
QGIS User Guide

Versão 2.6

QGIS Project

22/05/2015

1	Preâmbulo	3
2	Convenções	5
2.1	Convenções da Interface Gráfica	5
2.2	Convenções do Texto ou Teclado	5
2.3	Instruções específicas da Plataforma	6
3	Preâmbulo	7
4	Feições	9
4.1	Visualização de dados	9
4.2	Exploração de dados e compositores de mapas	9
4.3	Criar, editar, gerir e exportar dados	10
4.4	Análise de dados	10
4.5	Publicação de mapas na internet	10
4.6	Funcionalidades do QGIS expandida através de complementos	10
4.7	Console Python	11
4.8	Problemas conhecidos	12
5	O que há de novo no QGIS 2.6	13
5.1	Aplicação e opções de projeto	13
5.2	Provedor de arquivos	13
5.3	Compositor de impressão	13
5.4	Servidor QGIS	14
5.5	Simbologia	14
5.6	Interface de usuário	14
6	Iniciando	15
6.1	Instalação	15
6.2	Amostra de Dados	15
6.3	Sessão Amostra	16
6.4	Iniciar e Parar QGIS	17
6.5	Opções da Linha de Comandos	17
6.6	Projetos	19
6.7	Arquivo de Saída	20
7	QGIS GUI	21
7.1	Barra de Menu	22
7.2	Barra de Ferramentas	28
7.3	Legenda do Mapa	29
7.4	Visualização do mapa	31
7.5	Barra de Status	32

8	Ferramentas Gerais	33
8.1	Atalhos de teclado	33
8.2	Conteúdo da ajuda	33
8.3	Renderização	33
8.4	Medição	35
8.5	Identificar feições	37
8.6	Decorações	38
8.7	Ferramentas de anotação	40
8.8	Favoritos Espaciais	41
8.9	Projetos animados	43
9	Configuração QGIS	45
9.1	Painéis e Barras de Ferramentas	45
9.2	Propriedades do Projeto	46
9.3	Opções	46
9.4	Personalização	55
10	Trabalhando com Projeções	57
10.1	Visão geral do Suporte a Projeções	57
10.2	Especificação de Projeção Global	57
10.3	Definir Reprojeção Dinâmica Voo Livre (OTF)	59
10.4	Sistema de Referência de Coordenadas personalizado	60
10.5	transformações de datum padrão	61
11	QGIS Pesquisador	63
12	Trabalhando com Dados Vetoriais	65
12.1	Formatos de dados suportados	65
12.2	A Biblioteca de Símbolos	77
12.3	Janela de Propriedades de Vetor	80
12.4	Expressões	109
12.5	Editando	116
12.6	Ferramenta de Consulta	133
12.7	Calculadora de Campo	135
13	Trabalhando com Dados Raster	137
13.1	Trabalhando com dados raster	137
13.2	Diálogo de propriedades do Raster	138
13.3	Calculadora Raster	147
14	Trabalhando com dados OGC	149
14.1	QGIS como Cliente de Dados OGC	149
14.2	QGIS com Servidor de Dados OGC	158
15	Trabalhando com dados GPS	165
15.1	Complemento GPS	165
15.2	Rastreamento GPS em tempo real	169
16	Integração com SIG GRASS	175
16.1	Iniciando o complemento GRASS	175
16.2	Carregando camadas raster e vetorial GRASS	176
16.3	Localização e MAPSET GRASS	176
16.4	Importando dados para uma localização GRASS	179
16.5	O modelo de dados vetorial do GRASS	179
16.6	Criando uma nova camada vetorial GRASS	180
16.7	Digitalizando e editando uma camada vetorial GRASS	180
16.8	A ferramenta região GRASS	184
16.9	Ferramentas GRASS	184

17 QGIS estrutura de processamento	193
17.1 Introdução	193
17.2 A caixa de ferramentas	194
17.3 O modelador gráfico	203
17.4 A interface do processamento em lote	209
17.5 Usando os algoritmos do processamento a partir do Terminal Python.	211
17.6 Gerenciador do histórico	217
17.7 Escrevendo novos algoritmos de processamento com scripts python	218
17.8 Entregando os dados produzidos pelo algoritmo	220
17.9 Comunicação com o usuário	220
17.10 Documentando seus scripts	220
17.11 Exemplos de scripts	220
17.12 Melhores práticas para algoritmos de script escrito	221
17.13 Pré e pós-execução de ganchos de script	221
17.14 Configurando as aplicações externas	221
17.15 O QGIS Comando	228
18 Fornece algoritmos e processamento	231
18.1 Provedor de algoritmo GDAL	231
18.2 LAStools	264
18.3 Modeler Tools	289
18.4 OrfeoToolbox algorithm provider	291
18.5 QGIS algorithm provider	366
18.6 Provedor de algoritmo R	420
18.7 SAGA algorithm provider	430
18.8 TauDEM algorithm provider	601
19 Compositor de Impressão	633
19.1 Primeiros passos	634
19.2 Modo de Renderização	638
19.3 Itens do Compositor	640
19.4 Gerenciar ítems	663
19.5 Ferramentas de Reverter e Restaurar	665
19.6 Geração de Atlas	665
19.7 Criando um arquivo de Saída	667
19.8 Gerir o Compositor	669
20 Complementos	671
20.1 Complementos QGIS	671
20.2 Usando os Complementos Core QGIS	676
20.3 Complemento de Captura de Coordenadas	677
20.4 Complemento Gerenciador BD	677
20.5 Complemento dxf2shp	678
20.6 Complemento eVis	679
20.7 Complemento fTools	689
20.8 Complemento Ferramentas GDAL	692
20.9 Complemento Georreferenciador	695
20.10 Complemento de Interpolação	700
20.11 Complemento Edição Offine	701
20.12 Complemento GeoRaster Espacial Oracle	701
20.13 Complemento Análise do Terreno	704
20.14 Complemento Mapa de Densidade	705
20.15 Catálogo do Cliente MetaBusca	708
20.16 Complemento Menor Distância	712
20.17 Complemento de Consulta Espacial	713
20.18 Complemento SPIT	715
20.19 Complemento SQL Anywhere	715
20.20 Complemento Verificador de Topologia	717
20.21 Complemento Estatística Zonal	718

21 Ajuda e Suporte	721
21.1 Listas de Discursão	721
21.2 IRC	722
21.3 Rastreador de Erros	722
21.4 Blog	723
21.5 Plugins	723
21.6 Wiki	723
22 Apêndice	725
22.1 Licença Pública Geral GNU	725
22.2 GNU Licença de Documentação Gratuita	728
23 Referências Bibliográficas e Web	735
Índice	737

·
·

Preâmbulo

Este documento é o guia do usuário original do software descrito QGIS. O software e hardware descrito neste documento são, na maioria dos casos, marcas registradas e são, portanto, sujeitas a requisitos legais. QGIS está sujeito à Licença Pública Geral GNU. Encontre mais informações sobre o QGIS na página, <http://www.qgis.org>.

Os detalhes, dados e resultados neste documento podem ter sido escritos e verificados para melhorar o conhecimento e a responsabilidade dos autores e editores. No entanto, erros relativos ao conteúdo são possíveis.

Portanto, os dados não estão sujeitos a quaisquer direitos ou garantias. Os autores, editores e editoras não têm qualquer responsabilidade por falhas e suas consequências. Você é sempre bem-vindo para relatar possíveis erros.

Este documento foi elaborado com reStructuredText. Ele está disponível como código-fonte reST via [github](https://github.com) e on-line como HTML e PDF via <http://www.qgis.org/en/docs/>. As versões traduzidas deste documento também podem ser baixadas em vários formatos através da área de documentação do projeto QGIS. Para mais informações sobre a contribuir para este documento e sobre a tradução, por favor visite <http://www.qgis.org/wiki/>.

Links neste Documento

Este documento contém links internos e externos. Clicando sobre um link interno move-se dentro do documento, ao clicar em um link externo abre um endereço de internet. Em formato PDF, links internos e externos são mostradas em azul e são tratados pelo navegador do sistema. Em formato HTML, o navegador exibe e trata ambos de forma idêntica.

Usuário, Instalação e Código Guia dos Autores e Editores:

Tara Athan	Radim Blazek	Godofredo Contreras	Otto Dassau	Martin Dobias
Peter Ersts	Anne Ghisla	Stephan Holl	N. Horning	Magnus Homann
Werner Macho	Carson J.Q. Farmer	Tyler Mitchell	K. Koy	Lars Luthman
Claudia A. Engel	Brendan Morely	David Willis	Jürgen E. Fischer	Marco Hugentobler
Larissa Junek	Diethard Jansen	Paolo Corti	Gavin Macaulay	Gary E. Sherman
Tim Sutton	Alex Bruy	Raymond Nijssen	Richard Duivenvoorde	Andreas Neumann
Astrid Emde	Yves Jacolin	Alexandre Neto	Andy Schmid	Hien Tran-Quang

Copyright (c) 2004 - 2014 QGIS Equipe de Desenvolvimento

Internet: <http://www.qgis.org>

Licença deste documento


Permissão é garantida para cópia, distribuição e/ou modificação deste documento sobre os termos da Licença de Documentação Livre GNU, Versão 1.3 ou versão mais recente publicada pela Fundação Software Livre; sem Seções Invariantes, sem textos de Capa e sem textos de Contra-Capa. A cópia da licença é incluída no Apêndice *GNU Licença de Documentação Gratuita*.

Convenções

Esta seção descreve os estilos uniformes que serão utilizados ao longo deste manual.

2.1 Convenções da Interface Gráfica

Os estilos convenção da GUI, são destinados a similar a aparência gráfica. Em geral, um estilo vai refletir a aparência, para que um usuário possa verificar visualmente a GUI e encontrar algo semelhante com as instruções no manual.

- Menu Opções: *Camada* → *Adicionar uma camada Raster* ou *Configurações* → *Barra de Ferramentas* → *Digitalizar*
- Ferramenta:  Adiciona uma camada Raster
- Botão : **[Salvar como padrão]**
- Caixa de diálogo Título: *Propriedades da camada*
- Aba: *Geral*
- Caixa de seleção: *Renderizar*
- Botão de opção: *Postgis SRID* *EPSG ID*
- Selecione um número:
- Selecione uma frase:
- Procure pelo arquivo:
- Selecione uma cor:
- Barra deslizante:
- Entrada de texto:

Uma sombra indica um componente GUI clicável.

2.2 Convenções do Texto ou Teclado

Este manual também inclui estilos relacionados ao texto, comandos de teclado e de codificação para indicar diferentes entidades, tais como classes ou métodos. Estes estilos não correspondem à aparência real de qualquer texto ou codificação dentro do QGIS.



- Hyperlinks: <http://qgis.org>

- A Combinação das teclas: Press `Ctrl+B`, ou seja, pressione e segure a tecla `Ctrl` e, em seguida, pressione a tecla `B`.
- Nome de um arquivo: `lakes.shp`
- Nome de uma classe: **NewLayer**
- Método: `classFactory`
- Servidor: `myhost.de`
- Texto de Usuário: `qgis --help`



Linhas de código serão indicadas por uma fonte de tamanho fixo:

```
PROJCS["NAD_1927_Albers",  
GEOGCS["GCS_North_American_1927",
```


2.3 Instruções específicas da Plataforma


Sequências de GUI e pequenas quantidades de texto podem ser formatadas em linha: Clique   *Arquivo* **X** *QGIS* → *Sair para fechar o QGIS*. Isso indica que em plataformas Linux, Unix e Windows, você deve clicar primeiro no menu Arquivo, em seguida, Sair, enquanto nas plataformas Macintosh OS X, você deve clicar primeiro no menu QGIS e, em seguida Sair.

Grandes quantidades de texto podem ser formatados com uma lista:

-  Faz isso
-  Faz aquilo
- **X** Faz outro

ou um parágrafo:

 **X** Faz isso, isso e isso. Então faz isso, isso e isso e isso, isso e isso, e isso, isso e isso.

 Faz aquilo. E faz aquilo e aquilo e aquilo, e aquilo e aquilo e aquilo, e aquilo e aquilo e aquilo, e aquilo e aquilo e aquilo e aquilo, e aquilo e aquilo e aquilo.

Imagens que aparecem ao longo do guia do usuário foram criadas em diferentes plataformas, a plataforma é indicada pelo ícone específico da plataforma no final da legenda da figura.

Preâmbulo

Bem vindo ao mundo maravilhoso dos Sistemas de Informação Geográficas (SIG)!

QGIS é um Sistema de Informação Geográfica de Código Aberto. O projeto nasceu em maio de 2002 e foi criado como um projeto no SourceForge, em junho do mesmo ano. Nós trabalhamos duro para tornar o software SIG (que é um software proprietário tradicionalmente caro) uma perspectiva viável para qualquer pessoa com acesso básico a um computador pessoal. QGIS atualmente funciona na maioria das plataformas Unix, Windows e OS X. QGIS é desenvolvido usando o Qt toolkit (<http://qt.digia.com>) e C ++. Isso significa que QGIS é considerado amigável e tem uma interface gráfica (GUI) agradável e fácil de usar.

QGIS pretende ser um SIG de fácil utilização, fornecendo funções e feições comuns. O objetivo inicial do projeto era fornecer um visualizador de dados SIG. QGIS chegou ao ponto em sua evolução, onde ele está sendo usado por muitos para as suas necessidades de visualização de dados SIG diárias. QGIS suporta um grande número de formatos de dados raster e vetoriais, com o novo formato, suporta facilmente a sua adição de outros, usando a arquitetura de complementos.

QGIS é distribuído sob a GNU General Public License (GPL)/Licença Publica Geral. Desenvolvendo QGIS sob esta licença significa que você pode inspecionar e modificar o código-fonte, e garante que você, nosso feliz usuário, sempre terá acesso a um programa de SIG que é livre de custos e podem ser livremente modificado. Você deve ter recebido uma cópia completa da licença com a sua cópia do QGIS, e você também pode encontrá-la no Apêndice: ref: *gpl_appendix*.

Dica: Atualização da Documentação

A versão mais recente deste documento pode ser encontrada na área de documentação do QGIS website em <http://www.qgis.org/en/docs/>.

Feições

QGIS oferece muitas funcionalidades SIG comuns fornecidas com funções núcleo e complementos. Um breve resumo de seis categorias gerais de funções e complementos é apresentado a seguir, seguido de primeiros insights sobre o console Python integrado.

4.1 Visualização de dados

Pode ver ou sobrepor dados vetoriais e matriciais em diferentes formatos e projeções sem conversão para um formato interno ou comum. Os formatos suportados incluídos são:

- Tabelas ativadas espacialmente e visualização usando PostGIS, SpatiaLite e MS SQL Spatial, Oracle Spatial, formatos vetoriais suportados pela biblioteca OGR instalada, incluindo arquivos de forma ESRI, Map-Info, SDTS, GML e muitos mais. Consulte a seção *Trabalhando com Dados Vetoriais*.
- Formatos de imagens e Raster suportados pela biblioteca GDAL instalada (Geospatial Data Abstraction Library), como GeoTIFF, ERDAS IMG, ArcInfo ASCII GRID, JPEG, PNG e muitos mais. Consulte a seção *Trabalhando com Dados Raster*.
- Dados vetoriais e raster GRASS para base de dados GRASS (location.mapset). Ver seção *Integração com SIG GRASS*.
- Servidores de dados espaciais online como Serviços Web OGC, incluem WMS, WMTS, WCS, WFS, e WFS-T. Ver seção *Trabalhando com dados OGC*.

4.2 Exploração de dados e compositores de mapas

Você pode compor mapas e interativamente explorar dados espaciais com uma interface gráfica amigável. As muitas ferramentas úteis disponíveis na GUI incluem:

- Pesquisador QGIS
- Reprojecção On-the-fly
- Gerenciador BD
- Compositor de Mapas
- Painel de Vista Global
- Marcadores espaciais
- Ferramentas de anotação
- Identificar/selecionar feições
- Editar/ver/procurar atributos
- Rotulagem de feição Dado-definido

- Ferramentas de simbologia raster e vetorial dado definido
- Compositor de Atlas com camada grade
- rótulos de Norte, barra de escala e copyright para o mapa
- Suporte para salvamento e restauração de projetos

4.3 Criar, editar, gerir e exportar dados

Você pode criar, editar, gerenciar e exportar camadas vetoriais e raster em vários formatos. [lgg!](#) oferece a seguir:

- Ferramentas de digitalização para formatos suportados OGR e camadas vetoriais GRASS
- Habilitado para criar e editar camadas vetoriais shapefile e GRASS
- Complemento Georreferenciador para geocodificar imagens
- Ferramentas de GPS para importar e exportar formatos GPX, e converter outros formatos GPS para GPX ou baixar/carregar diretamente para uma unidade GPS (No Linux, usb: pode ser adicionado na lista de dispositivos GPS.)
- Suporte para visualização e edição de dados OpenStreetMap
- Habilitado para criar tabelas de base de dados espaciais a partir de shapefiles com complemento Gerenciador BD
- Tratamento melhorado de tabelas de bases de dados espaciais
- Ferramentas para gerenciamento de tabelas de atributos vetoriais
- Opção para salvar as imagens como imagens georreferenciadas

4.4 Análise de dados

Você pode executar análise de dados espaciais em bancos de dados espaciais e outros formatos OGR suportados. QGIS atualmente oferece análise vetorial, amostragem, geoprocessamento, geometria e ferramentas de gerenciamento de banco de dados. Você também pode usar as ferramentas do GRASS integradas, que incluem a funcionalidade GRASS completa de mais de 400 módulos. (Consulte a seção *Integração com SIG GRASS*.) Ou, você pode trabalhar com os Complementos de Processamento, que fornece um quadro de análise geoespacial poderoso para chamar algoritmos nativos e de terceiros a partir do QGIS, como GDAL, Saga, GRASS, fTools e muito mais. (Consulte a seção *Introdução*.)

4.5 Publicação de mapas na internet

QGIS pode ser usado como um cliente WMS, WMTS, WMS-C ou WFS and WFS-T, e um servidor WMS, WCS ou WFS. (Ver seção *Trabalhando com dados OGC*.) Além disso, você pode publicar seus dados na Internet usando um servidor web com UMN MapServer ou GeoServer instalada.

4.6 Funcionalidades do QGIS expandida através de complementos

QGIS pode ser adaptado às suas necessidades especiais, com a arquitetura de complemento extensível e bibliotecas que podem ser usadas para criar complementos. Você pode até mesmo criar novas aplicações com C++ ou Python!

4.6.1 Complementos Core

Complementos incluídos no programa

1. Captura de Coordenadas (Captura coordenadas com o mouse em diferentes SRC's)
2. Gerenciador BD (Troca, edição, e visualização de camadas e tabelas; execução de consultas SQL)
3. Diagrama Overlay (Coloque diagramas em camadas vetoriais)
4. Conversor Dxf2Shp (Converte arquivos DXF para shapefiles)
5. eVIS (Visualiza eventos)
6. fTools (Analiza e gerencia dados vetoriais)
7. GDALTools (Integração das Ferramentas GDAL no QGIS)
8. Georeferenciador GDAL (Adiciona a informação de projeção usando raster GDAL)
9. Ferramentas de GPS (Carrega e importa dados de GPS)
10. GRASS (Integra o SIG GRASS)
11. Mapa de calor (Gera uma mapa de calor raster a partir de dados de pontos)
12. Complemento Interpolação (Interpolação baseada em vértices de uma camada vetorial)
13. Edição Offline (Permitir a edição offline e sincronização com bancos de dados)
14. Oracle GeoRaster Espacial
15. Processamento (anteriormente chamado SEXTANTE)
16. Análise de terreno (Análise de terreno baseada em raster)
17. Caminho mais curto (Analisa uma rede com caminho mais curto)
18. Complemento de Consulta Espacial
19. SPIT (Importa shapefiles para PostgreSQL/PostGIS)
20. Complemento SQL Anywhere (Armazena camadas vetoriais dentro de base de dados SQL Anywhere)
21. Verificador de topologia (Encontra os erros topológicos em camadas vetoriais)
22. Complemento Estatística Zonal (Calcule contagem, soma e média de um raster para cada polígono de uma camada vetorial)

4.6.2 Complementos Externos Python

QGIS oferece um número crescente de complementos externos Python que são fornecidos pela comunidade. Esses complementos aloca-se no repositório oficial de complementos e podem ser facilmente instalados usando o instalador de complementos Python. Consulte a Seção *Diálogo de Complementos*.

4.7 Console Python

Para scripts, é possível tirar proveito de um Terminal Python integrado, que pode ser aberto a partir do menu: *Complementos* → *Terminal Python*. O terminal irá abrir como uma janela do utilitário não-modal. Para a interação com o ambiente QGIS, há a variável `qgis.utils iface`, que é uma instância de `QgsInterface`. Esta interface permite o acesso ao mapa em tela, menus, barras de ferramentas e outras partes do aplicativo QGIS.

Para mais informações sobre como trabalhar com o console Python e programação de complementos e aplicativos QGIS, por favor consulte http://www.qgis.org/html/en/docs/pyqgis_developer_cookbook/index.html.

4.8 Problemas conhecidos

4.8.1 Limitação no número de arquivos abertos

Se você está abrindo um grande projeto QGIS e você tem certeza de que todas as camadas são válidas, mas algumas camadas são sinalizadas como ruim, você provavelmente irá se confrontar com esta questão. O Linux (e outros sistemas operacionais, da mesma forma) tem um limite de arquivos abertos por processo. Limites de recursos por processo e hereditária. O comando `ulimit`, que é um shell integrado, muda os limites apenas para o processo de shell atual; o novo limite será herdado por quaisquer processos filhos.

Você pode ver todos os atuais limites de informações digitando

```
user@host:~$ ulimit -aS
```

Você pode ver o número permitido atual de arquivos abertos por processos com o seguinte comando em um console

```
user@host:~$ ulimit -Sn
```

Para alterar os limites para uma **sessão existente**, você pode ser capaz de usar algo como

```
user@host:~$ ulimit -Sn #number_of_allowed_open_files
user@host:~$ ulimit -Sn
user@host:~$ qgis
```

Para fixar isso sempre

Na maioria dos sistemas Linux, limites de recursos são definidos no login pelo módulo `pam_limits` de acordo com as definições contidas no `/etc/security/limits.conf` ou `/etc/security/limits.d/*.conf`. Você será capaz de editar os arquivos, se você tem privilégios de root (também via `sudo`), mas você vai precisar fazer login novamente para que as alterações tenham efeito.

Mais informações:

<http://www.cyberciti.biz/faq/linux-increase-the-maximum-number-of-open-files/> <http://linuxaria.com/article/open-files-in-linux?lang=en>

.

O que há de novo no QGIS 2.6

Esta versão contém novas características e se estende a interface de programação com respeito as versões anteriores. Se recomenda que utilize esta versão sobre as versões anteriores.

Esta versão inclui centenas de correções de erros e muitas novas características e melhoras que são descritas neste manual. Também pode revisar a lista de mudanças visuais em <http://changelog.linfinity.com/qgis/version/2.6.0/>.

5.1 Aplicação e opções de projeto

- **Nome do arquivo de projeto em propriedades:** Agora pode ver o caminho completo para o arquivo de projeto do QGIS no diálogo de propriedades do projeto.

5.2 Provedor de arquivos

- **Melhoras nas ferramentas de Exportação DXF:**
 - Visão de árvore e a seleção de atributos para a camada assinada no diálogo
 - apoio chegado a polígonos/HATCH
 - representar textos como MTEXT em lugar de TEXT (incluindo a fonte, inclinação e peso)
 - suporte as cores RGB quando não existe coincidência exata da cor
 - utilizar AutoCAD 2000 DXF (R15) no lugar de R12

5.3 Compositor de impressão

- **Atualizar extensão de tela do mapa a partir da extensão do desenho do mapa: Não articulado**
propriedades de um elemento de mapa na agora existe dois botões adicionais que lhe permite (1) estabelece a extensão da tela do mapa de acordo com a extensão de seu elemento e (2) a visão em visão do mapa, atualmente a extensão se estabelece em seu elemento Mapa.
- **Múltiplo suporte de rede:** Agora é possível ter mais de uma quadrícula de seu elemento de mapa. Cada rede é totalmente personalizável e pode-se assinalar um SRC diferente. Isto significa, por exemplo, agora pode-se ter um desenho de mapa com quadrícula tanto geográfica como projetada.
- **Exportação seletiva:** Para todos os elementos de seu desenho de impressão do mapa, baixa Opções de representação, pode excluir estes objetos de exportação do mapa.

5.4 Servidor QGIS

5.5 Simbologia

5.6 Interface de usuário

Iniciando

Este capítulo dá uma visão geral rápida sobre a instalação do QGIS, alguns dados de exemplo na web sobre o QGIS e rodar uma primeira seção simples visualizando camadas raster e vetoriais.

6.1 Instalação

A instalação do QGIS é muito simples. Estão disponíveis pacotes de instalação padrão para MS Windows e Mac OS X. Se proporcionam pacotes binários (rpm e deb) ou repositórios de software para adquirir o seu gerenciador de instalação de pacotes para diversos servidores de GNU/Linux. Consiga as últimas informações sobre pacotes binários na página do QGIS na internet em <http://download.qgis.org>.

6.1.1 Instalação à partir da fonte


Se necessitar compilar o QGIS a partir da fonte, por favor consulte as instruções de instalação. São distribuídas com o código fonte do QGIS em um arquivo chamado `INSTALL`. Também pode encontrar em linha de comando em <http://htmlpreview.github.io/?https://raw.githubusercontent.com/qgis/QGIS/master/doc/INSTALL.html>

6.1.2 Instalação no dispositivo de armazenamento externo


QGIS te permite definir uma opção `--configpath` que sobrepõe uma rota pré-determinada pela configuração do usuário (ex.: `~/ .qgis2` bajo Linux) e também força uma **QSettings** ao usar esse diretório. Ele te permite, por exemplo, levar uma instalação do QGIS em uma memória flash junto com todos os complementos e a configuração. Veja a seção *Menu Sistema* para informações adicionais.

6.2 Amostra de Dados

O guia de utilizador contém exemplos baseados no conjunto de amostra de dados do QGIS.

 O instalador do Windows tem uma opção para baixar o conjunto de dados de amostra do QGIS. Ao marcar a opção, os dados serão baixados em sua pasta *Meus Documentos* e ficarão localizados em uma pasta chamada *GIS Database*. Pode-se usar o Windows Explorador para mover esta pasta para um local adequado. Se não marcar a caixa de seleção para instalação do conjunto de dados de amostra durante a instalação inicial do QGIS, pode optar por uma das seguintes:

- Usar dados SIG que já possui
- Baixar dados de amostra de http://download.osgeo.org/qgis/data/qgis_sample_data.zip
- Desinstalar QGIS e reinstalar com a opção baixar dados marcada (só recomendado de as soluções anteriores não funcionarem)

 Para GNU/Linux e Mac OS X, ainda não existe disponível pacotes de instalação de dados no formato rpm, deb ou dmg. Para usar o conjunto de dados de amostra, baixe o arquivo `qgis_sample_data` como um arquivo ZIP de http://download.osgeo.org/qgis/data/qgis_sample_data.zip e descompacte os arquivos no seu sistema.

O banco de dados Alaska inclui todos dados SIG que voce usará nos exemplos e verá nas imagens do manual do usuário; também inclui uma pequena base de dados do GRASS. A projeção usada nos dados de exemplo do QGIS é Alaska Albers Equal Área em unidade pés. O código EPSG é 2964.




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

Se você pretende usar o QGIS como um visualizador gráfico para o GRASS, você pode encontrar uma seleção de localizações de amostra (ex.: Spearfish or South Dakota) no site oficial do SIG GRASS, <http://grass.osgeo.org/download/sample-data/>.

6.3 Sessão Amostra







Agora que você tem o QGIS instalado e um conjunto de dados de amostra disponível, nós gostaríamos de demonstrar uma corta e simples seção no QGIS. Usaremos a camada raster `qgis_sample_data/raster/landcover.img`, e o camada vetor `qgis_sample_data/gml/lakes.gml`.

6.3.1 Iniciar o QGIS

-  Iniciar QGIS teclando “QGIS” um uma linha de comando prompt, ou usando-se um binário pré-combinado, usando o menu de Aplicativos.
-  Inicie o QGIS usando o menu Iniciar ou o atalho do ambiente de trabalho, ou de duplo clique no arquivo de projeto QGIS.
-  de duplo clique no ícone na sua pasta de Aplicações.

6.3.2 Carregue camadas vetoriais ou raster a partir do conjunto de amostras de dados



1. Clique no ícone  Carregar Raster
2. Navegue até a pasta `qgis_sample_data/raster/`, selecione o arquivo ERDAS IMG `landcover.img` e clique [**Abrir**].

