Ako nije uključeno, rasterske ćelije moraju imati zajedničku ivicu da bi se smatrale povezanim (*4-povezane*). Ako je uključeno, i rasterske ćelije koje se dodiruju smatraju se povezanim (*8-povezane*).
Vektorizovano	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog (poligonskog) vektorskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Vektorizovano	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj

Python kod

ID algoritma: `gdal:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Preuredi kanale

Pravi novi raster koristeći izabrane kanale zadatog rasterskog sloja. Algoritam takođe omogućava promenu redosleda kanala u novonapravljenom rasteru.

Ovaj algoritam je izveden iz GDAL translate alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Izabrani kanal(i)	`BANDS`	[rasterski kanal] [lista] Podrazumevano: Nije postavljeno	Uređena lista kanala koji se koriste za pravljenje novog rastera
Pretvoreno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.907 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 0	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Koristi tip podataka ulaznog sloja • 1 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 2 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 3 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 6 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 7 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 8 — CInt16 (Kompleksni Int16) • 9 — CInt32 (Kompleksni Int32) • 10 — CFloat32 (Kompleksni Float32) • 11 — CFloat64 (Kompleksni Float64) • 12 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[raster]	Izlazni rasterski sloj sa preuređenim kanalima.

Python kod

ID algoritma: `gdal:rearrange_bands`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

RGB u PCT

Pretvara 24-bitnu RGB sliku u 8-bitnu paletiranu. Računa optimalnu tabelu pseudoboja za zadatu RGB sliku pomoću algoritma medijalnog preseka nad poduzorkovanim RGB histogramom. Zatim sliku pretvara u pseudobojenu sliku pomoću te tabele boja. Ovo pretvaranje koristi Flojd-Stajnbergovo rasipanje (difuziju greške) radi maksimalnog vizuelnog kvaliteta izlazne slike.

Ako želite da klasifikujete rastersku kartu i smanjite broj klasa, može biti korisno da prethodno poduzorkujete sliku ovim algoritmom.

Ovaj algoritam je izveden iz GDAL rgb2pct alatke.

Podrazumevani meni: *Raster ► Pretvaranje*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni (RGB) rasterski sloj
Broj boja	`NCOLORS`	[numerički: ceo broj] Podrazumevano: 2	Broj boja koje će rezultujuća slika sadržati. Moguća je vrednost od 2 do 256.
RGB u PCT	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rastera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
RGB u PCT	`OUTPUT`	[raster]	Izlazni rasterski sloj.

Python kod

ID algoritma: `gdal:rgbtopct`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Translate (pretvori format)

Pretvara rasterske podatke između različitih formata.

Ovaj algoritam je izveden iz GDAL translate alatke.

Podrazumevani meni: *Raster ► Pretvaranje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Zameni projekciju izlazne datoteke Opciono	`TARGET_CRS`	[crs]	Zadajte projekciju za izlaznu datoteku

наставак на следећој страни

Табела 24.911 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodeli zadatu NoData vrednost izlaznim kanalima Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše vrednost koja se koristi za NoData u izlaznom rasteru
Kopiraj sve podskupove podataka ove datoteke u pojedinačne izlazne datoteke	`COPY_SUBDATASETS`	[logička vrednost] Podrazumevano: False	Napravi pojedinačne datoteke za podskupove podataka
Pretvoreno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog (pretvorenog) rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

наставак на следећој страни

Табела 24.912 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 0	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Koristi tip podataka ulaznog sloja • 1 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 2 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 3 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 6 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 7 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 8 — CInt16 (Kompleksni Int16) • 9 — CInt32 (Kompleksni Int32) • 10 — CFloat32 (Kompleksni Float32) • 11 — CFloat64 (Kompleksni Float64) • 12 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[raster]	Izlazni (pretvoreni) rasterski sloj.

Python kod

ID algoritma: `gdal:translate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2.3 Izdvajanje iz rastera

Iseci raster po obuhvatu

Iseca bilo koju rastersku datoteku koju GDAL podržava na zadati obuhvat.

Ovaj algoritam je izveden iz GDAL translate alatke.

Podrazumevani meni: *Raster ► Izdvajanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni raster
Obuhvat isecanja	`EXTENT`	[obuhvat]	Obuhvat koji treba koristiti za izlazni raster. Samo pikseli unutar zadatog graničnog pravougaonika biće uključeni u izlaz. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Zameni projekciju za izlaznu datoteku	`OVERCRS`	[logička vrednost] Podrazumevano: False	Ako je uključeno, izlaznoj datoteci se dodeljuje CRS ulaznog sloja.
Dodeli zadatu NoData vrednost izlaznim kanalima Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše vrednost koja se umeće umesto NoData vrednosti u izlaznom rasteru
Isečeno (obuhvat)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.915 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 0	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Koristi tip podataka ulaznog sloja • 1 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 2 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 3 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 6 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 7 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 8 — CInt16 (Kompleksni Int16) • 9 — CInt32 (Kompleksni Int32) • 10 — CFloat32 (Kompleksni Float32) • 11 — CFloat64 (Kompleksni Float64) • 12 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno (obuhvat)	`OUTPUT`	[raster]	Izlazni rasterski sloj isečen zadatim obuhvatom

Python kod

ID algoritma: `gdal:cliprasterbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Iseci raster maskirnim slojem

Iseca bilo koji raster koji GDAL podržava vektorskim maskirnim slojem.

Ovaj algoritam izveden je iz GDAL warp alatke.

Podrazumevani meni: *Raster* ► *Izdvajanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni raster
Maskirni sloj	`MASK`	[vektor: poligon]	Vektorska maska za isecanje rastera
Izvorni CRS	`SOURCE_CRS`	[crs]	Zadajte koordinatni referentni sistem koji se koristi za ulazni raster
Ciljni CRS	`TARGET_CRS`	[crs]	Zadajte koordinatni referentni sistem koji se koristi za maskirni sloj
Ciljni obuhvat Opciono	`TARGET_EXTENT`	[obuhvat]	Obuhvat izlazne datoteke koja se pravi Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Dodeli zadatu NoData vrednost izlaznim kanalima Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše vrednost koja se umeće umesto NoData vrednosti u izlaznom rasteru
Napravi izlazni alfa kanal	`ALPHA_BAND`	[logička vrednost] Podrazumevano: False	Pravi alfa kanal za rezultat. Alfa kanal tada sadrži vrednosti prozirnosti piksela.
Uskladi obuhvat isečenog rastera sa obuhvatom maskirnog sloja	`CROP_TO_CUTLINE`	[logička vrednost] Podrazumevano: True	Ako je uključeno, primenjuje obuhvat vektorskog sloja na izlazni raster.

наставак на следећој страни

Табела 24.917 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Zadrži rezoluciju ulaznog rastera	`KEEP_RESOLUTION`	[logička vrednost] Podrazumevano: False	Rezolucija izlaznog rastera neće biti promenjena
Zadaj rezoluciju izlazne datoteke	`SET_RESOLUTION`	[logička vrednost] Podrazumevano: False	Da li treba zadati izlaznu rezoluciju (veličinu ćelije)
X rezolucija izlaznih kanala Opciono	`X_RESOLUTION`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Širina ćelija u izlaznom rasteru
Y rezolucija izlaznog kanala Opciono	`Y_RESOLUTION`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Visina ćelija u izlaznom rasteru
Koristi višenitnu implementaciju warp-a	`MULTITHREADING`	[logička vrednost] Podrazumevano: False	Koristiće se dve niti za obradu delova slike i istovremeno izvršavanje ulazno/izlaznih operacija. Imajte u vidu da samo izračunavanje nije višenitno.
Isečeno (maska)	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.918 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	DATA_TYPE	[nabrajanje] Podrazumevano: 0	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Koristi tip podataka ulaznog sloja • 1 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 2 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 3 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 6 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 7 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 8 — CInt16 (Kompleksni Int16) • 9 — CInt32 (Kompleksni Int32) • 10 — CFloat32 (Kompleksni Float32) • 11 — CFloat64 (Kompleksni Float64) • 12 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Dodatni parametri komandne linije Opciono	EXTRA	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno (maska)	OUTPUT	[raster]	Izlazni rasterski sloj isečen vektorskim slojem

Python kod

ID algoritma: `gdal:cliprasterbymasklayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izolinije

Izdvaja izolinije iz bilo kog visinskog rastera koji GDAL podržava.

Ovaj algoritam je izveden iz GDAL contour alatke.

Podrazumevani meni: *Raster ► Izdvajanje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni raster
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Rasterski kanal iz kojeg se prave izolinije
Interval između izolinija	`INTERVAL`	[numerički: decimalni] Podrazumevano: 10.0	Definiše interval između izolinija u zadatim jedinicama visinskog rastera (minimalna vrednost 0)
Naziv atributa (ako nije zadat, atribut visine se ne dodaje) Opciono	`FIELD_NAME`	[niska] Podrazumevano: 'ELEV'	Zadaje naziv atributa u koji se upisuje visina.
Pomeraj od nule u odnosu na koji se tumače intervali Opciono	`OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	
Izolinije	`OUTPUT`	[vektor: linija] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog vektorskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Napravi 3D vektor	`CREATE_3D`	[logička vrednost] Podrazumevano: False	Prisiljava pravljenje 3D vektora umesto 2D. Uključuje visinu na svakom temenu.
Tretiraj sve rasterske vrednosti kao važeće	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Zanemaruje sve NoData vrednosti u skupu podataka.
Ulazna vrednost piksela koja se tretira kao „NoData” Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše vrednost koja se umeće umesto NoData vrednosti u izlaznom rasteru
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodajte dodatne opcije GDAL komandne linije. Pogledajte odgovarajuću dokumentaciju GDAL alatke.

Izlazi

Oznaka	Назив	Tip	Opis
Izolinije	`OUTPUT`	[vektor: linija]	Izlazni vektorski sloj sa izolinijama

Python kod

ID algoritma: `gdal:contour`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Poligoni izolinija

Izdvaja poligone izolinija iz bilo kog visinskog rastera koji GDAL podržava.

Ovaj algoritam je izveden iz GDAL contour alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni raster
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Rasterski kanal iz kojeg se prave izolinije
Interval između izolinija	`INTERVAL`	[numerički: decimalni] Podrazumevano: 10.0	Definiše interval između izolinija u zadatim jedinicama visinskog rastera (minimalna vrednost 0)
Pomeraj od nule u odnosu na koji se tumače intervali Opciono	`OFFSET`	[numerički: decimalni] Podrazumevano: 0.0	
Naziv atributa za minimalnu visinu poligona izolinije Opciono	`FIELD_NAME_MIN`	[niska] Podrazumevano: 'ELEV_MIN'	Zadaje naziv atributa u koji se upisuje minimalna visina poligona izolinije. Ako nije zadat, atribut minimalne visine se ne dodaje.
Naziv atributa za maksimalnu visinu poligona izolinije Opciono	`FIELD_NAME_MAX`	[niska] Podrazumevano: 'ELEV_MAX'	Zadaje naziv atributa u koji se upisuje maksimalna visina poligona izolinije. Ako nije zadat, atribut maksimalne visine se ne dodaje.
Izolinije	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog vektorskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Napravi 3D vektor	`CREATE_3D`	[logička vrednost] Podrazumevano: False	Prisiljava pravljenje 3D vektora umesto 2D. Uključuje visinu na svakom temenu.
Tretiraj sve rasterske vrednosti kao važeće	`IGNORE_NODATA`	[logička vrednost] Podrazumevano: False	Zanemaruje sve NoData vrednosti u skupu podataka.
Ulazna vrednost piksela koja se tretira kao „NoData” Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše vrednost koja se umeće umesto NoData vrednosti u izlaznom rasteru
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodajte dodatne opcije GDAL komandne linije. Pogledajte odgovarajuću dokumentaciju GDAL alatke.

Izlazi

Oznaka	Назив	Tip	Opis
Izolinije	`OUTPUT`	[vektor: poligon]	Izlazni vektorski sloj sa poligonima izolinija

Python kod

ID algoritma: `gdal:contour_polygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2.4 Raster razno

Kreiranje pregleda (piramida)

Radi ubrzavanja iscrtavanja rasterskih slojeva mogu se napraviti pregledi (piramide). Pregledi su kopije podataka niže rezolucije koje QGIS koristi u zavisnosti od nivoa uvećanja.

Ovaj algoritam izveden je iz GDAL addo alatke.

Podrazumevani meni: *Raster* ► *Razno*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Ukloni sve postojeće preglede	`CLEAN`	[logička vrednost] Podrazumevano: False	Uklanja postojeće preglede iz rastera. Podrazumevano se ne uklanjaju.

Napredni parametri

Oznaka	Назив	Tip	Opis
Nivoi pregleda	`LEVELS`	[niska] Podrazumevano: '2 4 8 16'	Definiše broj nivoa pregleda izračunatih na osnovu originalne rezolucije ulaznog rasterskog sloja. Podrazumevano se uzimaju u obzir 4 nivoa.
Metod preuzorkovanja Opciono	`RESAMPLING`	[nabrajanje] Podrazumevano: 0	Izračunava preglede zadatim metodom preuzorkovanja. Mogući metodi preuzorkovanja su: • 0 – Najbliži sused (`nearest`) • 1 – Prosek (`average`) • 2 – Gausov (`gauss`) • 3 – Kubni (4x4 jezgro) (`cubic`) • 4 – Kubni B-splajn (4x4 jezgro) (`cubicspline`) • 5 – Lancoš (6x6 jezgro) (`lanczos`) • 6 – Prosek MP (`average_mp`) • 7 – Prosek u prostoru magnituda/faza (`average_magphase`) • 8 – Mod (`mode`)
Format pregleda Opciono	`FORMAT`	[nabrajanje] Podrazumevano: 0	Pregledi se mogu čuvati interno ili eksterno, kao GTiff ili ERDAS Imagine datoteka. Podrazumevano se čuvaju u izlaznom rasteru. Mogući formati su: • 0 – Interno (ako je moguće) • 1 – Eksterno (GTiff .ovr) • 2 – Eksterno (ERDAS Imagine .aux)
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Piramidizovano	`OUTPUT`	[raster]	Izlazni rasterski sloj sa pregledima

Python kod

ID algoritma: `gdal:overviews`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

Kreiranje virtuelnog rastera

Pravi VRT (virtuelni skup podataka) koji je mozaik liste ulaznih rastera koje GDAL podržava. Mozaikom možete spojiti više rasterskih datoteka.

Ovaj algoritam izveden je iz GDAL buildvrt alatke.

Podrazumevani meni: *Raster* ► *Razno*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Rasterski slojevi koje GDAL podržava.
Rezolucija	`RESOLUTION`	[nabrajanje] Podrazumevano: 0	Izlazna rezolucija mozaika. Podrazumevano se bira prosečna rezolucija rasterskih datoteka. Opcije: • 0 — Prosek (`average`) • 1 — Najviša (`highest`) • 2 — Najniža (`lowest`)
Smesti svaku ulaznu datoteku u zaseban kanal	`SEPARATE`	[logička vrednost] Podrazumevano: False	Sa 'True' možete zadati da svaka rasterska datoteka ide u zaseban naslagani kanal u VRT kanalu.
Dozvoli razliku u projekciji	`PROJ_DIFFERENCE`	[logička vrednost] Podrazumevano: False	Dozvoljava da izlazni kanali imaju različite projekcije izvedene iz projekcija ulaznih rasterskih slojeva.
Virtuelni	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodaj kanal alfa maske u VRT kada ga izvorni raster nema	`ADD_ALPHA`	[logička vrednost] Podrazumevano: False	Dodaje kanal alfa maske u VRT kada ga izvorni raster nema.
Zameni projekciju izlazne datoteke Opciono	`ASSIGN_CRS`	[crs] Podrazumevano: Nije postavljeno	Zamenjuje projekciju izlazne datoteke. Reprojektovanje se ne vrši.

наставак на следећој страни

Табела 24.930 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Algoritam preuzorkovanja	`RESAMPLING`	[nabrajanje] Podrazumevano: 0	Algoritam preuzorkovanja koji se koristi. Opcije: • 0 — Najbliži sused (`nearest`) • 1 — Bilinearni (2x2 jezgro) (`bilinear`) • 2 — Kubni (4x4 jezgro) (`cubic`) • 3 — Kubni B-splajn (4x4 jezgro) (`cubicspline`) • 4 — Lancoš (6x6 jezgro) (`lanczos`) • 5 — Prosek (`average`) • 6 — Mod (`mode`)
NoData vrednost(i) za ulazne kanale (razdvojene razmakom) Opciono	`SRC_NODATA`	[niska] Podrazumevano: Nije postavljeno	NoData vrednosti za ulazne kanale, razdvojene razmakom
Dodatni parametri komandne linije	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Virtuelni	`OUTPUT`	[raster]	Izlazni rasterski sloj

Python kod

ID algoritma: `gdal:buildvirtualraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

gdal2tiles

Generiše direktorijum sa malim pločicama i metapodacima, prema OSGeo Tile Map Service specifikaciji. Pogledajte i OpenGIS Web Map Tile Service Implementation Standard. Generišu se i jednostavne veb stranice sa pregledačima zasnovanim na Google Maps, OpenLayers i Leaflet. Da biste svoje karte razgledali na mreži u veb pregledaču, dovoljno je da generisani direktorijum otpremite na veb server.

Ovaj algoritam pravi i metapodatke potrebne za Google Earth (KML SuperOverlay), u slučaju da zadata karta koristi projekciju `EPSG:4326`.

Pri generisanju pločica koriste se ESRI world datoteke i ugrađeno georeferenciranje, ali sliku možete objaviti i bez ispravnog georeferenciranja.

Ovaj algoritam izveden je iz GDAL gdal2tiles alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Rasterski sloj koji GDAL podržava.
Profil isecanja pločica	`PROFILE`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — Mercator (`mercator`) • 1 — Geodetski (`geodetic`) • 2 — Raster (`raster`)
Nivoi uvećanja koji se iscrtavaju Opciono	`ZOOM`	[niska] Podrazumevano: ”	
Veb pregledač koji se generiše	`VIEWER`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — Svi (`all`) • 1 — GoogleMaps (`google`) • 2 — OpenLayers (`openlayers`) • 3 — Leaflet (`leaflet`) • 4 — Nijedan (`none`)
Naslov karte Opciono	`TITLE`	[niska] Podrazumevano: ”	
Autorska prava karte	`COPYRIGHT`	[niska] Podrazumevano: ”	
Izlazni direktorijum	`OUTPUT`	[folder] Podrazumevano: `[Save to temporary folder]`	Zadajte izlazni folder za pločice. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum

Napredni parametri

Oznaka	Назив	Tip	Opis
Metod preuzorkovanja	`RESAMPLING`	[nabrajanje] Podrazumevano: 0	Algoritam preuzorkovanja koji se koristi. Opcije: • 0 — Prosek (`average`) • 1 — Najbliži sused (`near`) • 2 — Bilinearni (2x2 jezgro) (`bilinear`) • 3 — Kubni (4x4 jezgro) (`cubic`) • 4 — Kubni B-splajn (4x4 jezgro) (`cubicspline`) • 5 — Lancoš (6x6 jezgro) (`lanczos`) • 6 — Antialijas (`antialias`)
Prostorni referentni sistem izvornih ulaznih podataka Opciono	`SOURCE_CRS`	[crs] Podrazumevano: Nije postavljeno	
Vrednost providnosti koja se dodeljuje ulaznim podacima Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	

наставак на следећој страни

Табела 24.933 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
URL adresa na kojoj će generisane pločice biti objavljene Opciono	`URL`	[niska] Podrazumevano: ”	
Google Maps API ključ (http://code.google.c Opciono	`GOOGLE_KEY`	[niska] Podrazumevano: ”	Vaš Google Maps API ključ.
Bing Maps API ključ (https://www.bingm Opciono	`BING_KEY`	[niska] Podrazumevano: ”	Vaš Bing Maps API ključ.
Generiši samo datoteke koje nedostaju	`RESUME`	[logička vrednost] Podrazumevano: False	
Generiši KML za Google Earth	`KML`	[logička vrednost] Podrazumevano: False	
Izbegni automatsko generisanje KML datoteka za EPSG:4326	`NO_KML`	[logička vrednost] Podrazumevano: False	

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT`	[folder]	Izlazni folder (za pločice)

Python kod

ID algoritma: `gdal:gdal2tiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Spajanje

Spaja rasterske datoteke na jednostavan način. Ovde možete koristiti tabelu pseudoboja iz ulaznog rastera i definisati tip izlaznog rastera. Sve slike moraju biti u istom koordinatnom sistemu.

Ovaj algoritam izveden je iz GDAL merge alatke.

Podrazumevani meni: *Raster* ► *Razno*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni slojevi	`INPUT`	[raster] [lista]	Ulazni rasterski slojevi
Preuzmi tabelu pseudoboja iz prvog sloja	`PCT`	[logička vrednost] Podrazumevano: False	Za bojenje se koristi tabela pseudoboja iz prvog sloja
Smesti svaku ulaznu datoteku u zaseban kanal	`SEPARATE`	[logička vrednost] Podrazumevano: False	Smesti svaku ulaznu datoteku u zaseban kanal
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 5	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Spojeno	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Ulazna vrednost piksela koja se tretira kao „NoData” Opciono	`NODATA_INPUT`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Zanemaruje piksele sa ovom vrednošću iz datoteka koje se spajaju

наставак на следећој страни

Табела 24.936 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodeli zadatu „NoData” vrednost izlazu Opciono	`NODATA_OUTPUT`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Dodeljuje zadatu NoData vrednost izlaznim kanalima.
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Spojeno	`OUTPUT`	[raster]	Izlazni rasterski sloj

Python kod

ID algoritma: `gdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Pansharpening

Vrši operaciju pan-sharpening. Može napraviti „klasičan” izlazni skup podataka (kao što je GeoTIFF) ili VRT skup podataka koji opisuje tu operaciju.