3. Se o arquivo não estiver na lista, verifique se está listado *Tipo de arquivos*  na parte inferior da caixa de diálogo e encontra-se no tipo correto, neste caso “Erdas Imagine Images (*.img, *.IMG)”.
4. Agora clique no ícone  Carregar Vetor
5.  *Arquivo* deveria estar selecionado como *Tipo de origem* na nova janela *Adicionar camada vetorial*. Agora clique [**Navegar**] para selecionar a camada vetorial.
6. Navegue na pasta `qgis_sample_data/gml/`, selecione ‘Geography Markup Language [GML] [OGR] (.gml,.GML)’ da lista *Tipo de arquivos* , a continuação selecione o arquivo GML `lakes.gml` e clique [**Abrir**]. No diálogo *Adicionar camada vetorial*, clique [**Abrir**]. O diálogo *Seletor do Sistema de Referência de Coordenadas* se abrirá com *NAD27 / Alaska Albers* selecionado, clicando [**Aceitar**].
7. Amplie um pouco até à sua área favorita com alguns lagos.
8. De duplo clique na camada `lakes` da legenda do mapa para abrir o diálogo *Propriedades*
9. Clique na janela *Estilo* e selecione um preenchimento de cor azul.
10. Clique na janela *Etiquetas* e marque a caixa de seleção  *Etiquetar esta camada com* para habilitar o etiqueta. Selecione o campo “NOMES” com o campo que contém as etiquetas.
11. Para melhoras a leitura das etiquetas, pode adicionar um buffer branco ao redor dando um clique no “Buffer” na lista da esquerda, marcando  *Desenhar buffer no texto* e escolhendo 3 como tamanho do buffer
12. Clique [**Aplicar**]. Confira se o resultado ficou bom e clique finalizar [**OK**].

Pode ver como é fácil visualizar camadas raster e vetoriais no QGIS. Vamos até a seção seguinte para aprender mais sobre as funções, características e configurações disponíveis e como usar-las.


6.4 Iniciar e Parar QGIS

Na seção *Sessão Amostra* você aprenderá como iniciar o QGIS. Repetiremos isto e verá que o QGIS também proporciona outras opções de linha de comandos.

-  Assumindo que o QGIS está instalado no PATH, inicie QGIS teclando `qgis` no console ao dando duplo clique no link do aplicativo QGIS (ou atalho) no desktop ou no menu do aplicativo.
-  Inicie o QGIS usando o menu Iniciar ou o atalho do ambiente de trabalho, ou de duplo clique no arquivo de projeto QGIS.
- **X** Duplo clique no ícone na sua pasta Aplicações. Se necessita iniciar QGIS na shell, execute `/path-to-installation-executable/Contents/MacOS/Qgis`.

Para finalizar o QGIS, clique na opção do menu   *Arquivo* **X** *QGIS* → *Sair*, ou use o atalho `Ctrl+Q`.

6.5 Opções da Linha de Comandos

 QGIS suportam um número de opções quando iniciam pelas linhas de comando. Para obter uma lista das opções, introduza `qgis --help` na linha de comando. A sentença de uso para o QGIS é:

```
qgis --help
QGIS - 2.6.0-Brighton 'Brighton' (exported)
QGIS is a user friendly Open Source Geographic Information System.
Usage: /usr/bin/qgis.bin [OPTION] [FILE]
OPTION:
  [--snapshot filename]  emit snapshot of loaded datasets to given file
  [--width width] width of snapshot to emit
  [--height height]      height of snapshot to emit
  [--lang language]      use language for interface text
```

```
[--project projectfile] load the given QGIS project
[--extent xmin,ymin,xmax,ymax] set initial map extent
[--nologo]             hide splash screen
[--noplugins]          don't restore plugins on startup
[--nocustomization]    don't apply GUI customization
[--customizationfile] use the given ini file as GUI customization
[--optionspath path]   use the given QSettings path
[--configpath path]    use the given path for all user configuration
[--code path]          run the given python file on load
[--defaultui]          start by resetting user ui settings to default
[--help]               this text
```

FILE:

Files specified on the command line can include rasters, vectors, and QGIS project files (.qgs):

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

Dica: Exemplo do Uso dos argumentos da linha de comandos

Inicie o QGIS para especificar um ou mais arquivos de dados na linha de comando. Por exemplo, assumindo ser o diretório `qgis_sample_data`, voce inicia QGIS com uma camada de arquivo vetorial e uma raster estabelecidos para que se carregue no inicio, usando-se os seguintes comandos: `qgis ./raster/landcover.img ./gml/lakes.gml`

Opção da linha de comandos --snapshot

Esta opção permite que possa criar uma captura de ecrã no formato PNG da visão atual. Isto vem a calhar quando tem vários projetos e quer gerar capturas de tela dos seus dados.

Currently, it generates a PNG file with 800x600 pixels. This can be adjusted using the `--width` and `--height` command line arguments. A filename can be added after `--snapshot`.

Opção da linha de comandos --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, `--lang=it` starts QGIS in italian localization. A list of currently supported languages with language code and status is provided at http://hub.qgis.org/wiki/quantum-gis/GUI_Translation_Progress.

Opção da linha de comandos --projeto

Iniciar o QGIS com um arquivo de projeto existente também é possível. Só adicione na linha de comando a opção `--project` seguido do nome de seu projeto e QGIS irá abrir com todas as camadas carregadas no arquivo indicado.

Opção da linha de comandos --extent

Use esta opção para iniciar com uma extensão de mapa específica. Necessita adicionar uma quadro delimitador da sua extensão na seguinte ordem, seguido por uma vírgula:

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

Opção da linha de comandos --nologo

Este argumento de linha de comando oculta a tela inicial quando você iniciar o QGIS.

Opção da linha de comandos --noplugins

Se você tiver problemas ao iniciar os complementos, poderá evitar iniciar-los com essa opção. Estarão sendo disponíveis depois no gerenciador de complementos.

Opção de linha de comando --customizationfile

Usando este argumento de linha de comando, você pode definir um arquivo GUI personalizado, que será aplicado ao iniciar.

Opção da linha de comandos `--nocustomization`

Usando este argumento de linha de comando, um arquivo GUI personalizado existente, não será aplicado ao iniciar.

Opção da linha de comandos `--optionspath`

Pode ter várias configurações e decidir qual utilizar ao iniciar QGIS com esta opção. Verá-se *Opções* para confirmar onde armazenar os arquivos de configuração o sistema operacional. Atualmente, não existe forma de especificar um arquivo para escrever a configuração; portanto pode criar uma cópia do arquivo de configuração original e trocar o nome. A opção específica de caminho ao diretório com os ajustes. Por exemplo, para utilizar o arquivo de configuração `/path/to/config/QGIS/QGIS2.ini`, use a opção.

```
--optionspath /path/to/config/
```

Opção da linha de comandos `--configpath`

Esta opção é similar a anterior, mas no entanto, anula o caminho predeterminado pela configuração do usuário (`~/ .qgis2`) e força **QSettings** para também usar esse diretório. Isto permite aos usuários, por exemplo, carregar uma instalação QGIS na unidade flash, junto com todos os complementos e configurações.

Opção de linha de comandos `--código`



Esta opção pode ser utilizada para executar um arquivo python dado diretamente depois que QGIS foi iniciado.


Por exemplo, quando se tem um arquivo python chamado `load_alaska.py` com o seguinte conteúdo:


```
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)
```

Supondo que esta no diretório onde o arquivo `load_alaska.py` se encontra, pode iniciar QGIS, cargue o arquivo raster `landcover.img` e de uma camada de nome 'Alaska' utilizando o seguinte comando: `qgis --code load_alaska.py`

6.6 Projetos

O estado de sua seção QGIS é considerado um projeto. QGIS trabalha em um projeto por vez. Configurações são consideradas por projeto, ou como padrão pré-determinado para novos projetos (ver seção *Opções*). QGIS pode salvar o estado de sua área de trabalho dentro do arquivo do projeto, usando a opção do menu *Projeto* →  *Salvar* ou *Projeto* →  *Salvar como...*



Carregue o projeto salvo em uma seção QGIS usando *Projeto* →  *Abrir...*, *Projeto* → *Novo a partir de um modelo* ou *Projeto* → *Abrir projeto recente* →.

Se você deseja cancelar sua seção e iniciar outra escolha *Projeto* →  *Novo*. Ou das opções do menu será solicitado que você salve o projeto existente se houve mudanças desde a última vez que ele foi aberto ou salvo.

O tipo de informação salva num arquivo de projeto inclui:

- Camadas adicionadas
- Propriedades da Camada, incluindo a simbolização
- Projeção para a vista do mapa
- Última extensão visualizada



O arquivo do projeto é salvo em formato XML, assim é possível editar o arquivo em outras versões do QGIS se conhecer o que está fazendo. O formato do arquivo tem sido atualizado constantemente comparado com as versões anteriores do QGIS. Os arquivos de projeto de versões antigas do QGIS não podem funcionar corretamente. Para fazer isto, na janela *Geral* aba *Configurações* → *Opções* você pode selecionar:

-  *Avisar para salvar o projeto e alterações de fontes de dados quando necessário*
-  *Avisar quando abrir um projeto salvo com uma versão antiga do lqg*

Sempre que salvar um projeto no QGIS 2.2 irá criar uma cópia de segurança do projeto.

6.7 Arquivo de Saída

Existem muitas maneiras de gerar uma saída para seção QGIS. Já discutimos isso na seção *Projetos*, salvando como um arquivo de projeto. Aqui apresentamos outras formas de produzir arquivos de saída.

- A opção do menu *Projecto* →  *Salvar como Imagem* abre um diálogo de arquivo onde pode selecionar nome, caminho e tipo de imagem (formato PNG ou JPG). Um arquivo world file com a extensão PNGW ou JPGW é salvo na mesma pasta georeferenciando a imagem.
- Opção do menu *Projeto* → *Exportar DXF ...* abrirá a janela onde deve definir o ‘Modo da simbologia’, a ‘Escala da simbologia’ e a camada vetorial que irá exportar para DXF.
- Na opção de menu *Projeto* →  *Novo compositor de impressão* abrirá a janela onde pode criar ou imprimir o mapa atual (ver seção *Compositor de Impressão*).

QGIS GUI

Quando QGIS começa, você será apresentado a interface gráfica, como mostrado nas figuras (os números de 1 a 5 em círculos amarelos são discutidas abaixo).

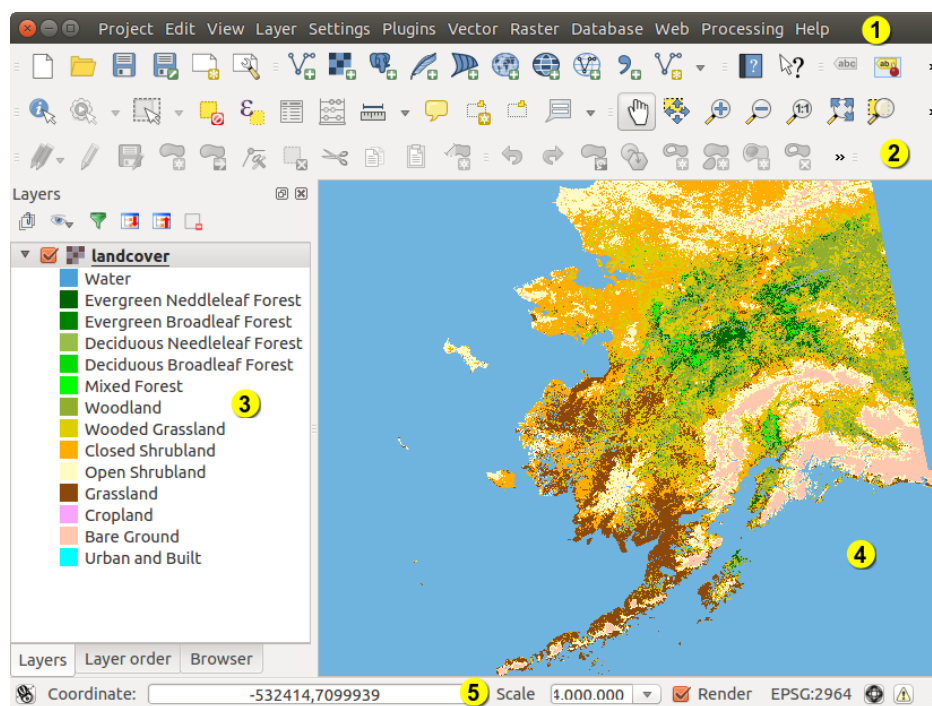


Figura 7.1: GUI do QGIS com dados amostra do Alaska 🐧

Nota: Suas decorações de janela (barra de título, etc) podem parecer diferentes dependendo do seu sistema operacional e gerenciador de janelas.

O GUI do QGIS é dividido em cinco áreas:

1. Barra de Menu
2. Barra de Ferramentas
3. Legenda do Mapa
4. Visualização do mapa
5. Barra de Status









Estes cinco componentes da interface do QGIS são descritos em mais detalhe nas seguintes seções. Mais duas seções apresentam atalhos de teclado e ajuda do contexto.

7.1 Barra de Menu























A barra de menu fornece acesso a diversas feições QGIS usando um menu hierárquico padrão. Os menus de nível superior e um resumo de algumas das opções do menu estão listados abaixo, juntamente com os ícones associados como eles aparecem na barra de ferramentas e atalhos de teclado. Os atalhos apresentados nesta seção são os padrões; no entanto, os atalhos de teclado também podem ser configurados manualmente usando o :guilabel: janela *Configurar atalhos*, aberto a partir de :menuselection:`Configurações -> Configurar Atalhos...`.


Embora a maioria das opções de menu tem uma ferramenta correspondente e vice-versa, os menus não são organizados exatamente como as barras de ferramentas. A barra de ferramentas que contém as ferramentas que estão listadas depois de cada opção habilitada no menu de entrada. Algumas opções de menu aparecem somente se o complemento correspondente for carregado. Para mais informações sobre as ferramentas e barras de ferramentas, consulte a seção :ref:`label_toolbars`.

7.1.1 Projeto




Opção de menu	Atalho	Referência	Barra de Ferramentas
 <i>Novo</i>	Ctrl+N	ver <i>Projetos</i>	<i>Projeto</i>
 <i>Abrir</i>	Ctrl+O	ver <i>Projetos</i>	<i>Projeto</i>
<i>Novo a partir do modelo -></i>		ver <i>Projetos</i>	<i>Projeto</i>
<i>Abrir Recente -></i>		ver <i>Projetos</i>	<i>Projeto</i>
 <i>Salvar</i>	Ctrl+S	ver <i>Projetos</i>	<i>Projeto</i>
 <i>Salvar Como...</i>	Ctrl+Shift+S	ver <i>Projetos</i>	<i>Projeto</i>
 <i>Salvar como Imagem...</i>		ver <i>Arquivo de Saída</i>	
<i>Exportar DXF ...</i>		ver <i>Arquivo de Saída</i>	
 <i>Novo compositor de impressão</i>	Ctrl+P	ver <i>Compositor de Impressão</i>	<i>Projeto</i>
 <i>Gerenciador de compositores ...</i>		ver <i>Compositor de Impressão</i>	<i>Projeto</i>
<i>Imprimir Compositores -></i>		ver <i>Compositor de Impressão</i>	
 <i>Sair do QGIS</i>	Ctrl+Q		

7.1.2 Editar


Opção de menu	Atalho	Referência	Barra de Ferramentas
 <i>Desfazer</i>	Ctrl+Z	ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Refazer</i>	Ctrl+Shift+Z	ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Recortar Feições</i>	Ctrl+X	ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 <i>Copiar feições</i>	Ctrl+C	ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 <i>Colar feições</i>	Ctrl+V	ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
<i>Colar feição como →</i>		Veja <i>Trabalhando com a Tabela de Atributos</i>	
 <i>Adicionar feição</i>	Ctrl+.	ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 <i>Mover Elemento(s)</i>		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 <i>Excluir Selecionado(s)</i>		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 <i>Rodar Elemento(s)</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Simplificar feições</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Adicionar Anel</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Adicionar Parte</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Preenchimento Anel</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Excluir Anel</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Excluir Parte</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Remodelar feições</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Curva de deslocamento</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Quebras Feições</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Partes Split</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Juntar Elementos Selecionados</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Juntar Atributos dos Elementos Selecionados</i>		ver <i>Digitalização Avançada</i>	<i>Digitalização Avançada</i>
 <i>Ferramenta de nós</i>		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>

Após ativar o  Modo de edição para uma camada, irá encontrar o ícone Adicionar Elemento no menu *Editar* dependendo do tipo de camada (ponto, linha ou polígono).



7.1.3 Editar (extra)

Opção de menu	Atalho	Referência	Barra de Ferramentas
 Adicionar feição		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 Adicionar feição		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>
 Adicionar feição		ver <i>Digitalizar uma camada existente</i>	<i>Digitalização</i>

7.1.4 Ver






















Opção de menu	Atalho	Referência	Barra de Ferramentas
 Panorâmica no Mapa			<i>Navegação no Mapa</i>
 Mover mapa para seleção			<i>Navegação no Mapa</i>
 Aproximar	Ctrl++		<i>Navegação no Mapa</i>
 Afastar	Ctrl+-		<i>Navegação no Mapa</i>
Selecionar →		ver <i>Selecionar e desselecionar feições</i>	<i>Atributos</i>
 Identificar feições	Ctrl+Shift+I		<i>Atributos</i>
Medir →		ver <i>Medição</i>	<i>Atributos</i>
 Ver tudo	Ctrl+Shift+F		<i>Navegação no Mapa</i>
 Aproximar à camada			<i>Navegação no Mapa</i>
 Aproximar à seleção	Ctrl+J		<i>Navegação no Mapa</i>
 Última visualização			<i>Navegação no Mapa</i>
 Próxima visualização			<i>Navegação no Mapa</i>
 Aproximar à Resolução			<i>Navegação no Mapa</i>
Natural			
Decorações →		ver <i>decorações</i>	
 Dicas do Mapa			<i>Atributos</i>
 Novo Favorito...	Ctrl+B	ver <i>Favoritos Espaciais</i>	<i>Atributos</i>
 Mostrar Favoritos	Ctrl+Shift+B	ver <i>Favoritos Espaciais</i>	<i>Atributos</i>
 Atualizar	Ctrl+R		<i>Navegação no Mapa</i>

7.1.5 Camada






Opção de menu	Atalho	Referência	Barra de F
Novo →		veja <i>Criando novas camadas Vetoriais</i>	<i>Gerenciar c</i>
Incorporar camadas e grupos...		ver <i>Projetos animados</i>	
 Adicionar Camada Vetorial	Ctrl+Shift+V	ver <i>Trabalhando com Dados Vetoriais</i>	<i>Gerenciar c</i>
 Adicionar camada raster	Ctrl+Shift+R	ver <i>Carregando dados raster no QGIS</i>	<i>Gerenciar c</i>

Continuação na próx


Tabela 7.1 – continuação da página anterior

Opção de menu	Atalho	Referência	Barra de F
 Adicionar Camada PostGIS	Ctrl+Shift+D	ver <i>Camadas PostGIS</i>	Gerenciar c
 Adicionar camada SpatiaLite	Ctrl+Shift+L	ver <i>Camadas SpatiaLite</i>	Gerenciar c
 Adicionar Camada Espacial MSSQL	Ctrl+Shift+M	veja <i>Camadas Espaciais MSSQL</i>	Gerenciar c
 Adicionar camada Oracle GeoRaster		ver <i>Complemento GeoRaster Espacial Oracle</i>	Gerenciar c
 Adicionar camada SQL Anywhere		ver <i>Complemento SQL Anywhere</i>	Gerenciar c
 Adicionar Camada WMS/WMTS	Ctrl+Shift+W	ver <i>Cliente WMS/WMTS</i>	Gerenciar c
 Adicionar Camada WCS		veja <i>WCS Cliente</i>	Gerenciar c
 Adicionar camada WFS		veja <i>WFS e WFS-T Cliente</i>	Gerenciar c
 Adicionar Camada de Texto Delimitado		ver <i>Arquivos de texto delimitado</i>	Gerenciar c
 Copiar estilo		veja <i>Menu Estilo</i>	
 Colar estilo		veja <i>Menu Estilo</i>	
 Abrir tabela de Atributos		Veja <i>Trabalhando com a Tabela de Atributos</i>	Atributos
 Opções de Ressalto		ver <i>Digitalizar uma camada existente</i>	Digitalizaçã
 Salvar edições da camada		ver <i>Digitalizar uma camada existente</i>	Digitalizaçã
 Edições Atuais →		ver <i>Digitalizar uma camada existente</i>	Digitalizaçã
Salvar como...			
Salvar arquivo vetor como...		ver <i>Trabalhando com a Tabela de Atributos</i>	
 Remover Camada(s)	Ctrl+D		
 Camada (s) Duplicada			
Definir SRC da(s) Camada(s)	Ctrl+Shift+C		
Definir o SRC do projeto a partir da camada			
Propriedades...			
Pesquisa...			
 Rotular			
 Adicionar para a Visão Geral	Ctrl+Shift+O		Gerenciar c
 Adicionar tudo para a Visão Geral			
 Remover tudo da Visão Geral			
 Mostrar todas as camadas	Ctrl+Shift+U		Gerenciar c
 Ocultar todas as camadas	Ctrl+Shift+H		Gerenciar c

7.1.6 Configurações





Opção de menu	Atalho	Referência	Barra de Ferramentas
<i>Painéis →</i> <i>Barra de Ferramentas →</i> <i>Mudar para o modo de tela inteira</i>	F 11	veja <i>Painéis e Barras de Ferramentas</i> veja <i>Painéis e Barras de Ferramentas</i>	
 <i>Propriedades do Projeto ...</i>	Ctrl+Shift+P	ver <i>Projetos</i>	
 <i>SRC Personalizado...</i>		ver <i>Sistema de Referência de Coordenadas personalizado</i>	
<i>Gerenciador de estilos...</i>		ver <i>Apresentação</i>	
 <i>Configurar atalhos...</i>			
 <i>Personalização ...</i>		ver <i>Personalização</i>	
 <i>Opções ...</i>		ver <i>Opções</i>	
<i>Opções de ajuste ...</i>			

7.1.7 Complementos

Opção de menu	Atalho	Referência	Barra de Ferramentas
 <i>Gerenciar e Instalar Complementos</i> <i>Terminal Python</i>		ver <i>Diálogo de Complementos</i>	

O começo do QGIS pela primeira vez não carrega todos os complementos core.

7.1.8 Vetor

Opção de menu	Atalho	Referência	Barra de Ferramentas
<i>Open Street Map →</i>		veja <i>Carregando vetores OpenStreetMap</i>	
 <i>Ferramenta de Análise →</i>		ver <i>Complemento fTools</i>	
 <i>Ferramenta de pesquisa →</i>		ver <i>Complemento fTools</i>	
 <i>Ferramenta de Geoprocessamento →</i>		ver <i>Complemento fTools</i>	
 <i>Ferramenta de Geometria →</i>		ver <i>Complemento fTools</i>	
 <i>Ferramenta de Gerenciamento de Dados →</i>		ver <i>Complemento fTools</i>	







O começo do QGIS pela primeira vez não carrega todos os complementos core.

7.1.9 Raster

Opção de menu	Atalho	Referência	Barra de Ferramentas
<i>Calculadora raster ...</i>		ver <i>Calculadora Raster</i>	







O começo do QGIS pela primeira vez não carrega todos os complementos core.


7.1.10 Processamento





Opção de menu	Atalho	Referência	Barra de Ferramentas
 <i>Caixa de Ferramentas</i>		veja <i>A caixa de ferramentas</i>	
 <i>Modelador Gráfico</i>		veja <i>O modelador gráfico</i>	
 <i>Historia e log</i>		veja <i>Gerenciador do histórico</i>	
 <i>Opções e configurações</i>		veja <i>Configurando a infraestrutura do processamento</i>	
 <i>Visualizador de resultados</i>		veja <i>Configurando as aplicações externas</i>	
 <i>Comandos</i>	Ctrl+Alt+M	veja <i>O QGIS Comando</i>	

Do começo do QGIS pela primeira vez não carrega todos os complementos core.

7.1.11 Ajuda

Opção de menu	Atalho	Referência	Barra de Ferramentas
 <i>Conteúdo da Ajuda</i>	F1		<i>Ajuda</i>
 <i>O que é isto?</i> <i>Documentação API</i> <i>Precisa de suporte comercial?</i>	Shift+F1		<i>Ajuda</i>
 <i>Página do QGIS</i>	Ctrl+H		
 <i>Verificar a versão do QGIS</i>			
 <i>Sobre</i>			
 <i>Patrocinadores</i>			

Por favor, note que para Linux , os itens do menu de barras listados acima são os padrões da janela gerenciadora do KDE. No GNOME, o menu *Configurações* tem um conteúdo diferente e seus itens têm de ser encontradas aqui:

 <i>Propriedades do Projeto</i>	<i>Projeto</i>
 <i>Opções</i>	<i>Editar</i>
 <i>Configurar Atalhos</i>	<i>Editar</i>
<i>Gerenciar estilo</i>	<i>Editar</i>
 <i>SRC personalizado</i>	<i>Editar</i>
<i>Painéis →</i>	<i>Exibir</i>
<i>Barra de Ferramentas →</i>	<i>Exibir</i>
<i>Mudar para o Modo de Tela Inteira</i>	<i>Exibir</i>
<i>Escala deslizante</i>	<i>Exibir</i>
<i>Live GPS tracking</i>	<i>Exibir</i>

7.2 Barra de Ferramentas

A barra de ferramentas permite o acesso à maioria das mesmas funções dos menus, além de ferramentas adicionais para interagir com o mapa. Cada item da barra de ferramentas pop-up tem ajuda disponível. Mantenha o mouse sobre o item e uma breve descrição a respeito da ferramenta será exibida.




Cada menu pode ser movido de acordo com suas necessidades. Além disso cada menu pode ser desligado com o botão direito do mouse sobre o menu de contexto, segurando o botão do mouse sobre a barra de ferramentas. (leia também *Painéis e Barras de Ferramentas*).

Dica: Restaurar barra de ferramentas

Se você tiver acidentalmente escondido todas as barras de ferramentas, você pode recuperá-las, escolhendo a opção do menu *Exibir* → *Barras de ferramentas* →. Se uma barra de ferramentas desaparece no Windows, o que parece ser um problema no QGIS de vez em quando, você deve remover a chave `\HKEY_CURRENT_USER\Software\QGIS\qgis\UI\state` no registro. Quando você reiniciar o QGIS, a chave estará escrita novamente com o estado padrão, e todas as barras serão visíveis novamente.

7.3 Legenda do Mapa

A área da legenda do mapa registra todas as camadas do projeto. A caixa de verificação de cada entrada da legenda pode ser utilizada para mostrar ou ocultar a camada. A barra de ferramentas da legenda na legenda do mapa esta lista permite **Adicionar grupo**, **Gerenciamento de visibilidade da camada** de todas as camadas ou gerenciamento de combinação de camadas pré-definidas, **Filtrar legenda pelo conteúdo do mapa**, Expandir tudo** ou


Comprimir tudo e **Eliminar grupo ou camada**. O botão  lhe permite adicionar vista **Preestabelecidos** na legenda. Isto significa que pode eleger se mostra alguma camada com organização específica e adiciona esta vista a lista de **Preestabelecidos**. Para adicionar uma vista preestabelecida simplesmente de um clique em , eleja *Adicionar preestabelecido...* do menu em cascata e de um nome ao preestabelecido. Depois verá uma lista com todos os preestabelecidos que pode chamar precionando o botão .

Todos os preestabelecidos adicionados estarão presentes no desenho de impressão afim de permitir a criação de um desenho de mapa de mapa com base em seus pontos de visão específicos (ver *Propriedades principais*).

Uma camada pode ser selecionada e arrastada para cima ou para baixo na legenda para mudar a Z-ordenação. Z-ordenação significa que as camadas listadas mais perto do topo da legenda são desenhadas sobre camadas listadas mais abaixo na legenda.


Nota: Este comportamento pode ser substituído pelo painel ‘Ordem das Camadas’.

Camadas na janela legenda podem ser organizadas em grupos. Há duas maneiras de fazer isso:

1. Precione o ícone  para adicionar um novo grupo. Escreva um nome para o grupo e precione `Enter`. Agora de um clique na camada existente e arraste-a para o grupo.
2. Selecione algumas camadas, clique direito na janela de legenda e escolha *Grupo selecionado*. As camadas selecionadas serão automaticamente colocadas em um novo grupo.

Para trazer uma camada de um grupo, você pode arrastá-la de fora, ou clique direito sobre ela e escolha *Faça o item toplevel*. Os grupos também podem ser aninhados dentro de outros grupos.

A caixa de seleção de um grupo vai mostrar ou ocultar todas as camadas do grupo com apenas um clique.

O conteúdo do menu de contexto do botão direito do mouse depende se o item de legenda selecionado é uma camada raster ou vetorial. Para camadas GRASS vetor,  :sup: *Alternar edição* não está disponível. Consulte a seção *Digitalizando e editando uma camada vetorial GRASS* para obter informações sobre a edição de camadas vetoriais GRASS.