Pogledajte GDAL Pansharpen.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Spektralni skup podataka	`SPECTRAL`	[raster]	Ulazni (spektralni) rasterski sloj
Pankromatski skup podataka	`PANCHROMATIC`	[raster]	Ulazni (pankromatski) rasterski sloj

наставак на следећој страни

Табела 24.938 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni (izoštreni) rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Algoritam preuzorkovanja	`RESAMPLING`	[nabrajanje] Podrazumevano: 2	Algoritam preuzorkovanja koji se koristi. Opcije: • 0 — Najbliži sused (`nearest`) • 1 — Bilinearni (2x2 jezgro) (`bilinear`) • 2 — Kubni (4x4 jezgro) (`cubic`) • 3 — Kubni B-splajn (4x4 jezgro) (`cubicspline`) • 4 — Lancoš (6x6 jezgro) (`lanczos`) • 5 — Prosek (`average`)
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[raster]	Izlazni (izoštreni) rasterski sloj

Python kod

ID algoritma: `gdal:pansharp`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterski kalkulator

Rasterski kalkulator komandne linije sa numpy sintaksom. Koristite bilo koju osnovnu aritmetiku koju numpy nizovi podržavaju, kao što su +, -, * i /, zajedno sa logičkim operatorima kao što je >. Imajte na umu da svi ulazni rasteri moraju imati iste dimenzije, ali se provera projekcije ne vrši.

Pogledajte dokumentaciju GDAL Raster Calculator alatke.

See also

Rasterski kalkulator, *Rasterski kalkulator (virtuelni)*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj A	`INPUT_A`	[raster]	Prvi ulazni rasterski sloj (obavezno)
Broj rasterskog kanala za A	`BAND_A`	[rasterski kanal]	Kanal za ulazni sloj A (obavezno)
Ulazni sloj B Opciono	`INPUT_B`	[raster] Podrazumevano: Nije postavljeno	Drugi ulazni rasterski sloj
Broj rasterskog kanala za B Opciono	`BAND_B`	[rasterski kanal]	Kanal za ulazni sloj B
Ulazni sloj C Opciono	`INPUT_C`	[raster] Podrazumevano: Nije postavljeno	Treći ulazni rasterski sloj
Broj rasterskog kanala za C Opciono	`BAND_C`	[rasterski kanal]	Kanal za ulazni sloj C
Ulazni sloj D Opciono	`INPUT_D`	[raster] Podrazumevano: Nije postavljeno	Četvrti ulazni rasterski sloj
Broj rasterskog kanala za D Opciono	`BAND_D`	[rasterski kanal]	Kanal za ulazni sloj D
Ulazni sloj E Opciono	`INPUT_E`	[raster] Podrazumevano: Nije postavljeno	Peti ulazni rasterski sloj
Broj rasterskog kanala za E Opciono	`BAND_E`	[rasterski kanal]	Kanal za ulazni sloj E
Ulazni sloj F Opciono	`INPUT_F`	[raster]	Šesti ulazni rasterski sloj
Broj rasterskog kanala za F Opciono	`BAND_F`	[rasterski kanal] Podrazumevano: Nije postavljeno	Kanal za ulazni sloj F

наставак на следећој страни

Табела 24.941 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izračunavanje u gdalnumeric sintaksi pomoću +-/* ili bilo koje funkcije numpy niza (npr. logical_and())	`FORMULA`	[niska] Podrazumevano: ”	Formula izračunavanja. Primeri: • `A*(A>0)` — daje vrednost rastera A ako je vrednost A veća od 0. Ako nije, daje 0. • `A*(A>0 and A>B)` — daje vrednost A ako je ta vrednost veća od 0 i veća od vrednosti B. Ako nije, daje 0. • `A*logical_or(A<=177, A>=185)` — daje vrednost A ako je A <= 177 ili A >= 185. Ako nije, daje 0. • `sqrt(A*A+B*B)` — Daje kvadratni koren zbira kvadrata vrednosti A i kvadrata vrednosti B.
Postavi izlaznu NoData vrednost Opciono	`NO_DATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Vrednost koja se koristi za NoData
Postupanje sa razlikama u obuhvatu `Dodato u 3.34`	`EXTENT_OPT`	[nabrajanje] Podrazumevano: 0	Određuje kako se postupa sa rasterima različitih obuhvata. Podržane opcije su: • 0 — Zanemari • 1 — Prekini sa greškom • 2 — Unija • 3 — Presek **Пажња** Korišćenje ovog parametra zahteva QGIS instaliran sa GDAL >= 3.3 (pogledajte meni *Pomoć* ► *O programu*).
Izlazni obuhvat Opciono	`INPUT`	[obuhvat]	Proizvoljan obuhvat izlaznog rastera. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax` **Пажња** Korišćenje ovog parametra zahteva QGIS instaliran sa GDAL >= 3.3 (pogledajte meni *Pomoć* ► *O programu*).

наставак на следећој страни

Табела 24.941 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznog rastera	`RTYPE`	[nabrajanje] Podrazumevano: 5	Definiše tip podataka izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Izračunato	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni (izračunati) rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ”	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Izračunato	`OUTPUT`	[raster]	Izlazni (izračunati) rasterski sloj

Python kod

ID algoritma: `gdal:rastercalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Informacije o rasteru

Program gdalinfo izlistava razne informacije o rasterskom skupu podataka koji GDAL podržava.

Ovaj algoritam izveden je iz GDAL info alatke.

Podrazumevani meni: *Raster* ► *Razno*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj
Prinudno izračunaj stvarne min/maks vrednosti za svaki kanal	`MIN_MAX`	[logička vrednost] Podrazumevano: False	Prinudno izračunava stvarne min/maks vrednosti za svaki kanal u skupu podataka
Pročitaj i prikaži statistiku slike (po potrebi prinudno izračunaj)	`STATS`	[logička vrednost] Podrazumevano: False	Čita i prikazuje statistiku slike. Prinudno je izračunava ako u slici nije sačuvana.
Potisni informacije o GCP-ovima	`NO_GCP`	[logička vrednost] Podrazumevano: False	Potiskuje ispis liste kontrolnih tačaka na terenu. Može biti korisno za skupove podataka sa velikim brojem GCP-ova, kao što su L1B AVHRR ili HDF4 MODIS, koji ih sadrže na hiljade.
Potisni informacije o metapodacima	`NO_METADATA`	[logička vrednost] Podrazumevano: False	Potiskuje ispis metapodataka. Neki skupovi podataka mogu sadržati mnogo niski metapodataka.
Informacije o sloju	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte HTML datoteku za izlaz. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	EXTRA	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Informacije o sloju	OUTPUT	[html]	HTML datoteka sa informacijama o ulaznom rasterskom sloju

Python kod

ID algoritma: `gdal:gdalinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Ponovno deljenje na pločice

Ponovo deli skup ulaznih pločica. Sve ulazne pločice moraju biti georeferencirane u istom koordinatnom sistemu i imati isti broj kanala. Opciono se generišu nivoi piramide.

Ovaj algoritam izveden je iz GDAL Retile alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazne datoteke	INPUT	[raster] [lista]	Ulazne rasterske datoteke
Širina pločice	TILE_SIZE_X	[numerički: ceo broj] Podrazumevano: 256	Širina pločica u pikselima (minimum 0)
Visina pločice	TILE_SIZE_Y	[numerički: ceo broj] Podrazumevano: 256	Visina pločica u pikselima (minimum 0)
Preklapanje u pikselima između uzastopnih pločica	OVERLAP	[numerički: ceo broj] Podrazumevano: 0	
Broj nivoa piramide koji se gradi	LEVELS	[numerički: ceo broj] Podrazumevano: 1	Minimum: 0

наставак на следећој страни

Табела 24.947 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	`OUTPUT`	[folder] Podrazumevano: `[Save to temporary folder]`	Zadajte izlazni folder za pločice. *Jedno od*: • Sačuvaj u privremeni direktorijum • Sačuvaj u direktorijum
CSV datoteka sa podacima o georeferenciranju pločica	`OUTPUT_CSV`	[datoteka] Podrazumevano: `[Skip output]`	Zadajte izlaznu datoteku za pločice. *Jedno od*: • Preskoči izlaz • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Izvorni koordinatni referentni sistem Opciono	`SOURCE_CRS`	[crs] Podrazumevano: Nije postavljeno	
Metod preuzorkovanja	`RESAMPLING`	[nabrajanje] Podrazumevano: 0	Algoritam preuzorkovanja koji se koristi. Opcije: • 0 — Najbliži sused (`nearest`) • 1 — Bilinearni (2x2 jezgro) (`bilinear`) • 2 — Kubni (4x4 jezgro) (`cubic`) • 3 — Kubni B-splajn (4x4 jezgro) (`cubicspline`) • 4 — Lancoš (6x6 jezgro) (`lanczos`)
Razdvajač kolona u CSV datoteci Opciono	`DELIMITER`	[niska] Podrazumevano: ';'	Razdvajač koji se koristi u CSV datoteci sa podacima o georeferenciranju pločica
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ''	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ''	Dodaje dodatne GDAL opcije komandne linije

наставак на следећој страни

Табела 24.948 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	DATA_TYPE	[nabrajanje] Podrazumevano: 5	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Izgradi samo piramide	ONLY_PYRAMIDS	[logička vrednost] Podrazumevano: False	
Koristi zaseban direktorijum za svaki red pločica	DIR_FOR_ROW	[logička vrednost] Podrazumevano: False	

Izlazi

Oznaka	Назив	Tip	Opis
Izlazni direktorijum	OUTPUT	[folder]	Izlazni folder za pločice.
CSV datoteka sa podacima o georeferenciranju pločica	OUTPUT_CSV	[datoteka]	CSV datoteka sa podacima o georeferenciranju pločica.

Python kod

ID algoritma: `gdal:retile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Indeks pločica

Pravi vektorski sloj sa slogom za svaku ulaznu rastersku datoteku, atributom koji sadrži naziv datoteke i poligonskom geometrijom koja ocrtava raster. Ovaj izlaz pogodan je za korišćenje sa MapServer-om kao rasterski indeks pločica.

Ovaj algoritam izveden je iz GDAL Tile Index alatke.

Podrazumevani meni: *Raster ► Razno*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazne datoteke	`LAYERS`	[raster] [lista]	Ulazne rasterske datoteke. Može ih biti više.
Naziv polja koje čuva putanju do indeksiranih rastera Opciono	`PATH_FIELD_NAME`	[niska] Podrazumevano: 'location'	Naziv izlaznog polja koje čuva putanju/lokaciju datoteka indeksiranih rastera.
Čuvaj apsolutnu putanju do indeksiranih rastera	`ABSOLUTE_PATH`	[logička vrednost] Podrazumevano: False	Određuje da li se apsolutna putanja do rasterskih datoteka čuva u datoteci indeksa pločica. Podrazumevano se nazivi rasterskih datoteka upisuju tačno onako kako su zadati u komandi.
Preskoči datoteke sa drugačijom projekcijom	`PROJ_DIFFERENCE`	[logička vrednost] Podrazumevano: False	Ubacuju se samo datoteke sa istom projekcijom kao datoteke već ubačene u indeks pločica. Podrazumevano se projekcija ne proverava i prihvataju se svi ulazi.
Indeks pločica	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Zadajte poligonski vektorski sloj u koji se upisuje indeks. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Transformiši geometrije u zadati CRS Opciono	`TARGET_CRS`	[crs]	Geometrije ulaznih datoteka biće transformisane u zadati ciljni koordinatni referentni sistem. Podrazumevano se prave jednostavni pravougaoni poligoni u istom koordinatnom referentnom sistemu kao ulazni rasteri.
Naziv polja u koje se čuva SRS svake pločice Opciono	`CRS_FIELD_NAME`	[niska]	Naziv polja u koje se čuva SRS svake pločice
Format u kome se upisuje CRS svake pločice	`CRS_FORMAT`	[nabrajanje] Podrazumevano: 0	Format CRS-a. Jedno od: • 0 – Auto (`AUTO`) • 1 – Well-known text (`WKT`) • 2 – EPSG (`EPSG`) • 3 – Proj.4 (`PROJ`)

Izlazi

Oznaka	Назив	Tip	Opis
Indeks pločica	`OUTPUT`	[vektor: poligon]	Poligonski vektorski sloj sa indeksom pločica.

Python kod

ID algoritma: `gdal:tileindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Viewshed

Izračunava raster vidljivosti iz ulaznog rasterskog DEM-a metodom definisanim u Wang2000 za tačku koju zadaje korisnik.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj nadmorskih visina
Broj kanala	`BAND`	[rasterski kanal] Podrazumevano: 1	Broj kanala koji se koristi kao nadmorska visina
Lokacija posmatrača	`OBSERVER`	[koordinate]	Lokacija posmatrača
Visina posmatrača	`OBSERVER_HEIGHT`	[numerički: decimalni] Podrazumevano: 1.0	Visina posmatrača, u jedinicama DEM-a
Visina cilja	`TARGET_HEIGHT`	[numerički: decimalni] Podrazumevano: 1.0	Visina ciljnog elementa, u jedinicama DEM-a
Najveće rastojanje od posmatrača za izračunavanje vidljivosti	`MAX_DISTANCE`	[numerički: decimalni] Podrazumevano: 100.0	Najveće rastojanje od posmatrača za izračunavanje vidljivosti, u jedinicama DEM-a
Izlaz	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Izlazni rasterski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATI-ON_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Dodatni parametri komandne linije	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Izlaz	`OUTPUT`	[raster]	Rasterski sloj koji prikazuje vidljivost.

Python kod

ID algoritma: `gdal:viewshed`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2.5 Rasterske projekcije

Dodela projekcije

Primenjuje koordinatni sistem na rasterski skup podataka.

Ovaj algoritam izveden je iz GDAL edit alatke.

Podrazumevani meni: *Raster ► Projekcije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT_LAYER`	[raster]	Ulazni rasterski sloj
Željeni CRS	`CRS`	[crs]	Projekcija (CRS) izlaznog sloja

Izlazi

Oznaka	Назив	Tip	Opis
Sloj sa projekcijom	`OUTPUT`	[raster]	Izlazni rasterski sloj (sa novim podacima o projekciji)

Python kod

ID algoritma: `gdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izdvajanje projekcije

Izdvaja projekciju rasterske datoteke i upisuje je u *world* datoteku sa ekstenzijom `.wld`.

Ovaj algoritam izveden je iz GDAL srsinfo alatke.

Podrazumevani meni: *Raster ► Projekcije*

Parametri

Oznaka	Назив	Tip	Opis
Ulazna datoteka	`INPUT_LAYER`	[raster]	Ulazni raster. Rasterski sloj mora biti zasnovan na datoteci, jer algoritam koristi putanju do rasterske datoteke kao lokaciju generisane `.wld` datoteke. Korišćenje rasterskog sloja koji nije zasnovan na datoteci dovešće do greške.
Napravi i .prj datoteku	`PRJ_FILE_CREATE`	[logička vrednost] Podrazumevano: False	Ako je ovo aktivirano, pravi se i `.prj` datoteka sa podacima o projekciji.

Izlazi

Oznaka	Назив	Tip	Opis
World datoteka	`WORLD_FILE`	[datoteka]	Tekstualna datoteka sa ekstenzijom `.wld` koja sadrži parametre transformacije za rastersku datoteku.
ESRI Shapefile prj datoteka	`PRJ_FILE`	[datoteka]	Tekstualna datoteka sa ekstenzijom `.prj` koja opisuje CRS. Biće `None` ako je *Napravi i .prj datoteku* False.

Python kod

ID algoritma: `gdal:extractprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Warp (reprojektovanje)

Reprojektuje rasterski sloj u drugi koordinatni referentni sistem (CRS). Mogu se izabrati rezolucija izlazne datoteke i metod preuzorkovanja.

Ovaj algoritam izveden je iz GDAL warp alatke.

Podrazumevani meni: *Raster ► Projekcije*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[raster]	Ulazni rasterski sloj za reprojektovanje
Izvorni CRS Opciono	`SOURCE_CRS`	[crs]	Definiše CRS ulaznog rasterskog sloja
Ciljni CRS Opciono	`TARGET_CRS`	[crs] Podrazumevano: `EPSG:4326`	CRS izlaznog sloja
Metod preuzorkovanja koji se koristi	`RESAMPLING`	[nabrajanje] Podrazumevano: 0	Metod preuzorkovanja vrednosti piksela koji se koristi. Opcije: • 0 — Najbliži sused • 1 — Bilinearni (2x2 jezgro) • 2 — Kubni (4x4 jezgro) • 3 — Kubni B-splajn (4x4 jezgro) • 4 — Lancoš (6x6 jezgro) • 5 — Prosek • 6 — Mod • 7 — Maksimum • 8 — Minimum • 9 — Medijana • 10 — Prvi kvartil (Q1) • 11 — Treći kvartil (Q3)
NoData vrednost za izlazne kanale Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Postavlja NoData vrednost za izlazne kanale. Ako se ne zada, NoData vrednosti kopiraju se iz izvornog skupa podataka.
Rezolucija izlazne datoteke u ciljnim georeferenciranim jedinicama Opciono	`TARGET_RESOLUTION`	[numerički: decimalni] Podrazumevano: Nije postavljeno	Definiše rezoluciju izlazne datoteke koja je rezultat reprojektovanja
Reprojektovano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATI-ON_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).
Tip izlaznih podataka	`DATA_TYPE`	[nabrajanje] Podrazumevano: 0	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Koristi tip podataka ulaznog sloja • 1 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 2 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 3 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 4 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 5 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 6 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 7 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 8 — CInt16 (Kompleksni Int16) • 9 — CInt32 (Kompleksni Int32) • 10 — CFloat32 (Kompleksni Float32) • 11 — CFloat64 (Kompleksni Float64) • 12 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)

наставак на следећој страни

Табела 24.961 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Georeferencirani obuhvat izlazne datoteke koja se pravi Opciono	`TARGET_EXTENT`	[obuhvat]	Postavlja georeferencirani obuhvat izlazne datoteke koja se pravi (podrazumevano u *Ciljni CRS*. U *CRS ciljnog obuhvata rastera*, ako je zadat). Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
CRS ciljnog obuhvata rastera Opciono	`TARGET_EXTENT_CRS`	[crs]	Zadaje CRS u kome se tumače koordinate date za obuhvat izlazne datoteke. Ne treba ga mešati sa ciljnim CRS-om izlaznog skupa podataka. To je pogodnost npr. kada su izlazne koordinate poznate u geodetskom CRS-u geografske dužine/širine, a rezultat se želi u projektovanom koordinatnom sistemu.
Koristi višenitnu implementaciju warp-a	`MULTITHREADING`	[logička vrednost] Podrazumevano: False	Koristiće se dve niti za obradu delova slike i istovremeno izvršavanje ulazno/izlaznih operacija. Imajte na umu da samo izračunavanje nije višenitno.
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: Nije postavljeno	Dodaje dodatne GDAL opcije komandne linije.

Izlazi

Oznaka	Назив	Tip	Opis
Reprojektovano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Reprojektovani izlazni rasterski sloj

Python kod

ID algoritma: `gdal:warpreproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

24.2.6 Pretvaranje vektora

Pretvaranje formata

Pretvara bilo koji vektorski sloj koji OGR podržava u drugi format koji OGR podržava.

Ovaj algoritam izveden je iz ogr2ogr alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj
Pretvori sve slojeve iz skupa podataka	`CONVERT_ALL_LAYERS`	[logička vrednost] Podrazumevano: False	Pretvara ceo skup podataka. Podržani izlazni formati za ovu opciju su `GPKG` i `GML`.
Pretvoreno	`OUTPUT`	[isto kao ulaz]	Specifikacija izlaznog vektorskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku… Za `Save to File` mora se zadati izlazni format. Podržani su svi GDAL vektorski formati. Za `Save to a Temporary File` koristi se podrazumevani QGIS vektorski format.

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Pretvoreno	`OUTPUT`	[isto kao ulaz]	Izlazni vektorski sloj

Python kod

ID algoritma: `gdal:convertformat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterizacija (prepisivanje atributom)

Prepisuje rasterski sloj vrednostima iz vektorskog sloja. Nove vrednosti dodeljuju se na osnovu vrednosti atributa vektorskog objekta koji se preklapa.

Ovaj algoritam izveden je iz GDAL rasterize alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Ulazni rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Polje koje se koristi za upisnu vrednost Opciono	`FIELD`	[polje tabele: numerički]	Definiše atributsko polje koje se koristi za postavljanje vrednosti piksela

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodaj upisne vrednosti postojećim vrednostima rastera	`ADD`	[logička vrednost] Podrazumevano: False	Ako je False, pikselima se dodeljuje vrednost izabranog polja. Ako je True, vrednost izabranog polja dodaje se vrednosti ulaznog rasterskog sloja.
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ”	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Rasterizovano	`OUTPUT`	[raster]	Prepisani ulazni rasterski sloj

Python kod

ID algoritma: `gdal:rasterize_over`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterizacija (prepisivanje fiksnom vrednošću)

Prepisuje delove rasterskog sloja fiksnom vrednošću. Pikseli koji se prepisuju biraju se na osnovu zadatog vektorskog sloja koji se preklapa.