Menu botão direito do mouse para camadas raster

- *Aproximar à extensão da camada*
- *Adicionar ao enquadramento*
- *Ampliar à Melhor Escala (100%)*
- *Esticar Usando o Enquadramento Atual*

- *Remover*
- *Duplicar*
- *Estabelecer escala de visibilidade da camada*
- *Definir SRC da Camada*
- *Definir SRC do projeto a partir da Camada*
- *Salvar como ...*
- *Salvar como Estilo de definição de camada*
- *Propriedades...*
- *Renomear*
- *Copiar Estilo*

Adicionalmente, de acordo com a posição da camada e seleção

- *Faça Item de Topo*
- *Agrupar Selecionados*

Menu botão direito do mouse para camadas vetor

- *Aproximar à Extensão da Camada*
- *Adicionar ao Enquadramento*
- *Remover*
- *Duplicar*
- *Estabelecer escala de visibilidade da camada*
- *Definir SRC da Camada*
- *Definir SRC do projeto a partir da Camada*
- *Abrir Tabela de Atributos*
- *Alternar Edição (não está disponível para camadas GRASS)*
- *Salvar Como ...*
- *Salvar como Estilo de definição de camada*
- *Filtrar*
- *Exibir Contagem de Elementos*
- *Propriedades...*
- *Renomear*
- *Copiar Estilo*

Adicionalmente, de acordo com a posição da camada e seleção

- *Faça Item de Topo*
- *Agrupar Selecionados*

Menu botão direito do mouse para grupo de camadas

- *Ampliação ao Grupo*
- *Remover*
- *Definir SRC do Grupo*
- *Renomear*
- *Adicionar grupo*

É possível selecionar mais de uma camada ou grupo ao mesmo tempo segurando a tecla `Ctrl` enquanto seleciona as camadas com o botão esquerdo do mouse. Pode mover todas as camadas selecionadas para um novo grupo ao mesmo tempo.

Você também é capaz de excluir mais de uma camada ou um grupo de uma só vez, selecionando várias camadas com `Ctrl` e pressionando `Ctrl+D` depois. Desta forma, todas as camadas ou grupos selecionados serão removidos da lista de camadas.

7.3.1 Trabalhando com a Ordem da legenda de camada independente

Há um painel que permite que você defina uma ordem de desenho independente para a legenda do mapa. Você pode ativar ele no menu *Exibir* → *Painéis* → *Ordem da camada*. Esta funcionalidade permite-lhe, por exemplo, ordenar suas camadas em ordem de importância, mas ainda exibi-las na ordem correta (ver [figura_layer_order](#)).

Checkando a caixa *Controle da ordem de renderização* por baixo da lista de camadas irá causar um voltar ao comportamento padrão.

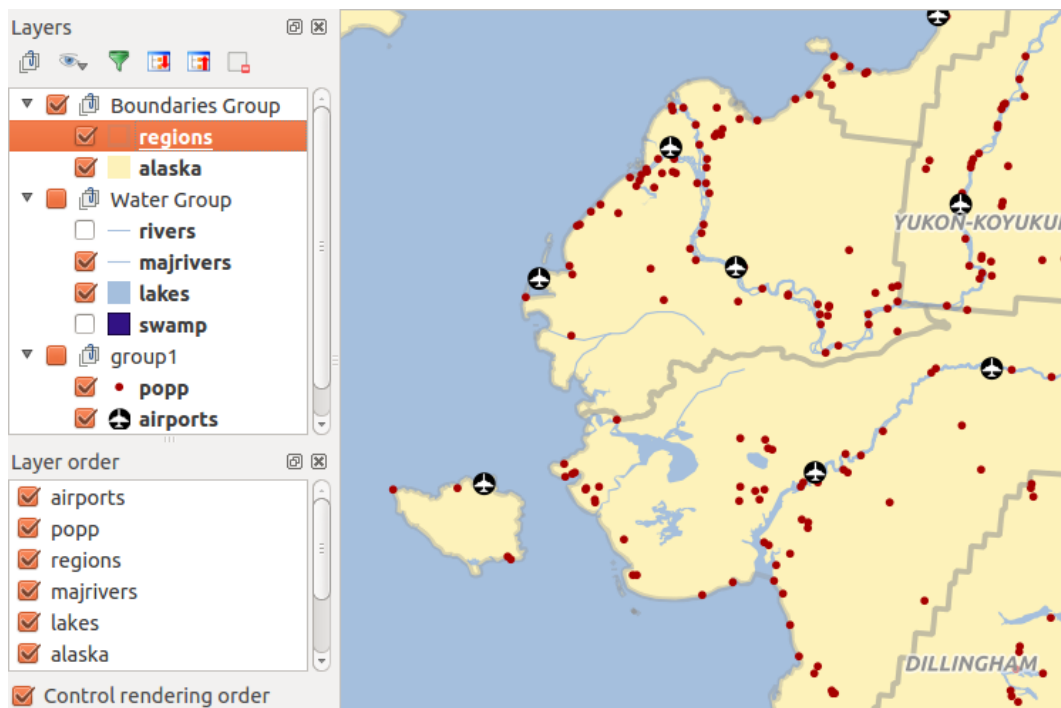


Figura 7.2: Definir a ordem da legenda de camada independente 

7.4 Visualização do mapa

Este é o “fim do negócio” do QGIS — mapas são exibidos nesta área! O mapa exibido nesta janela dependerá das camadas vetoriais e raster que você escolheu para carregar (ver seções a seguir para obter mais informações sobre como carregar camadas). A vista do mapa pode ser deslocada, mudando o foco da exibição do mapa para outra região e zoom in e out. Várias outras operações podem ser realizadas sobre o mapa, tal como descrito na descrição acima da barra de ferramentas. A vista do mapa e a legenda estão fortemente ligados uns aos outros — os mapas em vista refletem as alterações feitas na área de legenda.

Dica: Ampliando o mapa com a roda do mouse

Você pode usar a roda do mouse para zoom in e out no mapa. Coloque o cursor do mouse dentro da área do mapa e gire a roda para a frente (longe de você) para ampliar e para trás (para você) para diminuir o zoom. A posição do cursor do mouse é o centro onde o zoom ocorre. Você pode personalizar o comportamento do zoom roda do mouse usando o `:guilabel:` aba *Ferramentas do mapa* sob o menu *Configurações* → *Opções*.

Dica: Percorrendo o mapa com as setas e a barra de espaço


Você pode usar as setas do teclado para se deslocar no mapa. Coloque o cursor do mouse dentro da área do mapa e clique na seta para a direita para pan Leste, seta para a esquerda para pan Oeste, seta para cima para pan Norte e para baixo seta para deslocar Sul. Você também pode deslocar o mapa utilizando a barra de espaço ou clique na roda do mouse.

7.5 Barra de Status

A barra de status mostra sua posição atual nas coordenadas do mapa (por exemplo, metros ou graus decimais) como o ponteiro do mouse é movido através da visualização do mapa. Para a esquerda da tela de coordenadas na barra de status tem um pequeno botão que irá alternar entre mostrar posição coordenada ou como você está visualizando as extensões do mapa como pan e zoom in e out.


Próximo à coordenada mostrar que você encontra a exibição de escala. Ele mostra a escala de visualização do mapa. Se você ampliar ou QGIS mostra a escala atual. Este é um seletor de escala que permite que você escolha entre as escalas pré-definidas a partir de 1:500 até 1:1000000.

Uma barra de progresso na barra de status mostra o progresso de renderização como cada camada é atraída para a visualização do mapa. Em alguns casos, como a coleta de estatísticas em camadas raster, a barra de progresso será usada para mostrar o status de operações longas.

Se um novo complemento ou uma atualização de complemento está disponível, você verá uma mensagem na barra de status. No lado direito da barra de status existe uma pequena caixa que pode ser usada para impedir temporariamente que camadas sendo renderizadas sejam vistas no mapa (ver Seção [Renderização](#) abaixo). O ícone  imediatamente interrompe o processo de renderização atual mapa.

À direita da edição das funções, você encontra o código EPSG do SRC do projeto atual e um ícone projetor. Clicando sobre ele abre as propriedades de projeção do projeto atual.

Dica: Calculando a escala correta do seu Mapa da tela/visualização

Quando você inicia o QGIS, graus é a unidade padrão, e quer diz que qualquer QGIS coordenadas em sua camada está em graus. Para obter valores de escala correta, você pode mudar isso para metros manualmente no :guilabel: guia *Geral* em :menuselection: *Configurações* -> *Propriedades do projeto* ou você pode selecionar um projeto de coordenadas referência do Sistema SRC clicando no ícone  :sup: *status SRC* no canto inferior direito da barra de status. No último caso, as unidades estão definidas para que a projeção projeto específico, por exemplo, '+unidades=m'.

!Atualizadireitos!

Ferramentas Gerais

8.1 Atalhos de teclado

QGIS fornece atalhos de teclado padrão para muitas funções. Pode achá-los na seção *Barramenu_etiquetas*. Adicionalmente, o menu da opção *Settings* → *Configure Atalhos* permite mudar os atalhos de teclado padrão e adicionar nuevos atalhos as funções de QGIS.

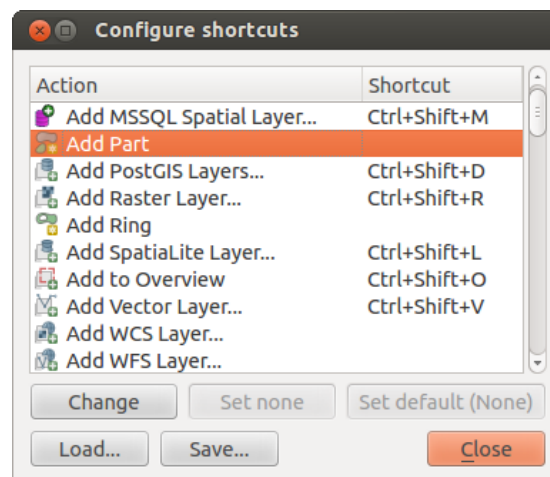


Figura 8.1: Defina as opções de atalho 🐧 (Gnome)

A configuração é simples. Apenas selecione uma função da lista e clique em **[Mudar]**, **[Definir nenhum]** ou **[Definir Padrão]**. Uma vez finalizada a sua configuração, pode salvar a mesma como arquivo XML e carregá-la em numa outra instalação QGIS.

8.2 Conteúdo da ajuda

Quando precisar de ajuda num tópico específico, pode acessar a ajuda de contexto **[Help]**, através do botão disponível na maioria dos diálogos— por favor, veja que plugins de terceiras partes podem apontar para páginas web específicas.

8.3 Renderização

Por padrão, QGIS representa todos as camadas visíveis toda vez que a tela do mapa é refeita. Os eventos que refazem a tela do mapa incluem:

- Adicionar uma camada
- Pan ou zoom
- Redimensionando a janela QGIS
- Mudanças na visibilidade de uma camada ou camadas

QGIS permite controlar o processo de representação em diferentes formas.

8.3.1 Escala dependente da renderização

A representação dependente da escala, permite especificar as escalas mínima e máxima as quais a camada será visível. Para definir uma visualização dependente da escala, abra o diálogo *Propriedades*, clicando duas vezes sobre a legenda da camada. Na aba *Geral*, clique na caixa *Visibilidade dependente da Escala* para ativar a função, depois coloque os valores de escala mínimo e máximo.

Pode-se determinar os valores da escala, primeiro fazendo um zoom ao nível desejado e verificando o valor da escala no barra QGIS .

8.3.2 Controlando a renderização do mapa

A visualização do mapa pode ser controlada de diversas formas, como descritas na sequencia.

Suspensão de edição

Para suspender a visualização do mapa, clique na caixa *Representação* na esquina inferior da barra de estado. Quando a caixa *Representação* no está marcada, QGIS não refaz a tela em resposta a qualquer dos eventos descritos na seção:ref:redesenho_eventos. Exemplos de quando você pode querer suspender a renderização incluem:

- Adicionando várias camadas e simbolizando antes do desenho.
- Adicionando uma ou mais camadas grandes e definindo a dependência de escala antes do desenho.
- Adicionando uma ou várias camadas grandes e definindo a escala de visualização antes do desenho.
- Qualquer combinação dos anteriores

Caixa de seleção :guilabel: caixa de seleção *Renderizar* permite renderização e causa uma atualização imediata na tela do mapa.

Opções de adicionar configurações da camada

Pode-se definir uma opção para que possa carregar novas camadas sem visualizá-las. Isto implica que a camada será adicionada ao mapa, mas a caixa de visibilidade na legenda estará desmarcada por padrão. Para definir esta opção, escolha o menu de opções *Definições* → *Opções* e clique em *Representação*. Desmarque a caixa *Por Padrão novas camadas adicionadas ao mapa serão visualizadas*. Qualquer camada adicionada depois ao mapa estará deligada (invisível) por padrão.

Parar renderização

Para parar o desenho no mapa, pressione a tecla `ESC`. Isto irá parar a atualização da tela do mapa e deixar o mapa parcialmente desenhado. Pode levar um pouco de tempo entre pressionar `ESC` e o tempo que o desenho do mapa será interrompido.

Nota: No momento não é possível parar a representação ou visualização — isto foi desabilitado no porta Qt4 devido a que a Interface do usuário (UI) ocasiona problemas e colapsa.

Atualização da tela durante a edição do mapa

Pode-se definir uma opção para a atualização da tela do mapa na medida em que as feições são desenhadas. Como padrão, QGIS não apresenta qualquer feição numa camada até que a totalidade da camada tenha sido representada. Para atualizar a visualização na medida em que as feições são lidas da base de dados, escolha a opção do menu *Definições* → *Opções* e clique na aba *Representação*. Coloque a contagem de feições a um valor apropriado para atualizar a tela enquanto realiza a representação. Colocando um valor de 0, desabilita a atualização enquanto desenha (este é o padrão). Colocar um valor muito baixo resultará num desempenho ruim, já que a tela do mapa estará continuamente se atualizando a medida que lê as feições. Um valor sugerido para começar é 500.

Influência da qualidade da edição

Para melhorar a qualidade da visualização do mapa, tem-se duas opções. Escolher a opção do menu *Definições* → *Opções*, clicar na aba *Representação* e selecionar ou desabilitar as seguintes caixas:

- Fazer linhas aparecem com menos definição para não peder performance ao renderizar*
- Ajustar problemas com polígonos preenchidos incorretamente*

Acelerando a visualização

Existem duas definições que permitem melhorar a velocidade da visualização. Abra a opção de diálogo QGIS *Definições* → *Opções*, e na aba *Representação* e selecionar ou desabilitar as seguintes caixas:


- Ativar buffer anterior*. Isso proporciona melhor desempenho gráfico ao custo de perder a possibilidade de cancelar a visualização e além disso, desenhar feições. Se não está selecionado, pode-se definir o *Número de feições a desenhar antes de atualizar a tela*, senão esta opção está desabilitada.
- :guilabel: ‘ Usar o cache de visualização quando possível para agilizar re-desenhos ‘


8.4 Medição

Medindo em dados com sistemas de coordenadas projetadas (ex. UTM) e não projetadas. Se o mapa carregado está definido em sistema de coordenadas geográficas (latitude/longitude), os resultados de medir uma linha ou área serão incorrectos. Para corrigir isto, precisamos definir um sistema de coordenadas de mapa apropriado (veja a seção *Trabalhando com Projeções*). Todos os módulos de medição também usam as definições de atrair do módulo de digitalização. Isso é útil se deseja medir ao longo de linhas ou áreas em camadas vectoriais.

Para escolher uma ferramenta de medição, clique em **ImAçãoMedir!** e selecione a ferramenta que deseje usar.

8.4.1 Medição de linha, áreas ou ângulos

 **Measure Line**: QGIS is able to measure real distances between given points according to a defined ellipsoid. To configure this, choose menu option *Settings* → *Options*, click on the *Map tools* tab and select the appropriate ellipsoid. There, you can also define a rubberband color and your preferred measurement units (meters or feet) and angle units (degrees, radians and gon). 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. To stop measuring, click your right mouse button.

 **Measure Area**: Areas can also be measured. In the measure window, the accumulated area size appears. In addition, the measuring tool will snap to the currently selected layer, provided that layer has its snapping tolerance set (see section *Configurando a Tolerância de Atracção e Raio de Pesquisa*). So, if you want to measure exactly

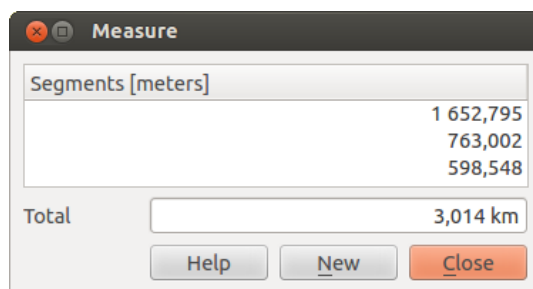


Figura 8.2: Medição de Distancia 🐧 (Gnome)

along a line feature, or around a polygon feature, first set its snapping tolerance, then select the layer. Now, when using the measuring tools, each mouse click (within the tolerance setting) will snap to that layer.

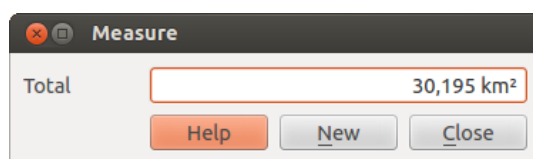



Figura 8.3: Medição de Área 🐧 (Gnome)

 **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 measure is displayed in a pop-up dialog.

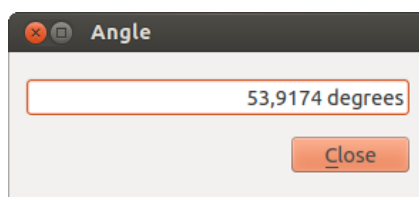








Figura 8.4: Medição de ângulo 🐧 (Gnome)

8.4.2 Selecionar e desselecionar feições

A barra de ferramentas | qg | oferece várias formas de selecionar feições na tela do mapa. Para selecionar uma ou mais feições, basta clicar sobre | mActionSelect | e selecionar sua ferramenta:


-  Selecionar uma única feição
-  Selecionar Feições através de Retangulo
-  Selecionar Feições através de Polígono
-  Selecionar Feições por Desenho Livre
-  Selecionar Feições por Raio

Para desselecionar todas as feições selecionadas clique em  Desselecionar todas feições da camada.

 Selecionar feição usando uma expressão permite ao usuário selecionar feição usando diálogo expressão. Veja *capítulo vector_expressions* para algum exemplo.

Os usuários podem salvar a seleção de feições em uma **Novo Arquivo de Camada Vetor** ou **** Nova Camada Vetor**** usando *Editar* → *Colar feições como...* e escolher o modo desejado.

8.5 Identificar feições

A ferramenta Identificar permite que você interaja com a tela do mapa e obtenha informações sobre as feições, em uma janela pop-up. Para identificar feições, use: menuselection: *Ver* -> *Identificar feições* ou pressione: kbd: *Ctrl + Shift + I*, ou clique no ícone *mActionIdentify* : sup: *Identificar feições* na barra de ferramentas.

Se você clicar em várias feições, o diálogo: guilabel: *Identificar resultados* irá listar informações sobre todos as feições selecionadas. O primeiro item é o número da feição na lista de resultados, seguido do nome da camada. Em seguida, o seu primeiro item relacionado será o nome do campo com o seu valor. Finalmente, todas as informações sobre a feição são apresentadas.

Essa janela pode ser personalizada para exibir campos personalizados, mas por padrão ele irá exibir apenas três tipos de informação:

- **Ações:** Ações podem ser adicionados às janelas de identificação de feições. Ao clicar na etiqueta da ação, a mesma será executada. Por padrão, apenas uma ação para ver feições para edição, é adicionada .
- **Derivada:** Esta informação é calculada ou derivada de outras informações. Você pode encontrar coordenadas clicadas, coordenadas X e Y, área em unidades de mapa e perímetro em unidades do mapa de polígonos, comprimento em unidades do mapa de linhas e identificação das feições.
- **Atributos dos dados:** essa é a lista de campos de atributos dos dados.

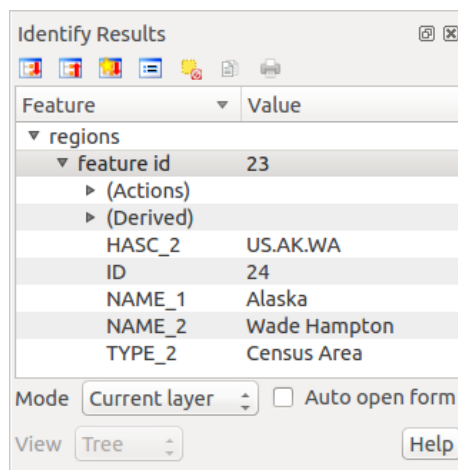








Figura 8.5: Janela identificar feição  (Gnome)

Na parte de baixo da janela, vai encontrar cinco ícones:

-  Expandir árvore
-  Fechar árvore
-  Comportamento Padrão
-  Copiar atributos
-  Imprimir resposta HTML selecionada

Outras funções podem ser encontrados no menu de contexto do item identificado. Por exemplo, do menu de contexto, você pode:


- Ver o formulário da feição
- Zoom para feição

- Copiar feição: Copiar todos os atributos e a geometria da feição
- Seleção de feição de alternância: Adicionar identificador de feição à seleção
- Copiar o valor do atributo: Copiar apenas o valor do atributo que clicou.
- Copiar atributos da feição: copiar apenas os atributos
- Limpar resultados: apaga os resultados na janela
- Limpar destaques: Remover feiçõesdestacadas no mapa
- Destaque todos
- Destaque a camada
- Ativar camada: escolha uma camada a ser ativada
- Propriedades da camada: Abre a janela de propriedades da camada
- Estender tudo
- Encolher tudo

8.6 Decorações

As Decorações de | qg | incluem a Gride ou malha, o Etiqueta de Direitos Autorais, a Seta do Norte e a Barra de Escala. Eles são usados para “decorar” o mapa, adicionando elementos cartográficos.

8.6.1 Malha

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

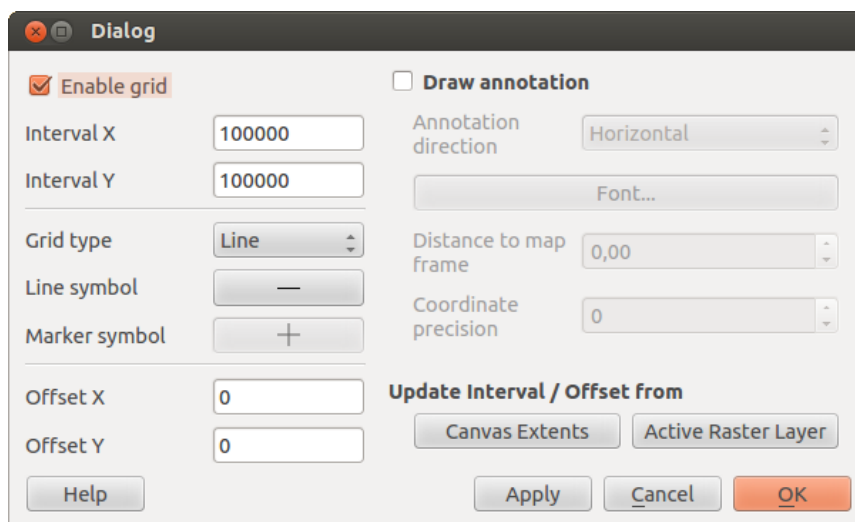





Figura 8.6: O Diálogo Malha 

1. Selecionar do menu *Ver* → *Decorações* → *Malha*. O diálogo abre (veja [figure_decorations_1](#)).
2. Marque a guia  *Abilitar Malha* e coloque as definições da malha de acordo as camadas carregadas na tela do mapa.
3. Marque a guia  *Desenhar anotações* e coloque as definições de anotações, de acordo as camadas carregadas na tela do mapa.
4. Clique [**A**plicar] para verificar que a aparência é a esperada

5. Clique [OK] para fechar o diálogo.

8.6.2 Rótulo Copyright

 Copyright label adds a copyright label using the text you prefer to the map.

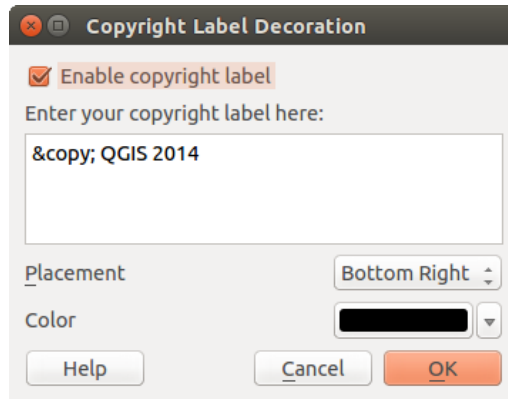




Figura 8.7: O diálogo de Direitos de Cópia Linux

1. Selecionar no menu *Ver* → *Decorações* → *Etiqueta de Propriedade Intelectual*. O diálogo abre (veja [figure_decorations_2](#)).
2. Entre o texto que deseja colocar no mapa. Pode usar HTML como mostrado no exemplo.
3. Escolha o lugar da etiqueta a partir do combo de caixas *Localização* .
4. Confirme que a caixa *Abilitar Etiqueta Direito de Cópia* está selecionada.
5. Clique [OK].

No exemplo acima, que é padrão, QGIS coloca um símbolo de direitos de cópia, seguido pela data, na esquina inferior direita da tela do mapa.

8.6.3 Seta Norte

 North Arrow places a simple north arrow on the map canvas. At present, there is only one style available. You can adjust the angle of the arrow or let QGIS set the direction automatically. If you choose to let QGIS determine the direction, it makes its best guess as to how the arrow should be oriented. For placement of the arrow, you have four options, corresponding to the four corners of the map canvas.

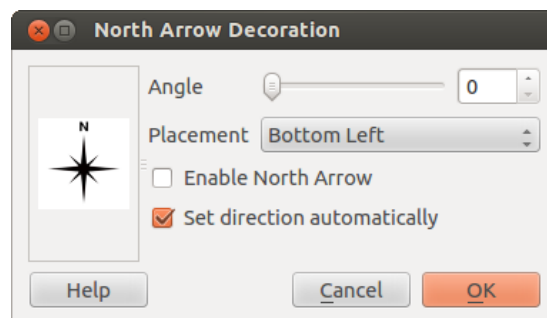




Figura 8.8: Janela de indicação do Norte 

8.6.4 Barra de Escala

 Scale Bar adds a simple scale bar to the map canvas. You can control the style and placement, as well as the labeling of the bar.

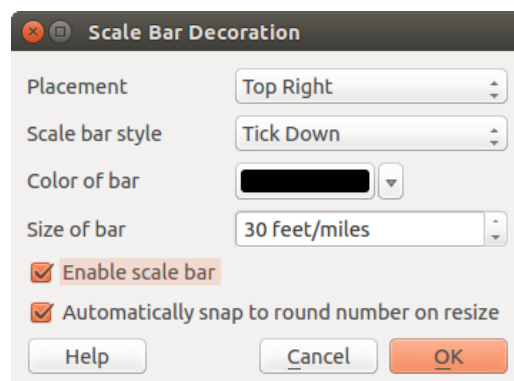





Figura 8.9: A janela da barra de escala 

QGIS only supports displaying the scale in the same units as your map frame. So if the units of your layers are in meters, you can't create a scale bar in feet. Likewise, if you are using decimal degrees, you can't create a scale bar to display distance in meters.


Para adicionar uma barra de escala:

1. Selecione a partir do menu: *menuselection: Ver -> Decorações -> Barra de Escala*. O diálogo abre (veja [figure_decorations_4](#)).
2. Selecione a localização a partir do combo de caixas *Localização* .
3. Selecione o estilo a partir do combo de caixas *Estilo da Barra de Escala* .
4. Selecione a cor da barra *Cor da bar* **selecione cor!** ou use a cor preto padrão.
5. Definir o tamanho da barra e sua etiqueta *Tamanho da barra* **selecione número!**.
6. Confirme que a caixa *Abilitar barra escala*, está clicada.
7. Opcionalmente, marque : *guilabel: automaticamente arredondará para número inteiro no redimensionamento*.
8. Clique [OK].

Dica: Configurações de Decorações

Quando salva um projeto `.qgs`, qualquer mudança que tenha efetuado ao Gríde, Seta do Norte, Barra de Escala e Direitos de Cópia, serão salvos no projeto e restaurados na próxima vez que carregue o projeto.

8.7 Ferramentas de anotação

A ferramenta  Anotação de Texto na barra de ferramentas dos atributos, dão a possibilidade de colocar texto formatado na legenda da tela do mapa QGIS. Use a ferramenta *Anotação de Texto* e clique na tela do mapa.