Ovaj algoritam izveden je iz GDAL rasterize alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Ulazni rasterski sloj	`INPUT_RASTER`	[raster]	Ulazni rasterski sloj
Fiksna vrednost za upis	`BURN`	[numerički: decimalni] Podrazumevano: 0.0	Vrednost koja se upisuje

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodaj upisne vrednosti postojećim vrednostima rastera	`ADD`	[logička vrednost] Podrazumevano: False	Ako je False, pikselima se dodeljuje fiksna vrednost. Ako je True, fiksna vrednost dodaje se vrednosti ulaznog rasterskog sloja.
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ”	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Rasterizovano	`OUTPUT`	[raster]	Prepisani ulazni rasterski sloj

Python kod

ID algoritma: `gdal:rasterize_over_fixed_value`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Rasterizacija (vektor u raster)

Pretvara vektorske geometrije (tačke, linije i poligone) u rastersku sliku.

Ovaj algoritam izveden je iz GDAL rasterize alatke.

Podrazumevani meni: *Raster ► Pretvaranje*

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Polje koje se koristi za upisnu vrednost Opciono	`FIELD`	[polje tabele: numerički]	Definiše atributsko polje iz koga se biraju atributi za piksele
Fiksna vrednost za upis Opciono	`BURN`	[numerički: decimalni] Podrazumevano: 0.0	Fiksna vrednost koja se upisuje u kanal za sve objekte.
Upisna vrednost izdvojena iz „Z" vrednosti objekta	`USE_Z`	[logička vrednost] Podrazumevano: False	Označava da se upisna vrednost izdvaja iz „Z" vrednosti objekta. Radi sa tačkama i linijama (linearna interpolacija duž svakog segmenta). Za poligone radi ispravno samo ako su ravni (ista Z vrednost za sva temena)
Jedinice veličine izlaznog rastera	`UNITS`	[nabrajanje] Podrazumevano: 0	Jedinice koje se koriste pri definisanju veličine/rezolucije izlaznog rastera. Jedno od: • 0 — Pikseli • 1 — Georeferencirane jedinice
Širina/vodoravna rezolucija	`WIDTH`	[numerički: decimalni] Podrazumevano: 0.0	Postavlja širinu (ako su jedinice veličine „Pikseli") ili vodoravnu rezoluciju (ako su jedinice veličine „Georeferencirane jedinice") izlaznog rastera. Najmanja vrednost: 0.0.
Visina/uspravna rezolucija	`HEIGHT`	[numerički: decimalni] Podrazumevano: 0.0	Postavlja visinu (ako su jedinice veličine „Pikseli") ili uspravnu rezoluciju (ako su jedinice veličine „Georeferencirane jedinice") izlaznog rastera.

наставак на следећој страни

Табела 24.972 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Izlazni obuhvat Opciono	`EXTENT`	[obuhvat]	Obuhvat izlaznog rasterskog sloja. Ako se obuhvat ne zada, koristi se najmanji obuhvat koji pokriva izabrane referentne slojeve. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Dodeli zadatu NoData vrednost izlaznim kanalima Opciono	`NODATA`	[numerički: decimalni] Podrazumevano: 0.0	Dodeljuje zadatu NoData vrednost izlaznim kanalima
Rasterizovano	`OUTPUT`	[raster] Podrazumevano: `[Save to temporary file]`	Specifikacija izlaznog rasterskog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku… Za `Save to File` mora se zadati izlazni format. Podržani su svi GDAL rasterski formati. Za `Save to a Temporary File` koristi se podrazumevani QGIS rasterski format.

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ”	Za dodavanje jedne ili više opcija kreiranja koje upravljaju rasterom koji se pravi (boje, veličina bloka, kompresija datoteke…). Radi praktičnosti možete se osloniti na unapred definisane profile (pogledajte *odeljak o opcijama GDAL drajvera*). Grupna obrada i Dizajner modela: više opcija razdvojite uspravnom crtom (`|`).

наставак на следећој страни

Табела 24.973 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Tip izlaznih podataka	DATA_TYPE	[nabrajanje] Podrazumevano: 5	Definiše format izlazne rasterske datoteke. Opcije: • 0 — Byte (Osmobitni ceo broj bez znaka (quint8)) • 1 — Int16 (Šesnaestobitni ceo broj sa znakom (qint16)) • 2 — UInt16 (Šesnaestobitni ceo broj bez znaka (quint16)) • 3 — UInt32 (Tridesetdvobitni ceo broj bez znaka (quint32)) • 4 — Int32 (Tridesetdvobitni ceo broj sa znakom (qint32)) • 5 — Float32 (Tridesetdvobitni broj u pokretnom zarezu (float)) • 6 — Float64 (Šezdesetčetvorobitni broj u pokretnom zarezu (double)) • 7 — CInt16 (Kompleksni Int16) • 8 — CInt32 (Kompleksni Int32) • 9 — CFloat32 (Kompleksni Float32) • 10 — CFloat64 (Kompleksni Float64) • 11 — Int8 (Osmobitni ceo broj sa znakom (qint8)) Dostupne opcije zavise od GDAL verzije sa kojom je QGIS izgrađen (pogledajte meni *Pomoć* ► *O programu*)
Prethodno inicijalizuj izlaznu sliku vrednošću Opciono	INIT	[numerički: decimalni]	Prethodno inicijalizuje kanale izlazne slike ovom vrednošću. Ne označava se kao NoData vrednost u izlaznoj datoteci. Ista vrednost koristi se u svim kanalima.
Obrni rasterizaciju	INVERT	[logička vrednost] Podrazumevano: False	Upisuje fiksnu upisnu vrednost, ili upisnu vrednost pridruženu prvom objektu, u sve delove slike koji nisu unutar zadatog poligona.
Dodatni parametri komandne linije Opciono	EXTRA	[niska] Podrazumevano: ”	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Rasterizovano	OUTPUT	[raster]	Izlazni rasterski sloj

Python kod

ID algoritma: `gdal:rasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

24.2.7 Vektorska geoobrada

Bafer vektora

Pravi bafere oko objekata vektorskog sloja.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Naziv kolone geometrije	`GEOMETRY`	[niska] Podrazumevano: 'geometry'	Naziv kolone geometrije ulaznog sloja koja se koristi
Rastojanje bafera	`DISTANCE`	[numerički: decimalni] Podrazumevano: 10.0	Minimum: 0.0
Stopi po atributu Opciono	`FIELD`	[polje tabele: bilo koji] Podrazumevano: Nije postavljeno	Polje koje se koristi za stapanje
Stopi rezultate	`DISSOLVE`	[logička vrednost] Podrazumevano: False	Ako je uključeno, rezultat se stapa. Ako nije zadato polje za stapanje, svi baferi se stapaju u jedan objekat.
Napravi jedan objekat za svaku geometriju u bilo kojoj vrsti kolekcije geometrija u izvornoj datoteci	`EXPLODE_COLLECTIONS`	[logička vrednost] Podrazumevano: False	
Bafer	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni sloj bafera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: '' (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Bafer	`OUTPUT`	[vektor: poligon]	Izlazni sloj bafera

Python kod

ID algoritma: `gdal:buffervectors`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Iseci vektor po obuhvatu

Iseca bilo koju vektorsku datoteku koju OGR podržava na zadati obuhvat.

Ovaj algoritam je izveden iz GDAL ogr2ogr alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Obuhvat isecanja	`EXTENT`	[obuhvat]	Definiše granični pravougaonik koji treba koristiti za izlaznu vektorsku datoteku. Mora biti definisan u koordinatama ciljnog CRS-a. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Isečeno (obuhvat)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni (isečeni) sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno (obuhvat)	`OUTPUT`	[isto kao ulaz]	Izlazni (isečeni) sloj. Podrazumevani format je „ESRI Shapefile”.

Python kod

ID algoritma: `gdal:clipvectorbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Iseci vektor maskirnim slojem

Iseca bilo koji vektorski sloj koji OGR podržava poligonskim maskirnim slojem.

Ovaj algoritam je izveden iz GDAL ogr2ogr alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Maskirni sloj	`MASK`	[vektor: poligon]	Sloj koji se koristi kao obuhvat isecanja za ulazni vektorski sloj.
Isečeno (maska)	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Save to temporary file]`	Izlazni (maskirani) sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Isečeno (maska)	`OUTPUT`	[isto kao ulaz]	Izlazni (maskirani) sloj. Podrazumevani format je „ESRI Shapefile”.

Python kod

ID algoritma: `gdal:clipvectorbypolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Stapanje

Stapa (kombinuje) geometrije koje imaju istu vrednost za zadati atribut / polje. Izlazne geometrije su višedelne.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni sloj koji se stapa
Polje za stapanje Opciono	`FIELD`	[polje tabele: bilo koji]	Polje ulaznog sloja koje se koristi za stapanje
Naziv kolone geometrije	`GEOMETRY`	[niska] Podrazumevano: ’geometry’	Naziv kolone geometrije ulaznog sloja koja se koristi za stapanje.
Stopljeno	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Napravi jedan objekat za svaku geometriju u bilo kojoj vrsti kolekcije geometrija u izvornoj datoteci	`EXPLODE_COLLECTIONS`	[logička vrednost] Podrazumevano: False	Napravi jedan objekat za svaku geometriju u bilo kojoj vrsti kolekcije geometrija u izvornoj datoteci
Zadrži ulazne atribute	`KEEP_ATTRIBUTES`	[logička vrednost] Podrazumevano: False	Zadrži sve atribute iz ulaznog sloja
Prebroj stopljene objekte	`COUNT_FEATURES`	[logička vrednost] Podrazumevano: False	Prebroji stopljene objekte i uključi broj u izlazni sloj.
Izračunaj površinu i obim stopljenih objekata	`COMPUTE_AREA`	[logička vrednost] Podrazumevano: False	Izračunaj površinu i obim stopljenih objekata i uključi ih u izlazni sloj
Izračunaj min/max/sumu/srednju vrednost za atribut	`COMPUTE_STATISTICS`	[logička vrednost] Podrazumevano: False	Izračunaj statistiku (min, max, sumu i srednju vrednost) za navedeni numerički atribut i uključi je u izlazni sloj
Numerički atribut za koji se računa statistika Opciono	`STATISTICS_ATTRIBUTE`	[polje tabele: numerički]	Numerički atribut za koji se računa statistika
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Stopljeno	`OUTPUT`	[isto kao ulaz]	Izlazni sloj sa višedelnom geometrijom (sa stopljenim geometrijama)

Python kod

ID algoritma: `gdal:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Odmaknuta kriva

Odmiče linije za zadato rastojanje. Pozitivna rastojanja odmiču linije ulevo, a negativna udesno.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski sloj
Naziv kolone geometrije	`GEOMETRY`	[niska] Podrazumevano: 'geometry'	Naziv kolone geometrije ulaznog sloja koja se koristi
Rastojanje odmicanja (levo: pozitivno, desno: negativno)	`DISTANCE`	[numerički: decimalni] Podrazumevano: 10.0	
Odmaknuta kriva	`OUTPUT`	[vektor: linija] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni linijski sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: '' (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Odmaknuta kriva	`OUTPUT`	[vektor: linija]	Izlazni sloj odmaknutih krivih

Python kod

ID algoritma: `gdal:offsetcurve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Jednostrani bafer

Pravi bafer sa jedne strane (desne ili leve) linija u linijskom vektorskom sloju.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski sloj
Naziv kolone geometrije	`GEOMETRY`	[niska] Podrazumevano: 'geometry'	Naziv kolone geometrije ulaznog sloja koja se koristi
Rastojanje bafera	`DISTANCE`	[numerički: decimalni] Podrazumevano: 10.0	
Strana bafera	`BUFFER_SIDE`	[nabrajanje] Podrazumevano: 0	Jedno od: • 0 — Desno • 1 — Levo
Stopi po atributu Opciono	`FIELD`	[polje tabele: bilo koji] Podrazumevano: Nije postavljeno	Polje koje se koristi za stapanje
Stopi sve rezultate	`DISSOLVE`	[logička vrednost] Podrazumevano: False	Ako je uključeno, rezultat se stapa. Ako nije zadato polje za stapanje, svi baferi se stapaju u jedan objekat.
Napravi jedan objekat za svaku geometriju u bilo kojoj vrsti kolekcije geometrija u izvornoj datoteci	`EXPLODE_COLLECTIONS`	[logička vrednost] Podrazumevano: False	
Jednostrani bafer	`OUTPUT`	[vektor: poligon] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni sloj bafera. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: '' (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Jednostrani bafer	`OUTPUT`	[vektor: poligon]	Izlazni sloj bafera

Python kod

ID algoritma: `gdal:onesidebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju

algoritama obrade iz Python konzole.

Tačke duž linija

Generiše tačku na svakoj liniji linijskog vektorskog sloja na zadatom rastojanju od početka. Rastojanje se zadaje kao deo dužine linije.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: linija]	Ulazni linijski sloj
Naziv kolone geometrije	`GEOMETRY`	[niska] Podrazumevano: 'geometry'	Naziv kolone geometrije ulaznog sloja koja se koristi
Rastojanje od početka linije izraženo kao deo dužine linije	`DISTANCE`	[numerički: decimalni] Podrazumevano: 0.5	
Tačke duž linija	`OUTPUT`	[vektor: tačka] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni tačkasti sloj. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: '' (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
Tačke duž linije	`OUTPUT`	[vektor: tačka]	Izlazni tačkasti sloj

Python kod

ID algoritma: `gdal:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

24.2.8 Razni vektorski alati

Napravi virtuelni vektor

Pravi virtuelni vektorski sloj koji sadrži skup vektorskih slojeva.

Ovaj algoritam je posebno koristan kada drugom algoritmu treba više slojeva, a prihvata samo jedan `vrt` u kojem su slojevi navedeni.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni izvori podataka	`INPUT`	[vektor: bilo koji] [lista]	Izaberite vektorske slojeve koje želite da koristite za pravljenje virtuelnog vektora
Napravi „unified“ VRT	`UNIONED`	[logička vrednost] Podrazumevano: False	Označite ako želite da spojite sve vektore u jednu `vrt` datoteku
Virtuelni vektor	`OUTPUT`	[isto kao ulaz] Podrazumevano: `[Save to temporary file]`	Zadajte izlazni sloj koji sadrži samo duplikate. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku…

Izlazi

Oznaka	Назив	Tip	Opis
Virtuelni vektor	`OUTPUT`	[vektor: bilo koji]	Izlazni virtuelni vektor napravljen od izabranih izvora

Python kod

ID algoritma: `gdal:buildvirtualvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvršavanje SQL-a

Pokreće jednostavan ili složen upit sa SQL sintaksom nad izvornim slojem. Rezultat upita biće dodat kao novi sloj.

Ovaj algoritam je izveden iz GDAL ogr2ogr alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Ulazni vektorski sloj koji OGR podržava
SQL izraz	`SQL`	[niska]	Definiše SQL upit, na primer `SELECT * FROM my_table WHERE name is not null`.

наставак на следећој страни

Табела 24.998 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
SQL dijalekat	`DIALECT`	[nabrajanje] Podrazumevano: 0	SQL dijalekat koji se koristi. Jedno od: • 0 — Nijedan • 1 — OGR SQL • 2 — SQLite
SQL rezultat	`OUTPUT`	[vektor: bilo koji]	Specifikacija izlaznog sloja. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku… Za `Save to File` mora se zadati izlazni format. Podržani su svi GDAL vektorski formati. Za `Save to a Temporary File` koristiće se podrazumevani format izlaznog vektorskog sloja.

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Oznaka	Назив	Tip	Opis
SQL rezultat	`OUTPUT`	[vektor: bilo koji]	Vektorski sloj koji je upit kreirao

Python kod

ID algoritma: `gdal:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u PostgreSQL (dostupne veze)

Uvozi vektorske slojeve u PostgreSQL bazu podataka na osnovu dostupne veze. Veza mora prethodno biti *ispravno definisana*. Imajte u vidu da su polja „Sačuvaj korisničko ime“ i „Sačuvaj lozinku“ uključena. Zatim možete koristiti algoritam.

Ovaj algoritam je izveden iz GDAL ogr2ogr alatke.

Parametri

Oznaka	Назив	Tip	Opis
Baza podataka (naziv veze)	DATABASE	[niska]	PostgreSQL baza podataka na koju se povezuje
Ulazni sloj	INPUT	[vektor: bilo koji]	Vektorski sloj koji OGR podržava, za izvoz u bazu podataka
Kodiranje shapefile-a Opciono	SHAPE_ENCODING	[niska] Podrazumevano: ”	Postavlja kodiranje koje se primenjuje na podatke
Izlazni tip geometrije	GTYPE	[nabrajanje] Podrazumevano: 0	Definiše izlazni tip geometrije. Jedno od: • 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
Dodeli izlazni CRS Opciono	A_SRS	[crs] Podrazumevano: Nije postavljeno	Definiše izlazni CRS tabele baze podataka
Reprojektuj u ovaj CRS pri izvozu Opciono	T_SRS	[crs] Podrazumevano: Nije postavljeno	Reprojektuje/transformiše u ovaj CRS pri izvozu
Zameni izvorni CRS Opciono	S_SRS	[crs] Podrazumevano: Nije postavljeno	Zamenjuje CRS ulaznog sloja
Šema (naziv šeme) Opciono	SCHEMA	[niska] Podrazumevano: ’public’	Definiše šemu za tabelu baze podataka
Tabela u koju se izvozi (ostavite prazno za korišćenje naziva sloja) Opciono	TABLE	[niska] Podrazumevano: ”	Definiše naziv tabele koja će biti uvezena u bazu podataka. Podrazumevano je naziv tabele naziv ulazne vektorske datoteke.
Primarni ključ (novo polje) Opciono	PK	[niska] Podrazumevano: ’id’	Definiše koje atributsko polje će biti primarni ključ tabele baze podataka
Primarni ključ (postojeće polje, koristi se ako je gornja opcija prazna) Opciono	PRIMARY_KEY	[polje tabele: bilo koji] Podrazumevano: Nije postavljeno	Definiše koje atributsko polje u izvezenom sloju će biti primarni ključ tabele baze podataka

наставак на следећој страни

Табела 24.1001 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Naziv kolone geometrije Opciono	`GEOCOLUMN`	[niska] Podrazumevano: 'geom'	Definiše u kom atributskom polju baze podataka će se nalaziti informacija o geometriji
Dimenzije vektora Opciono	`DIM`	[nabrajanje] Podrazumevano: 0 (2D)	Definiše dimenzije koordinata uvezenih vektorskih podataka. Jedno od: • 0 — 2 (`XY`) • 1 — 3 (`XYZ`) • 2 — 4 (`XYZM`)
Tolerancija rastojanja za uprošćavanje Opciono	`SIMPLIFY`	[niska] Podrazumevano: ”	Definiše toleranciju rastojanja za uprošćavanje vektorskih geometrija koje se uvoze. Podrazumevano nema uprošćavanja.
Maksimalno rastojanje između 2 čvora (zgušnjavanje) Opciono	`SEGMENTIZE`	[niska] Podrazumevano: ”	Maksimalno rastojanje između dva čvora. Koristi se za pravljenje međutačaka. Podrazumevano nema zgušnjavanja.
Izaberi objekte po obuhvatu (definisanom u CRS-u ulaznog sloja) Opciono	`SPAT`	[obuhvat] Podrazumevano: Nije postavljeno	Možete izabrati objekte iz zadatog obuhvata koji će biti u izlaznoj tabeli. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Iseci ulazni sloj pomoću gornjeg (pravougaonog) obuhvata	`CLIP`	[logička vrednost] Podrazumevano: False	Ulazni sloj biće isečen obuhvatom koji ste prethodno definisali
Izaberi objekte pomoću SQL „WHERE“ iskaza (npr: column=”value”) Opciono	`WHERE`	[niska] Podrazumevano: ”	Definiše SQL „WHERE“ iskazom koji objekti treba da budu izabrani iz ulaznog sloja
Grupiši N objekata po transakciji (Podrazumevano: 2000) Opciono	`GT`	[niska] Podrazumevano: ”	Možete grupisati ulazne objekte u transakcije gde N definiše veličinu. Podrazumevano N ograničava veličinu transakcije na 20000 objekata.
Prepiši postojeću tabelu	`OVERWRITE`	[logička vrednost] Podrazumevano: True	Ako u bazi podataka postoji tabela istog naziva i ako je ova opcija postavljena na True, tabela će biti prepisana.

наставак на следећој страни

Табела 24.1001 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodaj na postojeću tabelu	`APPEND`	[logička vrednost] Podrazumevano: False	Ako je uključeno / True, vektorski podaci biće dodati na postojeću tabelu. Nova polja pronađena u ulaznom sloju se zanemaruju. Podrazumevano se pravi nova tabela.
Dodaj i dodaj nova polja na postojeću tabelu	`ADDFIELDS`	[logička vrednost] Podrazumevano: False	Ako je aktivirano, vektorski podaci biće dodati na postojeću tabelu, neće se praviti nova tabela. Nova polja pronađena u ulaznom sloju dodaju se u tabelu. Podrazumevano se pravi nova tabela.
Ne pročišćavaj nazive kolona/tabela	`LAUNDER`	[logička vrednost] Podrazumevano: False	Sa uključenom ovom opcijom možete sprečiti podrazumevano ponašanje (pretvaranje naziva kolona u mala slova, uklanjanje razmaka i drugih neispravnih znakova).
Ne pravi prostorni indeks	`INDEX`	[logička vrednost] Podrazumevano: False	Sprečava pravljenje prostornog indeksa za izlaznu tabelu. Podrazumevano se prostorni indeks dodaje.
Nastavi nakon greške, preskačući objekat sa greškom	`SKIPFAILURES`	[logička vrednost] Podrazumevano: False	Nastavi nakon greške, preskačući objekat sa greškom.
Proveri ispravnost geometrija prema specifikaciji Simple Features	`MAKEVALID`	[logička vrednost] Podrazumevano: False	Primenjuje operacije čišćenja na geometrijama kako bi bile ispravne prema pravilima specifikacije Simple Features.
Unapredi u višedelnu	`PROMOTETOMULTI`	[logička vrednost] Podrazumevano: True	Pretvara tip geometrije objekata u višedelni u izlaznoj tabeli
Zadrži širinu i preciznost ulaznih atributa	`PRECISION`	[logička vrednost] Podrazumevano: True	Izbegava izmenu atributa kolona radi usklađivanja sa ulaznim podacima
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Ovaj algoritam nema izlaz.

Python kod

ID algoritma: `gdal:importvectorintopostgisdatabaseavailableconnections`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Izvoz u PostgreSQL (nova veza)

Uvozi vektorske slojeve u PostgreSQL bazu podataka. Mora se napraviti nova veza sa PostgreSQL bazom podataka.

Ovaj algoritam je izveden iz GDAL ogr2ogr alatke.

Parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: bilo koji]	Vektorski sloj koji OGR podržava, za izvoz u bazu podataka
Kodiranje shape-file-a Opciono	`SHAPE_ENCODING`	[niska] Podrazumevano: ”	Postavlja kodiranje koje se primenjuje na podatke
Izlazni tip geometrije	`GTYPE`	[nabrajanje] Podrazumevano: 0	Definiše izlazni tip geometrije. Jedno od: • 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
Dodeli izlazni CRS Opciono	`A_SRS`	[crs] Podrazumevano: Nije postavljeno	Definiše izlazni CRS tabele baze podataka
Reprojektuj u ovaj CRS pri izvozu Opciono	`T_SRS`	[crs] Podrazumevano: Nije postavljeno	Reprojektuje/transformiše u ovaj CRS pri izvozu
Zameni izvorni CRS Opciono	`S_SRS`	[crs] Podrazumevano: Nije postavljeno	Zamenjuje CRS ulaznog sloja
Host Opciono	`HOST`	[niska] Podrazumevano: ’localhost’	Naziv hosta baze podataka
Port Opciono	`PORT`	[niska] Podrazumevano: ’5432’	Broj porta na kojem sluša server PostgreSQL baze podataka
Korisničko ime Opciono	`USER`	[niska] Podrazumevano: ”	Korisničko ime koje se koristi za prijavu na bazu podataka
Naziv baze podataka Opciono	`DBNAME`	[niska] Podrazumevano: ”	Naziv baze podataka
Lozinka Opciono	`PASSWORD`	[niska] Podrazumevano: ”	Lozinka koja se koristi sa korisničkim imenom za povezivanje sa bazom podataka

наставак на следећој страни

Табела 24.1002 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Šema (naziv šeme) Opciono	`SCHEMA`	[niska] Podrazumevano: 'public'	Definiše šemu za tabelu baze podataka
Naziv tabele, ostavite prazno za korišćenje ulaznog naziva Opciono	`TABLE`	[niska] Podrazumevano: ''	Definiše naziv tabele koja će biti uvezena u bazu podataka. Podrazumevano je naziv tabele naziv ulazne vektorske datoteke.
Primarni ključ (novo polje) Opciono	`PK`	[niska] Podrazumevano: 'id'	Definiše koje atributsko polje će biti primarni ključ tabele baze podataka
Primarni ključ (postojeće polje, koristi se ako je gornja opcija prazna) Opciono	`PRIMARY_KEY`	[polje tabele: bilo koji] Podrazumevano: Nije postavljeno	Definiše koje atributsko polje u izvezenom sloju će biti primarni ključ tabele baze podataka
Naziv kolone geometrije Opciono	`GEOCOLUMN`	[niska] Podrazumevano: 'geom'	Definiše u kom atributskom polju se čuva informacija o geometriji
Dimenzije vektora Opciono	`DIM`	[nabrajanje] Podrazumevano: 0 (2D)	Definiše dimenzije koordinata uvezenih vektorskih podataka. Jedno od: • 0 — 2 (`XY`) • 1 — 3 (`XYZ`) • 2 — 4 (`XYZM`)
Tolerancija rastojanja za uprošćavanje Opciono	`SIMPLIFY`	[niska] Podrazumevano: ''	Definiše toleranciju rastojanja za uprošćavanje vektorskih geometrija koje se uvoze. Podrazumevano nema uprošćavanja.
Maksimalno rastojanje između 2 čvora (zgušnjavanje) Opciono	`SEGMENTIZE`	[niska] Podrazumevano: ''	Maksimalno rastojanje između dva čvora. Koristi se za pravljenje međutačaka. Podrazumevano nema zgušnjavanja.
Izaberi objekte po obuhvatu (definisanom u CRS-u ulaznog sloja) Opciono	`SPAT`	[obuhvat] Podrazumevano: Nije postavljeno	Možete izabrati objekte iz zadatog obuhvata koji će biti u izlaznoj tabeli. Dostupni metodi su: • Izračunaj iz sloja…: koristi obuhvat sloja učitanog u tekućem projektu • Izračunaj iz karte rasporeda…: koristi obuhvat *stavke karte u rasporedu* u aktivnom projektu • Izračunaj iz obeleživača…: koristi obuhvat sačuvanog *obeleživača* • Koristi obuhvat platna karte • Nacrtaj na platnu: klikom i prevlačenjem nacrtajte pravougaonik koji ograničava područje koje se uzima u obzir • Unesite koordinate kao `xmin, xmax, ymin, ymax`
Iseci ulazni sloj pomoću gornjeg (pravougaonog) obuhvata	`CLIP`	[logička vrednost] Podrazumevano: False	Ulazni sloj biće isečen obuhvatom koji ste prethodno definisali

наставак на следећој страни

Табела 24.1002 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Polja koja se uključuju (ostavite prazno za korišćenje svih polja) Opciono	FIELDS	[niska] [lista] Podrazumevano: []	Definiše polja koja se zadržavaju iz uvezene vektorske datoteke. Ako nijedno nije izabrano, uvoze se sva polja.
Izaberi objekte pomoću SQL „WHERE“ iskaza (npr: column=”value”) Opciono	WHERE	[niska] Podrazumevano: ”	Definiše SQL „WHERE“ iskazom koji objekti treba da budu izabrani za izlaznu tabelu
Grupiši N objekata po transakciji (Podrazumevano: 2000) Opciono	GT	[niska] Podrazumevano: ”	Možete grupisati ulazne objekte u transakcije gde N definiše veličinu. Podrazumevano N ograničava veličinu transakcije na 20000 objekata.
Prepiši postojeću tabelu	OVERWRITE	[logička vrednost] Podrazumevano: True	Ako u bazi podataka postoji tabela istog naziva i ako je ova opcija postavljena na True, tabela će biti prepisana.
Dodaj na postojeću tabelu	APPEND	[logička vrednost] Podrazumevano: False	Ako je uključeno / True, vektorski podaci biće dodati na postojeću tabelu. Nova polja pronađena u ulaznom sloju se zanemaruju. Podrazumevano se pravi nova tabela.
Dodaj i dodaj nova polja na postojeću tabelu	ADDFIELDS	[logička vrednost] Podrazumevano: False	Ako je aktivirano, vektorski podaci biće dodati na postojeću tabelu, neće se praviti nova tabela. Nova polja pronađena u ulaznom sloju dodaju se u tabelu. Podrazumevano se pravi nova tabela.
Ne pročišćavaj nazive kolona/tabela	LAUNDER	[logička vrednost] Podrazumevano: False	Sa uključenom ovom opcijom možete sprečiti podrazumevano ponašanje (pretvaranje naziva kolona u mala slova, uklanjanje razmaka i drugih neispravnih znakova).
Ne pravi prostorni indeks	INDEX	[logička vrednost] Podrazumevano: False	Sprečava pravljenje prostornog indeksa za izlaznu tabelu. Podrazumevano se prostorni indeks dodaje.
Nastavi nakon greške, preskačući objekat sa greškom	SKIPFAILURES	[logička vrednost] Podrazumevano: False	Nastavi nakon greške, preskačući objekat sa greškom.
Proveri ispravnost geometrija prema specifikaciji Simple Features	MAKEVALID	[logička vrednost] Podrazumevano: False	Primenjuje operacije čišćenja na geometrijama kako bi bile ispravne prema pravilima specifikacije Simple Features.
Unapredi u višedelnu	PROMOTETOMULTI	[logička vrednost] Podrazumevano: True	Pretvara tip geometrije objekata u višedelni u izlaznoj tabeli
Zadrži širinu i preciznost ulaznih atributa	PRECISION	[logička vrednost] Podrazumevano: True	Izbegava izmenu atributa kolona radi usklađivanja sa ulaznim podacima

наставак на следећој страни

Табела 24.1002 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Dodatne opcije kreiranja Opciono	`CREATION_OPTIONS` (za QGIS <= 3.42 ovo je bilo `OPTIONS`)	[niska] Podrazumevano: '' (bez dodatnih opcija)	Dodatne GDAL opcije kreiranja.

Izlazi

Ovaj algoritam nema izlaz.

Python kod

ID algoritma: `gdal:importvectorintopostgisdatabasenewconnection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Informacije o vektoru

Pravi informacionu datoteku koja navodi informacije o izvoru podataka koji OGR podržava. Izlaz će biti prikazan u prozoru „Rezultat“ i može se upisati u HTML datoteku. Informacije uključuju tip geometrije, broj objekata, prostorni obuhvat, informacije o projekciji, listu atributa i pripadajućih tipova, listu relacija i domena polja, i još mnogo toga.

Ovaj algoritam je izveden iz GDAL ogrinfo alatke.

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Omogući izlistavanje svih slojeva u skupu podataka `Dodato u 3.40`	`ALL_LAYERS`	[logička vrednost] Podrazumevano: False	Ako je uključeno, QGIS će ispisati informacije o svim slojevima u ulaznom skupu podataka umesto informacija samo o prvom sloju. Zgodno kada se kao ulaz koristi višeslojni format (GeoPackage, GML, KML, SpatiaLite, SQLite, …).
Samo zbirni izlaz	`SUMMARY_ONLY`	[logička vrednost] Podrazumevano: True	Izostavlja izlistavanje pojedinačnih objekata i prikazuje samo zbirne informacije kao što su projekcija, šema, broj objekata i obuhvati.
Potisni informacije o metapodacima	`NO_METADATA`	[logička vrednost] Podrazumevano: False	Izostavlja ispis metapodataka. Neki skupovi podataka mogu sadržati mnogo niski metapodataka.

наставак на следећој страни

Табела 24.1003 – настављено са претходне стране

Oznaka	Назив	Tip	Opis
Informacije o sloju	`OUTPUT`	[html] Podrazumevano: `[Save to temporary file]`	Zadajte izlaznu HTML datoteku koja sadrži informacije o datoteci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku… Ako HTML datoteka nije definisana, izlaz će biti upisan u privremenu datoteku

Napredni parametri

`Dodato u 3.40`

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Informacije o sloju	`OUTPUT`	[html]	Izlazna HTML datoteka koja sadrži informacije o datoteci.

Python kod

ID algoritma: `gdal:ogrinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

Informacije o vektoru (JSON)

`Dodato u 3.40`

Pravi informacionu datoteku koja navodi informacije o izvoru podataka koji OGR podržava. Izlaz će biti prikazan u prozoru „Rezultat“ i može se upisati u `.JSON` datoteku. Informacije uključuju tip geometrije, broj objekata, prostorni obuhvat, informacije o projekciji, listu atributa i pripadajućih tipova, listu relacija i domena polja, i još mnogo toga.

Ovaj algoritam je izveden iz GDAL ogrinfo alatke.

Пажња

Pokretanje ovog algoritma zahteva QGIS instaliran sa GDAL >= 3.7 (pogledajte meni *Pomoć* ► *O programu*).

Parametri

Osnovni parametri

Oznaka	Назив	Tip	Opis
Ulazni sloj	`INPUT`	[vektor: geometrija]	Ulazni vektorski sloj
Omogući izlistavanje svih slojeva u skupu podataka	`ALL_LAYERS`	[logička vrednost] Podrazumevano: False	Ako je uključeno, QGIS će ispisati informacije o svim slojevima u ulaznom skupu podataka umesto informacija samo o prvom sloju. Zgodno kada se kao ulaz koristi višeslojni format (GeoPackage, GML, KML, SpatiaLite, SQLite, …).
Omogući izlistavanje objekata	`FEATURES`	[logička vrednost] Podrazumevano: False	
Potisni informacije o metapodacima	`NO_METADATA`	[logička vrednost] Podrazumevano: False	Izostavlja ispis metapodataka. Neki skupovi podataka mogu sadržati mnogo niski metapodataka.
Informacije o sloju	`OUTPUT`	[datoteka] Podrazumevano: `[Save to temporary file]`	Zadajte izlaznu JSON datoteku koja sadrži informacije o datoteci. *Jedno od*: • Sačuvaj u privremenu datoteku • Sačuvaj u datoteku… Ako JSON datoteka nije definisana, izlaz će biti upisan u privremenu datoteku

Napredni parametri

Oznaka	Назив	Tip	Opis
Dodatni parametri komandne linije Opciono	`EXTRA`	[niska] Podrazumevano: ” (bez dodatnih opcija)	Dodaje dodatne GDAL opcije komandne linije

Izlazi

Oznaka	Назив	Tip	Opis
Informacije o sloju	`OUTPUT`	[datoteka]	Izlazna *.JSON* datoteka koja sadrži informacije o datoteci.

Python kod

ID algoritma: `gdal:ogrinfojson`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

ID algoritma se prikazuje kada pređete mišem preko algoritma u kutiji sa alatkama Obrade. *Rečnik parametara* daje NAZIVe i vrednosti parametara. Pogledajte *Using processing algorithms from the console* za detalje o pokretanju algoritama obrade iz Python konzole.

ГЛАВА 25

Dodaci

25.1 QGIS Plugins

QGIS has been designed with a plugin architecture. This allows many new features and functions to be easily added to the application. Some of the features in QGIS are actually implemented as plugins.

25.1.1 Core and External plugins

QGIS plugins are implemented either as **Core Plugins** or **External Plugins**.

Core Plugins are maintained by the QGIS Development Team and are automatically part of every QGIS distribution. They are written in one of two languages: **C++** or **Python**.

Most of External Plugins are currently written in Python. They are stored either in the 'Official' QGIS Repository at https://plugins.qgis.org/plugins/ or in external repositories and are maintained by the individual authors. Detailed documentation about the usage, minimum QGIS version, home page, authors, and other important information are provided for the plugins in the Official repository. For other external repositories, documentation might be available with the external plugins themselves. External plugins documentation is not included in this manual.

To install or activate a plugin, go to *Plugins* menu and select  *Manage and install plugins...*. Installed external python plugins are placed under the `python/plugins` folder of the active *user profile* path.

Paths to Custom C++ plugins libraries can also be added under *Settings* ► *Options* ► *System*.

25.1.2 The Plugins Dialog

The Settings tab

At the bottom of the left panel, the  *Settings* tab is the main place you can configure which plugins can be displayed in your application. You can use the following options:

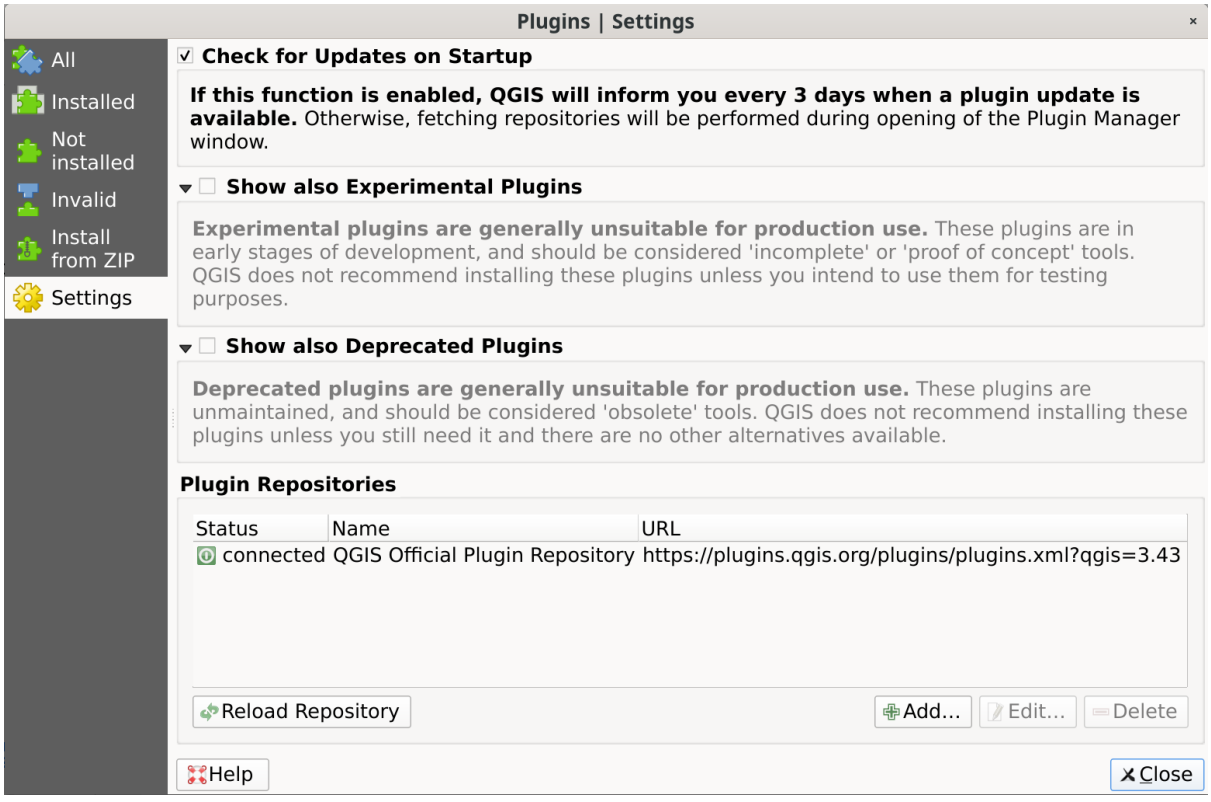
-  *Check for Updates on Startup*. QGIS will automatically check for plugin updates at startup, at most once every 3 days. Whenever an installed plugin has an update available, QGIS will inform you.

- *Show also Experimental Plugins*. QGIS will show you plugins in early stages of development, which are generally unsuitable for production use. For these plugins, you can install either the stable or the experimental version, and at any moment switch from one to the other.
- *Show also Deprecated Plugins*. These plugins are usually unmaintained because they have replacement functions in QGIS, a lack of maintainers, they rely on functions that are no longer available in QGIS… They are generally unsuitable for production use and appear grayed in the plugins list.

By default, in the *Plugin Repositories* section, QGIS provides you with its official plugin repository with the URL `https://plugins.qgis.org/plugins/plugins.xml?qgis=version` (where `<version>` represents the exact QGIS version you are running). To add external author repositories, click *Add…* and fill in the *Repository Details* form with a name and the URL. The URL can be of `http://` or `file://` protocol type.

The default QGIS repository is an open repository and you don't need any authentication to access it. You can however deploy your own plugin repository and require an authentication (basic authentication, PKI). You can get more information on QGIS authentication support in *Authentication* chapter.

If you do not want one or more of the added repositories, they can be disabled from the Settings tab via the *Edit…* button, or completely removed with the *Delete* button.



Слика25.1: The *Settings* tab

Browsing the plugins

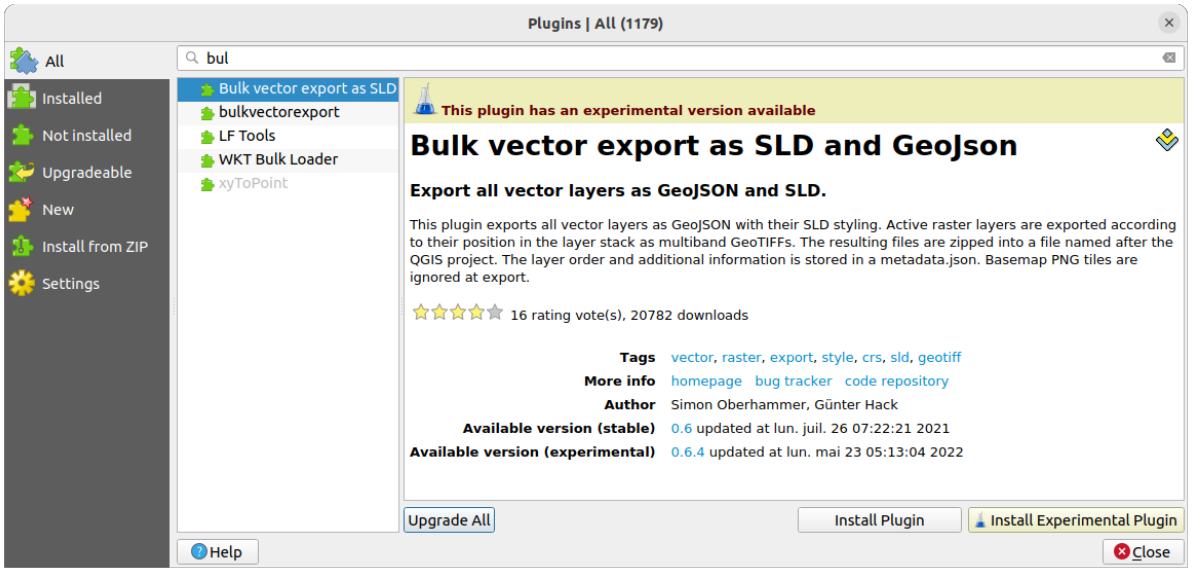
The tabs

The upper tabs in the *Plugins* dialog provide you with lists of plugins based on their install, creation or update status. Depending on the plugins settings, available tabs can be:

- *All*: shows all the available plugins in the enabled repositories

- *Installed*: shows both the plugins you installed and the core plugins that are installed by default and you can not uninstall
- *Not installed*: shows uninstalled or not yet installed plugins in the enabled repositories
- *New*: shows plugins released since the last *Check for Updates on Startup*
- *Upgradeable*: shows installed plugins that have published a more recent version in the repository
- *Invalid*: shows all installed plugins that are currently broken for any reason (missing dependency, errors while loading, incompatible functions with QGIS version…)

At the top of the tabs, a *Search* function helps you find any plugin using metadata information (author, name, description, tag,…).