Dê um duplo clique no item para abrir o diálogo com várias opções. Tem um editor de texto para entrar texto formatado e definir outros itens. Por exemplo, tem uma forma de colocar um item numa posição do mapa (mostrada por um símbolo marcador) or para ter um item numa posição da tela (não relacionada ao mapa).O artigo pode ser

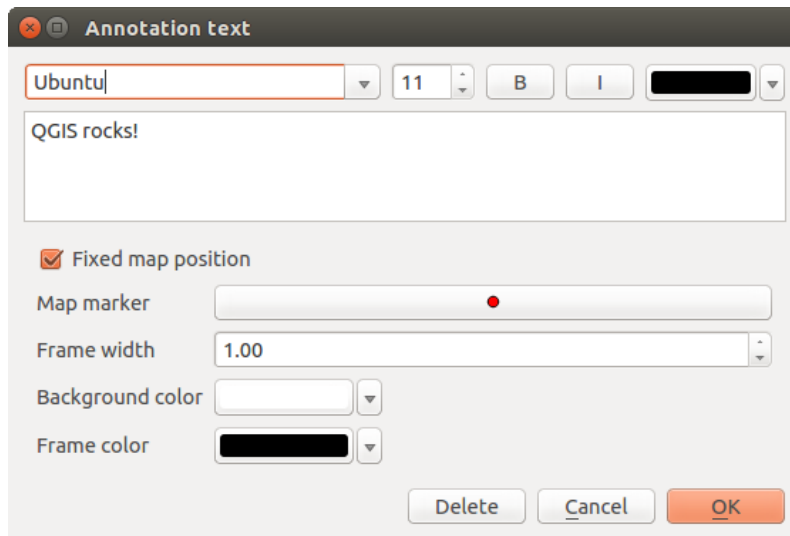




Figura 8.10: Janela texto de anotação 🐧


movido pela posição do mapa (arrastando o marcador do mapa) ou movendo apenas o balão. Os ícones são parte do tema GIS e podem ser usados como padrão em outros temas também.

A ferramenta  Mover Anotação permite mover a anotação na tela do mapa.


8.7.1 Anotações HTML

A ferramenta  Anotação HTML na barra de ferramentas dos atributos, dá a possibilidade de colocar conteúdo de em arquivo HTML numa caixa de texto ou na tela do mapa QGIS. Para usar a ferramenta *Anotação HTML*, clique na tela do mapa e adicione o caminho para o arquivo HTML no diálogo.

8.7.2 Anotações SVG

A ferramenta  Anotação SVG na barra de ferramentas dos atributos, fornece a possibilidade de colocar um símbolo SVG em uma caixa de texto na tela do mapa QGIS. Para usar a ferramenta *Anotação SVG*, clique na tela do mapa e adicione o caminho para o arquivo SVG no diálogo.

8.7.3 Anotação de formulário

Além disso, pode-se criar seu próprio formulário de anotação. A ferramenta  Formulário de Anotação é útil para mostrar os atributos de uma camada vectorial em formulário personalizado do tipo Qt Designer (veja [figure_custom_annotation](#)). Isto é semelhante aos formulários da ferramenta *Identificar feições*, mas exibido em um item de anotação. Veja também o vídeo <https://www.youtube.com/watch?v=0pDBuSbQ02o> de Tim Sutton para informações adicionais.

Nota: Se você pressionar `Ctrl+T` enquanto uma ferramenta *Anotação* está ativa (mover anotação, texto de anotação, formulário de anotação), os estados de visibilidade dos itens serão invertidos.

8.8 Favoritos Espaciais

Favoritos espaciais permite que você “marque” uma localização geográfica e volte a ela mais tarde.

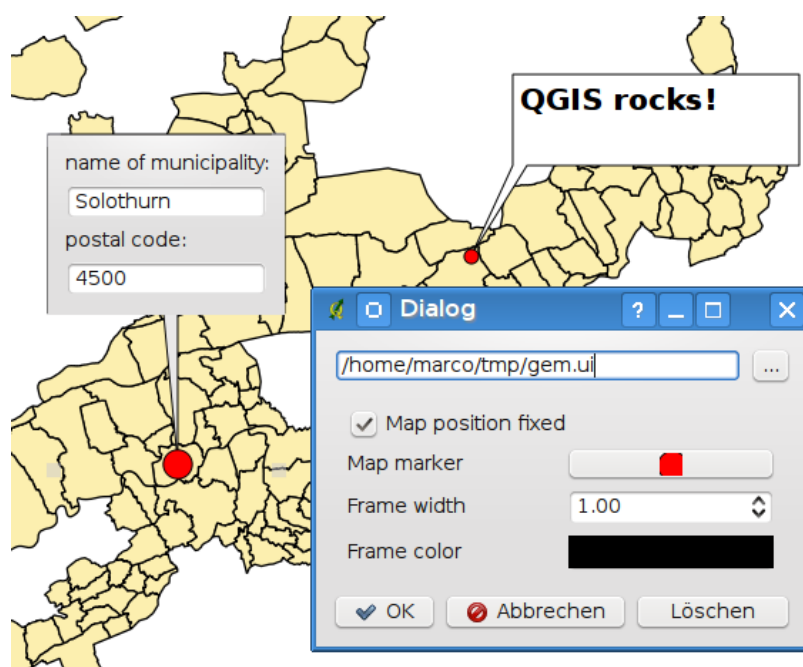


Figura 8.11: Desenho personalizado qt do formulário de anotação 🐧

8.8.1 Novo Favorito

Para criar um novo favorito:

1. Zoom ou pan na área de interesse.
2. Selecione a opção de menu *Exibir* → *Novo Favorito* ou pressione `Ctrl+B`.
3. Adicione um nome descritivo para o favorito (no máximo 255 caracteres).
4. Pressione `Enter` para adicionar um favorito ou **[Delete]** para remover o favorito.

Note que você pode ter vários favoritos com o mesmo nome.

8.8.2 Trabalhando com favoritos

Para usar ou gerenciar os marcadores, selecione a opção do menu *Ver* → *Mostrar marcadores*. O diálogo *Marcadores Geospaciais* permite focar ou apagar um marcador. Não é possível editar o nome ou as coordenadas de um marcador.

8.8.3 Aproximar para um favorito

A partir do :guilabel: diálogo *Favoritos Geoespaciais*, selecione o marcador desejado, clicando sobre ele, em seguida, clique **[Aproximar para]**. Você também pode ampliar até um marcador, clicando duas vezes sobre ele.

8.8.4 Deletando um favorito


Para excluir um marcador do diálogo *Marcadores Geospaciais*, clique nele e logo clique em **[Apagar]**. Confirme sua escolha clicando **[Sim]**, ou cancele a operação clicando **[Não]**.

8.9 Projetos animados

Se deseja incorporar conteúdo de outros arquivos de projetos no seu projeto, pode escolher *Camada → Incorporar Camadas e Grupos*.

8.9.1 Incorporando camadas

Os seguintes diálogos permitem incorporar camadas de outros projetos. Aqui um pequeno exemplo:

1. Pressione  para olhar para outro projeto do conjunto de dados Alasca.
2. Selecione o arquivo do projeto **:arquivo:'coberturavegetal'**. Pode ver agora o conteúdo do projeto (veja [figure_embed_dialog](#)).
3. Pressione **Ctrl** e clique nas camadas **:arquivo:'coberturavegetal'** e **:arquivo:'regiões'**. Pressione **[OK]**. As camadas selecionadas serão incorporadas na legenda do mapa e na tela do mapa.

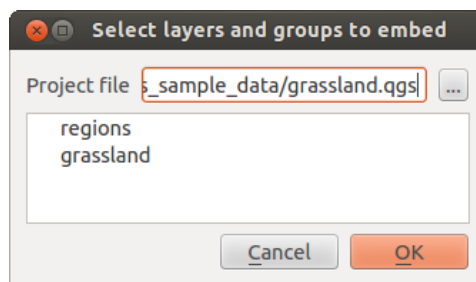



Figura 8.12: Selecionar camadas e grupos para incorporar 

Enquanto as camadas incorporadas são editáveis, não podem ser mudadas suas propriedades de estilo e rotulação.

8.9.2 Removendo camadas incorporadas

Clique com botão direito na camada incorporada e escolha  Apagar.

Configuração QGIS

O QGIS é altamente personalizável através do menu *Configurações*. Escolha entre Painéis, Caixa de Ferramentas, Propriedades do Projeto, Opções e Personalização.

Nota: QGIS segue as diretrizes de desktop para a localização das opções e elementos das propriedades do projeto. Por consequência relacionada com o sistema operacional será utilizando, a localização de alguns dos elementos descritos anteriormente poderiam estar situados no menu :menuselection`Exibir` (Painéis e Barra de ferramentas) ou em *Projeto* para Opções.

9.1 Painéis e Barras de Ferramentas

No menu *Painéis*→ pode desligar os widgets do QGIS. O menu *Caixa de Ferramentas*→ fornece a possibilidade de trocar ativar ou desativar grupos de ícones na barra de ferramentas do QGIS (veja [figure_panels_toolbars](#)).

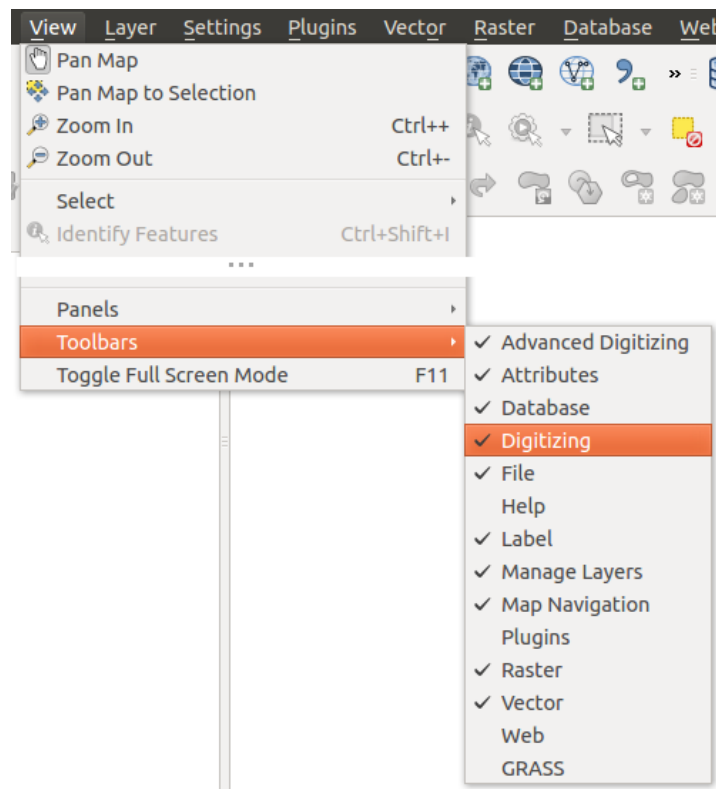







Figura 9.1: Os Painéis e o Menu de Barra de Ferramentas 




Dica: Ativando o Enquadramento QGIS

No QGIS pode usar o painel do enquadramento que fornece a extensão total das camadas adicionadas. Pode ser selecionada no menu  *Configurações* → *Painéis* ou  *Ver* → *Painéis*. Dentro da vista existe um retângulo a mostrar a extensão atual do mapa. Isto permite rapidamente determinar que área do mapa está a ver atualmente. É de notar que os rótulos não serão renderizados no enquadramento do mapa mesmo que seja ativado a rotulagem. Se clicar e arrastar o retângulo vermelho no enquadramento que mostra a atual extensão, este irá atualizar de acordo como o mapa principal.

Dica: Mostrar Mensagens de Registro



É possível seguir as mensagens do QGIS. Pode ativar o  *Registro de Mensagens* no menu  *Configurações* → *Painéis* ou  *Vier* → *Painéis* e seguir as mensagens que aparecem nos diferentes separadores durante o carregamento e operação.

9.2 Propriedades do Projeto

Na janela de propriedades do projeto baixo  *Configuração* → *Propriedades do projeto* (kde) ou   *Projeto* → *Propriedades do projeto* (Gnome), pode estabelecer opções específicas do projeto. Isto inclui:

- No menu *Geral* podem ser definidos o título do projeto, a cor de seleção e fundo, unidades da camadas, precisão, e os caminhos relativos onde serão salvas as camadas. Se a transformação SRC estiver ligada pode escolher o cálculo de distâncias recorrendo ao elipsóide. Pode definir as unidades do enquadramento (apenas usado quando a transformação SRC está desativada) e a precisão das casas decimais a usar. Pode definir também uma escala de projeto, que rescreverá sobre as escalas globais pré-definidas.
- O menu *SRC* permite que escolha o Sistema de Coordenadas Referência para o projeto, e para ativar a reprojeção on-the-fly das camadas matriciais e vetoriais na exibição de camadas de diferentes SRC.
- Com o terceiro menu *Identificar camadas* pode definir (ou desativar) que camadas irão responder à ferramenta identificar. (Veja o parágrafo das “Ferramentas de Mapa” da seção *Opções* Seção para ativar a identificação de múltiplas camadas).
- O menu *Estilos padrão* te permite controlar como as novas camadas se distribuem quando não tem um estilo existente .qml definido. Também pode estabelecer o nível de transparência por defeito para novas camadas e se seus símbolos devem ter cores indefinidas para marcar. Também tem uma seção adicional onde pode definir cores específicas para o projeto em execução. Pode encontrar as cores adicionadas no menu em cascata da aba de diálogo de cor presente em cada representação.
- O separador *Servidor OWS* permite definir a informação sobre as Capacidades do WMS e WFS, a extensão e as restrições SRC do Servidor QGIS.
- O menu *Macros* é para editar macros Python para os projetos. Atualmente, apenas estão disponíveis três macros: `openProject()`, `saveProject()` e `closeProject()`.
- O menu *Relations* é usado para definir relações 1:n. As relações são definidas no diálogo das propriedades do projeto. Quando existirem relações para uma camada, um novo elemento de interface do utilizador na vista de formulário (p. ex. quando identificar um elemento e abrir o seu formulário) irá listar os elementos relacionados. Isto fornece uma maneira poderosa para expressar p. ex. o histórico de inspeção ao longo de um segmento de tubagem ou estrada. Poderá encontrar mais informação sobre suporte de relações 1:n na Seção *Criando uma ou mais relações*.

9.3 Opções

 Algumas opções básicas para QGIS podem ser selecionadas usando o diálogo *Options*. Selecione a opção de menu *Configurações* →  *Opções*. Os separadores onde poderá personalizar as suas opções são descritos abaixo.

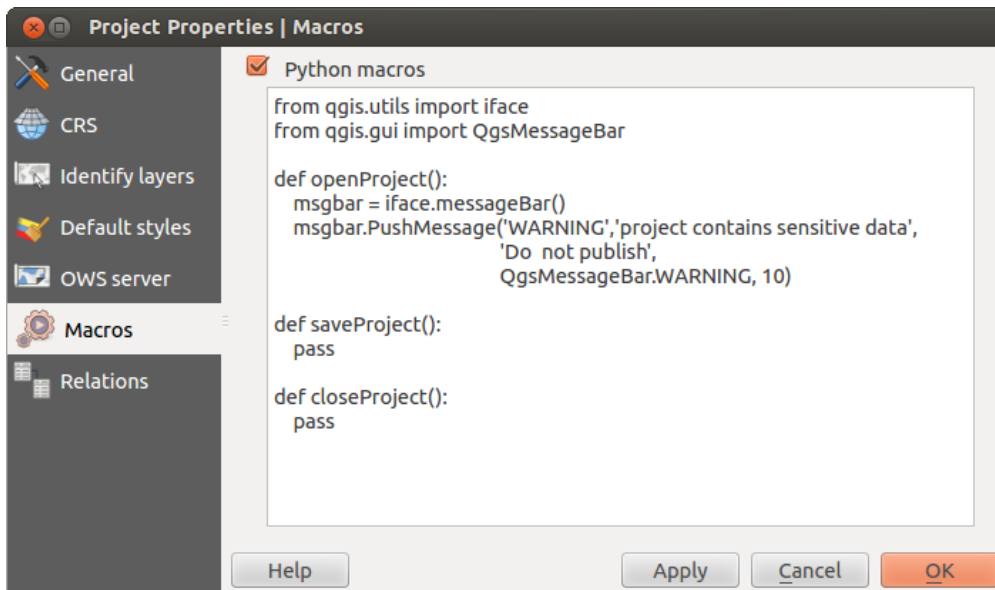









Figura 9.2: Configurações de Macro no QGIS

9.3.1 Menu Geral

Aplicação

- Selecione a *Estilo* (Necessário reiniciar o QGIS)  e escolha entre ‘Oxygen’, ‘Windows’, ‘Motif’, ‘CDE’, ‘Plastique’ e ‘Cleanlux’ .
- Definir o *Tema de Ícone* . Atualmente a opção ‘default’ é possível.
- Definir o *Tamanho do Ícone* .
- Definir a *Fonte*. Escolha entre *QT padrão* e uma fonte definida pelo utilizador.
- Altera o *Tempo limite para mensagens ou diálogos* .
- *Não exibir a janela inicial*
- *Mostrar dicas ao iniciar*
- *Títulos da caixa de grupos a negrito*
- *QGIS-estilo das caixas de grupo*
- *Use diálogos seletores de cor para atualizações ao vivo*

Arquivos de projeto

- *Abrir o projeto no arranque*  (escolha entre ‘Novo’, ‘Mais recente’ e ‘Específico’). Quando escolher ‘Específico’ use  para definir um projeto.
- *Criar um novo projeto como projeto padrão*. Tem a possibilidade de carregar em *Usar projeto atual com padrão* ou em *Resetar padrão*. Pode navegar através dos seus arquivos e definir um diretório onde encontra os modelos de projeto definidos pelo usuário. Isto será adicionado a *Projeto* → *Novo do modelo*. Primeiro ative *Criar novo projeto como projeto padrão* e a seguir salve o projeto dentro da pasta de modelos de projeto.
- *Avisar para salvar projeto e alterações de fontes de dados quando necessário*

- *Avisar quando abrir um projeto salvo com uma versão antiga do lqg!*
- *Enable macros* . Esta opção foi criada para lidar com as macros que são escritas para executar uma ação nos eventos do projeto. Pode escolher entre ‘Never’, ‘Ask’, ‘For this session only’ e ‘Always (not recommended)’.

9.3.2 Menu Sistema

Ambiente

Variáveis de ambiente do sistema podem ser vistas agora, e muitas configuradas, no grupo **Environment** (ver [figure_environment_variables](#)). Isto é útil para plataformas, tais como Mac, onde um aplicativo GUI não herda necessariamente o ambiente do utilizador da linha de comandos. Também é útil para configuração e visualização de variáveis de ambiente para os conjuntos de ferramentas externas controlados pela caixa de ferramentas de Processamento (p. ex., SAGA, GRASS), e para ativar a saída de depuração (“debugging”) para seções específicas do código-fonte.

- *Utilize variáveis personalizadas (Reinício necessário - incluem separadores)*. Você pode [Adicionar] e [remover] variáveis. Variáveis de ambiente já definidas são exibidos em *variáveis de ambiente atual*, e é possível filtrá-las ativando *Mostrar somente variáveis específicas-QGIS*.

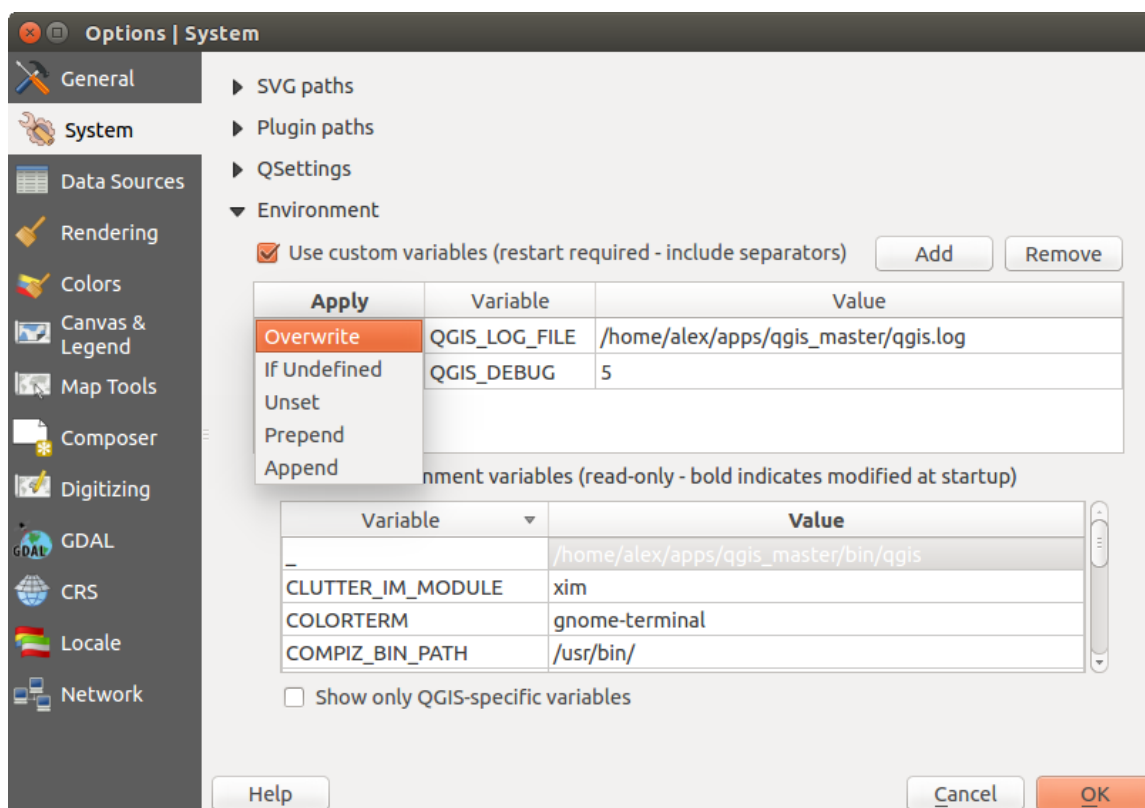





Figura 9.3: Variáveis de ambiente do Sistema no QGIS

Diretórios dos complementos



[Adicionar] ou [Remover] Caminho(s) para pesquisar bibliotecas de complementos C++ adicionais

9.3.3 Menu Fonte de Dados

Atributos dos elementos e tabela

- *Abre a tabela de atributos na janela principal (Necessário reiniciar o QGIS)*
- *Copiar geometria na representação WKT da tabela de atributos.* Ao utilizar  copiar linhas selecionadas para área de transferência do diálogo :guilabel: *tabela de Atributo*, isso tem como resultado que as coordenadas dos pontos ou vértices também são copiados para a área de transferência.
- *Comportamento da tabela de atributos* . Existem três possibilidades: ‘Mostrar todos os elementos’, ‘Mostrar elementos selecionados’ e ‘Mostrar elementos visíveis no mapa’.
- *Cache da linha da Tabela de atributos*  1,00. Esse cache de linha, torna possível para salvar as últimas linhas de atributos N carregadas, de modo que o trabalho com a tabela de atributos seja mais rápido. O cache será excluído ao fechar a tabela de atributos.
- *Representação para valores NULL.* Aqui, pode definir um valor para os campos de dados que contêm valores NULL.

Manipulação das fontes de dados

- *Pesquisar por atributos válidos na janela do navegador* . Pode escolher entre ‘Verificar extensão’ e ‘Verificar conteúdo do arquivo’.
- *Pesquisar por conteúdo de arquivos compactados (.zip) na janela do navegador* . ‘Não’, ‘Verificação básica’ e ‘Verificação completa’ são possíveis.
- *Solicitar subcamadas raster ao abrir.* Alguns rasters suportam subcamadas — elas são chamadas de sub-datasets no GDAL. Um exemplo são os arquivos netCDF — se há muitas variáveis netCDF, GDAL verá cada variável como um subdataset. A opção permite que você controle como lidar com subcamadas quando um arquivo é aberto com subcamadas. Você tem as seguintes opções:
 - ‘Sempre’: Perguntar sempre (se existem subcamadas)
 - ‘Se necessário’: Perguntar se a camada não tem bandas, mas tem subcamadas
 - ‘Nunca’: Nunca pede, não irá carregar nada
 - ‘Carregar tudo’: Nunca pede, mas carrega todas as subcamadas
- *Ignora a declaração de codificação do shapefile.* Se o shapefile tiver informação de codificação, este será ignorado pelo QGIS.
- *Adicionar uma camada PostGIS com duplo clique e selecione em modo estendido*
- *Adicione camadas Oracle com duplo clique e selecione em modo estendido*

9.3.4 Menu de Renderização

Comportamento da renderização

- *Por padrão novas camadas adicionadas ao mapa devem ser exibidas*
- *Use tornar cache sempre que possível para acelerar redesenhos*
- *Representação de camadas em paralelo utilizando muitos núcleos CPU*
- *Máximo de núcleos para utilizar*
- *Intervalo de atualização do mapa (por padrão 250 ms)*
- *Ativar simplificação de camadas por padrão para camadas recém adicionadas*
- *Simplificação de limiar*
- *Simplifique no provedor ao lado se possível*

- *Escala máxima a que a camada deve simplificar*





Qualidade de renderização

- *Faz com que as linhas apareçam menos irregulares, em detrimento de algum desempenho do desenho*

Matriciais

- Com a *Seleção de banda RGB* pode definir o número para a banda Vermelha, Verde e Azul.

Melhora de contraste

- *Banda cinza única* . Uma banda cinza única pode ter 'Sem estender', 'Estender para MinMax', 'Estender e Cortar para MinMax' e também 'Cortar para MinMax'.
- *Cor de Multi banda (byte/banda)* . As opções são 'Não estender', 'Estender para MinMax', 'Estender e cortar para MinMax' e 'Cortar para MinMax'.
- *Cor de Multi banda (>byte/banda)* . As opções são 'Não estender', 'Estender para MinMax', 'Estender e cortar para MinMax' e 'Cortar para MinMax'.
- *Limites (mínimo/máximo)* . As opções são 'Corte de contagem de pixel cumulativa', 'Mínimo/Máximo', 'Média +/- desvio padrão'.
- *Limites de contagem cumulativa de pixels de corte*
- *Multiplicador do desvio-padrão*

Corrigindo Erros

- *Atualiza o mapa da tela*

9.3.5 Menu de Cores


Este menu permite adicionar algumas cores personalizadas que possa encontrar em cada janela de diálogo de cor da representação. Verá um conjunto de cores predefinidas na janela: pode eliminar o editar todos eles. Por outra parte pode adicionar a cor que desejar e realizar alguma operação como copiar e colar. Finalmente pode exportar a cor estabelecida como um arquivo `gpl` ou importar-los.

9.3.6 Menu de Janela e Legenda

Aparência padrão do mapa (reescrita pelas propriedades do projeto)

- Define a *Cor da seleção* e a *Cor de fundo*.

Legenda da camada



- *Duplo clique na legenda* . Pode 'Abrir propriedades da camada' ou 'Abrir tabela de atributos' com duplo clique.
- As seguintes *Estilos de itens de legenda* são possíveis:
 - *Tornar maiúsculo os nomes da camada*
 - *Tornar negrito os nomes da camada*
 - *Tornar negrito os nomes dos grupos*
 - *Mostrar nomes de atributos de classificação*
 - *Criar ícones matriciais (pode ser lento)*
 - *Adicionar novas camadas ao grupo atual ou selecionado*

9.3.7 Menu Ferramentas de Mapa


Este menu oferece algumas opções com respeito ao funcionamento da *Ferramenta de Identificação*.

- *Raio de busca para identificar e visualizar avisos no mapa* é um fator de tolerância expressada como uma porcentagem do eixo do mapa. Isto significa que a ferramenta de identificação representara os resultados sempre e quando clicar dentro desta tolerância.
- *Cor de realce* lhe permite eleger com que cor devem ser identificados os objetos espaciais que estão destacados.
- *Buffer* expressado como uma porcentagem do eixo do mapa, determina uma distância de separação que representa a partir do contorno mais destacado para identificar.
- *Eixo mínimo* expresso como uma porcentagem do eixo do mapa, determina a grossura do contorno e como deve ser o objeto destacado.

Ferramenta de medida

- Define *Cor do elástico* para as ferramentas de medida
- Define *Casas decimais*
- *Manter unidade base*
- *Unidades de medição preferidas*  ('Metros', 'Pés', 'Milhas Nauticas' ou 'Graus')
- *Unidades de ângulo preferida*  ('Graus', 'Radianos' ou 'Grados')

Movendo e ampliando

- Define a *Ação da roda do rato*  ('Visualizar', 'Visualizar e Centrar', 'Aproximar ao cursor do rato', 'Nada')
- Define o *factor de aproximação/afastamento* para a roda do mouse

Escalas pré-definidas


Aqui, você encontrará uma lista de escalas pré-definidas. Com os botões [+] e [-] que você pode adicionar ou remover suas escalas individuais.

9.3.8 Menu do Compositor

Composição padrão

Você pode definir a fonte *Padrão* aqui.