Слика25.2: Searching a plugin from the *All* tab

The Plugins

Select a plugin and you will have some metadata displayed in the right panel:

- information on whether the plugin is experimental or has an experimental version available (if *Show also Experimental Plugins* is checked)
- summary and description
- rating vote(s) (you can vote for your preferred plugin!)
- tags
- some useful links to the home page, tracker and code repository
- author(s)
- version(s) available with link to download page in the repository, or path to local folder for installed plugins

The *Plugin Manager* dialog allows you to interact with the latest version of the plugins. When enabled, the experimental version can be shown only if it is more recent than the latest stable version. Depending on the active tab, whether the selected plugin is installed, you will be given some of the following options:

- *Install*: installs the latest stable version of the selected plugin
- *Install Experimental Plugin*: installs the experimental version of the selected plugin

- *Reinstall Plugin*: installs the same stable version of the plugin e.g. after it has failed to load
- *Reinstall Experimental Plugin*: installs the same experimental version of the plugin e.g. after it has failed to load
- *Upgrade Plugin*: upgrades selected plugin to its latest stable version
- *Upgrade Experimental Plugin*: upgrades selected plugin to its experimental version
- *Upgrade All*: upgrades all installed plugins to their more recent stable or experimental version (depending on whether their previously installed version was stable or experimental).
- *Downgrade Plugin*: moves from the experimental version of the plugin to its previous stable version
- *Downgrade Experimental Plugin*: moves from an experimental version of the plugin to its latest published experimental version. This may occur when playing with a not yet published version.
- *Uninstall Plugin*: removes the installed plugin from the user profile

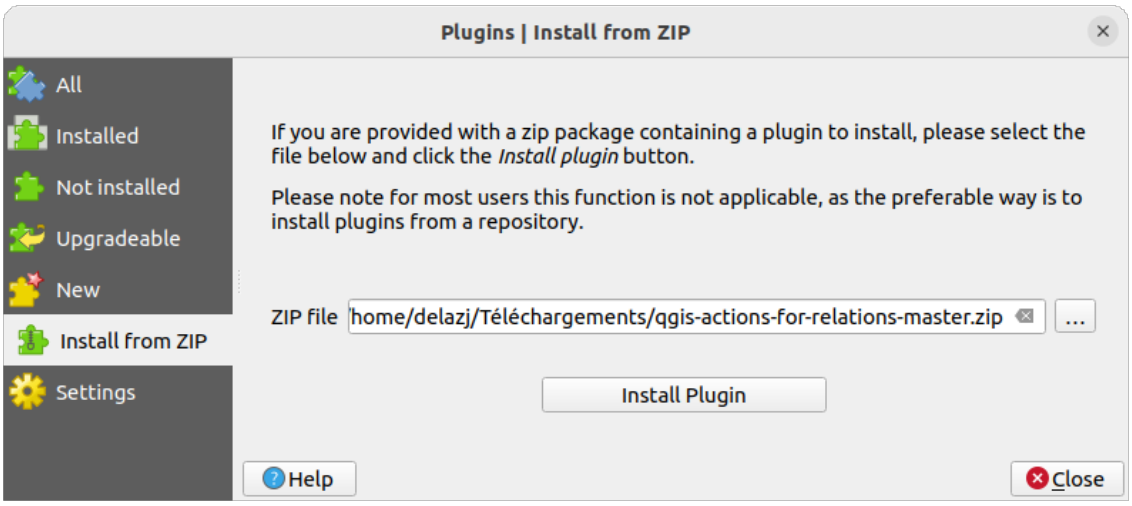
An installed plugin displays a checkbox on its left. Uncheck it to temporarily deactivate the plugin.

Right-click on a plugin in the list and you will be able to sort the plugins list by various metadata. The new order applies to all the tabs. Sort options are:

- *Sort by Name*
- *Sort by Downloads*
- *Sort by Vote*
- *Sort by Status*
- *Sort by Date Created*
- *Sort by Date Updated*

The Install from ZIP tab

The *Install from ZIP* tab provides a file selector widget to import plugins in a zipped format, e.g. plugins downloaded directly from their repository. Encrypted files are supported.

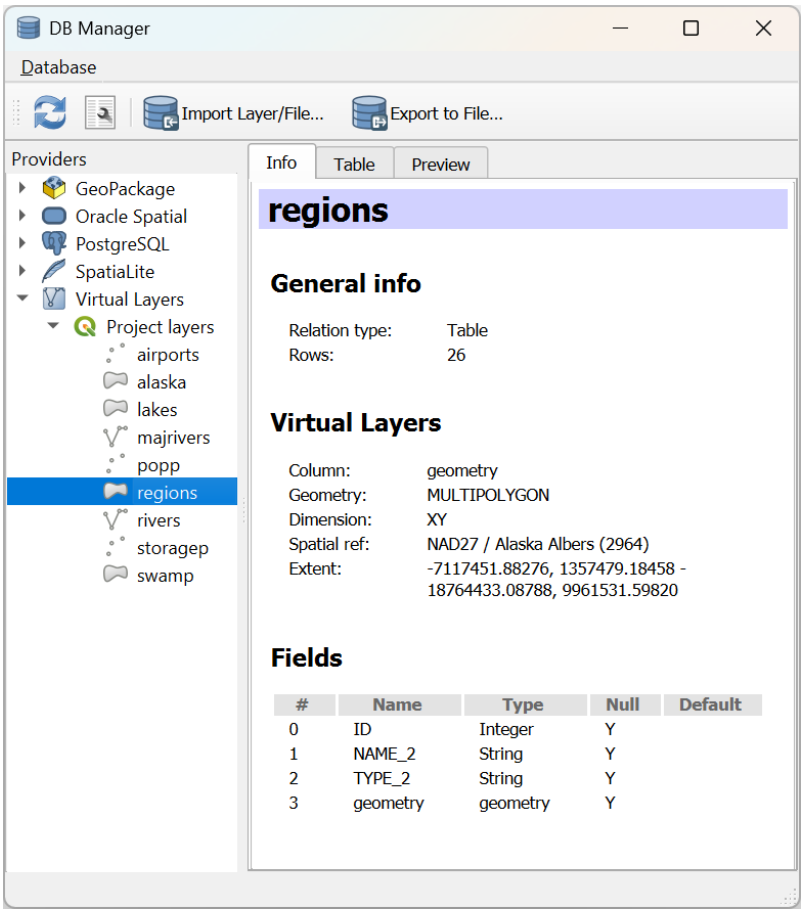


Слика25.3: The *Install from zip* tab

25.2 Korišćenje osnovnih QGIS dodataka

25.2.1 DB Manager Plugin

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.



Слика25.4: DB Manager dialog

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.

The *Schema* menu includes tools to create and delete (only if empty) schemas and, if topology is available (e.g. with PostGIS topology), to start a *TopoViewer*.

The *Table* menu allows you to create and edit tables and to delete tables and views. It is also possible to empty tables and to move tables between schemas. You can *Run Vacuum Analyze* for the selected table. *Vacuum* reclaims space and makes it available for reuse, and *analyze* updates statistics that is used to determine the most efficient way to execute a query. *Change Logging…* allows you to add change logging support to a table. Finally, you can *Import Layer/File…* and *Export to File…*.

Белешка

Using the DB Manager it is possible to add comments for tables and columns of a PostgreSQL Database.

The *Providers* window lists all existing databases supported by QGIS. With a double-click, you can connect to the database. With the right mouse button, you can rename and delete existing schemas and tables. Tables can also be added to the QGIS canvas with the context menu.

If connected to a database, the **main** window of the DB Manager offers four tabs. The *Info* tab provides information about the table and its geometry, as well as about existing fields, constraints and indexes. It allows you to create a spatial index on a the selected table. The *Table* tab shows the table, and the *Preview* tab renders the geometries as preview. When you open an *SQL Window*, it will be placed in a new tab.

Working with the SQL Window

You can use the DB Manager to execute SQL queries against your spatial database. Queries can be saved and loaded, and there the *SQL Query Builder* will help you formulate your queries. You can even view spatial output by checking *Load as new layer* and specifying *Column(s) with unique values* (IDs), *Geometry column* and *Layer name (prefix)*. It is possible to highlight a portion of the SQL to only execute that portion when pressing `Ctrl+R` or clicking the *Execute* button.

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:

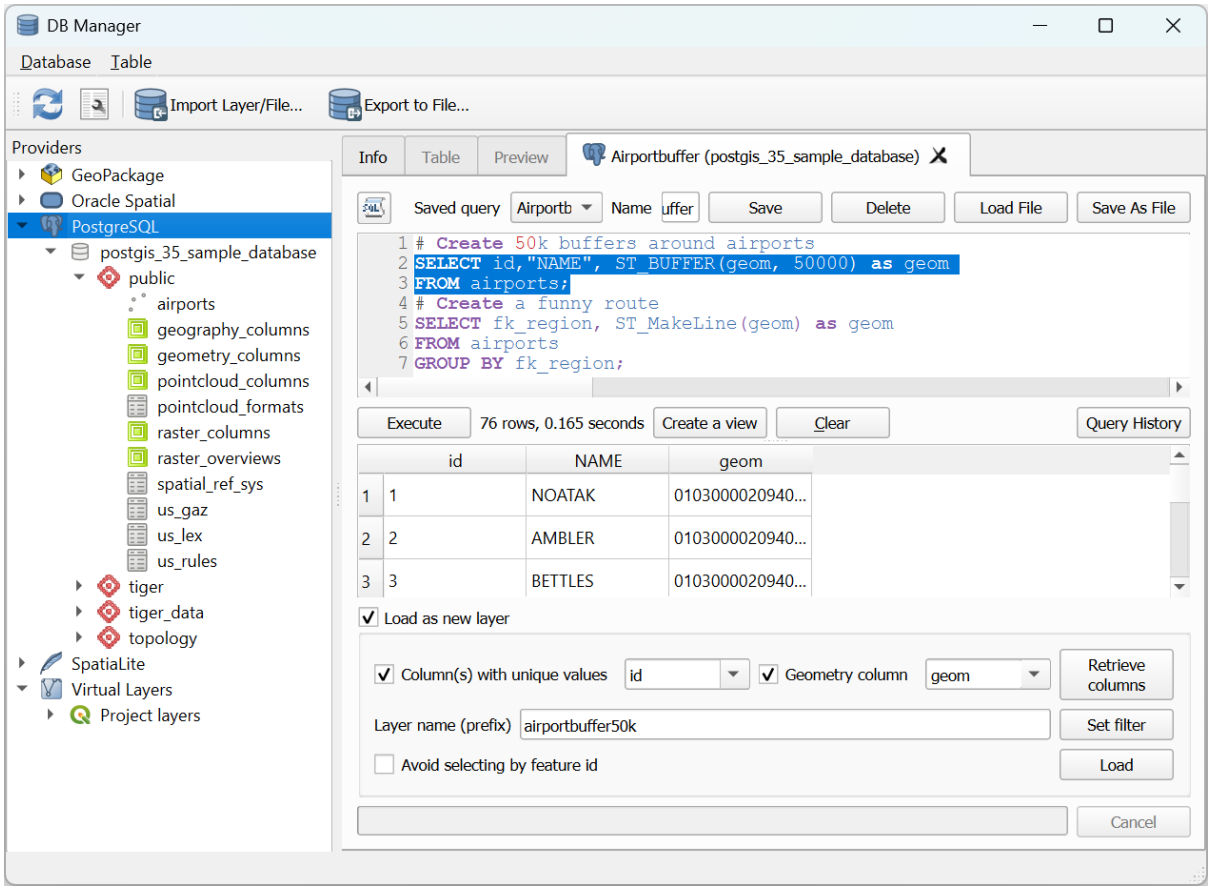
```
SELECT * FROM places WHERE name REGEXP '^A';
```

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.

The *Query History* button stores the last 20 queries of each database and provider.

Double clicking on an entry will add the string to the SQL window.



Слика25.5: Executing SQL queries in the DB Manager SQL window

Белешка

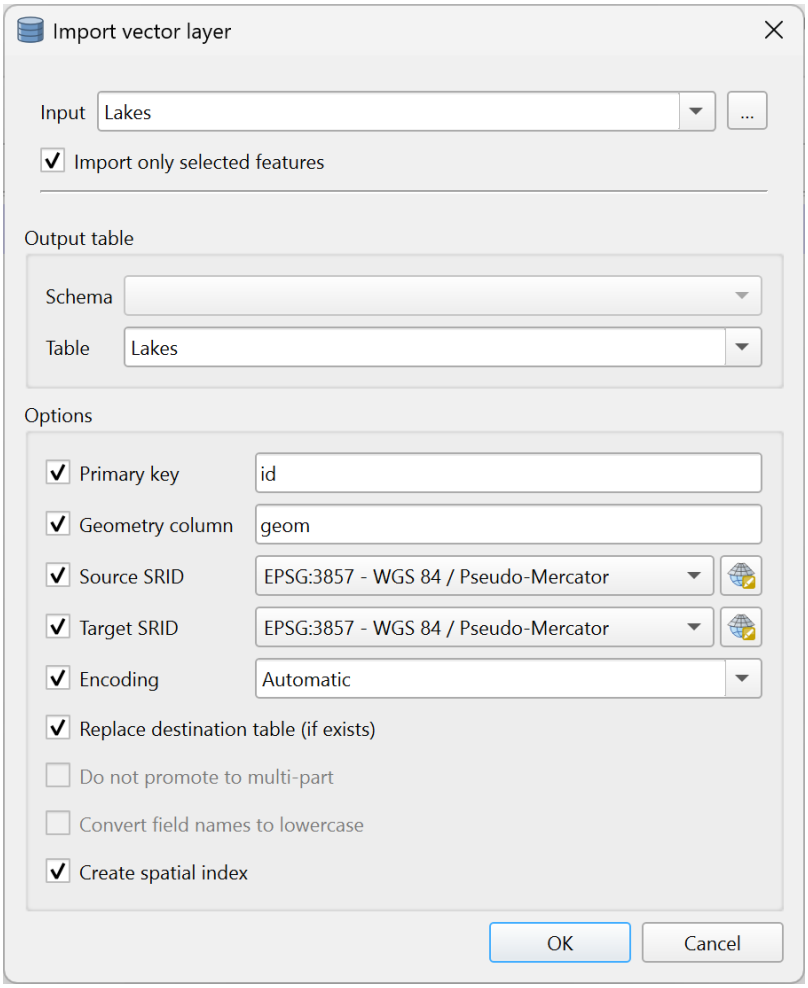
The SQL Window can also be used to create Virtual Layers. In that case, instead of selecting a database, select **QGIS Layers** under **Virtual Layers** before opening the SQL Window. See *Creating virtual layers* for instructions on the SQL syntax to use.

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.



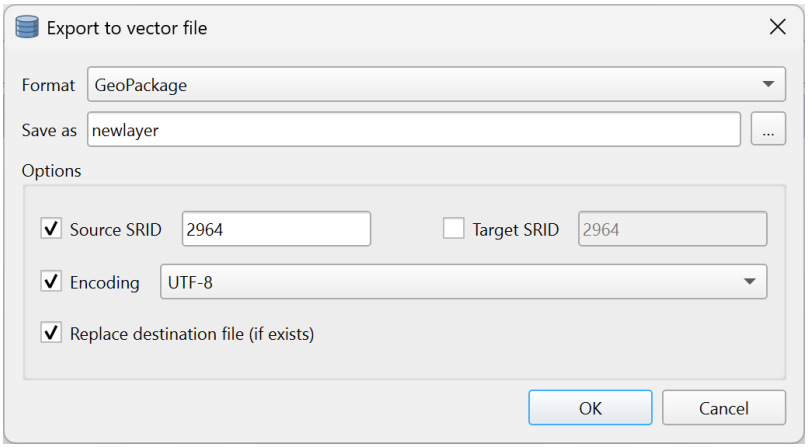
Слика25.6: Importing a vector layer into a spatial database using DB Manager

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.



Слика25.7: Exporting a table to a vector file using DB Manager

25.2.2 Geometry Checker Plugin

Geometry Checker is a powerful core plugin to check and fix the geometry validity of a layer. It is available from the *Vector* menu (*Check Geometries…*).

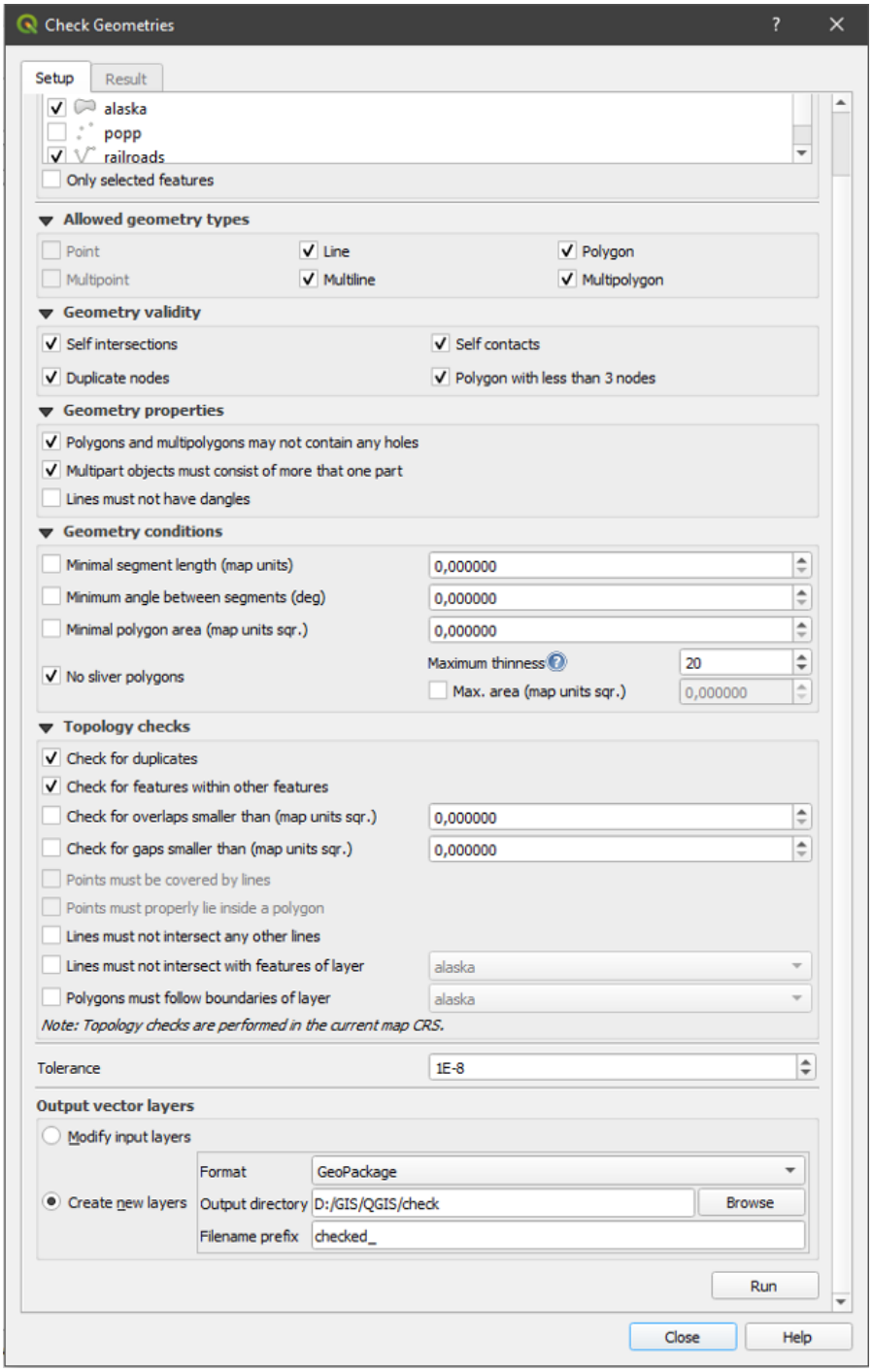
Configuring the checks

The *Check Geometries* dialog shows different grouped settings in the first tab (*Setup*):

- *Input vector layers*: to select the layers to check. A *Only selected features* checkbox can be used to restrict the checking to the geometries of the selected features.
- *Allowed geometry types* gives the chance to restrict the geometry type of the input layer(s) to:
  - Point
  - Multipoint
  - Line
  - Multiline
  - Polygon
  - Multipolygon
- *Geometry validity*. Depending on geometry types you can choose between:
  - *Self intersections*
  - *Duplicate nodes*
  - *Self contacts*
  - *Polygon with less than 3 nodes.*
- *Geometry properties*. Depending on geometry types, different options are available:

  - *Polygons and multipolygons may not contain any holes*
  - *Multipart objects must consist of more than one part*
  - *Lines must not have dangles*
- *Geometry conditions*. Allows you to add some condition to validate the geometries with:
  - *Minimal segment length (map units)* 1,00
  - *Minimum angle between segment (deg)* 1,00
  - *Minimal polygon area (map units sqr.)* 1,00
  - *No sliver polygons* with a *Maximum thinness* 1,00 and a *Max. area (map units sqr.)* 1,00
- *Topology checks*. Depending on geometry types, many different options are available:
  - *Checks for duplicates*
  - *Checks for features within other features*
  - *Checks for overlaps smaller than* 1,00
  - *Checks for gaps smaller than* 1,00
  - *Points must be covered by lines*
  - *Points must properly lie inside a polygon*
  - *Lines must not intersect any other lines*
  - *Lines must not intersect with features of layer*
  - *Polygons must follow boundaries of layer*
- *Tolerance*. You can define the tolerance of the check in map layer units.
- *Output vector layer* gives the choice to:
  - *Modify input layer*
  - *Create new layers*

When you are happy with the configuration, you can click on the *Run* button.