Aparência do Gride

- Define o *Estilo do Gride*  ('Sólido', 'Pontos', 'Cruzamentos')
- Define a *Cor..*

Grice padrão

- Define o *Espaçamento*
- Define o *Espaçamento do Grice* para x e y
- Define o *Tolerância de Atração*

Guia padrão

- Define o *Tolerância de Atração*

9.3.9 Menu Digitalizar


Criação de elementos

- *Suprimir atributos de janelas pop-up depois de cada elemento criado*
- *Reutilizar últimos valores de atributos inseridos*
- *Validar geometrias.* Edição de linhas complexas e polígonos com muitos nós pode resultar em edição muito lenta. Isso ocorre porque os procedimentos de validação padrão no QGIS pode levar bastante tempo. Para acelerar o processamento, é possível selecionar a validação de geometria GEOS (a partir de GEOS 3.3) ou desligá-la. Validação geometria GEOS é muito mais rápida, mas a desvantagem é que apenas o primeiro problema de geometria será relatado.


Elástico

- Define a Borracha *Espessura da linha* e *Cor da linha*


Ajuste

- *Abrir opções de atração na janela principal (necessário reiniciar o QGIS)*
- Define o *Modo de atração padrão*  ('Ao vértice', 'Ao segmento', 'Ao vértice e segmento', 'Desligado')
- Define *Tolerância de atração pré-definida* em unidades de mapa ou pixels
- Define o *Raio de pesquisa para editar vértices* em unidades de mapa ou pixels

Marcadores de Vértices

- *Mostrar marcadores apenas para elementos selecionados*
- Define o vértice do *Estilo do Marcador*  ('Cruz' (padrão), 'Circulo semi-transparente' ou 'Nenhum')
- Definir o vértice *Tamanho do Marcador*

Ferramenta de curva de afastamento

A 3 opções seguintes referem a ferramentas  Curva Offset no *Digitalização Avançada*. Através das várias configurações, é possível influenciar a forma da linha de deslocamento. Estas opções são possíveis a partir de GEOS 3.3.

- *Juntar estilo*
- *Quadrante do Segmento*
- *Limite quadrante*

9.3.10 Menu GDAL

GDAL é uma biblioteca de troca de dados para arquivos raster. Neste guia, você pode *Edite criar opções e:* guilabel: *Editar Opções Pirâmides* dos formatos raster. Definir que drive GDAL está sendo utilizado para um formato raster, como em alguns casos, mais do que um drive GDAL está disponível.

9.3.11 Menu SRC

SRC padrão para novos projetos

- *Não permitir projeção 'on the fly'*
- *Habilitar automaticamente a reprojeção 'on the fly' se a camada tiver SRC diferente*
- *Habilitar reprojeção 'on the fly' por padrão*

- Selecione um SRC em *Iniciar sempre um novo projeto com este SRC*

SRC para novas camadas

Esta área permite que você defina a ação a ser tomada quando uma nova camada é criada, ou quando uma camada sem SRC é carregada.

- *Pergunta por SRC*
- *Usar SRC do projeto*
- *Utilizar SRC padrão mostrado em baixo*

Transformação de datum padrão

- *Pergunte pela transformação do datum quando nenhum padrão for definido*
- Se você trabalhou com a transformação SRC 'on-the-fly' você pode ver o resultado da transformação na janela abaixo. Você pode encontrar informações sobre 'Fonte da SRC' e 'Destino do SRC', bem como 'Transformação de Datum Fonte' e 'Transformação de Datum de Destino'.

9.3.12 Menu local

- *Sobrepôr idioma do sistema e Idioma a utilizar em alternativa*
- Informação sobre a região do sistema ativo local


9.3.13 Menu rede

Geral

- Define *Pesquisa de endereço WMS*, padrão como `http://geopole.org/wms/search?search=%1\&type=rss`
- Define *Tempo esgotado para pedidos de rede (ms)* - o padrão é 60000
- Define *Período padrão de validade para a quadrícula WMSC/WMTS (horas)* - o padrão é 24
- Define *Máxima repetição no caso de erros na requisição de mosaico*
- Define *Usuário-Agente*

Configurações de cache

Define a *Pasta* e *Tamanho* para o cache.

- *Usar proxy para acessar à web* e defina 'Máquina', 'Porta', 'Usuário', e 'Palavra-chave'.
- Configura o *Tipo de proxy*  de acordo com as necessidades.
 - *Default Proxy*: Proxy é determinado baseando-se na definição do proxy da aplicação em uso
 - *Socks5Proxy*: Proxy genérico para qualquer tipo de ligação. Suporta TCP, UDP, unindo a uma porta (ligações de entrada) e autenticação.
 - *HttpProxy*: Implementado usando o comando "LIGAR", apenas suporta ligações TCP de saída; suporta autenticação.
 - *HttpCachingProxy*: Implementado usando comandos HTTP normais, é útil apenas em pedidos no contexto do HTTP.
 - *FtpCachingProxy*: Implementado usando um proxy FTP, é útil no contexto de pedidos FTP.

Alguns URLs excluídos podem ser adicionados na caixa de texto debaixo das configurações de proxy (veja [Figure_Network_Tab](#)).

Se você precisar de informações mais detalhadas sobre as diferentes configurações de proxy, consulte o manual da documentação da biblioteca QT subjacente a <http://doc.trolltech.com/4.5/qnetworkproxy.html#ProxyType-enum>.

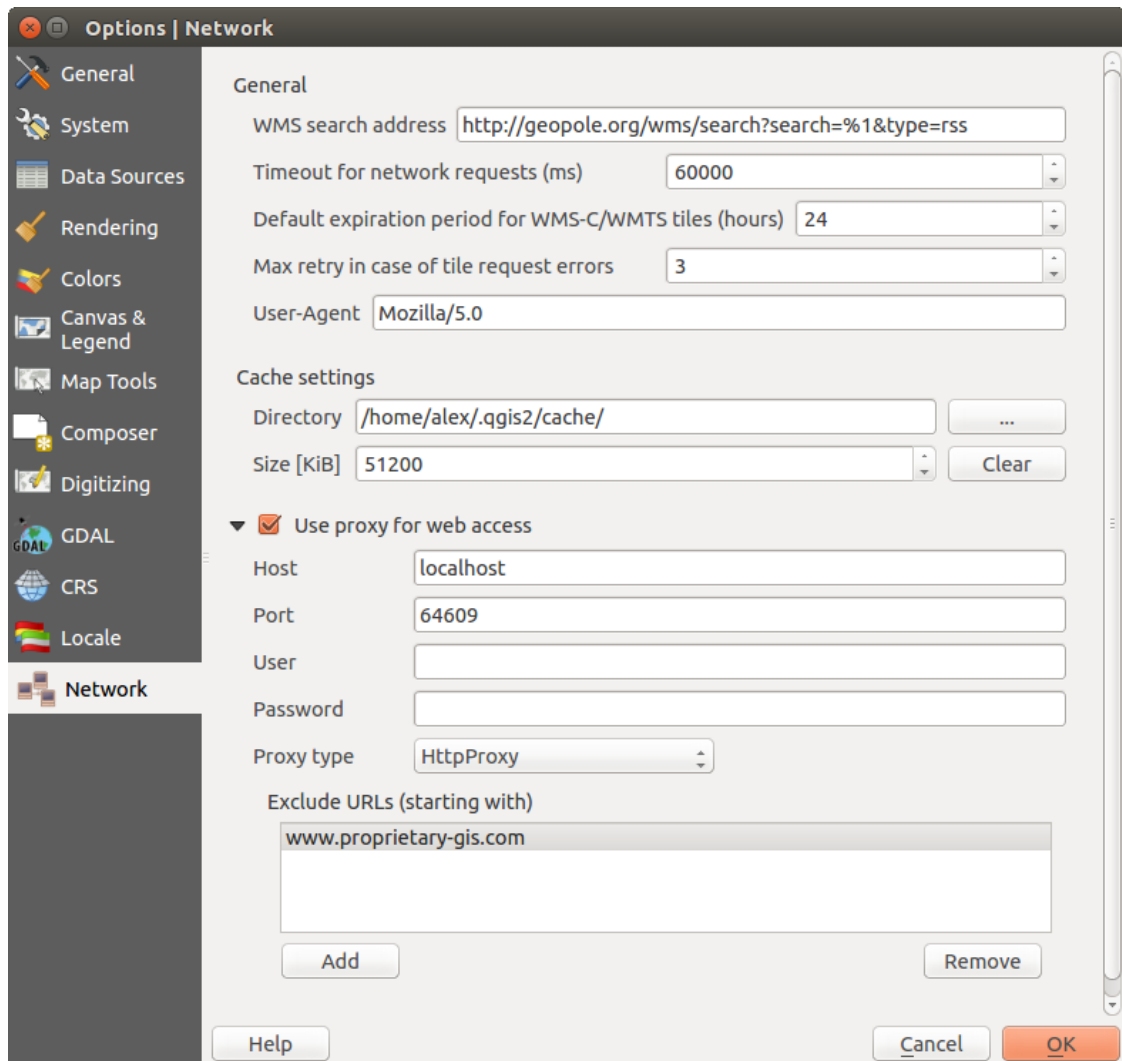




Figura 9.4: Configurações-proxy no QGIS

Dica: Usando Proxies

Usando proxies, por vezes, pode ser complicado. É útil proceder por “tentativa e erro”, com os tipos de proxy acima, verifique para ver se eles conseguem no seu caso.

Pode modificar as opções de acordo com as suas necessidades. Algumas das alterações pode requerer o reinício do QGIS antes de ser efetiva.

-  A configuração é salva em arquivo de texto: \$HOME/.config/QGIS/QGIS2.conf
- **X** pode encontrar as configurações em: \$HOME/Library/Preferences/org.qgis.qgis.plist
- As configurações  são armazenadas no registo em: HKEY\CURRENT_USER\Software\QGIS\qgis

9.4 Personalização

A ferramenta de personalização permite que (des)ative a maioria dos elementos na interface de utilizador QGIS. Isto pode ser muito útil se tiver um número elevado de módulos instalados que nunca usa e que estão preenchendo o ecrã.

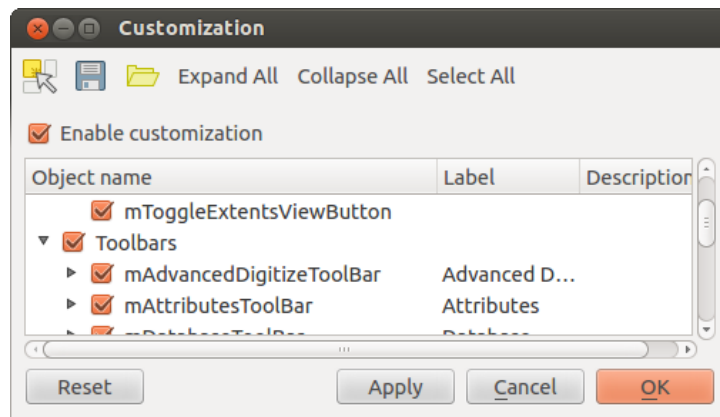




Figura 9.5: A janela de Personalização 

QGIS A Personalização está dividida em cinco grupos. No *Menus* pode esconder as entradas na Barra Menu. No *Painel* pode encontrar o painel de janelas. As janelas do Painel são aplicações que pode ser iniciadas e usadas como flutuantes, janelas de topo de nível ou contidas na janela principal do QGIS como um widget ancorado (veja also *Painéis e Barras de Ferramentas*). Na *Barra de Estado* características como a informação das coordenadas podem ser desativadas. Na *Caixa de Ferramentas* pode (des)ativar os ícones da barra de ferramentas do QGIS e em *Widgets* pode (des)ativar janelas assim como os seus botões.

Com  Mudar para pegar widgets na aplicação principal, você pode clicar em elementos QGIS que deseja ser escondida e localize a entrada correspondente na personalização (ver [figura_customization](#)). Você também pode salvar suas várias configurações também para diferentes casos de uso. Antes das alterações serem aplicadas, você precisa reiniciar QGIS.

Trabalhando com Projeções

O QGIS permite que os utilizadores definam de forma global e para todo o projecto um SC (Sistema de Coordenadas) para temas que não têm um SC pré-definido. Também permite que o utilizador defina sistemas de coordenadas personalizados e suporta a projecção dinâmica no ecrã de temas com diferentes SCs conseguindo sobrepondo-os correctamente.

10.1 Visão geral do Suporte a Projeções

O QGIS suporta aproximadamente 2.700 SCs conhecidos. Definições para cada SC são guardados numa base de dados SQLite que é instalada com o QGIS. Normalmente, não terá necessidade de manipular a base de dados directamente. De facto, ao fazê-lo poderá causar problemas. SCs definidos manualmente são guardados numa base de dados do utilizador. Veja a secção *Sistema de Referência de Coordenadas personalizado* para informação sobre a gestão dos seus sistemas de coordenadas.


Os SCs disponíveis no QGIS são baseados nos definidos pelo Grupo Europeu de Pesquisa Petrolífera (EPSG) e pelo Instituto Geográfico Nacional Francês (IGNF) e são em grande parte derivados das tabelas de referência espacial usadas pelo GDAL. Os identificadores EPSG presentes na base de dados podem ser usados para especificar um SC no QGIS.

Para poder usar a projecção dinâmica, os seus dados devem conter informação sobre o seu sistema de coordenadas ou vocês deverá definir um sistema global, para temas ou para o projeto. Para temas PostGIS, o QGIS usa o identificador de referência espacial que foi especificado aquando da criação do tema. Para dados suportados pelo OGR, o QGIS recorre à presença de um meio reconhecível para especificar o SRC. No caso de shapefiles, isto significa um arquivo contendo um texto bem-conhecido (WKT) especificando o SC. Este arquivo de projecção tem o mesmo nome base que o shapefile e uma extensão `.prj`. Por exemplo, um shapefile chamado `alaska.shp` teria um arquivo de projecção correspondente chamado `alaska.prj`.

Sempre que seleccionar um novo SC, as unidades do temas serão mudadas automaticamente no separador *Geral* das **ImActionsOptions** na janela de *Propriedades do Projeto*, no menu *Projeto* (Gnome, OSX) ou *Definições* (KDE, Windows).

10.2 Especificação de Projecção Global

O QGIS inicia cada novo projeto usando a projecção global pré-definida. O SRC global por omissão é o EPSG:4326 - WGS 84 (`proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs`), e vem pré-definido no QGIS. Este valor pode ser alterado via o botão **[Selecionar...]** na primeira secção, que é usada para definir o sistema de coordenadas por omissão para novos projetos, como mostrado na [figure_projection_1](#). Este opção pode ser gravada para uso em sessões subsequentes do QGIS.

Quando usa temas que não têm um SC, tem de definir como o QGIS reage a estes temas. Isto pode ser feito globalmente ou por projeto no separador *SC* no *Definições* →  *Opções*.

As opções mostradas na [figure_projections_1](#) são:

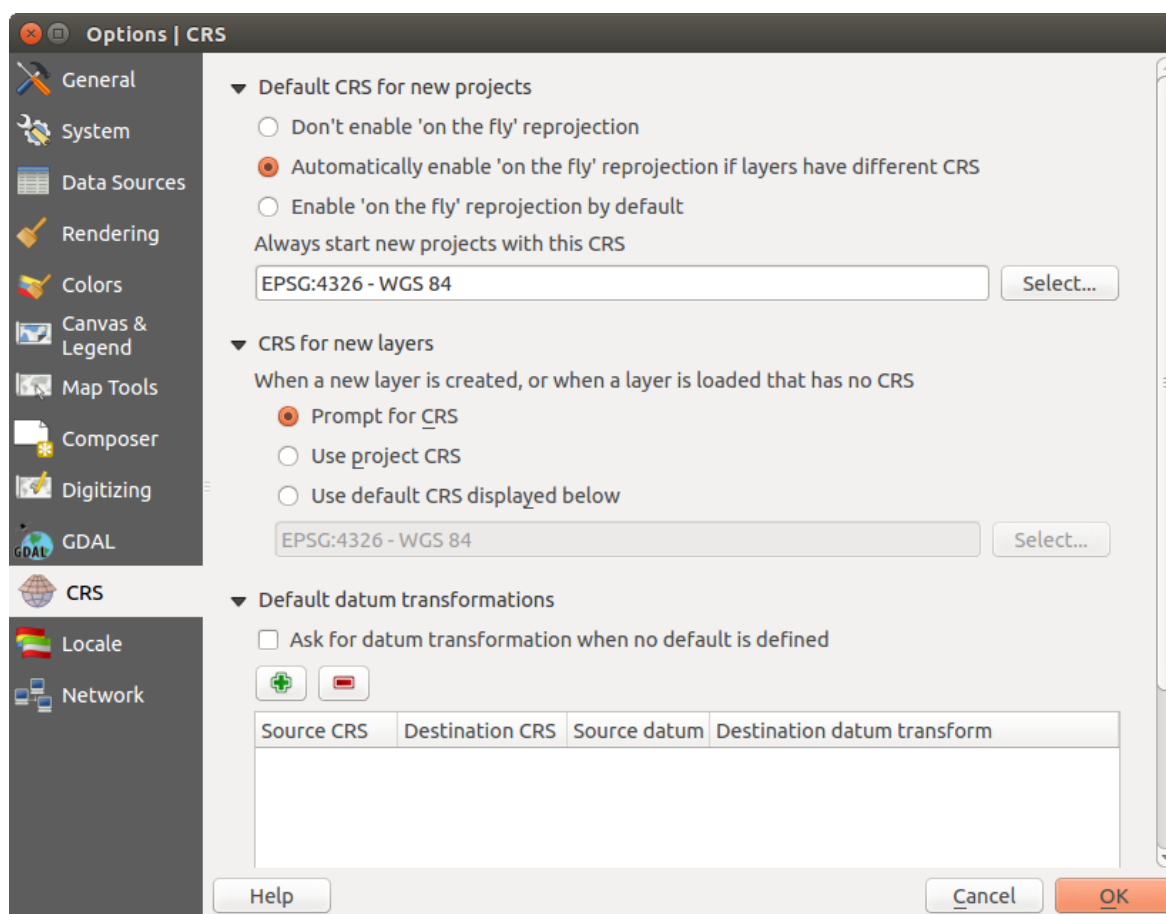


Figura 10.1: Separador SRC na Janela de Opções QGIS 🐧


- *Entrada de SRC*
- *Usar SRC do projeto*
- *Usar o SRC pré-definido mostrado abaixo*

Se quer definir o sistema de coordenadas para um dado tema sem informação de SRC, pode também fazê-lo no separador *Geral* da janela de propriedade raster e vetor (ver *Menu Geral* para rasters e *Menu Geral* para vectores). Se o seu tema já tem um SRC definido, este será mostrado como na *Janela de Propriedades do Vetor*.



Dica: SRC na Legenda do Mapa


Clicando com o botão direito num tema na Legenda do Mapa (seção *Legenda do Mapa*) mostra dois atalhos SRC. *Definir SRC do tema* abre a janela de Seleção de Sistema de Referência de Coordenadas (ver [figure_projection_2](#)). *Definir SRC do projeto a partir do Tema* redefine o SRC do projeto usando o SRC do tema.

10.3 Definir Reprojecção Dinâmica Voo Livre (OTF)

O QGIS suporta reprojecção dinâmica OTF para dados raster e vetor. Contudo, OTF não está ativo por omissão. Para usar projecção dinâmica, deve ativar a caixa de opção *Ativar transformação dinâmica de SRC* no separador *SRC* na janela  *Propriedades do Projeto*.

Há três formas de fazer isto:

1. Selecione  **menuselecion:‘Propriedades do Projeto‘** no menu *Projecto* (Gnome, OSX) ou *Definições* (KDE, Windows).
2. Clique no ícone  estado SRC no canto inferior direito da barra de estado.
3. Ative a projecção dinâmica por omissão no separador *SRC* da janela *Opções* selecionando a *Ativar reprojecção dinâmica por omissão* ou *Automaticamente ativar a reprojecção dinâmica se temas têm diferentes SRC*.

Se tem já carregado um tema e quer ativar a projecção dinâmica, a melhor prática é abrir o separador *SRC* na janela *Propriedades do Projeto*, seleccionar um SRC, e ativar a caixa de selecção *Ativar transformação dinâmica de SRC*. O ícone  Estado SRC deixará de estar desativado (cinzento), e todos os temas serão dinamicamente reprojctados para o SRC mostrado junto ao ícone.

O separador *SC* na janela *Propriedades do Projeto* contem cinco componentes importantes, como mostrado na [Figure_projections_2](#) e descritos abaixo:

1. **Aivar transformação ‘dinâmica’ de SRC** — Esta caixa de selecção é usada para ativar ou desativar a projecção dinâmica. Quando desligada, cada tema é desenhado usando as coordenadas lidas dos seus dados, e os componentes descritos abaixo estão inativos. Quando ligada, as coordenadas de cada tema são projetadas para o sistema de coordenadas definido para o mapa.
2. **Filtro** — Se conhece o código EPSG, o identificador, ou o nome para um sistema de coordenadas, pode usar a função de pesquisa para o encontrar. Introduza o código EPSG, o identificador ou o nome.
3. **Sistemas de coordenadas usados recentemente** — Se tem certos SRCs que usa frequentemente no seu trabalho SIG diário, estes serão mostrados nesta lista. Clique num destes itens para seleccionar o SRC correspondente.
4. **Sistemas de referência de coordenadas do mundo** — Esta é a lista de todos os SRCs suportados pelo QGIS, incluindo sistemas de coordenadas Geográficas, Projetadas, e Personalizados. Para definir um SRC, seleccione-o na lista expandindo o nó apropriado e seleccionado o SRC. O SRC ativo está pré-seleccionado.
5. **Texto PROJ.4** — Este é um texto de SRC usado pelo motor de projeções PROJ.4. Este texto é de apenas leitura e fornecido para fins de informação.

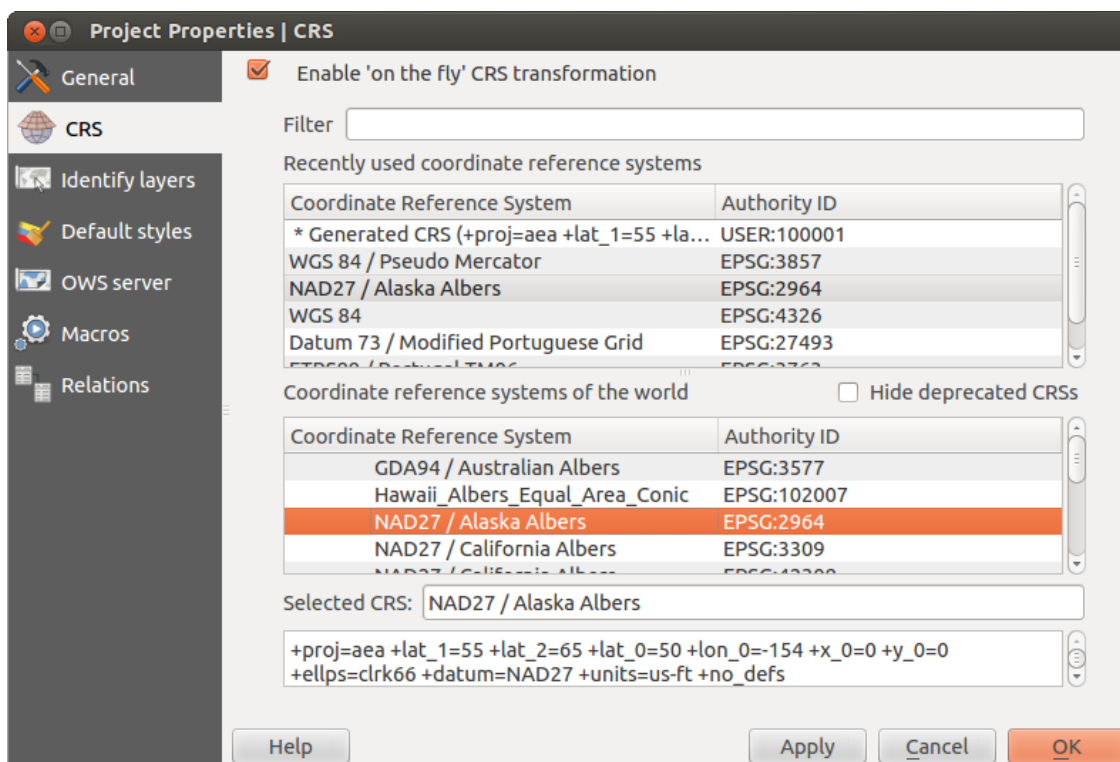


Figura 10.2: Janela de Propriedades do Projeto 🐧


Dica: Janela de Propriedades do Projeto

Se abrir a janela *Propriedades do Projeto* a partir do menu **menuselecion: 'Projeto'**, deve clicar no separador *SRC* para ver as definições de SRC.

Abriendo a janela a partir do ícone  Estado SRC abrirá automaticamente o separador *SRC*.

10.4 Sistema de Referência de Coordenadas personalizado

Se o QGIS não fornece o sistema de referência de coordenadas que necessita, pode definir um SRC personalizado.


Para definir um SRC, seleccione  *SRC personalizado...* a partir do menu *Definições*. SRCs personalizados são salvos na sua base de dados de utilizador QGIS. Além dos seus SRCs, esta base de dados também contem os seus marcadores espaciais e outros dados personalizados.

Definindo um SRC personalizado no QGIS requer uma boa compreensão da biblioteca de projeções PROJ.4. Para começar, consulte “Procedimentos cartográficos projeção para o Ambiente UNIX - Manual do Usuário” por Gerald I. Evenden, Serviço Geológico dos EUA Abra o Arquivo Relatório 90-284, 1990 (disponível em <ftp://ftp.remotesensing.org/proj/OF90-284.pdf>).

Este manual descreve o uso do `proj.4` e utilidades de linha de comando relacionados. Os parâmetros cartográficos usados com o `proj.4` são descritos no manual do utilizador e são os mesmo que os usados pelo QGIS.

A janela *Definição de Sistema de Referência de Coordenadas personalizado* exige apenas dois parâmetros para definir um SRC ao usuário:

1. Um nome descritivo
2. Os parâmetros cartográficos do formato PROJ.4

Para criar um novo SRC, clique no botão  Adicionar novo SRC e digite um nome descritivo e os parâmetros do SRC.

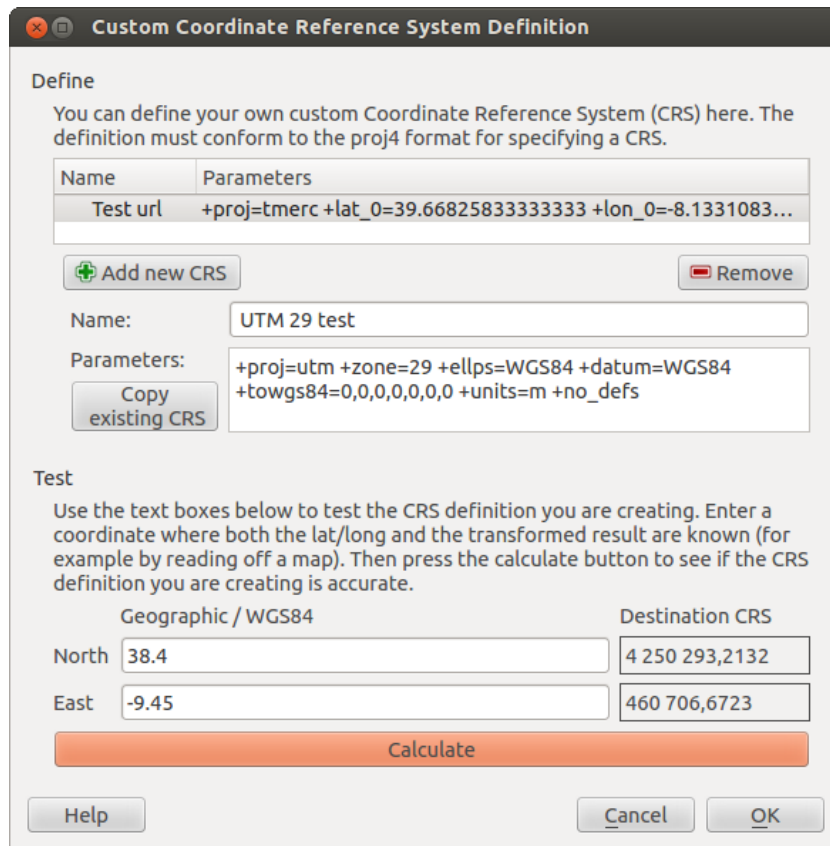


Figura 10.3: Janela de SRC personalizado 🐧

Note-se que o :guilabel: *Parâmetros* deve começar com um bloco + proj=, para representar o novo sistema de referência de coordenadas.

Você pode testar os parâmetros SRC para ver se eles dão bons resultados. Para fazer isso, digite valores conhecidos de latitude e longitude em WGS 84 *Norte* e :guilabel: *Campos Leste*, respectivamente. Clique em **[Calcular]**, e comparar os resultados com os valores conhecidos em seu sistema de referência de coordenadas.

10.5 transformações de datum padrão

depende OTF para ser capaz de transformar dados em um 'SRC padrão', e QGIS usa WGS84. Para alguns SRC há uma série de transformações disponíveis. QGIS permite que você defina a transformação utilizada para outros fins QGIS usa uma transformação padrão.

Na aba *SRC Configurações* → *Opções* você pode:

- QGIS está configurado para perguntar-lhe quando ele precisa definir uma transformação usando :guilabel: 'Pergunte qual a transformação do datum quando nenhum padrão for definido'
- editar uma lista dos padrões do usuário para transformações.

QGIS pede qual a transformação irá usar ao abrir uma caixa de diálogo exibindo texto PROJ.4 descrevendo as transformações de origem e destino. Mais informações podem ser encontradas a respeito de uma transformação. Os padrões do usuário podem ser salvos, selecionando *Lembrar seleção*.

QGIS Pesquisador

O Pesquisador QGIS é um painel do QGIS que permite você navegar facilmente nos seus arquivos e gerenciar seus dados geográficos. Você pode acessar os arquivos vetoriais mais conhecidos (ex.: ESRI shapefile ou arquivos MapInfo), base de dados (ex.: PostGIS, Oracle, Spatialite ou MSSQL Spatial) e conexões WMS/WFS. Você também pode visualizar os seus dados GRASS (para obter os dados no QGIS, veja :ref: sec_grass)

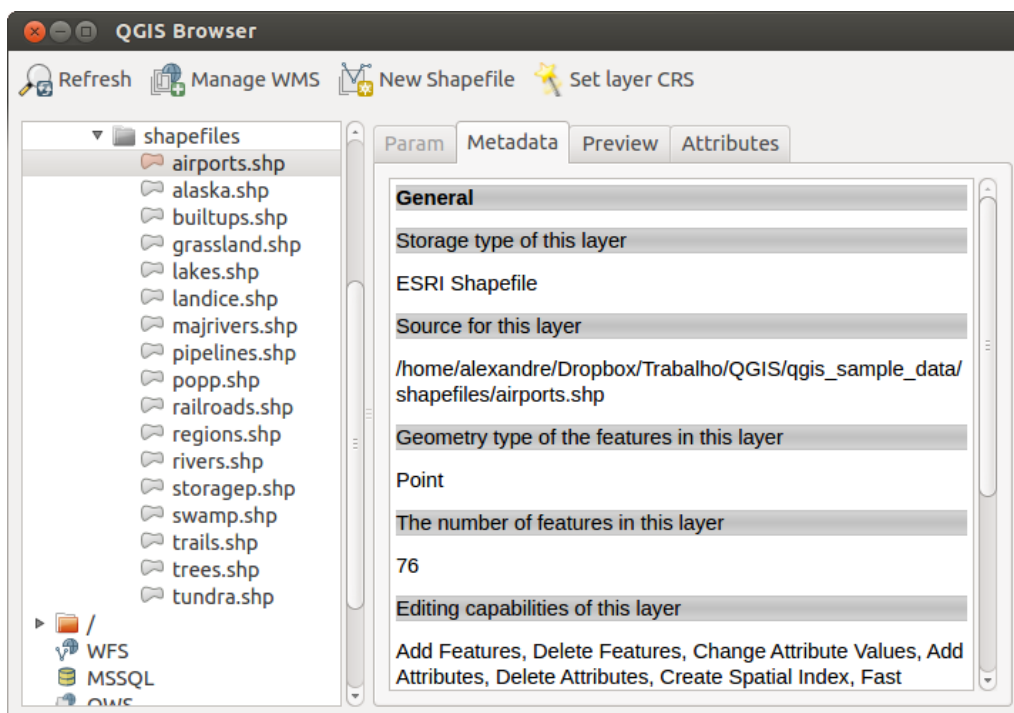



Figura 11.1: QGIS navegador como aplicação padrão 🐧


Use o Navegador QGIS para pré-visualizar seus dados. As funções de arraste e soltar torna mais fácil colocar os dados no visualizador de mapas e na legenda do Mapa.

1. Ative o Pesquisador QGIS: Clique com o direito do mouse na barra de ferramentas e marque a caixa de seleção *Pesquisador* ou selecione a partir de *Configurações* → *Paineis*.
2. Arraste o painel para a janela de legenda e solte-o.
3. Clique no separador *Pesquisar*.
4. Pesquise em seu arquivos de sistemas e escolha a pasta *shapefile* do diretório *qgis_sample_data*.
5. Pressione a tecla :kbd: *Shift* e selecione os arquivos :file: *airports.shp* e *alaska.shp*.
6. Pressione o botão esquerdo do mouse, então arraste e solte os arquivos para a tela do mapa.

7. Clique com o direito do mouse na camada e escolha *Definir o SRC do projeto a partir da camada*. Para mais informações veja *Trabalhando com Projeções*.

8. Clique em  Ampliação Total para tornar todas as camadas visíveis.


Existe um segundo pesquisador disponível em *Configurações* → *Painéis*. Isto é útil quando você precisa mover arquivos ou camadas entre diferentes locais.

1. Ative o segundo Pesquisador QGIS: Clique com o direito do mouse na barra de ferramentas e marque a caixa de seleção  *Pesquisador (2)* ou selecione a partir de *Configurações* → *Painéis*.


2. Arraste o painel para a janela de legenda.

3. Navegue para aba *Pesquisador (2)* e pesquise pelo shapefile no seu sistema de arquivos.

4. Selecione um arquivo com o botão esquerdo do mouse. Agora você pode usar o ícone



 Adicionar Camadas Seleccionadas para adicioná-las no seu projeto atual.

O QGIS procura automaticamente o Sistema Referência de Coordenadas (SRC) e aplica o zoom na extensão da camada se você estiver trabalhando com um projeto QGIS em branco. Se já existirem arquivos no seu projeto, o arquivo será apenas adicionado e no caso que ele tenha a mesma projeção SRC, ele será visualizado. Se o arquivo tiver outro SRC e outra extensão de camada, você deve primeiro clicar com o botão direito do mouse na camada e escolher **:guilabel:'Definir SRC do Projeto na Camada'**. Em seguida escolha :guilabel:'Zoom na extensão da camada'.

A função  Filtrar arquivos trabalha com níveis de diretório. Pesquise a pasta que deseja filtrar os arquivos e procure com uma palavra ou palavra-chave. O Pesquisador irá mostrar apenas nomes de arquivos correspondentes - outros dados não serão exibidos.

Também é possível rodar o Pesquisador QGIS como uma aplicação padrão.

Iniciar o Pesquisador qgis

-  Escreva na linha de comando “qbrowser”.
-  Inicie o Pesquisador QGIS utilizando o Menu Iniciar ou tecla de atalho no Desktop.
- **X** O Pesquisador QGIS está disponível a partir da pasta de Aplicações.

Na *figure_browser_standalone_metadata* você pode visualizar as funcionalidades padrão do Pesquisador QGIS. A aba :guilabel: *Param* fornece os detalhes de suas conexões de dados, like PostGIS ou MSSQL Spatial. A aba *Metadados* contém informações gerais sobre o arquivo. (veja *Menu Metadados*). Com a aba *Pré-visualização* você pode visualizar seus arquivos sem ter que importá-los para um projeto QGIS. Também é possível pré-visualizar os atributos dos seus arquivos na aba *Atributos*.

Trabalhando com Dados Vetoriais

Atualização

12.1 Formatos de dados suportados

QGIS usa a biblioteca OGR para ler e escrever formatos de dados vetoriais, incluindo arquivos de formato ESRI, MapInfo e formatos de arquivo MicroStation, AutoCAD DXF, PostGIS, SpatiaLite, Oracle Spatial e bancos de dados MSSQL espaciais, e muitos mais. Vetor grama e suporte ao PostgreSQL é fornecido pela nativas | qg | provedores de plugins. Dados vetoriais também podem ser carregados no modo de leitura de arquivos ZIP e GZIP em QGIS. A partir da data deste documento, 69 formatos vetoriais são suportados pela biblioteca OGR (ver OGR-SOFTWARE-SUITE em: ref: *literature_and_web*). A lista completa está disponível em http://www.gdal.org/ogr/ogr_formats.html.

Nota: Nem todos os formatos mencionados podem trabalhar no QGIS por várias razões. Por exemplo, alguns exigem bibliotecas comerciais externas, ou a instalação de seu sistema operacional GDAL/OGR pode não ter sido construída para suportar o formato que você deseja usar. Apenas os formatos que foram bem testados irão aparecer na lista de tipos de arquivos ao carregar um raster no QGIS. Outros formatos não testados podem ser carregados selecionando * . * .

Procedimentos com dados vetoriais GRASS são descritos na Seção *Integração com SIG GRASS*.

Esta seção descreve como trabalhar com vários formatos comuns: arquivos ESRI, camadas PostGIS, camadas SpatiaLite, vetores OpenStreetMap e dados Comma Separated (CSV). Muitos dos recursos disponíveis no | QG | independentemente da fonte de dados vetoriais você pode trabalhar sozinho e isso inclui identificar, selecionar, rotular e atribuir funções.

12.1.1 Shapefiles ESRI

O formato de arquivo vetorial padrão usado pelo QGIS é o Shapefile ESRI. O suporte é fornecido pela OGR Simple Feature Library (<http://www.gdal.org/ogr/>).

Um shapefile consiste na verdade de um conjunto de vários arquivos. Os três seguintes são necessários:

1. `.shp` arquivo que contém as formas vetoriais.
2. `.dbf` arquivo que contém os atributos no formato dBase..
3. `.shx` arquivos index.

Shapefiles também podem incluir um arquivo com a extensão `.prj` que contém as informações de projeção. Embora seja muito útil um arquivo de projeção, não é obrigatória. Um conjunto de dados shapefile pode conter arquivos adicionais. Para mais detalhes veja a especificação técnica ESRI em: <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>.

Carregando um Shapefile

Para carregar um arquivo comece o | QG | e clique em mActionAddOgrLayer: Adicionar um Vetor ou simplesmente pressione as teclas 'Ctrl + Shift + V'. Isso fará com que você crie uma nova janela (ver [figura_vetor_1](#)).

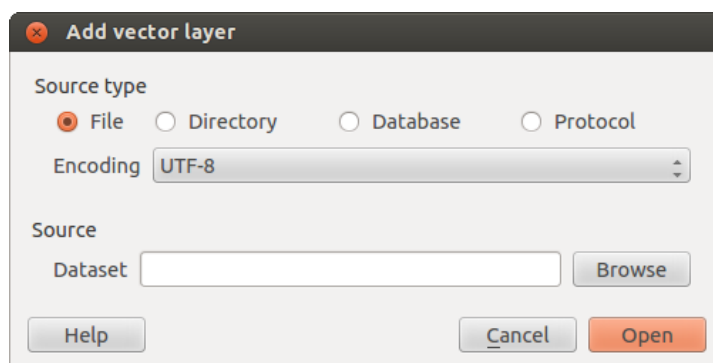


Figura 12.1: Diálogo Adicionando Camada Vetorial

Entre as opções disponíveis verificar : guilabel: *Arquivo*. Clique no [**Browse**]. Isso fará com que um diálogo de arquivo padrão aberto (ver [figure_vector_2](#)), que permite que você navegue no sistema de arquivos e carregue um shapefile ou outra fonte de dados suportados. A caixa de seleção: guilabel: 'Filtro' permite você pré-selecionar alguns formatos de arquivos OGR suportados.

Você também pode selecionar a codificação para o shapefile, se desejar.

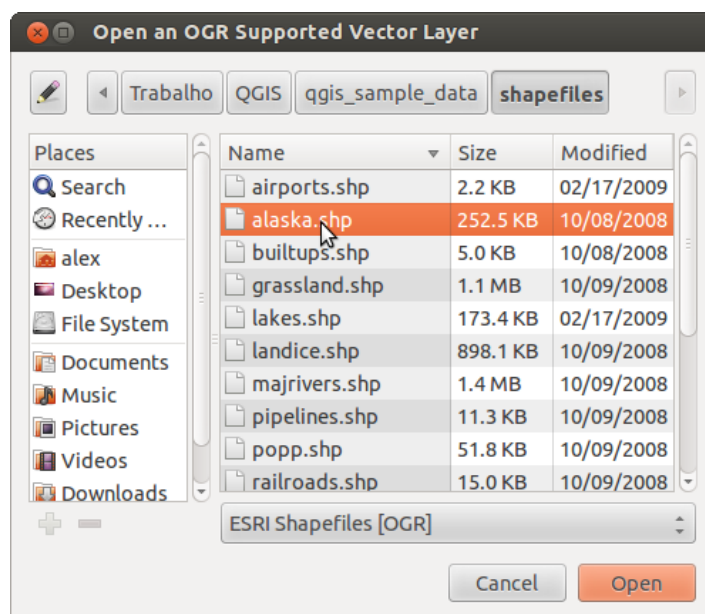


Figura 12.2: Diálogo abrir Camada Vetorial OGR Suportada

Ao selecionar um shapefile a partir da lista e clicando em [**Abrir**] o QGIS o carregará. A [Figure_vector_3](#) exibe o QGIS depois de carregar o arquivo `alaska.shp`.

Dica: Cores das camadas

Quando você adiciona uma camada ao mapa, é atribuída uma cor aleatória. Na adição de mais de uma camada de uma vez, são atribuídas cores diferentes para cada camada.

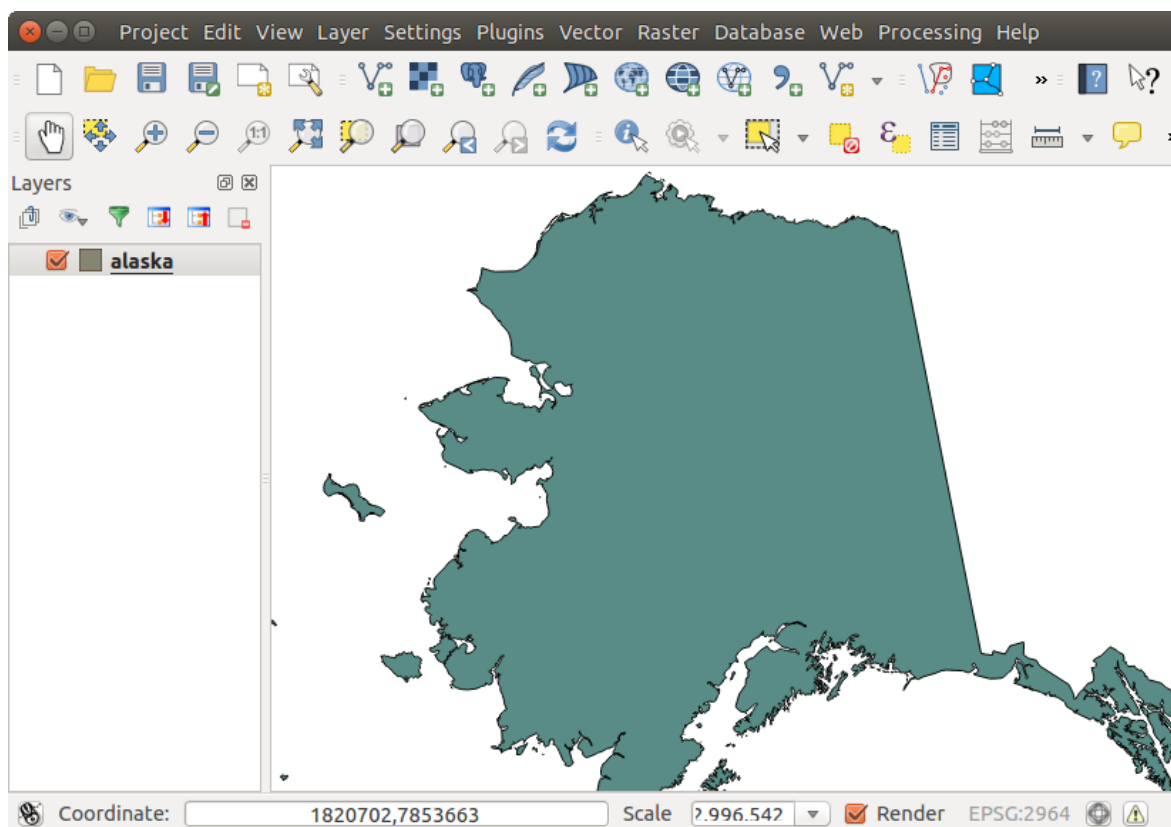


Figura 12.3: QGIS com Shapefile do Alaska carregado 🐧

Uma vez carregado, você pode aplicar zoom em todo o shapefile usando as ferramentas de navegação do mapa. Para alterar o estilo de uma camada, abra a caixa de diálogo *Propriedades da Camada* clicando duas vezes no nome da camada ou clicando com o botão direito do mouse sobre o nome na legenda e escolha *Propriedades* a partir do menu context. Consulte a Seção *vector_style_tab* para mais informações sobre a definição da simbologia das camadas vetoriais.

Dica: Carregando camadas e armazenando projetos em unidades externas no OS X

No OS X, unidades portáteis que são armazenadas além do disco rígido principal não aparecem no *File* → *Open Project* como esperado. Para resolver isso estamos trabalhando em uma forma mais nativa do OS X com caixa de diálogo abrir/salvar. Como alternativa, você pode digitar */Volumes* no campo Nome do arquivo e pressione *kbd: Voltar*. Depois, você pode navegar para unidades externas ou armazenadas em rede.

Melhorando o desempenho de Shapefiles

Para melhorar o desempenho do desenho de um shapefile, você pode criar um índice espacial. Um índice espacial irá melhorar a velocidade do zoom e visão panorâmica. Os índices espaciais utilizados pelo QGIS estão na extensão *.qix*.

Siga estes passos para criar o índice:




- Carregue um shapefile clicando no ícone **Add Vector Layer** na barra de ferramentas ou pressionando *kbd: Ctrl + Shift + V*.
- Abra a caixa de diálogo *Propriedades da camada* clicando duas vezes sobre o nome do shapefile na legenda ou com o botão direito do mouse e escolha *Propriedades* a partir do menu de contexto.
- Na guia *guiabel: Geral*, clique no botão **[Criar índice espacial]**.

Problema ao carregar um arquivo shape .prj

Se você carregar um arquivo shapefile com o **arquivo:'.prj'** e QGIS o sistema não é capaz de ler a referência de coordenadas a partir desse arquivo, você tem que definir a projeção adequada manualmente dentro guilabel: 'Geral' da caixa de diálogo *Propriedades da camada*. Isto é devido ao fato de que arquivos `.prj` muitas vezes não fornecem os parâmetros de projeção completos, conforme usado no QGIS e listados na guia *CRS*.

Por essa razão, se você criar um novo shapefile no QGIS, dois arquivos de projeção diferentes são criados. Um arquivo `.prj` com os parâmetros limitados de projeção, compatível com o software ESRI, e um arquivo `.qpj`, fornecendo os parâmetros completos dos SRC usados. Se o QGIS encontra um arquivo `.qpj`, ele será usado em vez do arquivo `prj`.

12.1.2 Carregando uma camada MapInfo

 Para carregar uma camada MapInfo, clique no botão  Adicionar Camada Vetor da barra de ferramenta; ou tipo `Ctrl+Shift+V`, mudar o arquivo do tipo filtro *Arquivos de Tipo*  : para 'Arquivo Mapinfo [OGR] (*.mif *.tab *.MIF *.TAB)' e selecione a camada MapInfo que voce irá carregar.

12.1.3 Carregando uma camada ArcInfo Binary

Para carregar um Arquivo/Informação clique no mActionAddOgrLayer: Adicionar Vetor no botão da barra de ferramentas ou pressione as teclas 'Ctrl + Shift + V' para abrir o guilabel: Adicionar uma informação ao Vetor. Selecione o botão do guilabel e o Diretório do guilabel. Altere o tipo de arquivo do guilabel de SelectString para Arquivo/Informação. Navegue até o diretório que contém o arquivo e selecione.

Da mesma forma, você pode carregar arquivos vetoriais baseados em um diretório no formato UK National Transfer, bem como os formatos TIGER do Census Bureau US.

12.1.4 Arquivos de texto delimitado

Dados em tabelas é um formato muito comum e amplamente utilizado devido à sua simplicidade e facilidade de acesso - os dados podem ser visualizados e editados até mesmo em um editor de texto simples. Um arquivo de texto delimitado é uma tabela de atributos com cada coluna separados por um caracter definido e cada linha separada por uma quebra de linha. A primeira linha geralmente contém os nomes das colunas. Um tipo comum de arquivo de texto delimitado é um arquivo CSV (Comma Separated Values), com cada coluna separado por uma vírgula.

Esses arquivos de dados também podem conter informações sobre a posição em duas formas principais:

- Coordenadas de ponto separadas por colunas
- Texto bem conhecido (WKT) representado por geometria

QGIS allows you to load a delimited text file as a layer or ordinal table. But first check that the file meets the following requirements:

1. O arquivo deve ter uma linha de cabeçalho delimitada de nomes de campo. Esta deve ser a primeira linha do arquivo de texto.
2. A linha de cabeçalho deve conter campo (s) com a definição da geometria. Estes campo (s) pode ter qualquer nome.
3. As coordenadas X e Y (se a geometria for definida pelas coordenadas) deverá ser especificada como números. O sistema de coordenadas não é importante.


Como um exemplo de um arquivo de texto válido, nós importamos o arquivo de elevação de dados de pontos `elevp.csv` que vem com o conjunto de dados da amostra do QGIS (ver seção: ref: *label_sampledata*):

```
X;Y;ELEV
-300120;7689960;13
-654360;7562040;52
1640;7512840;3
[...]
```

Alguns itens a serem observados sobre o arquivo de texto:

1. O arquivo de texto exemplo usa ; (ponto e vírgula) como delimitador. Qualquer caractere pode ser usado para delimitar os campos.
2. A primeira linha é a linha de cabeçalho. Ela contém os campos X, Y e ELEV.
3. Sem aspas (") são usados para delimitar os campos de texto.
4. A coordenada X está contida no campo X.
5. A coordenada Y está contida no campo Y.

Carregando um arquivo de texto delimitado

Clique no ícone da barra de ferramentas  na barra de ferramentas *Gerenciar camadas* para abrir o diálogo *Criar uma camada a partir de um arquivo de texto delimitado*, como mostrado na [figure_delimited_text_1](#).

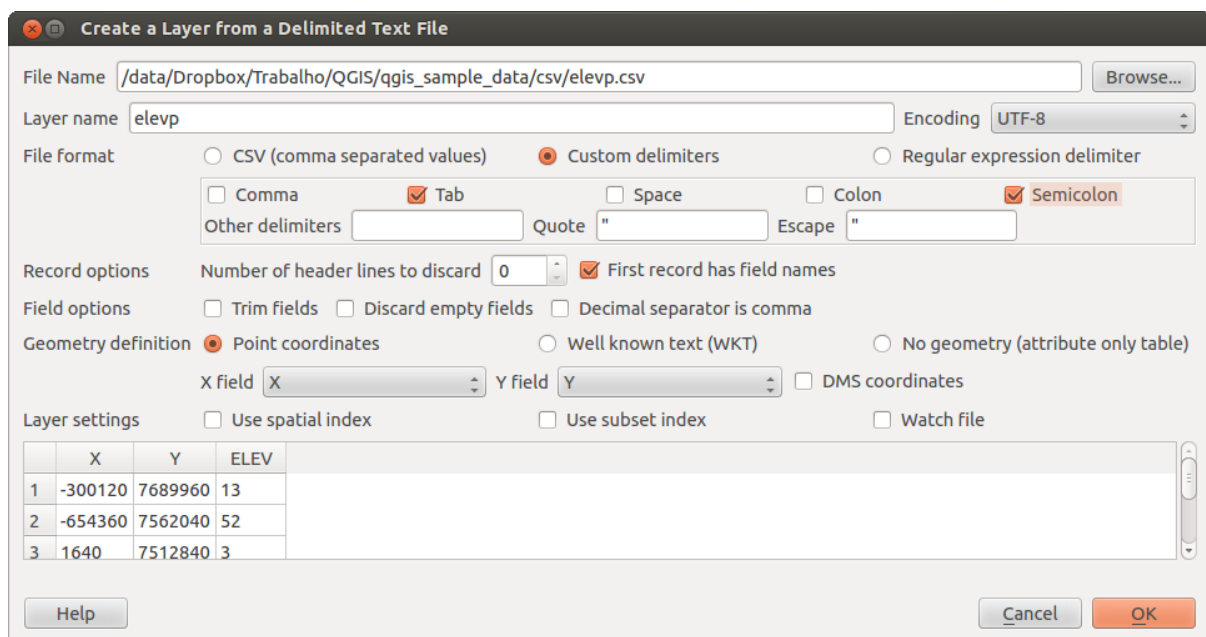



Figura 12.4: Diálogo Texto Delimitado 

Primeiro, selecione o arquivo a ser importado (por exemplo: arquivo 'qgis_sample_data/csv/elevp.csv') clicando no botão do [navegador]. Quando o arquivo for selecionado, o QGIS tenta analisar o arquivo com o delimitador usado mais recentemente. Para ativar o QGIS para analisar corretamente o arquivo, é importante selecionar o delimitador correto. Você pode especificar um delimitador ativando pelo botão no guilabel, pelo delimitador personalizado ou pelo texto no guilabel: 'Expressão'. Por exemplo, para alterar o delimitador de tabulação, use "t" (esta é uma expressão para o caractere de tabulação).




Uma vez que o arquivo é analisado, defina o guilabel, selecione o botão: 'definição de geometria'. As coordenadas são o ponto 'X' e o campo 'Y' das listas suspensas. As coordenadas são definidas como graus/minutos/segundos, ative a caixa de seleção no guilabel 'DMS'.

Por fim, digite um nome para a camada (por exemplo: arquivo 'elevp'), como mostrado na [figura_texto_1](#). Para adicionar a camada ao mapa, clique **[OK]**. O arquivo de texto delimitado agora se comporta como qualquer outra camada de mapa em QGIS.

Há também uma opção de ajuda que permite cortar espaços iniciais e finais dos campos no guilabel: 'campos da guarnição'. Além disso, é possível no guilabel: descartar campos vazios. Se necessário, você pode forçar uma vírgula para ser o separador decimal ativando no guilabel: 'separador decimal por vírgula'.

Se a informação espacial é representada por WKT, ative o botão no guilabel: na opção 'Texto' e selecione o campo com a definição WKT para ponto, linha ou polígono. Se o arquivo contém dados não-espaciais, ative o botão no guilabel: 'Não' e ele será carregado como uma tabela.

Adicionalmente, você pode ativar:


-  *Utilize índice espacial* para melhorar o desempenho de exibição e espacialmente a seleção das feições.
-  *Utilize índice de subconjunto.*
-  *Assista arquivo* para observar as alterações no arquivo por outros aplicativos enquanto o QGIS está sendo executado.

12.1.5 Dados OpenStreetMap

Nos últimos anos, o projeto OpenStreetMap ganhou popularidade porque, em muitos países há dados geográficos gratuitos, tais como mapas de estradas digitais estão disponíveis. O objetivo do projeto OSM é criar um mapa livre e editável do mundo a partir de dados de GPS, fotografia aérea ou conhecimento local. Para apoiar este objetivo, o QGIS fornece suporte para dados OSM.

Carregando vetores OpenStreetMap

QGIS integra importação OpenStreetMap como funcionalidade principal.

- Para conectar-se aos dados do servidor OSM e de download, abra o menu: 'Vetor -> OpenStreetMap -> Carregar dados'. Você pode pular esta etapa se você já obteve um arquivo XML usando .osm, JOSM, API ou qualquer outra fonte.
- No menu 'Vetor -> OpenStreetMap -> Importar topologia de um arquivo XML' irá converter o seu: arquivo .osm em um banco de dados SpatiaLite e criar uma conexão de banco de dados correspondente.
- O menu 'Vetor -> OpenStreetMap -> Exportar topologia para SpatiaLite' permite que você abra a conexão de banco de dados, selecione o tipo de dados que você quer (pontos, linhas ou polígonos) e escolher as tags para importação. Isso cria uma camada de geometria SpatiaLite que você pode adicionar ao seu projeto, clicando no  'Adicionar SpatiaLite Layer' na barra de ferramentas ou selecionando a | mActionAddSpatiaLiteLayer | : 'Adicionar Camada SpatiaLite' opção no menu Camada (ver seção:: ref: label SpatiaLite).

12.1.6 Camadas PostGIS

Camadas PostGIS são armazenados em um banco de dados PostgreSQL. As vantagens do PostGIS são a indexação espacial, filtragem e recursos de consulta. Usando o PostGIS, funções vetoriais, como selecionar e identificar permitem um trabalho com mais precisão do que com camadas OGR no QGIS.