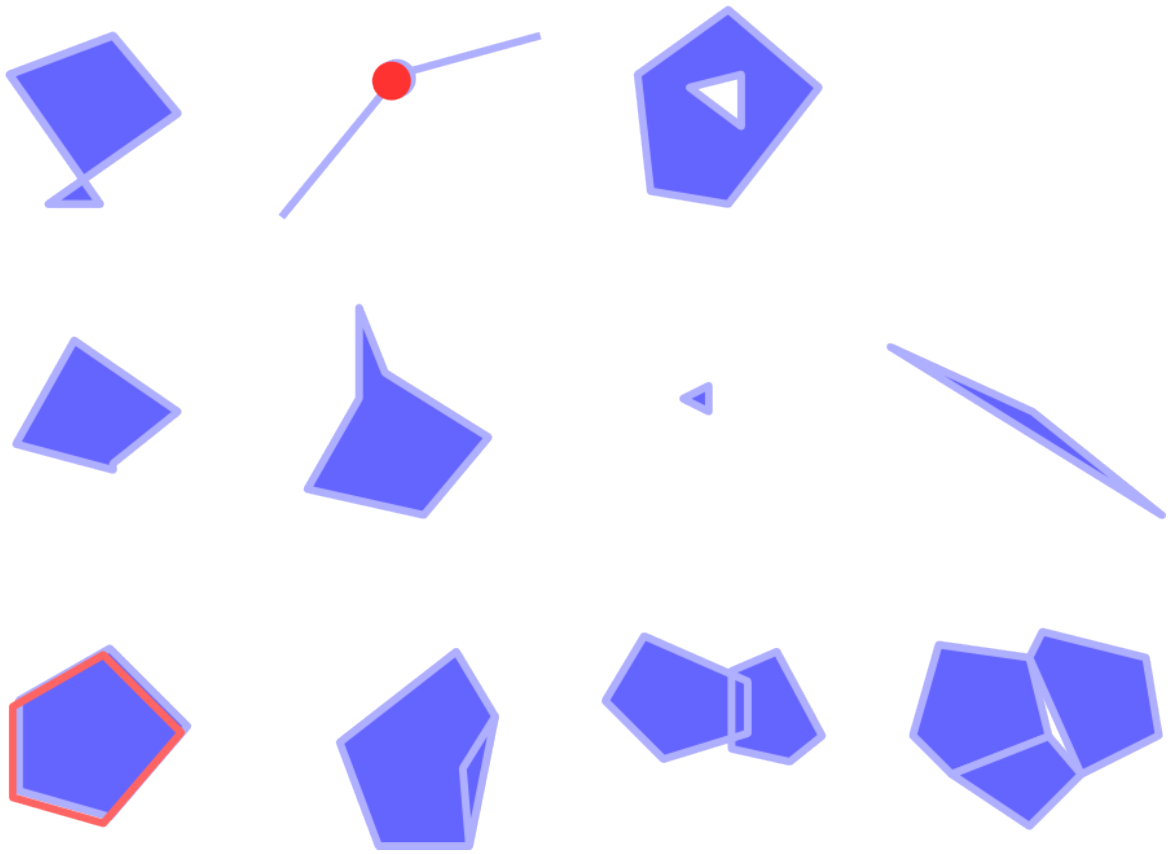
Слика25.8: The Geometry Checker Plugin

The *Geometry Checker Plugin* can find the following errors:

- Self intersections: a polygon with a self intersection
- Duplicate nodes: two duplicates nodes in a segment
- Holes: hole in a polygon
- Segment length: a segment length lower than a threshold
- Minimum angle: two segments with an angle lower than a threshold
- Minimum area: polygon area lower than a threshold

- Silver polygon: this error come from very small polygon (with small area) with a large perimeter
- Duplicates features
- Feature within feature
- Overlaps: polygon overlapping
- Gaps: gaps between polygons

The following figure shows the different checks made by the plugin.



Слика25.9: Some checks supported by the plugin

Analysing the results

The results appear in the second tab (*Result*) and as an overview layer of the errors in the canvas (its name has the default prefix `checked_`). A table lists the *Geometry check result* with one error per row and columns containing: the layer name, an ID, the error type, then the coordinates of the error, a value (depending on the type of the error) and finally the resolution column which indicates the resolution of the error. At the bottom of this table, you can *Export* the error into different file formats. You also have a counter with the number of total errors and fixed ones.

You can select a row to see the location of the error. You can change this behavior by selecting another action between *Error* (default), *Feature*, *Don't move*, and *Highlight selected features*.

Below the zoom action when clicking on the table row, you can:

- *Show selected features in attribute table*
- *Fix selected errors using default resolution*
- *Fix selected errors, prompt for resolution method* You will see a window to choose the resolution's method among which:

  - Merge with neighboring polygon with longest shared edge
  - Merge with neighboring polygon with largest area
  - Merge with neighboring polygon with identical attribute value, if any, or leave as is
  - Delete feature
  - No action
- *Error resolution settings* allows you to change the default resolution method depending on the error type

Савет

Fix multiple errors

You can fix multiple errors by selecting more than one row in the table with the *CTRL + click* action.

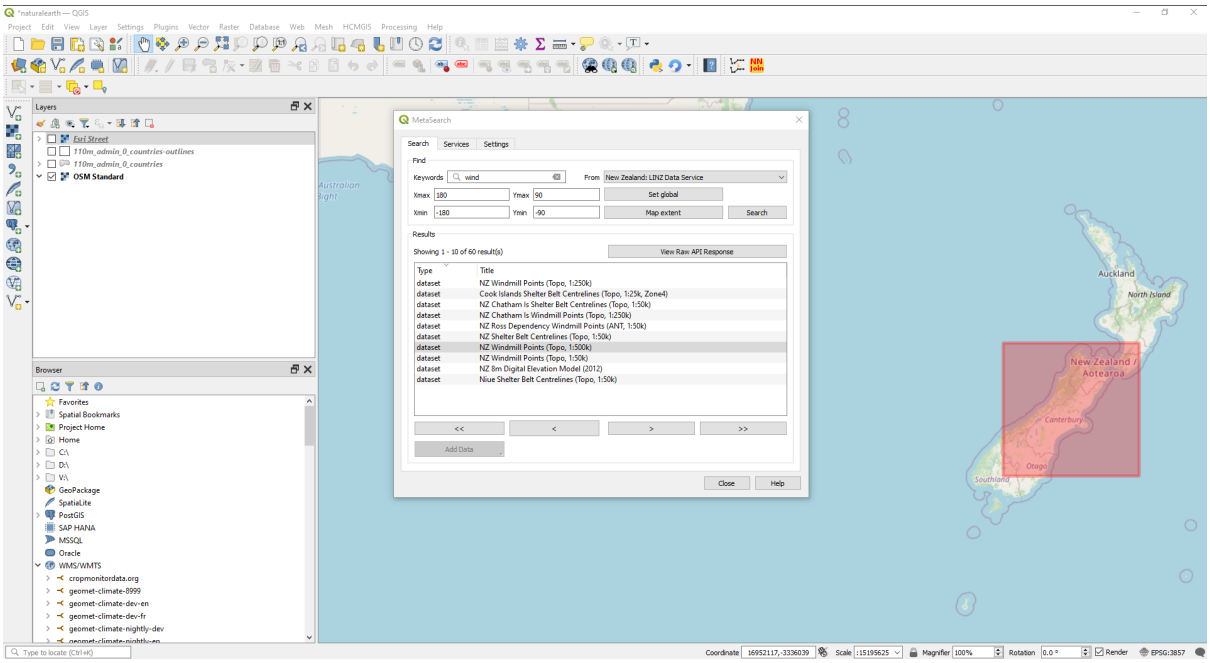
Finally, you can choose which *Attribute to use when merging features by attribute value*.

25.2.3 MetaSearch klijent kataloga

Introduction

MetaSearch is a QGIS plugin to interact with metadata catalog services, supporting both the OGC API - Records and OGC Catalog Service for the Web (CSW) standards.

MetaSearch provides an easy and intuitive approach and user-friendly interface to searching metadata catalogs within QGIS.



Слика25.10: Search and results of Services in MetaSearch

Working with Metadata Catalogs in QGIS

MetaSearch is included by default in QGIS, with all of its dependencies, and can be enabled from the QGIS Plugin Manager.

OGC API - Records

OGC API - Records is an OGC (Open Geospatial Consortium) standard for the discovery of geospatial resources on the Web.

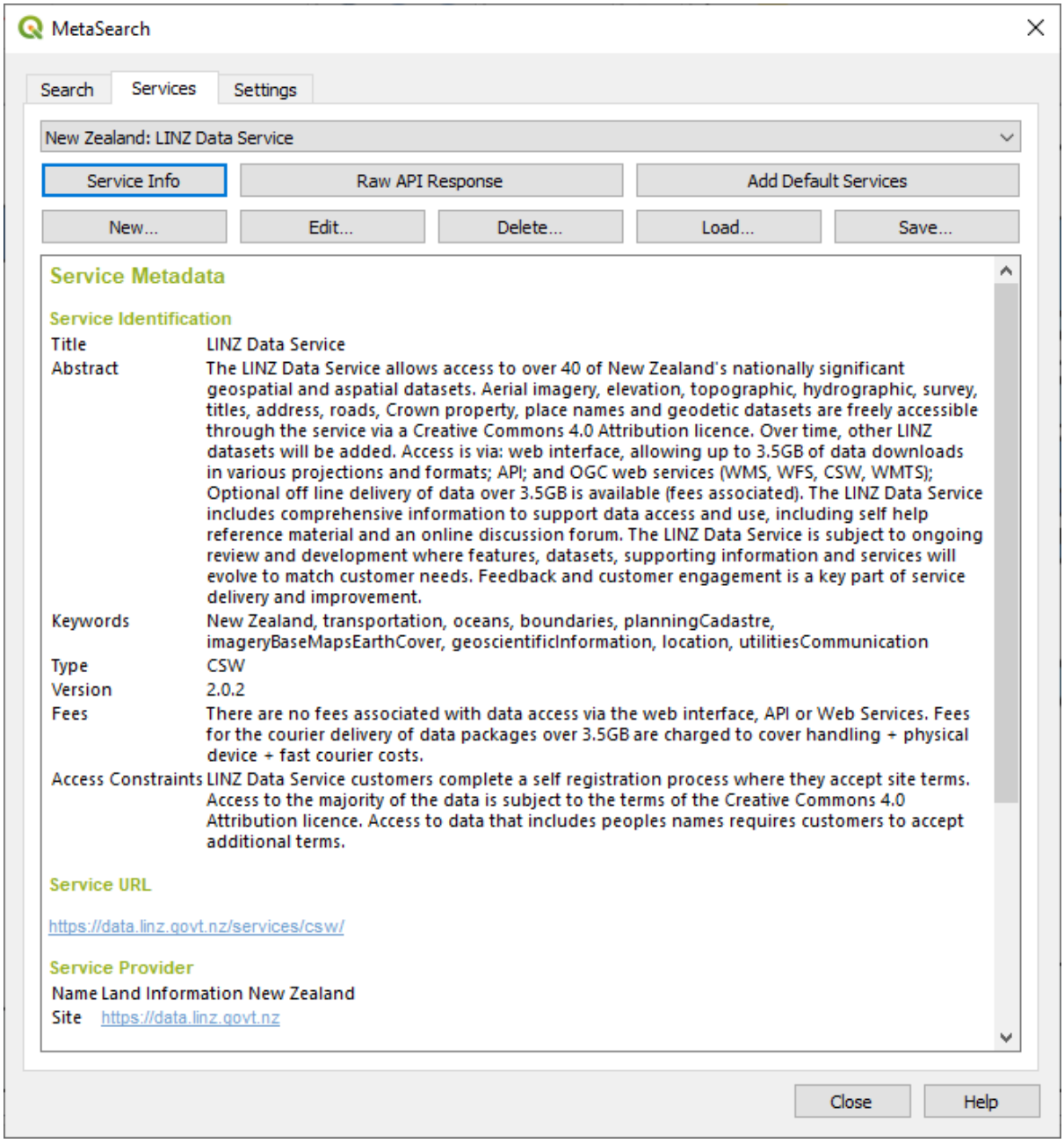
CSW (Catalog Service for the Web)

CSW (Catalog Service for the Web) is an OGC (Open Geospatial Consortium) specification that defines common interfaces to discover, browse and query metadata about data, services, and other potential resources.

Startup

To start MetaSearch, click the  icon or select *Web* ► *MetaSearch* ► *MetaSearch* via the QGIS main menu. The MetaSearch dialog will appear. The main GUI consists of three tabs: *Services*, *Search* and *Settings*.

Managing Catalog Services



Слика25.11: Managing Catalog Services

The *Services* tab allows the user to manage all available catalog services. MetaSearch provides a default list of Catalog Services, which can be added by pressing the *Add Default Services* button.

To find all listed Catalog Service entries, click the dropdown select box.

To add a Catalog Service entry:

1. Click the *New* button
2. Enter a *Name* for the service, as well as the *URL* (endpoint) and select the *Catalog Type*. Note that for OGC CSW 2.0.2 Catalogs, only the base URL is required (not a full GetCapabilities URL). For OGC API - Records Catalogs, the URL should be the path to the collection endpoint
3. If the Catalog requires authentication, enter the appropriate *User name* and *Password* credentials.

4. Click *OK* to add the service to the list of entries.

To edit an existing Catalog Service entry:

1. Select the entry you would like to edit
2. Click the *Edit* button
3. Modify the relevant parameters
4. Click *OK*.

To delete a Catalog Service entry, select the entry you would like to delete and click the *Delete* button. You will be asked to confirm deleting the entry.

MetaSearch allows loading and saving connections to an XML file. This is useful when you need to share settings between applications. Below is an example of the XML file format.

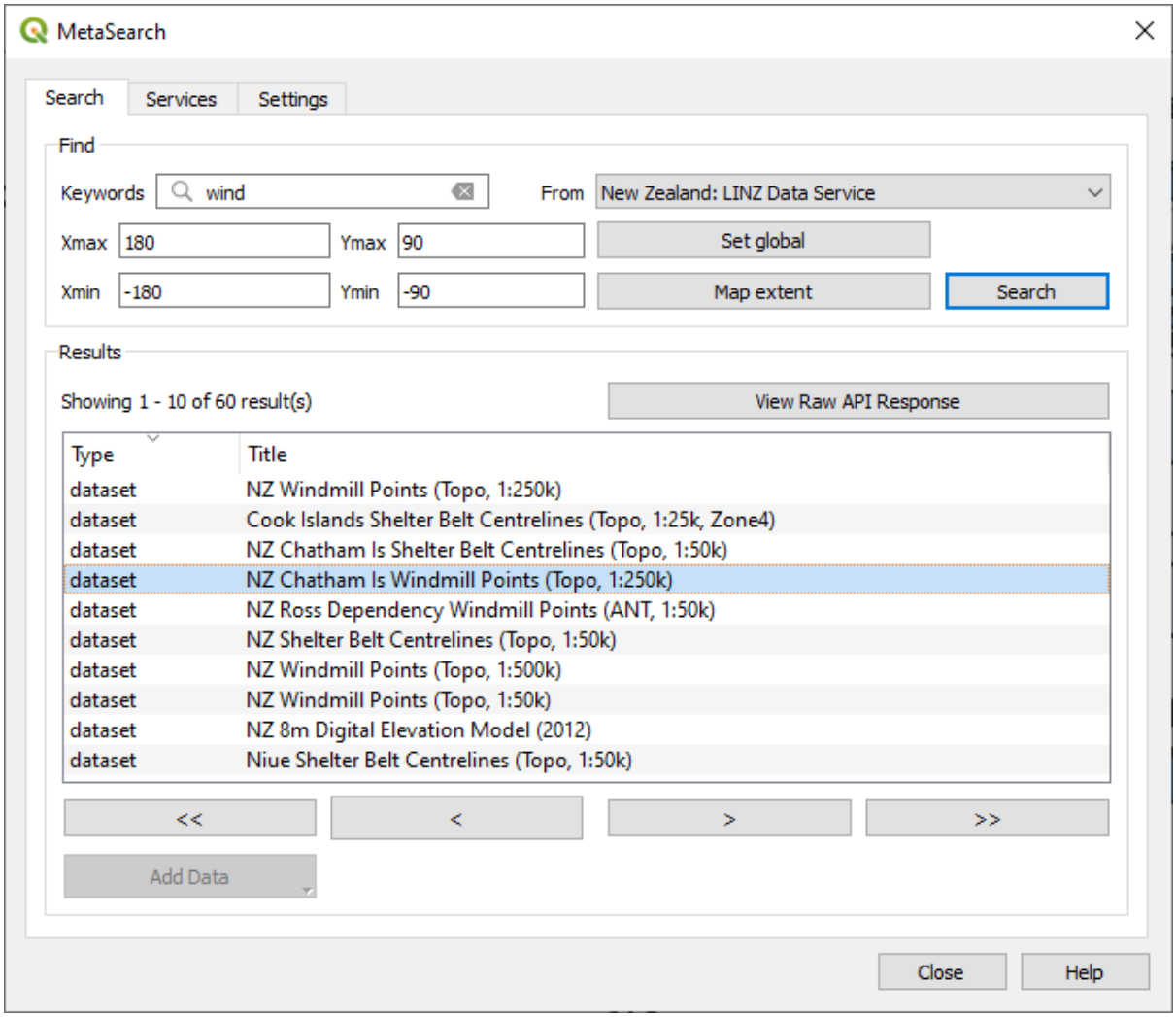
```
<?xml version="1.0" encoding="UTF-8"?>
<qgsCSWConnections version="1.0">
    <csw type="OGC CSW 2.0.2" name="Data.gov CSW" url="https://catalog.data.gov/
↪csw-all"/>
    <csw type="OGC CSW 2.0.2" name="Geonorge - National CSW service for Norway"␣
↪url="https://www.geonorge.no/geonetwork/srv/eng/csw"/>
    <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>
```

To load a list of entries:

1. Click the *Load* button. A new window will appear.
2. Click the *Browse* button and navigate to the XML file of entries you wish to load.
3. Click *Open*. The list of entries will be displayed.
4. Select the entries you wish to add from the list and click *Load*.

Click the *Service Info* button to display information about the selected Catalog Service such as service identification, service provider and contact information. If you would like to view the raw API response, click the *Raw API Response* button. A separate window will open displaying server information in raw JSON or XML format.

Searching Catalog Services



Слика25.12: Searching catalog services

The *Search* tab allows the user to query Catalog Services for data and services, set various search parameters and view results.

The following search parameters are available:

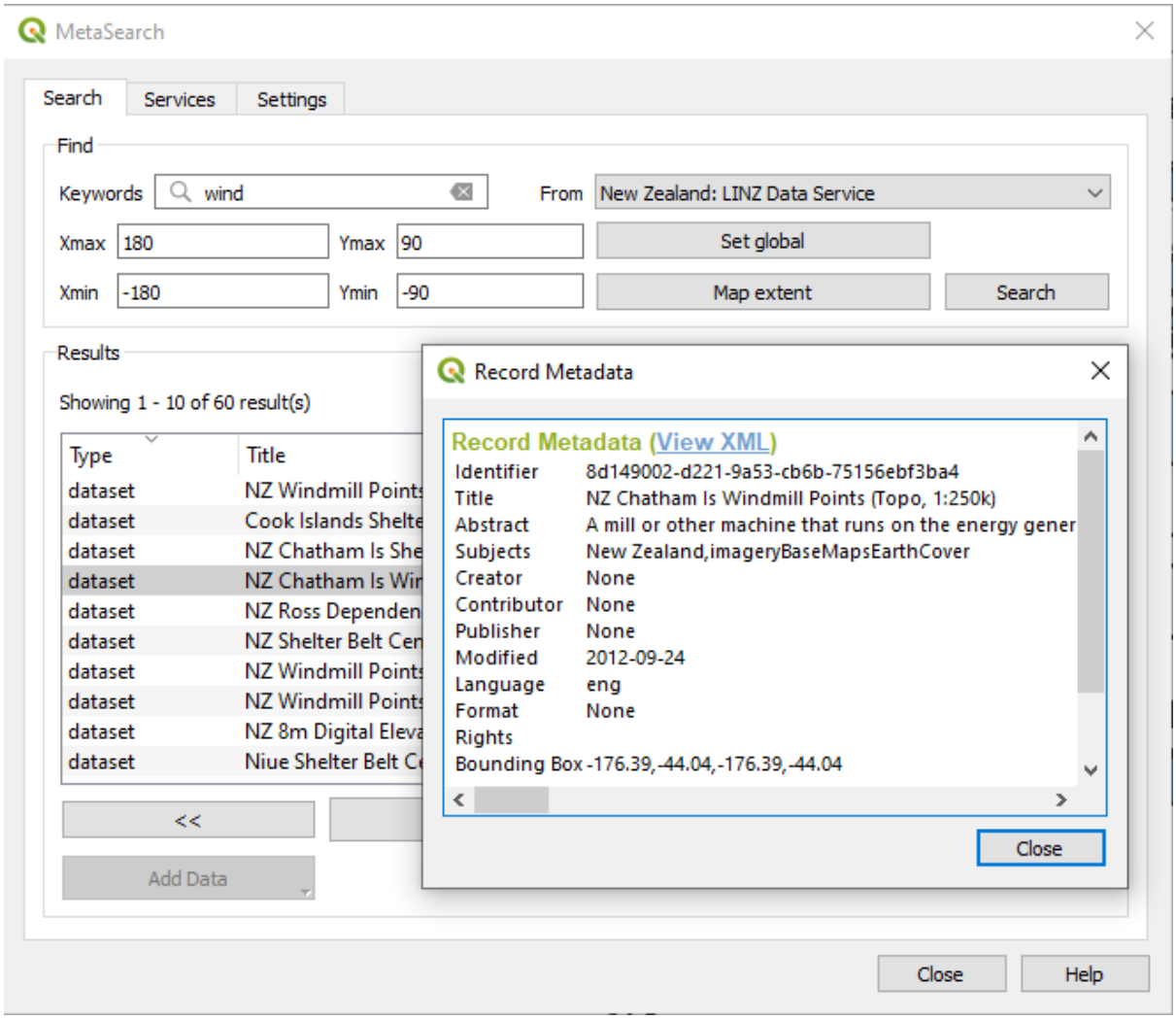
- *Keywords*: free text search keywords;
- *From*: the Catalog Service to perform the query against;
- **Bounding box**: the spatial area of interest to filter, defined by *Xmax*, *Xmin*, *Ymax*, and *Ymin*. Click *Set Global* to do a global search, click *Map Extent* to do a search in the visible area, or enter values manually.