Criando uma conexão armazenada

A primeira vez que você usar uma fonte de dados PostGIS, você deve criar uma conexão com o banco de dados PostgreSQL que contém os dados. Comece clicando no botão | mActionAddPostgisLayer | : sup: 'Adicionar PostGIS', selecionando a | mActionAddPostgisLayer | : 'Adicionar Camada PostGIS' ou digitando as teclas: 'Ctrl + Shift + D'. Você também pode abrir a guilabel: 'Adicionar Vetor' e selecione o botão banco de dados. A guilabel

‘Adicionar Tabela PostGIS’ será exibida. Para acessar o gerenciador de conexão, clique no **[novo]** para exibir o guilabel: ‘Criar uma nova conexão PostGIS’. Os parâmetros necessários para uma conexão são:


- **Nome:** Um nome para esta conexão. Pode ser o mesmo que o da *base de dados*.
- **Serviço:** parâmetro de serviço a ser usado como alternativa ao nome do host/porta (e, potencialmente, banco de dados). Esta pode ser definida em: arquivo ‘pg_service.conf’.
- **Host:** Nome do host no banco de dados. Este deve ser um nome de host usado para abrir uma conexão telnet ou ping. Se o banco de dados está no mesmo computador como **QGIS**, basta digitar ‘localhost’.
- **Porta:** Número da porta do servidor de banco de dados PostgreSQL atende. A porta padrão é 5432.
- **Banco de dados:** Nome do banco de dados.
- **Modo SSL:** A conexão SSL será negociado com o servidor. Note-se que a velocidade no PostGIS por camada pode ser alcançada desativando o SSL na conexão. As seguintes opções estão disponíveis:
 - Desativar: tentar Apenas uma conexão SSL não criptografada.
 - Permitir: Tente uma conexão não-SSL. Se isso não funcionar, tente uma conexão SSL.
 - Prefira (o padrão): Tente uma conexão SSL. Se isso não funcionar, tente uma conexão não-SSL.
 - Exigir: Apenas tentar uma conexão SSL.
- **Nome do usuário:** Nome do usuário usado para logar na base de dados.
- **Senha:** Senha usada com *Nome do Usuário* para conectar ao banco de dados.

Opcionalmente, você pode ativar as seguintes caixas de seleção:


- *Salvar nome do usuário*
- *Salvar Senha*
- *Apenas olhar na tabela geometry_columns*
- : guilabel: *Não resolve tipo de colunas sem restrições (geometria)*
- :guilabel: ‘Apenas olhar no esquema ‘público’
- *Também listar tabelas sem geometrias*
- *Usar metadados atribuídos da tabela*

Uma vez que todos os parâmetros e as opções estão configuradas, você pode testar a conexão clicando no botão **[Testar Conexão]**.

Carregando uma camada PostGIS

 Uma vez que você tem uma ou mais conexões definidas, você pode carregar as camadas do banco de dados PostgreSQL. Claro que isso exige ter dados no PostgreSQL. Consulte a Seção [Importação de dados no PostgreSQL](#) para entender sobre a importação de dados para o banco de dados.

Para colocar uma camada no PostGIS, execute os seguintes passos:

- Se o: guilabel: diálogo ‘Adicionar camadas PostGIS ‘ já não estiver aberto, selecionar o  :menuselection: ‘Adicionar PostGIS Camada ...’ opção no: menuselection: Menu *camada* ou digitando: kbd: ‘Ctrl + Shift + D ‘ abre o diálogo.
- Escolha a conexão a partir da lista solta a baixo e clique em **[Conectar]**.
- Marque ou desmarque *Listar tabelas sem geometrias*
- Opcionalmente, use no guilabel: ‘Opções de pesquisa’ para definir quais recursos carregar a partir da uma camada, ou usar o botão **** [Construtor de consultas] ****.

- Encontre a camada(s) que deseja adicionar na lista de camadas disponíveis.
- Selecione-a clicando sobre ela. Você pode selecionar várias camadas pressionando a tecla `Shift` enquanto clica. Consulte a Seção [Ferramenta de Consulta](#) para obter informações sobre como usar o Query Builder PostgreSQL para melhor definição da camada.
- Clique no botão **[Add]** para adicionar a camada ao mapa.

Dica: Camadas PostGIS

Normalmente uma camada PostGIS é definida por um caminho na tabela. A partir da versão 0.9.0 em diante, o QGIS permite carregar camadas que não têm um caminho na tabela `geometry_columns`. Isto inclui ambas as tabelas e exibições. A definição de uma visão espacial fornece um meio poderoso para visualizar seus dados. Consulte o manual do PostgreSQL para obter informações sobre a criação de pontos de vista diversos.

Alguns detalhes sobre as camadas PostgreSQL

Esta seção contém alguns detalhes sobre como o QGIS acessa camadas do PostgreSQL. Na maioria das vezes o QGIS simplesmente deve fornecer-lhe uma lista de tabelas do banco de dados que podem ser carregadas, e carregá-las mediante solicitação. No entanto, se você tiver problemas para carregar uma tabela do PostgreSQL no QGIS, as informações abaixo podem ajudá-lo a entender as mensagens do QGIS e dar-lhe orientações para alterar a tabela de PostgreSQL ou ver definições para permitir o QGIS carregá-la.

QGIS requer que as camadas PostgreSQL contendam uma coluna que pode ser usada como uma chave única para a camada. Para tabelas, isso geralmente significa que a tabela precisa de uma chave primária, ou uma coluna com uma restrição exclusiva sobre ela. No QGIS, esta coluna tem de ser do tipo `int4` (um inteiro de 4 bytes). Alternativamente, a coluna `ctid` pode ser usada como chave primária. Se uma tabela não tem esses itens, a coluna `oid` será usado em seu lugar. O desempenho será melhorado se a coluna é indexada (note que as chaves primárias são automaticamente indexadas pelo PostgreSQL).


Se o banco de dados PostgreSQL é um ponto de referência, existe o mesmo requisito, mas não têm chaves ou colunas primárias com restrições exclusivas sobre eles. Você tem que definir um campo de chave primária (tem que ser inteiro) no `!QG!` antes de você carregar. Se uma coluna adequada não existe como requisito, `!QG!` não carregará. Se isso ocorrer, a solução é alterar o modo de exibição para ele incluir uma coluna adequada (um tipo inteiro com chave primária ou uma restrição exclusiva, de preferência indexado).

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. It can make sense to disable this option when you use expensive views.

12.1.7 Importação de dados no PostgreSQL

Os dados podem ser importados para PostgreSQL/PostGIS utilizando várias ferramentas, incluindo o plugin SPIT e as ferramentas de linha de comando `shp2pgsql` e `ogr2ogr`.

Gerenciador BD

QGIS comes with a core plugin named  `DB Manager`. It can be used to load shapefiles and other data formats, and it includes support for schemas. See section [Complemento Gerenciador BD](#) for more information.

`shp2pgsql`

PostGIS inclui um utilitário chamado `shp2pgsql` que pode ser usado para importar arquivos em um banco de dados habilitado para PostGIS. Por exemplo, para importar um arquivo denominado 'lakes.shp' em um banco de dados PostgreSQL chamado "gis_data", use o seguinte comando:

```
shp2pgsql -s 2964 lakes.shp lakes_new | psql gis_data
```

Isso cria uma nova camada chamada `lakes_new` no banco de dados `gis_data`. A nova camada terá um identificador de referência espacial (SRID) de 2964. Consulte a Seção *Trabalhando com Projeções* para mais informações sobre os sistemas de referência espacial e projeções.

Dica: Exportando conjuntos de dados do PostGIS

Assim como a ferramenta de importação **shp2pgsql**, há também uma ferramenta para exportar conjuntos de dados PostGIS como shapefiles: **pgsql2shp**. Este é enviado dentro de sua distribuição PostGIS.


ogr2ogr

Junto ao **shp2pgsql** e ao **SPIT** existe uma outra ferramenta para a inserção de dados geográficos no PostGIS: **ogr2ogr**. Esta é parte da instalação do GDAL.

Para importar um shapefile no PostGIS, faça o seguinte :

```
ogr2ogr -f "PostgreSQL" PG:"dbname=postgis host=myhost.de user=postgres
password=topsecret" alaska.shp
```

Isto irá importar o shapefile `alaska.shp` para o banco de dados PostGIS `postgis` usando o usuário `postgres` com a senha `topsecret` no servidor `myhost.de`.

Note que o OGR deve ser construído com PostgreSQL para suportar PostGIS. Você pode ver isso digitando (in )

```
ogrinfo --formats | grep -i post
```

Se você preferir usar o PostgreSQL com o comando **CÓPIA** em vez do padrão **INSERIR**, você pode exportar a seguinte variável de ambiente (pelo menos disponível em `linux` e `OSX` !):

```
export PG_USE_COPY=YES
```

ogr2ogr **** não cria índices espaciais como `shp2pgsql` **. Você precisa criá-los manualmente, usando o comando SQL `CRIAR` como um passo extra (como descrito na próxima seção: ref: ‘melhorando a performance do vetor’).**

melhorando o Desempenho

Recuperando recursos a partir de um banco de dados PostgreSQL pode ser demorado, especialmente através da rede. Você pode melhorar o desempenho do desenho das camadas PostgreSQL, garantindo que o índice espacial PostGIS existe em cada camada do banco de dados. PostGIS suporta a criação de um índice GiST (Pesquisa em forma de Árvore) para acelerar as pesquisas espaciais dos dados (informações a partir da documentação disponível no PostGIS <http://postgis.refrains.net>).

A sintaxe para criar um índice GiST é:

```
CREATE INDEX [indexname] ON [tablename]
  USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

Observe que para grandes tabelas, a criação do índice pode levar um longo tempo. Uma vez que o índice é criado, você deve realizar um `VACUUM ANALYZE`. Consulte a documentação do PostGIS (PostGIS-PROJECT *Referências Bibliográficas e Web*) para mais informações.

A seguir um exemplo de criação de um índice GiST:

```
gsherman@madison:~/current$ psql gis_data
Welcome to psql 8.3.0, the PostgreSQL interactive terminal.
```

```
Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help with psql commands
```