Clicking the *Search* button will search the selected Metadata Catalog. Search results are displayed in a list, and can be sorted by clicking on the column header. You can navigate through search results with the directional buttons below the search results.

Select a result and:

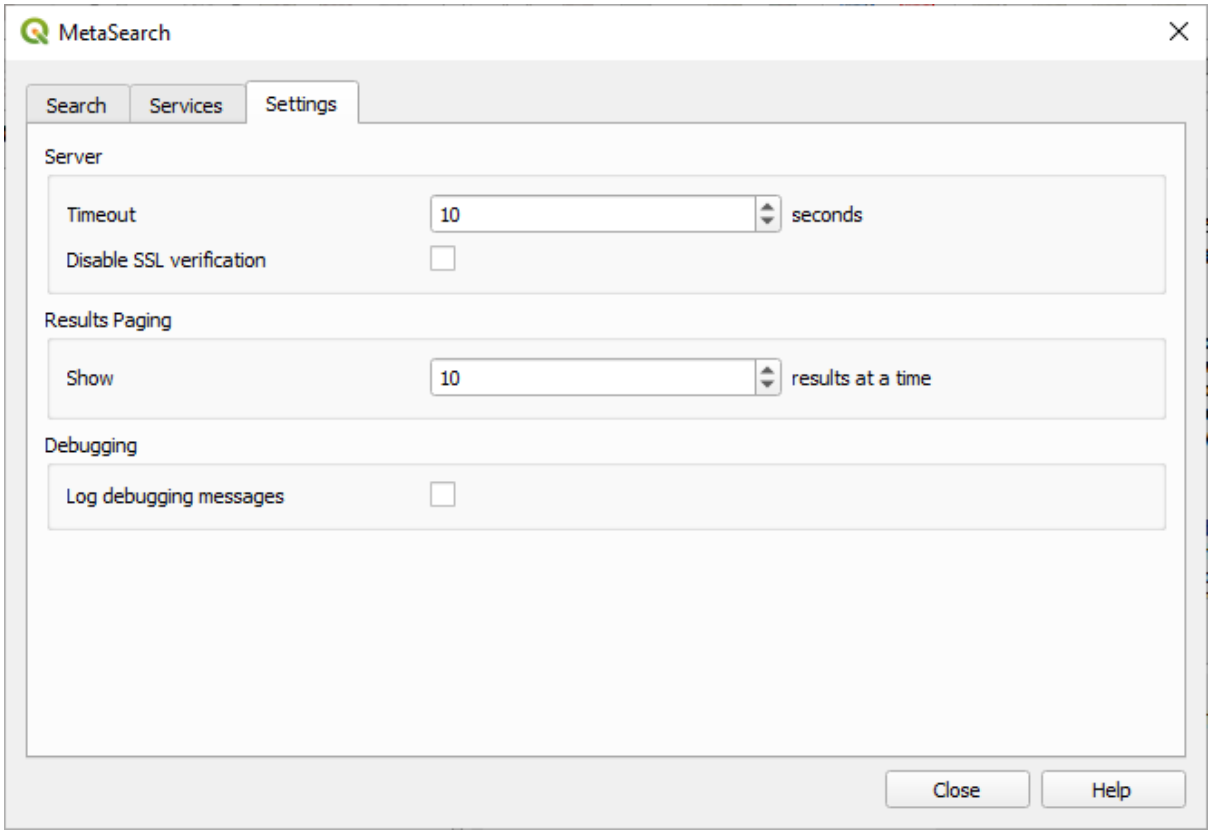
- Click the *View Raw API Response* button to open a window with the service response in raw JSON or XML format.
- If the metadata record has an associated bounding box, a footprint of the bounding box will be displayed on the map.

- Double-click the record to display the record metadata with any associated access links. Clicking a link opens the link in the user's web browser.
- If the record is a supported web service (WMS/WMTS, WFS, WCS, ArcGIS REST Service, etc.) or a GIS File, the *Add Data* button will be enabled. When clicking this button, MetaSearch will verify if this is a valid OWS. The service will then be added to the appropriate QGIS connection list, and the appropriate connection dialog will appear. If you choose *Add GIS File* the linked GIS file will be added to the current project.



Слика25.13: Metadata record display

Podešavanja



Слика25.14: MetaSearch settings

You can fine tune MetaSearch with the following *Settings*:

- *Server Timeout*: when searching metadata catalogs, the number of seconds for blocking connection attempt. Default value is 10.
- *Disable SSL verification*: option to switch off ssl verification.
- *Results paging*: when searching metadata catalogs, the number of results to show per page. Default value is 10.
- *Log debugging messages*: option to enable the logging of debugging messages in the *Log Messages Panel*.

Catalog Server Errors

In some cases, the Catalog will work in a web browser, but not in MetaSearch. This may be due to the Catalog server's configuration/setup. Catalog server providers should ensure URLs are consistent and up to date in their configuration (this is common in HTTP -> HTTPS redirection scenarios). Please see the pycsw FAQ item for a deeper explanation of the issue and fix. Although the FAQ item is pycsw specific it can also apply in general to other Catalog servers.

25.2.4 Dodatak za oflajn uređivanje

Za prikupljanje podataka, uobičajena je situacija rad sa laptopom ili mobilnim telefonom oflajn na terenu. Po povratku u mrežu, izmene je potrebno sinhronizovati sa glavnim izvorom podataka (npr. PostgreSQL bazom podataka). Ako više osoba istovremeno radi na istim skupovima podataka, teško je ručno spojiti izmene, čak i ako ljudi ne menjaju iste objekte.

Dodatak  Oflajn uređivanje automatizuje sinhronizaciju kopiranjem sadržaja izvora podataka u SpatiaLite ili GeoPackage bazu podataka i čuvanjem oflajn izmena u za to namenjene tabele. Nakon ponovnog povezivanja na mrežu, moguće je primeniti oflajn izmene na glavni skup podataka.

Da biste koristili dodatak:

1. Otvorite projekat sa nekim vektorskim slojevima (npr. iz Esri Shapefile, PostgreSQL ili WFS-T izvora podataka).
2. Pod pretpostavkom da ste već omogućili dodatak (pogledajte *Core and External plugins*) idite na *Baza podataka* ► *Oflajn uređivanje* ►  *Pretvori u oflajn projekat*. Otvara se istoimeni dijalog.
3. Izaberite *Tip skladišta*. Može biti tipa baze podataka *GeoPackage* ili *SpatiaLite*.
4. Koristite dugme *Pregledaj* da biste naznačili lokaciju baze podataka u kojoj će se čuvati *Oflajn podaci*. To može biti postojeća datoteka ili ona koju treba napraviti.
5. U odeljku *Izaberi udaljene slojeve* označite slojeve koje biste želeli da sačuvate. Sadržaj slojeva se čuva u tabele baze podataka.

   Белешка

   Pošto ciljni formati baza podataka nemaju nativnu podršku za liste, dodatak za oflajn uređivanje transformiše {niska, broj} polja liste u niskovna polja gde su vrednosti razdvojene zarezima. Ovo omogućava čitanje i uređivanje sadržaja tih polja kada ste oflajn.

   Ako biste želeli da rukujete i poljem iz originalnog sloja i oflajn sloja, možete se osloniti na funkcije izraza *try()* i *array*, npr.:

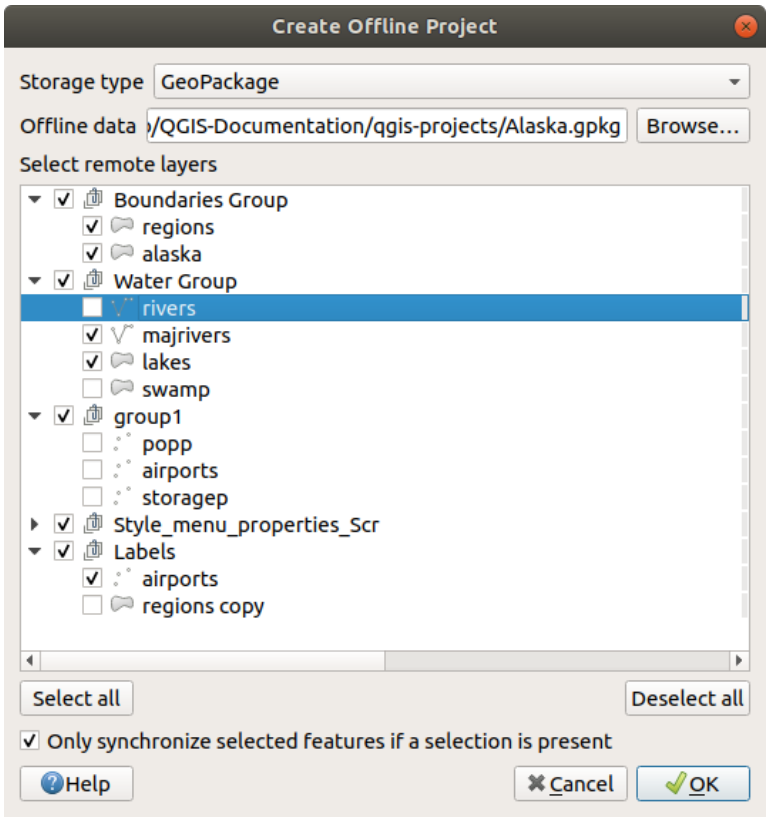
   ```
   try(array_contains("field",1),array_contains(string_to_array("field"),1))
   ```

6. Možete označiti  *Sinhronizuj samo izabrane objekte ako je izbor prisutan* čime se omogućava čuvanje i rad samo na podskupu. To može biti neprocenjivo u slučaju velikih slojeva.

   To je sve!
7. Sačuvajte svoj projekat i ponesite ga na teren.
8. Uredite slojeve oflajn.
9. Nakon ponovnog povezivanja, otpremite izmene pomoću *Baza podataka* ► *Oflajn uređivanje* ►  *Sinhronizuj*.

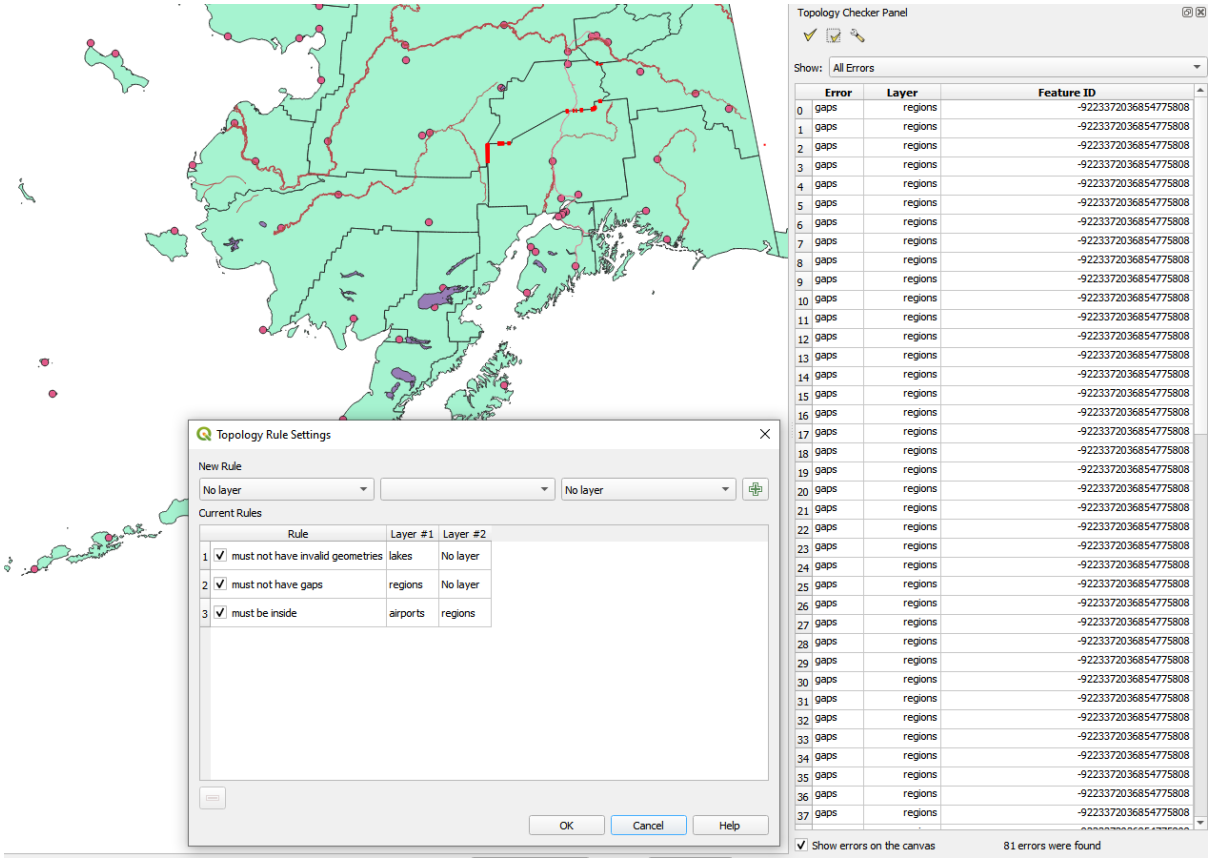
Белешка

Slojevi koji se koriste oflajn označeni su ikonom u panelu *Slojevi*.



Слика25.15: Napravi oflajn projekat

25.2.5 Topology Checker Plugin



Слика25.16: The Topology Checker Plugin

Topology describes the relationships between points, lines and polygons that represent the features of a geographic region. With the Topology Checker plugin, you can look over your vector files and check the topology with several topology rules. These rules check with spatial relations whether your features 'Equal', 'Contain', 'Cover', are 'CoveredBy', 'Cross', are 'Disjoint', 'Intersect', 'Overlap', 'Touch' or are 'Within' each other. It depends on your individual questions which topology rules you apply to your vector data (e.g., normally you won't accept overshoots in line layers, but if they depict dead-end streets you won't remove them from your vector layer).

QGIS has a built-in topological editing feature, which is great for creating new features without errors. But existing data errors and user-induced errors are hard to find. This plugin helps you find such errors through a list of rules.

To enable the *Topology checker* plugin:

1. Go to *Plugins* menu
2. Open *Manage and Install plugins* and choose *Installed*
3. Enable *Topology checker*
4. Close the *Plugin manager* dialog. A *Topology checker* entry is added to the *Vector* menu.

After you enabled *Topology checker* open it and choose *Configure* to create your topology rules.

On **point layers** the following rules are available:

- *Must be covered by*: Here you can choose a vector layer from your project. Points that aren't covered by the given vector layer occur in the 'Error' field.
- *Must be covered by endpoints of*: Here you can choose a line layer from your project.

- *Must be inside*: Here you can choose a polygon layer from your project. The points must be inside a polygon. Otherwise, QGIS writes an 'Error' for the point.
- *Must not have duplicates*: Whenever a point is represented twice or more, it will occur in the 'Error' field.
- *Must not have invalid geometries*: Checks whether the geometries are valid.
- *Must not have multi-part-geometries*: All multi-part points are written into the 'Error' field.

On **line layers**, the following rules are available:

- *End points must be covered by*: Here you can select a point layer from your project.
- *Must not have dangles*: This will show the overshoots in the line layer.
- *Must not have duplicates*: Whenever a line feature is represented twice or more, it will occur in the 'Error' field.
- *Must not have invalid geometries*: Checks whether the geometries are valid.
- *Must not have multi-part geometries*: Sometimes, a geometry is actually a collection of simple (single-part) geometries. Such a geometry is called multi-part geometry. If it contains just one type of simple geometry, we call it multi-point, multi-linestring or multi-polygon. All multi-part lines are written into the 'Error' field.
- *Must not have pseudos*: A line geometry's endpoint should be connected to the endpoints of two other geometries. If the endpoint is connected to only one other geometry's endpoint, the endpoint is called a pseudo node.

On **polygon layers**, the following rules are available:

- *Must contain*: Polygon layer must contain at least one point geometry from the second layer.
- *Must not have duplicates*: Polygons from the same layer must not have identical geometries. Whenever a polygon feature is represented twice or more it will occur in the 'Error' field.
- *Must not have gaps*: Adjacent polygons should not form gaps between them. Administrative boundaries could be mentioned as an example (US state polygons do not have any gaps between them…).
- *Must not have invalid geometries*: Checks whether the geometries are valid. Some of the rules that define a valid geometry are:
  - Polygon rings must close.
  - Rings that define holes should be inside rings that define exterior boundaries.
  - Rings may not self-intersect (they may neither touch nor cross one another).
  - Rings may not touch other rings, except at a point.
- *Must not have multi-part geometries*: Sometimes, a geometry is actually a collection of simple (single-part) geometries. Such a geometry is called multi-part geometry. If it contains just one type of simple geometry, we call it multi-point, multi-linestring or multi-polygon. For example, a country consisting of multiple islands can be represented as a multi-polygon.
- *Must not overlap*: Adjacent polygons should not share common area.
- *Must not overlap with*: Adjacent polygons from one layer should not share common area with polygons from another layer.

When you create a *New rule* click on the  Add rule to include it to the *Current rules*. You can enable or disable individual rules by clicking on the checkbox. Right-clicking over a rule provides the following options:

- *Select All* the rules
- *Activate* or *Deactivate* the selected rules
- *Toggle activation* of selected rules
- *Delete* selected rules. This can also be achieved with the Delete selected rules button.

Press *OK* and then choose from the *Topology checker* panel:

- Validate All: applies the active rules to all the features of the involved layer(s)
- or Validate Extent: applies the active rules to the features of the involved layer(s), within the current map canvas. The button is kept pushed and the results will update as the map canvas extent changes.

Errors will show up in the table of results containing type of error, layer and feature ID. Use *Filter errors by rule* menu to filter the errors to a specific error type.

Check *Show errors on the canvas* to show error location on the canvas. Clicking a row in the table will zoom the map canvas to the concerned feature, where you can use *QGIS digitizing tools* to fix the error.

Ispod je lista osnovnih dodataka koji se isporučuju uz QGIS. Nisu nužno podrazumevano omogućeni.

Ikona	Dodatak	Opis	Referenca u priručniku
	DB Manager	Upravljanje bazama podataka unutar QGIS-a	*DB Manager Plugin*
	Provera geometrije	Provera i popravka grešaka u vektorskim geometrijama	*Geometry Checker Plugin*
	GRASS 7	Funkcionalnosti GRASS-a	*GRASS GIS Integration*
	GRASS GIS provajder	Funkcionalnosti GRASS GIS-a za Obradu	*GRASS GIS Integration*
	MetaSearch klijent kataloga	Rad sa servisima kataloga metapodataka (CSW)	*MetaSearch klijent kataloga*
	Uređivanje van mreže	Uređivanje van mreže i sinhronizacija sa bazom podataka	*Dodatak za oflajn uređivanje*
	Obrada	Okvir za obradu prostornih podataka	*QGIS okvir za obradu*
	Provera topologije	Pronalaženje topoloških grešaka u vektorskim slojevima	*Topology Checker Plugin*

Белешка

Da biste koristili osnovne dodatke GRASS 7 i GRASS GIS provajder, oni moraju biti podešeni. Informacije možete pronaći *ovde*.

25.3 QGIS Python console

As you will see later in this chapter, QGIS has been designed with a plugin architecture. Plugins can be written in Python, a very famous language in the geospatial world.

QGIS brings a Python API (see PyQGIS Developer Cookbook for some code sample) to let the user interact with its objects (layers, feature or interface). QGIS also has a Python console.

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 The Interactive Console

The console is a Python interpreter that allows you to execute Python commands. Modules from QGIS (analysis, core, gui, server, processing, 3d) and Qt (QtCore, QtGui, QtNetwork, QtWidgets, QtXml) as well as Python's math, os, re and sys modules are already imported and can be used directly.

The interactive console is composed of a toolbar, an input area and an output one.



Слика25.17: The Python Console

Traka sa alatkama

The toolbar proposes the following tools:

- Clear Console to wipe the output area;
- Run Command available in the input area: same as pressing `Enter`;
- Show Editor: toggles *The Code Editor* visibility;
- Options...: opens a dialog to configure *console properties*;
- Help... provides a menu to access various documentation:
  - *Python Console Help* (the current page)
  - PyQGIS API documentation
  - PyQGIS Cookbook
- Dock Code Editor to dock or undock the panel in QGIS interface

Input area

The Console input area main features are:

- Code completion, highlighting syntax and calltips for the following APIs:
  - Python
  - PyQGIS
  - PyQt5
  - QScintilla2
  - osgeo-gdal-ogr
  - osgeo-geos

- `Ctrl+Alt+Space` to view the auto-completion list if enabled in the *Python settings*;
- Execute code snippets from the input area by typing and pressing `Enter` or *Run Command*;
- Execute code snippets from the output area using the *Enter Selected* from the contextual menu or pressing `Ctrl+E`;
- Browse the command history from the input area using the `Up` and `Down` arrow keys and execute the command you want;
- `Ctrl+Shift+Space` to view the command history: double-clicking a row will execute the command. The *Command History* dialog can also be accessed from context menu of input area;
- Save and clear the command history. The history will be saved into the `console_history.txt` file under the active *user profile* folder;
- Type the following special commands:
  - `?` to show a help of the Python Console
  - `_api` to open QGIS C++ API documentation or `_api(object)` for a specific object documentation (in QGIS C++ API or Qt API documentation)
  - `_pyqgis` to open QGIS Python API documentation or `_pyqgis(object)` for a specific object documentation (in QGIS Python API or Qt API documentation)
  - `_cookbook` to open PyQGIS Cookbook.
  - `!` followed by a command to execute Shell commands from the Python Console. The console will start a subprocess, and forward its output to the Python Console Output. While the subprocess is running, the Python Console Input switches to STDIN mode and forwards entered character to the child process. This makes it possible to send confirmation when the child program asks for it. When the Console is in STDIN mode, pressing `Ctrl+C` kills the subprocess. It is also possible to affect the result of a command to a variable with the syntax `var = !cmd`

```
>>> !echo QGIS Rocks!
QGIS Rocks

>>> !gdalinfo --version
GDAL 3.10.3, released 2025/04/01

>>> !pip install black
# Install black python formatter using pip (if available)

>>> sql_formats = !ogrinfo --formats | grep SQL
>>> sql_formats
['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']
```

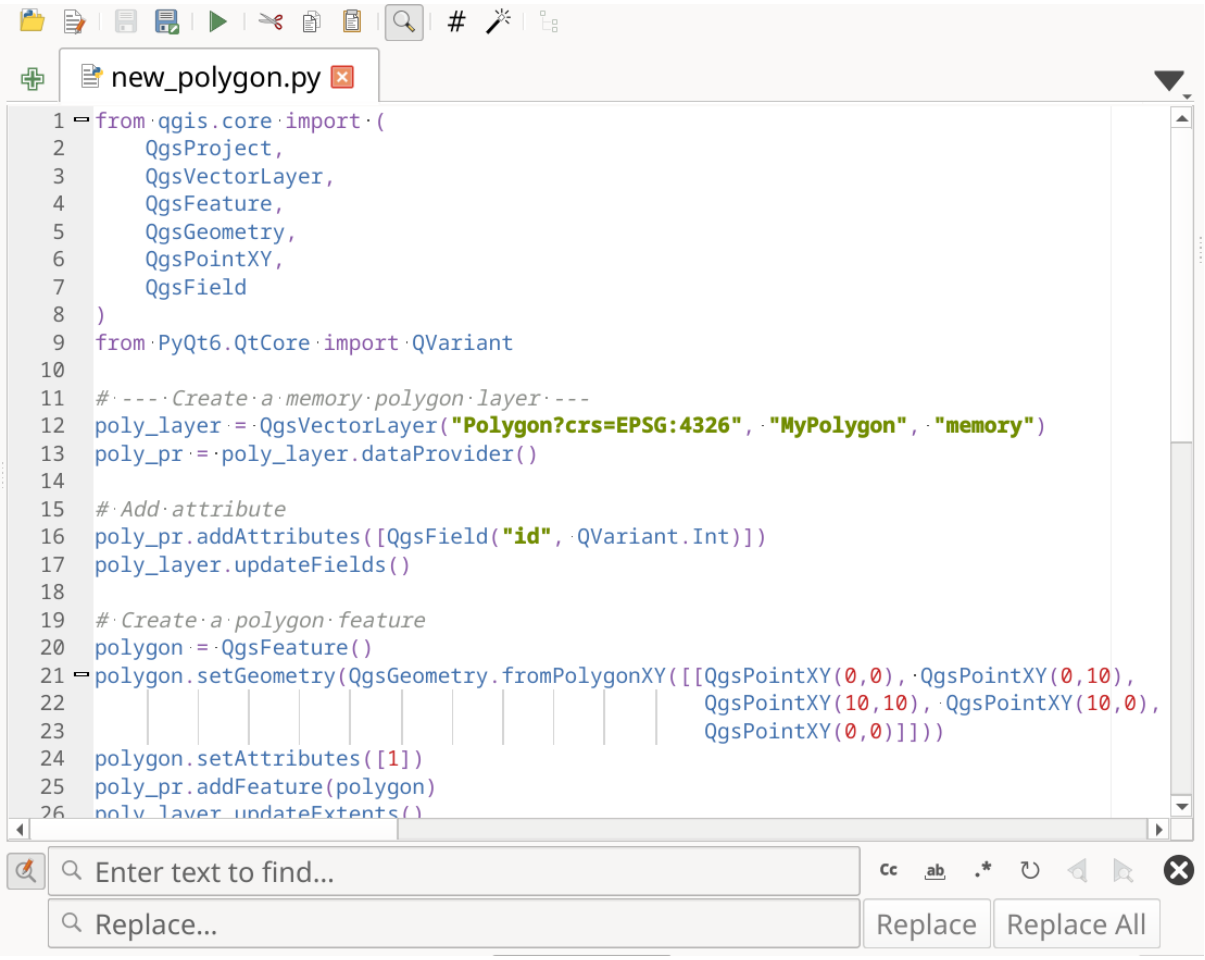
Савет

Reuse executed commands from the output panel

You can execute code snippets from the output panel by selecting some text and pressing `Ctrl+E`. No matter if selected text contains the interpreter prompt (`>>>`, `...`).

25.3.2 The 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.



Слика25.18: The Python Console 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.

Савет

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.

Табела25.1: Python Console Editor Tools

Tool name	Function	Location
Open script…	Loads a Python file in the code editor dialog, as a new tab	Traka sa alatkama
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. • Use the default Desktop Environment shortcut to find next/previous (`Ctrl+G` and `Shift+Ctrl+G`); • Automatically find first match when typing in find box; • Set initial find string to selection when opening find; • 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. • Pressing `Esc` closes the find bar.	
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.	

наставак на следећој страни

Табела 25.1 – настављено са претходне стране

Tool name	Function	Location
Object inspector…	Shows and hides a dedicated browser with a tree structure for classes and functions available in the script. Click an item for 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 settings* dialog.	

ГЛАВА 26

Dodaci

26.1 Appendix C: QGIS File Formats

26.1.1 QGS/QGZ - The QGIS Project File Format

The **QGS** format is an XML format for storing QGIS projects. The **QGZ** format is a compressed (zip) archive containing a QGS file and a QGD file. The **QGD** file is the associated sqlite database of the qgis project that contain auxiliary data for the project. If there are no auxiliary data, the QGD file will be empty.

A QGIS file contains everything that is needed for storing a QGIS project, including:

- project title
- project CRS
- the layer tree
- snapping settings
- relations
- the map canvas extent
- project models
- legend
- mapview docks (2D and 3D)
- the layers with links to the underlying datasets (data sources) and other layer properties including extent, SRS, joins, styles, renderer, blend mode, opacity and more.
- project properties

The figures below show the top level tags in a QGS file and the expanded `ProjectLayers` 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>
```

Слика26.1: The top level tags in a QGS file

```
−<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>
```

Слика26.2: The expanded top level ProjectLayers tag of a QGS file

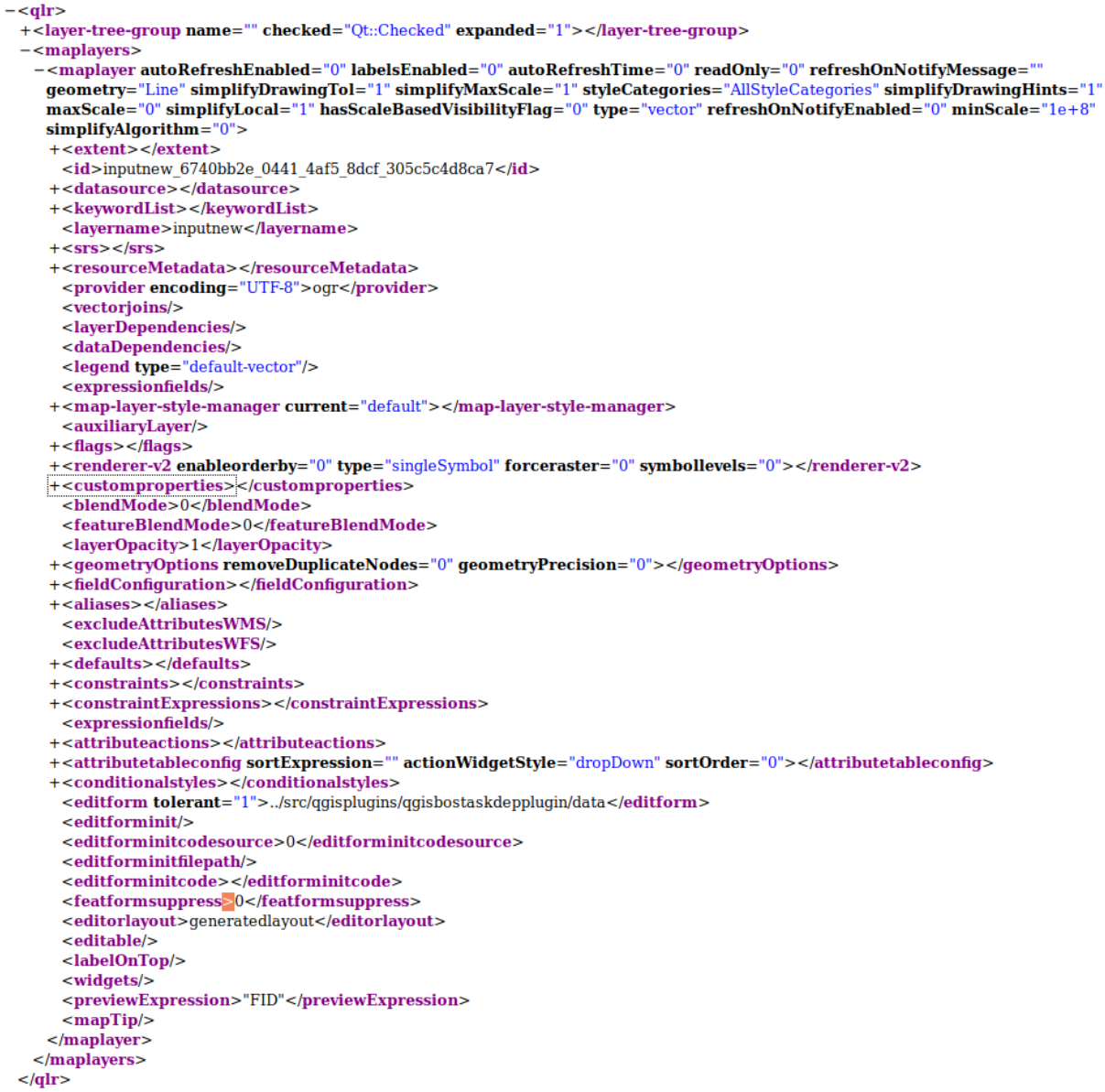
26.1.2 QLR - The QGIS Layer Definition file

A Layer Definition file (QLR) is an XML file that contains a pointer to the layer data source in addition to QGIS style information for the layer.

The use case for this file is simple: To have a single file for opening a data source and bringing in all the related style information. QLR files also allow you to mask the underlying datasource in an easy to open file.

An example of QLR usage is for opening MS SQL layers. Rather than having to go to the MS SQL connection dialog, connect, select, load and finally style, you can simply add a .qlr file that points to the correct MS SQL layer with all the necessary style included.

In the future a .qlr file may hold a reference to more than one layer.



Слика26.3: The top level tags of a QLR file

26.1.3 QML - The QGIS Style File Format

QML is an XML format for storing layer styling.

A QML file contains all the information QGIS can handle for the rendering of feature geometries including symbol definitions, sizes and rotations, labelling, opacity and blend mode and more.

The figure below shows the top level tags of a QML file (with only `renderer_v2` and its `symbol` tag expanded).

```
−<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>
```

Слика26.4: The top level tags of a QML file (only the renderer_v2 tag with its symbol tag is expanded)

26.2 Appendix D: QGIS R script syntax

Contributed by Matteo Ghetta - funded by Scuola Superiore Sant'Anna

Writing R scripts in Processing is a bit tricky because of the special syntax.

A Processing R script starts with defining its **Inputs** and **Outputs**, each preceded with double hash characters (`##`).

Before the inputs, the group to place the algorithm in can be specified. If the group already exists, the algorithm will be added to it, if not, the group will be created. In the example below, the name of the group is *My group*:

```
##My Group=group
```

26.2.1 Inputs

All input data and parameters have to be specified. There are several types of inputs:

- vector: `##Layer = vector`
- vector field: `##F = Field Layer` (where *Layer* is the name of an input vector layer the field belongs to)
- raster: `##r = raster`
- table: `##t = table`
- number: `##Num = number`
- string: `##Str = string`
- boolean: `##Bol = boolean`
- elements in a dropdown menu. The items must be separated with semicolons `;`: `##type=selection point;lines;point+lines`

26.2.2 Izlazi

As for the inputs, each output has to be defined at the beginning of the script:

- vector: `##output= output vector`
- raster: `##output= output raster`
- table: `##output= output table`
- plots: `##output_plots_to_html` (##showplots in earlier versions)
- To show R output in the *Result Viewer*, put > in front of the command whose output you would like to show.

26.2.3 Syntax Summary for QGIS R scripts

A number of input and output parameter types are offered.

Input parameter types

Parametar	Syntax example	Returning objects
vector	Layer = vector	sf object (or SpatialDataFrame object, if ##load_vector_using_rgdal is specified)
vector point	Layer = vector point	sf object (or SpatialDataFrame object, if ##load_vector_using_rgdal is specified)
vector line	Layer = vector line	sf object (or SpatialDataFrame object, if ##load_vector_using_rgdal is specified)
vector polygon	Layer = vector polygon	sf object (or SpatialPolygonsDataFrame object, if ##load_vector_using_rgdal is used)
multiple vector	Layer = multiple vector	sf object (or SpatialDataFrame objects if ##load_vector_using_rgdal is specified)
table	Layer = table	dataframe conversion from csv, default object of `read.csv` function
field	Field = Field Layer	name of the Field selected, e.g. `"Area"`
raster	Layer = raster	RasterBrick object, default object of `raster` package
multiple raster	Layer = multiple raster	RasterBrick objects, default object of `raster` package
number	N = number	integer or floating number chosen
string	S = string	string added in the box
longstring	LS = longstring	string added in the box, could be longer then the normal string
selection	S = selection first;second;third	string of the selected item chosen in the dropdown menu
crs	C = crs	string of the resulting CRS chosen, in the format: `"EPSG:4326"`
extent	E = extent	Extent object of the `raster` package, you can extract values as `E@xmin`
point	P = point	when clicked on the map, you have the coordinates of the point
file	F = file	path of the file chosen, e.g. „/home/matteo/file.txt”
folder	F = folder	path of the folder chosen, e.g. „/home/matteo/Downloads”

A parameter can be **OPTIONAL**, meaning that it can be ignored.

In order to set an input as optional, you add the string `optional` **before** the input, e.g:

```
##Layer = vector
##Field1 = Field Layer
##Field2 = optional Field Layer
```

Output parameter types

Parametar	Syntax example
vector	Output = output vector
raster	Output = output raster
table	Output = output table
file	Output = output file

Белешка

You can save plots as `png` from the *Processing Result Viewer*, or you can choose to save the plot directly from the algorithm interface.

Script body

The script body follows R syntax and the **Log** panel can help you if there is something wrong with your script.

Remember that you have to load all additional libraries in the script:

```
library(sp)
```

26.2.4 Primeri

Example with vector output

Let's take an algorithm from the online collection that creates random points from the extent of an input layer:

```
##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)
```

Explanation (per line in the script):

1. `Point pattern analysis` is the group of the algorithm
2. `Layer` is the input **vector** layer
3. `Size` is a **numerical** parameter with a default value of 10
4. `Output` is the **vector** layer that will be created by the algorithm
5. `library(sp)` loads the **sp** library
6. `spatpoly = as(Layer, "Spatial")` translate to an sp object
7. Call the `spsample` function of the `sp` library and run it using the input defined above (`Layer` and `Size`)
8. Create a *SpatialPointsDataFrame* object using the `SpatialPointsDataFrame` function
9. Create the output vector layer using the `st_as_sf` function

That's it! Just run the algorithm with a vector layer you have in the QGIS Legend, choose the number of random point. The resulting layer will be added to your map.

Example with raster output

The following script will perform basic ordinary kriging to create a raster map of interpolated values from a specified field of the input point vector layer by using the `autoKrige` function of the `automap` R package. It will first calculate the kriging model and then create a raster. The raster is created with the `raster` function of the raster R package:

```
##Basic statistics=group
##Layer=vector point
##Field=Field Layer
##Output=output raster
```

(наставак на следећој страни)

(настављено са претходне стране)

```
##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
```

By using `##load_vector_using_rgdal`, the input vector layer will be made available as a `SpatialDataFrame` objects, so we avoid having to translate it from an `sf` object.

Example with table output

Let's edit the `Summary Statistics` algorithm so that the output is a table file (csv).

The script body is the following:

```
##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
```

The third line specifies the **Vector Field** in input and the fourth line tells the algorithm that the output should be a table.

The last line will take the `Stat` object created in the script and convert it into a `csv` table.

Example with console output

We can use the previous example and instead of creating a table, print the result in the **Result Viewer**:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
Summary_statistics<-data.frame(rbind(
sum(Layer[[Field]]),
length(Layer[[Field]]),
length(unique(Layer[[Field]])),
min(Layer[[Field]]),
max(Layer[[Field]]),
max(Layer[[Field]])-min(Layer[[Field]]),
```

(наставак на следећој страни)

(настављено са претходне стране)

```
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
```

The script is exactly the same as the one above except for two edits:

1. no output specified (the fourth line has been removed)
2. the last line begins with >, telling Processing to make the object available through the result viewer

Example with plot

To create plots, you have to use the `##output_plots_to_html` parameter as in the following script:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
##output_plots_to_html
####output_plots_to_html
qqnorm(Layer[[Field]])
qqline(Layer[[Field]])
```

The script uses a field (`Field`) of a vector layer (`Layer`) as input, and creates a *QQ Plot* (to test the normality of the distribution).

The plot is automatically added to the Processing *Result Viewer*.

26.3 Dodatak E: Mrežne veze QGIS aplikacije

Ovo je lista unapred konfigurisanih/automatizovanih mrežnih veza koje QGIS uspostavlja. Neke veze pokreće korisnik jer zahtevaju radnju korisnika pre nego što se ostvare, a druge se dešavaju automatski.

Назив	Svrha	Režim UX	Server	Informacije poslate serveru	Informacije sačuvane na serveru
qgis.org					
Pomoć za Python API	Pregledanje PyQGIS dokumentacije	Pokreće korisnik	https://qgis.org/pyqgis/3.44/search	IP, verzija QGIS-a, OS	IP u dnevniku servera
version.qgis.					
Provera nove verzije	Obaveštava korisnika o dostupnosti novih verzija QGIS-a	Automatski	https://version.qgis.org	IP, verzija QGIS-a, OS	IP u dnevniku servera
feed.qgis.					
QGIS Feed	Preuzima QGIS vesti iz feed-a	Automatski	https://feed.qgis.org	IP, verzija QGIS-a, kôd jezika, vremenska oznaka poslednjeg preuzimanja, OS	IP u dnevniku servera; verzija QGIS-a, OS i IP se agregiraju i koriste za prikupljanje neke statistike
plugins.qgis					
Provera ažuriranja dodataka	Obaveštava korisnika o ažuriranjima dodataka	Pokreće korisnik/automa (podesivo)	https://plugins.qgis.org	IP, verzija QGIS-a, OS	IP u dnevniku servera
Lista dodataka	Preuzima listu dodataka	Pokreće korisnik	https://plugins.qgis.org	IP, verzija QGIS-a, OS	IP u dnevniku servera
Instalacija dodatka	Preuzima i instalira paket dodatka	Pokreće korisnik	https://plugins.qgis.org	IP, verzija QGIS-a, OS	Uvećava brojač preuzimanja dodatka za jedan
Stilovi	Navodi stilove koje su doprineli korisnici	Pokreće korisnik	https://hub.qgis.org/styles	IP, verzija QGIS-a, OS	Uvećava brojač preuzimanja za jedan
Treće strane					
Podaci o terenu	Pravi DEM za 3D prikaze	Pokreće korisnik	https://s3.amazonaws.com/elevation-tiles-prod terrarium/{z}/{x}/{y}	IP, verzija QGIS-a, OS	pogledajte Amazon TOS
Google Map Geocoder	Servisi geokodiranja	Pokreće korisnik	https://maps.googleapis.com/maps/api/geocode/json	IP, verzija QGIS-a, OS	Pogledajte TOS Google maps API-ja
Nominatim Geocoder	Servisi geokodiranja	Pokreće korisnik	https://nominatim.qgis.org/ui/search.html	IP, verzija QGIS-a, OS	
Geodetska mreža	Dodaje novu PROJ mrežu	Pokreće korisnik	https://cdn.proj.org	IP, verzija PROJ-a	Dnevnici pristupa se trajno brišu nakon jednog dana

ГЛАВА 27

Literatura i veb reference

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.