```

\g or terminate with semicolon to execute query
\q to quit

gis_data=# CREATE INDEX sidx_alaska_lakes ON alaska_lakes
gis_data=# USING GIST (the_geom GIST_GEOMETRY_OPS);
CREATE INDEX
gis_data=# VACUUM ANALYZE alaska_lakes;
VACUUM
gis_data=# \q
gsherman@madison:~/current$

```

12.1.8 Camadas vetoriais que ultrapassam 180 |graus| de longitude

Muitos pacotes de SIG não envolve mapas vetoriais com um sistema geográfico de referência (latitude/longitude) cruzando a linha de longitude de 180 graus (http://postgis.refrations.net/documentation/manual-2.0/ST_Shift_Longitude.html). Como resultado, se abrirmos esse mapa em |QG|, vamos ver dois pontos, locais distintos, que deve aparecer perto um do outro. Na *Figura_vetor_4*, o ponto minúsculo no canto esquerdo da tela do mapa (Ilhas Chatham) deve estar dentro da grade, à direita das principais ilhas da Nova Zelândia.



Figura 12.5: Mapa em lat/long atravessando a linha de longitude de 180 |graus|

A solução alternativa é transformar os valores de longitude utilizando PostGIS e a função `** ST_Shift_Longitude**`. Esta função lê cada ponto/vértice dos componentes com recursos de geometria, e se coordenar a longitude < 0 graus, se acrescenta 360 graus a ele. O resultado é 0 graus - 360 graus a serem plotados em um 180 graus em um mapa centralizado.

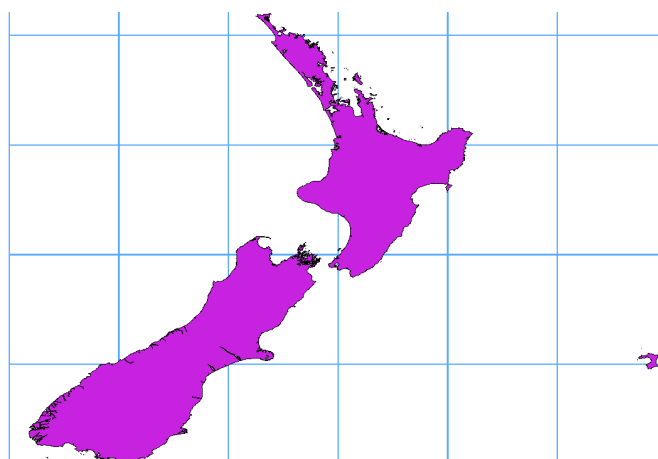


Figura 12.6: Cruzando 180 |graus| longitude aplicação da função `** ST_Shift_Longitude **`

Uso

- Importar dados para o PostGIS (ref: 'vetor_import_data_in_postgis'), utilizando, por exemplo, o complemento Gerenciador de DB.

- Use a interface do PostGIS para emitir o seguinte comando (neste exemplo, “tabela” é o nome real da tabela PostGIS): “gis_data = # update TABELA set the_geom = ST_Shift_Longitude (the_geom);”.
- Se tudo correu bem, você deve receber uma confirmação sobre o número de recursos que foram atualizados. Então você vai ser capaz de carregar o mapa e veja a diferença (Figura_vetor_5).

12.1.9 Camadas SpatialLite

A primeira vez que você carregar dados de um banco de dados SpatialLite, comece clicando no `mActionAddSpatialLiteLayer` arquivo: ‘Adicionar Camada SpatialLite, ou selecionando a `mActionAddSpatialLiteLayer`: ‘Adicionar Camada SpatialLite’ ... opção do menu de seleção ou digitando as teclas: ‘Ctrl + Shift + L’. Isso abrirá uma janela que permitirá que você se conecte a um banco de dados SpatialLite já conhecido por `QGIS`, que você pode escolher entre o menu flutuante, ou definir uma nova conexão com um novo banco de dados. Para definir uma nova conexão, clique em `** [Novo] **` e use o navegador para mostrar o banco de dados SpatialLite, que é um arquivo com extensão `.sqlite`.

Se você quiser salvar uma camada do vetor para o formato SpatialLite, você pode fazer isso com um clique direito na legenda. Em seguida, clique em ‘Salvar como...’, defina o nome do arquivo final, e selecione ‘SpatialLite’ como formato e oSRC. Além disso, você pode selecionar ‘SQLite’ como formato e, em seguida, adicionar “SpatialLite = SIM” no campo opção da fonte de dados OGR. Isto diz OGR para criar um banco de dados SpatialLite. Veja também http://www.gdal.org/ogr/drv_sqlite.html.

QGIS também suporta visualizações editáveis em SpatialLite.




Criando uma nova camada SpatialLite

Se você quiser criar uma nova camada SpatialLite, consulte a seção *Criando uma nova camada SpatialLite*.

Dica: Gestão complementos de dados SpatialLite

Para o gerenciamento de dados SpatialLite, você também pode usar vários complementos em Python: QSpatialLite, Gerenciamento SpatialLite ou Gerenciamento de DB (plugin de núcleo, recomendado). Se necessário, eles podem ser baixados e instalados com o Plugin Installer.

12.1.10 Camadas Espaciais MSSQL

 QGIS also provides native MS SQL 2008 support. The first time you load MSSQL Spatial data, begin by clicking on the  Add MSSQL Spatial Layer toolbar button or by selecting the  Add MSSQL Spatial Layer... option from the *Layer* menu, or by typing `Ctrl+Shift+M`.

12.1.11 Camadas Espaciais Oracle

As feições espaciais em usuários de próteses Oracle Espacial na gestão de dados geográficos e de localização em um tipo nativo dentro de um banco de dados Oracle. O QGIS agora tem suporte para essas camadas.

Criando uma conexão armazenada

A primeira vez que você usar uma fonte de dados Oracle Espacial, você deve criar uma conexão com o banco de dados que contém os dados. Comece clicando no `mActionAddOracleLayer` arquivo: ‘Adicionar Oracle Espacial’, selecionando a `mActionAddOracleLayer`: ‘Adicionar Oracle Camada Espacial ...’ opção no menu ou digitando as teclas: ‘Ctrl + Shift + O’. Para acessar o gerenciador de conexão, clique em `**[novo]**` para exibir o guilabel: ‘Criar uma nova conexão com Oracle Espacial’. Os parâmetros necessários para uma conexão são:

- **Nome:** Um nome para esta conexão. Pode ser o mesmo que *base de dados*

- **Base de dados:** SID ou SERVICE_NAME da instância Oracle.
- **Host:** Nome do host no banco de dados. Este deve ser um nome de host usado para abrir uma conexão telnet ou ping. Se o banco de dados está no mesmo computador como **QGIS**, basta digitar *'localhost'*.
- **Port:** Número da porta do servidor de banco de dados PostgreSQL. A porta padrão é 1521.
- **Usuário:** Nome de usuário usado para acessar o banco de dados.
- **Senha:** Senha usada com *Nome do Usuário* para conectar ao banco de dados.



Opcionalmente, você pode ativar caixas de seleção a seguir:

- **caixa:** 'Salvar Nome de Usuário' Indica salvar o nome de usuário do banco de dados na configuração da conexão.
- **caixa:** 'Salvar senha' Indica se é para salvar a senha do banco de dados nas configurações de conexão.
- : guilabel: *Olhe somente na tabela de metadados* Restringe as tabelas exibidas para aqueles que estão na visão `all_sdo_geom_metadata`. Isso pode acelerar a exibição inicial de tabelas espaciais.
- **caixa:** 'Verificar apenas as tabelas do usuário' Ao procurar tabelas espaciais, restringir a pesquisa para tabelas que são de propriedade do usuário.
- : guilabel: *Também lista tabelas sem geometria* Indica que tabelas sem geometria também devem ser listados por padrão.
- *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.
- : guilabel: *Somente tipos de geometria existente* listar apenas os tipos de geometria existentes e não oferecem para adicionar outros.

Uma vez que todos os parâmetros e as opções estão configuradas, você pode testar a conexão clicando no botão **[Testar Conexão]**.

Dica: QGIS Configurações do Usuário e Segurança

Dependendo do seu ambiente de computacional, armazenamento de senhas nas suas configurações QGIS pode ser um risco de segurança. As senhas são salvas em texto claro na configuração do sistema e nos arquivos do projeto! Suas configurações personalizadas para o QGIS são armazenadas com base no sistema operacional:

-  As configurações são armazenadas em seu diretório home em `~/.qgis2`.
-  As configurações são armazenadas no registro.

Carregando uma camada Espacial Oracle



Uma vez que você tem uma ou mais conexões definidas, você pode carregar camadas do banco de dados Oracle. Claro, isso exige ter dados em Oracle.

Para carregar uma camada Espacial Oracle, execute os seguintes passos:

- Se a caixa de diálogo Oracle Espacial já estiver aberta clique em: 'Adicionar' arquivo: `mActionAddOracleLayer`.
- Escolha a conexão a partir da lista solta a baixo e clique em **[Conectar]**.
- Marque ou desmarque *Listar tabelas sem geometrias*

- Opcionalmente, use o guilabel: ‘Opções de pesquisa’ para definir quais recursos será carregado a partir da camada ou use o botão de **[Consulta]**.
- Encontre a camada(s) que deseja adicionar na lista de camadas disponíveis.
- Selecione-o clicando sobre ele. Você pode selecionar várias camadas, mantendo pressionada a tecla ‘Shift’ enquanto clica. Consulte a seção ref: ‘vetor_query_builder’ para obter informações sobre o uso de Oracle Consultas para definir a camada.
- Clique no botão **[Add]** para adicionar a camada ao mapa.

Dica: Camadas Espacial Oracle

Normalmente, uma camada Espacial Oracle é definida por uma entrada na tabela **USER_SDO_METADATA**.

12.2 A Biblioteca de Símbolos

12.2.1 Apresentação

A Biblioteca de Símbolos é o lugar onde os usuários podem criar símbolos genéricos para serem usados em vários projetos QGIS. Ele permite aos usuários exporte e importe símbolos, grupos de símbolos e adicione, edite e remova os símbolos. Você pode abri-la com a guia *Configurações* → *Biblioteca de Estilo* ou do **Estilo** na camada vetor *Propriedades*.


Compartilhe e importe símbolos

Os usuários podem exportar e importar símbolos em dois formatos principais: qml (formato QGIS) e SLD (padrão OGC). Note-se que formato SLD não é totalmente suportado pelo QGIS.

usuário compartilhar item exibe uma lista suspensa que permitir que os símbolos de importação ou de exportação do usuário.

Grupos e grupos inteligentes

Os grupos são categorias de símbolos e grupos inteligentes são grupos dinâmicos.

Para criar um grupo, clique com o botão direito em um grupo existente ou sobre os principais diretório **Grupos** no lado esquerdo da biblioteca. Você também pode selecionar um grupo e clique no botão  adicionar item.

Para adicionar um símbolo em um grupo, você pode clique direito em um símbolo, em seguida, escolha: *menuselection: Aplicar ao grupo* e, em seguida, o nome do grupo adicionado antes. Há uma segunda maneira de adicionar vários símbolos em grupo: basta selecionar um grupo e clicar *mActionChangeLabelProperties* e escolher **Símbolos do Grupo**. Todos os símbolos exibem uma caixa de seleção que permite que você adicione o símbolo para os grupos selecionados. Quando terminar, você pode clicar no mesmo botão e escolha **Finalizar Agrupamento**.

Criar **Símbolos inteligentes** é semelhante à criação de grupo, mas neste selecione **Grupos Inteligentes**. A caixa de diálogo permite que o usuário escolha a expressão para selecionar símbolos de forma a aparecer no grupo inteligente (contém algumas tags, membro de um grupo, tem um texto com seu nome, etc.)

Adicionar, editar, remover símbolo

Com o: guilabel: *Gerenciador de Estilos* do menu **[Símbolo]** | *SelectString*, você pode gerenciar seus símbolos. Você pode | *mActionSignPlus* **!:** **sup:** ‘**adicionar item**’, **edição:** *sup:editar item*, | *mActionSignMinus* **!:**

sup:‘remove item’ e usuário Compartilhar item. Símbolos ‘marcador’, símbolos ‘Linha’, padrões de ‘Preenchimento’ e ‘rampas de cores’ podem ser usados para criar os símbolos. Os símbolos são atribuídos a ‘todos os símbolos’, ‘Grupos’ ou ‘grupos inteligentes’.

Para cada tipo de símbolos, você encontrará sempre a mesma estrutura de diálogo:

- na parte superior do lado esquerdo uma representação símbolo
- sob a representação símbolo a árvore símbolo mostrar as camadas símbolo
- à direita você pode configurar algum parâmetro (unidade, transparência, cor, tamanho e rotação)
- sob estes parâmetros você encontra algum símbolo da biblioteca de símbolos

A árvore símbolo permitem adicionar, remover ou proteger novo símbolo simples. Você pode mover para cima ou para baixo da camada de símbolo.

Mais definições detalhadas podem ser feitas ao clicar no segundo nível no diálogo : guilabel: *Símbolos da camada*. Você pode definir: guilabel: *Símbolo da camada* que são combinados depois. Um símbolo pode ser composta por vários: guilabel: *Símbolo da camada*. Configurações serão mostradas mais adiante neste capítulo.

Dica: Note-se que uma vez que você definir o tamanho nos níveis mais baixos do diálogo : guilabel: *Símbolos das camadas*, o tamanho de todo o símbolo pode ser alterado com o menu *Tamanho* no primeiro nível novamente. O tamanho dos níveis mais baixos muda consequentemente, enquanto a proporção de tamanho é mantido.

12.2.2 Marcador de Símbolos

Marcadores de Símbolos possuem vários tipos símbolo da camada:

- Marcador Elipse
- Marcador da Fonte
- Marcado Simples (padrão)
- Marcadores SVG
- Marcadores de campos vetoriais

A seguintes configurações são possíveis:

- : guilabel: *Tipo do Símbolo da camada*: Você tem a opção de usar marcadores de elipse, marcadores de fonte, marcadores simples, marcadores SVG e marcadores de campo vetorial.
- *cores*
- *Tamanho*
- *Estilo de contorno*
- *Largura do contorno*
- *Ângulo*
- : guilabel: ‘Deslocamento X,Y’: Você pode mudar o símbolo na direção -x ou -y.
- *Ancorar ponto*
- *Definir propriedades dos dados ...*

12.2.3 Símbolos de linha

Símbolos marcador de linha têm apenas dois tipos símbolo da camada:

- Marcadores de linha
- Linha simples (padrão)

O tipo camada símbolo padrão desenha uma linha simples, enquanto a outra tela um marcador de pontos regulares na linha. Você pode escolher diferentes localizações de vértices, intervalos ou pontos centrais. O Marcação de Linha pode ser compensado ao longo da linha ou linha de deslocamento. Finalmente, *rotação* permite que você mude a orientação do símbolo.

A seguintes configurações são possíveis:

- *Coloração*
- *Espessura da caneta*
- *Deslocamento*
- *Estilo da caneta*
- *Estilo da união*
- *Estilo da capa*
- *Use custom dash pattern*
- *Unidade padrão do traço*
- *Definir propriedades dos dados ...*

12.2.4 Símbolos do polígono

Marcador de Símbolos do Polígono tem também vários tipos símbolo da camada:

- Preenchimento do centroide
- Preenchimento em gradiente
- Preenchimento de linha padrão
- Preenchimento de ponto padrão
- Preenchimento SVG
- Preenchimento Estourar forma
- Preenchimento Simples (padrão)
- Contorno: linha de marcação (o mesmo que marcador de linha)
- Contorno: linha simples (o mesmo que marcador de linha)

A seguintes configurações são possíveis:

- *Cores para a borda e o preenchimento.*
- *Estilo de preenchimento*
- *Estilo de borda*
- *Largura da borda*
- *Deslocamento X,Y*
- *Definir propriedades dos dados ...*

Usando a caixa de combinação de cores, você pode arrastar e soltar o botão para uma cor do botão, copiar e colar a cor, escolher a cor de algum lugar, escolher uma cor da paleta de cores ou usadas recentemente ou padrão. A caixa de combinação permite que você preencha o feiçõ com transparência. Você também pode simplesmente cliquem no botão para abrir o diálogo paleta. Note que você pode importar a cor de algum software externo como o GIMP.

‘Preenchimento Gradiente’ *Tipo da camada de Símbolo* permite selecionar entre as configurações | radiobuttonon | : guilabel:Duas cores e | radiobuttonoff | : guilabel: Rampa de cores. Você pode usar a | caixa | : guilabel: centroide da feiçõ como: guilabel:Ponto de Referência. Todos os preenchimentos ‘Preenchimento Gradiente’ *Tipo da camada Símbolo* também está disponível através do menu *Símbolo* da Renderização categorizadas e graduadas e

através do menu *Propriedades da Regra* do renderizador baseada em regras. Outra possibilidade é a de escolher um ‘preenchimento forma estourada’, que é um preenchimento gradiente tamponado, onde um gradiente é desenhado a partir do limite de um polígono em direção ao centro do polígono. Parâmetros configuráveis incluem distância do limite de sombra, uso de rampas de cor ou simples dois gradientes de cor, indefinição opcional do preenchimento e deslocamentos.

É possível apenas desenhar bordas do polígonos dentro do polígono. Usando ‘Contorno: linha simples’ selecionando | caixa !: guilabel: *Desenhar linha só dentro polígono*.

12.2.5 Rampa de cores

Você pode criar uma rampa de cores personalizada escolha: menuselection: *Nova rampa de cor ...* a partir do menu suspenso *Cor de degradê*. Uma caixa de diálogo pedirá o tipo da cor de degradê: Gradiente, Aleatório, Compositor de cores, ou cpt-city. Os três primeiros têm opções para o número de etapas e / ou múltiplas paragens na rampa de cores. Você pode usar a opção | caixa !: guilabel: *Inverter* quanto classificar os dados com uma rampa de cores. Veja *figure_symbology_3* para um exemplo de rampa de cores personalizada e *figure_symbology_3a* para o diálogo cpt-city.

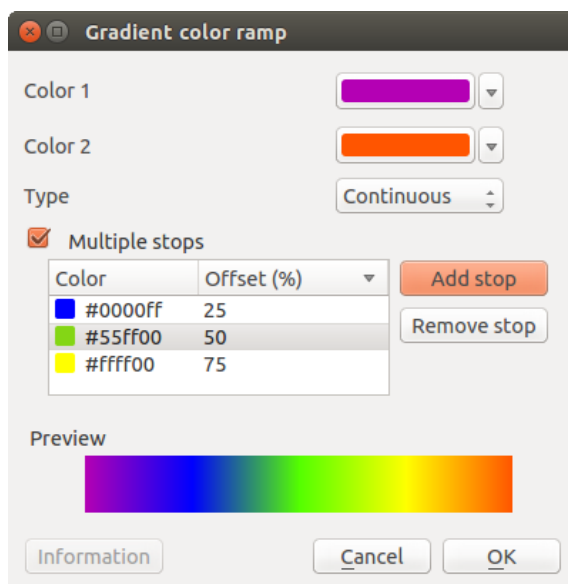



Figura 12.7: Exemplo de cor gradiente de rampa personalizado com paradas múltipla 

A opção cpt-city abrirá um novo diálogo com centenas de temas incluídos ‘fora da caixa’

12.3 Janela de Propriedades de Vetor

A: guilabel: ‘Propriedades da camada diálogo para uma camada vetorial fornece informações sobre a camada, as configurações de simbologia e opções de rotulagem. Se sua camada de vetor foi carregado a partir de um armazenamento de dados PostgreSQL / PostGIS, você também pode alterar o SQL subjacente para a camada invocando o: guilabel: ‘diálogo Query Builder no: guilabel: guia Geral. Para acessar o: guilabel: *Propriedades da camada* de diálogo, clique duas vezes em uma camada na legenda ou clique com o botão direito sobre a camada e selecione: menuselection: ‘Propriedades’ no menu pop-up.

12.3.1 Menu Estilo

O menu Estilo lhe oferece uma ferramenta completa para renderização e simbolizar seus dados vetoriais. Você pode usar: menuselection: *Renderização de Camada* -> ferramentas que são comuns a todos os dados do vetor,

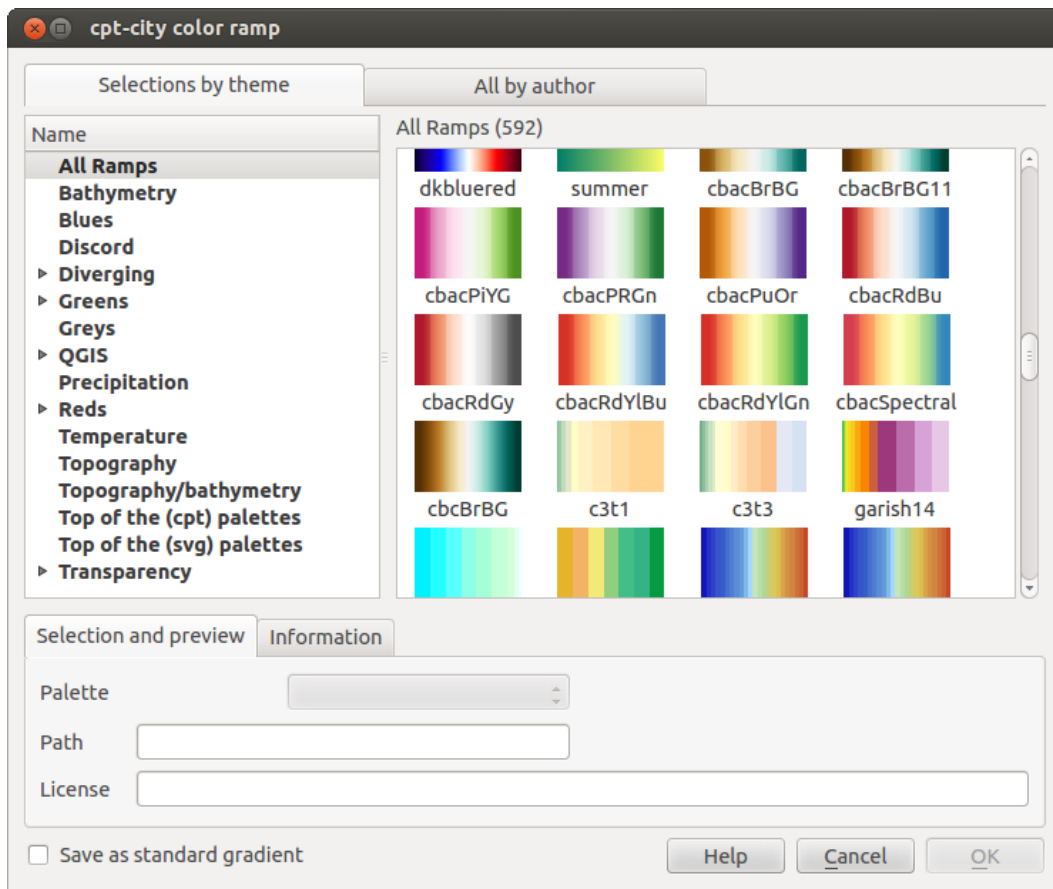


Figura 12.8: diálogo cpt-city com centenas de rampas de cores 🐧

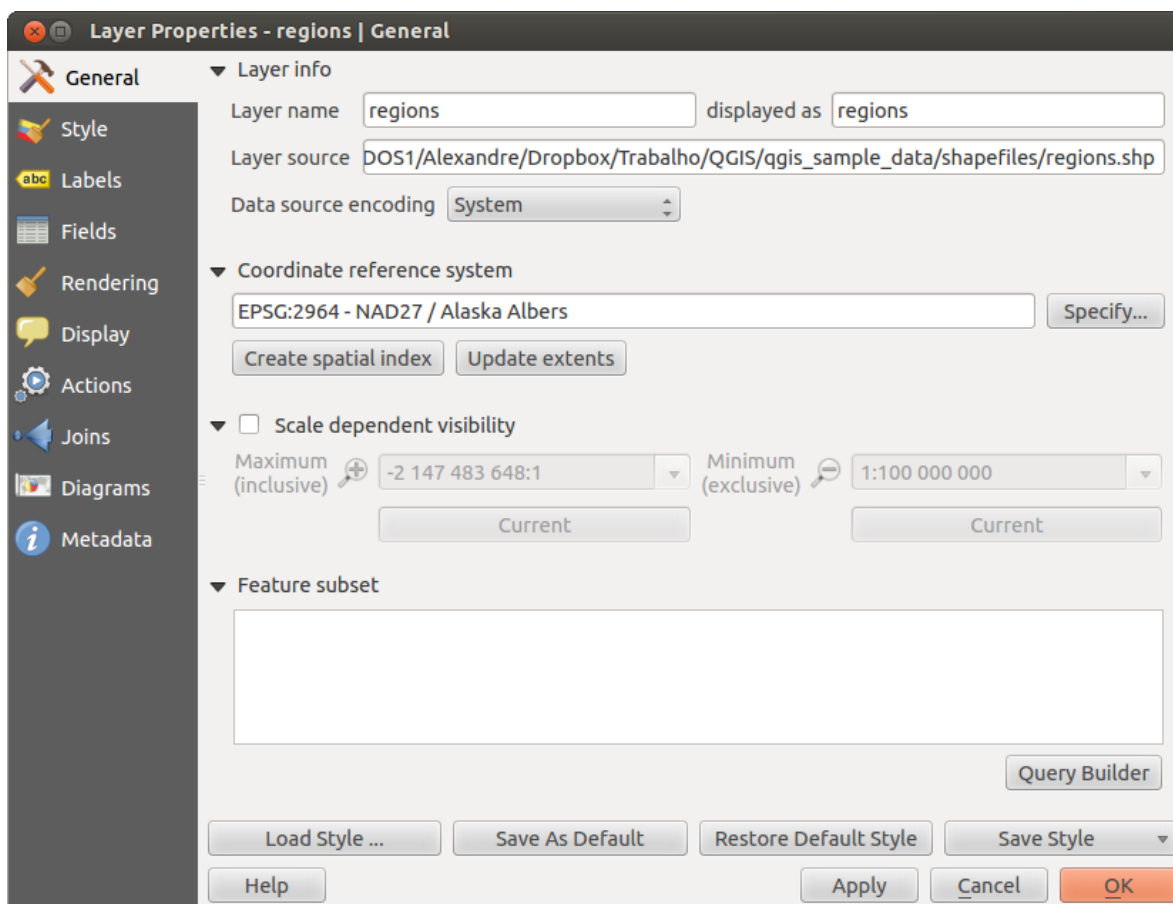



Figura 12.9: Janela de Propriedades do Vetor 

bem como ferramentas de simbolização especiais que foram projetados para os diferentes tipos de dados vetoriais.

Renderização

O renderizador é responsável por desenhar um feição junto com o símbolo correto. Existem quatro tipos de renderizadores de: símbolos únicos, categorizados, graduados e baseado em regras. Não há nenhum representante de cor contínua, porque é na verdade apenas um caso especial de renderização graduada. Os renderizadores categorizados e graduados podem ser criados, especificando um símbolo e uma rampa de cores - que vai definir as cores de símbolos de forma adequada. Para camadas de ponto, há um renderizador de deslocamento de ponto disponível. Para cada tipo de dados (pontos, linhas e polígonos), tipos de símbolo de camada vetor estão disponíveis. Dependendo do processador escolhido, o menu *Estilo* fornece diferentes seções adicionais. No canto inferior direito da janela de simbologia, existe o botão **[Símbolo]**, que dá acesso ao Gerenciador de estilo (veja: ref: *vector_style_manager*). O Gerenciador de estilo permite que você edite e remova símbolos existentes e adicione novos.

Depois de ter feito as alterações necessárias, o símbolo pode ser adicionado à lista de estilos de símbolos atual (usando **[Símbolo]**  *Salvar na biblioteca de símbolos*), e então ele pode ser facilmente utilizado no futuro. Além disso, você pode usar o botão **[Salvar Estilo]** **SelectString** para salvar o símbolo como um arquivo de estilo de camada (.qml) ou arquivo SLD (.sld) do lqgl. SLDs podem ser exportados a partir de qualquer tipo de processador - símbolos únicos, categorizados, graduados ou baseado em regras - mas quando a importação de um SLD, é criado um único símbolo ou renderizador baseado em regras. Isso significa que os estilos categorizados ou graduados são convertidos para à base da regra. Se você quiser preservar esses representantes, você tem que ficar com o formato QML. Por outro lado, pode ser muito útil, por vezes, para ter esta maneira fácil de converter estilos para baseado em regras.

Se você alterar o tipo de processador ao definir o estilo de uma camada de vetor as configurações feitas para o símbolo serão mantidas. Esteja ciente de que este procedimento só funciona para uma mudança. Se você repetir a alteração do tipo de renderizador as configurações para o símbolo irão se perder.

Se a fonte de dados da camada é um banco de dados (PostGIS ou SpatiaLite por exemplo), você pode salvar o seu estilo de camada dentro de uma tabela do banco de dados. Apenas Clique na caixa combinada *Salvar estilo* e escolha o item **Salvar na base de dados** em seguida, preencha a caixa de diálogo para definir um nome de estilo, adicione uma descrição, um arquivo ui e se o estilo é um estilo padrão. Ao colocar uma camada de banco de dados, se um estilo já existe para essa camada, QGIS carregará a camada e seu estilo. Você pode adicionar vários estilos no banco de dados. Apenas um será o estilo padrão de qualquer maneira.

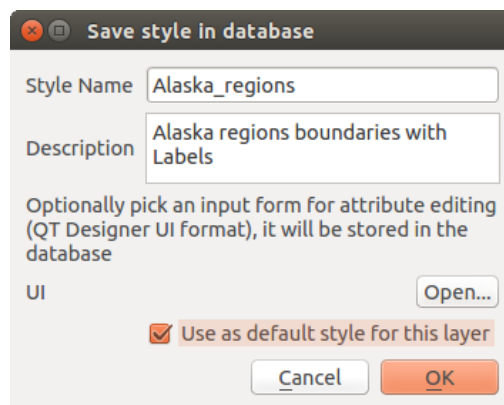


Figura 12.10: Diálogo Salvar estilo na base de dados 

Dica: Seleção e mudança de símbolos múltiplos

A simbologia permite que você selecione vários símbolos e clique no direito para mudar de cor, transparência, tamanho ou largura das entradas selecionadas.

Renderizando Símbolo Individual

O Renderizador de Único Símbolo é usado para processar todos as características da camada usando um único símbolo definido pelo usuário. As propriedades, que podem ser ajustados no: *guilabel: Menu estilo*, dependem em parte do tipo de camada, mas todos os tipos podem compartilhar essa estrutura de diálogo. Na parte superior esquerda do menu, há uma pré-visualização do símbolo atual para ser processado. Na parte direita do menu, há uma lista de símbolos já definidos para o estilo atual, preparado para ser utilizado, selecionando-os na lista. O símbolo atual pode ser modificada usando o menu do lado direito. Se você clicar no primeiro nível no diálogo *Símbolo das camadas* no lado esquerdo, é possível definir parâmetros básicos como: *guilabel:tamanho*, *transparência*, *cor* e *Rotação*. Aqui, as camadas são unidas em conjunto.

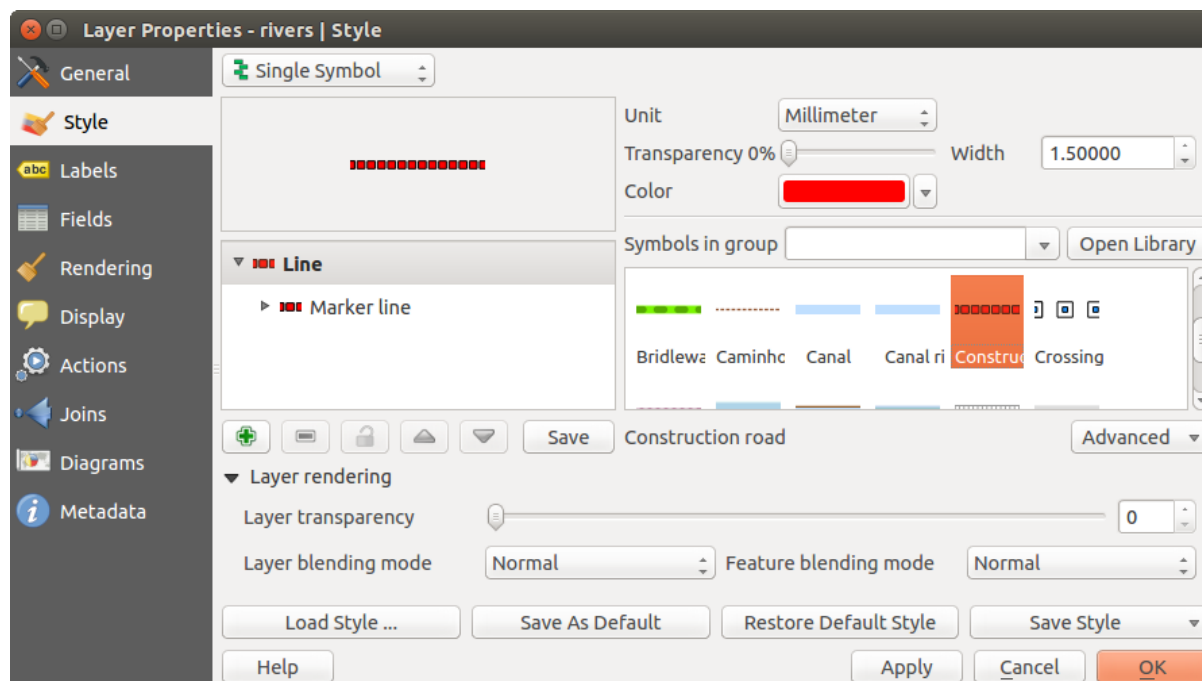


Figura 12.11: Propriedades da linha de símbolos simples 🐧

Renderizador Categorizado

O Renderizador Baseado em Regras é usado para processar todos os recursos a partir de uma camada, usando o único símbolo definido pelo usuário cuja cor reflete o valor do atributo de uma feição selecionada. A: *guilabel: menu Estilo* permite que você selecione:

- O atributo (usando a caixa de listagem Coluna ou a função *☰* :*guilabel: Definir expressão da coluna*, veja *Expressões*)
- Símbolo (usando a aba Símbolo)
- As cores (usando a rampa de cores caixa lista)

Em seguida, clique no botão **Classificar** para criar classes de valor distinto na coluna atributo. Cada classe pode ser desativada desmarcando a caixa de seleção à esquerda do nome da classe.

Você pode mudar símbolo, valor e / ou rótulo do clique, clicando apenas duas vezes no item que deseja alterar.

Clique com direito mostrará um menu de contexto com **Copiar/Colar**, **Mude a cor**, **Mudar transparência**, **Mudar unidade de saída**, **Mudar largura do símbolo**.

O botão **** [Avançado] **** no canto inferior direito da caixa de diálogo permite que você defina os campos que contenham informações de rotação e escala de tamanho. Para maior comodidade, o centro do menu lista os valores de todos os atributos selecionados no momento em conjunto, incluindo os símbolos que serão renderizados.

O exemplo da [figure_symbology_2](#) mostra a categoria usando janela de renderização para camada rios dos dados de exemplo do QGIS.

Renderização Graduada

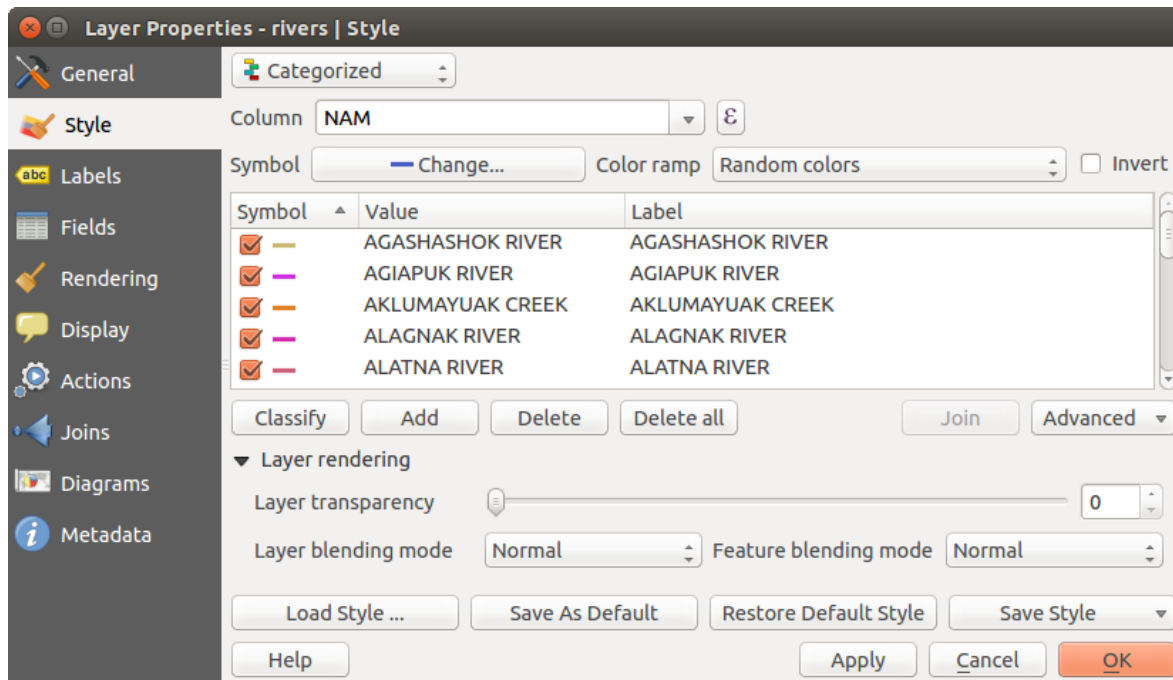


Figura 12.12: Opções de simbologia categorizada 🐧

O Renderizador Baseado em Regras é usado para processar todos os recursos a partir de uma camada, usando um único símbolo definido pelo usuário cuja cor reflete o valor do atributo do recurso selecionado para uma classe.

Como o Renderizador Categorizado, o Renderizador Graduado permite definir a rotação e escala de tamanho de colunas especificadas.

Além disso, análoga à Edição Categorizada, a guia :guilabel: *Estilo* permite que você selecione:

- A coluna atributo (usando a caixa de listagem Coluna ou o **E...** *Definir expressão da função*, veja: ref: ‘ capítulo vector_expressions ‘)
- O símbolo (usando o botão Propriedades do Símbolo)
- As cores (usando a lista de rampa de cores)

Além disso, você pode especificar o número de classes e também o modo de classificação das feições dentro das classes (usando a lista Modo). Os modos disponíveis são:

- Intervalo Igual: cada classe tem o mesmo tamanho (por exemplo, valores de 0 a 16 e com 4 classes, cada classe tem um tamanho 4);
- Quantil: cada classe terá o mesmo número de elementos no interior (a idéia de uma caixa de plotagem);
- Quebras naturais (Jenks): a variância dentro de cada classe é mínima, enquanto a variação entre as classes é máxima;
- Desvio Padrão: classes são construídas de acordo com os valores do desvio padrão;
- Quebras perfeitas: o mesmo de quebras naturais, mas o número extremos de cada classe são inteiros.

A caixa de listagem na parte central da lista do menu *Estilo* as classes juntamente com as suas faixas, etiquetas e símbolos que serão renderizados.

Clique no botão **Classifique** para criar classes, utilizando o modo escolhido. Cada classe pode ser desativada desmarcando a caixa de seleção à esquerda do nome da classe.

Você pode mudar símbolo, valor e / ou rótulo do clique, clicando apenas duas vezes no item que deseja alterar.

Clique com direito mostrará um menu de contexto com **Copiar/Colar**, **Mude a cor**, **Mudar transparência**, **Mudar unidade de saída**, **Mudar largura do símbolo**.

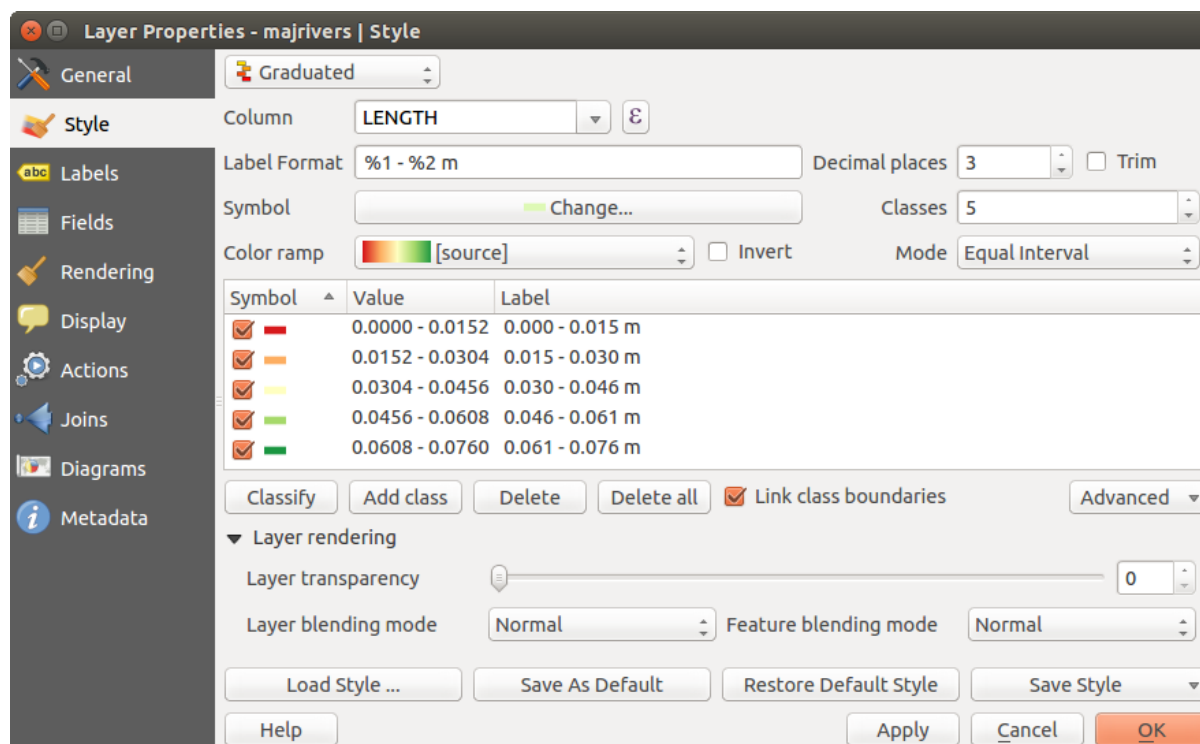


Figura 12.13: Opções de simbologia graduada 🐧

O exemplo da in [figure_symbology_4](#) mostra a janela de renderização graduada para camada rios dos dados de amostra do QGIS.


Dica: Mapas Temáticos usando uma expressão

Mapas temáticos categorizados e graduados podem agora ser criados usando o resultado de uma expressão. No diálogo para propriedades de camadas vetoriais, o atributo selecionador foi aumentada com a `mActionMIcon-ExpressionEditorOpen` !: `guiabel: function Definir expressão da coluna`. Então agora você não precisa mais escrever o atributo de classificação para uma nova coluna na sua tabela de atributos, se você quiser que o atributo de classificação para ser um composto de vários campos, ou uma fórmula de algum tipo.

Renderizador baseado em regras

O renderizador baseado em regras é utilizado para renderizar todas as feições de uma camada utilizando símbolos baseados em regras, cujas cores refletem a classificação do atributo de uma feição selecionada para uma classe. As regras são baseadas em declarações SQL. A janela permite agrupar as regras através de filtros e escala e você pode decidir se quer habilitar níveis de símbolos ou usar somente a primeira regra correspondente.

O exemplo da [figure_symbology_5](#) mostra a janela de renderização baseada em regra para a camada rios dos dados de amostra do QGIS.

Para criar uma regra, ativar uma linha existente, clicando duas vezes sobre ela, ou clique em '+' e clique sobre a nova regra. No diálogo *Propriedade da Regra*, você pode definir um rótulo para a regra. Pressione o botão  para abrir o construtor de expressão de texto. Na **Lista de funções**, clique em: `guiabel:Campos e Valores` para ver todos os atributos da tabela de atributos a serem pesquisados. Para adicionar um atributo na calculadora de campo no campo **Expressão**, clique duas vezes em seu nome na lista *Campos e Valores*. Geralmente, você pode usar nos campos diversos, valores e funções para construir a expressão de cálculo, ou você pode simplesmente digitá-las na caixa (veja: ref: *vector_expressions*). Você pode criar uma nova regra, copiando e colando uma regra existente com o botão direito do mouse. Você também pode usar a regra de 'ELSE' que será executada se nenhuma das outras regras desse nível forem usadas. Desde QGIS 2.6 a etiqueta para as regras aparece em uma pseudo árvore na legenda do mapa. Basta fazer duplo clique nas regras da legenda do mapa e do menu Estilo que as propriedades da camada aparecerão mostrando a regra de que é o fundo para o símbolo na pseudo árvore.

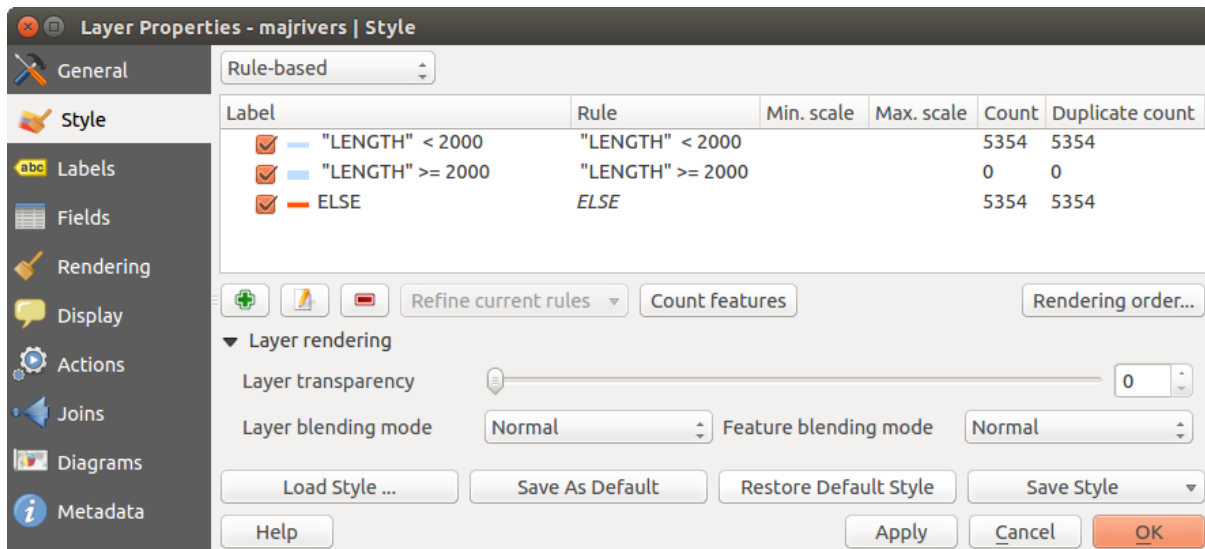



Figura 12.14: Opções simbologia baseada em regra 

Deslocamento de ponto

O Renderizador de Deslocamento de Ponto oferece a visualização de todas as feições de uma camada de pontos, mesmo se eles tem a mesma localização. Para para isso, os símbolos dos pontos são postos em um círculo de deslocamento ao redor de um símbolo.


Dica: Exportar simbologia vetorial



Você tem a opção de exportar simbologia de um vetor QGIS do Google *.kml, *.dxf e MapInfo* em arquivos de tabulação. Basta abrir o menu do botão direito do mouse da camada e clique em *Salvar seleção como* → para especificar o nome do arquivo de saída e seu formato. Na caixa de diálogo, use o menu de *Exportar Simbologia* para salvar a simbologia quer como *Simbologia da feição* → ' ou como *:menuselection:'Simbologia para camada símbolo* →. Se você já usou camadas de símbolos, recomenda-se usar a segunda configuração.

Polígono invertido




Renderizador de polígono invertido permite ao usuário definir um símbolo para preencher fora da camada do polígono. Como antes, você pode selecionar um subrenderizador. Estes subrenderizadores são os mesmos que para os principais representantes.

Seletor de Cores

Independentemente do tipo de estilo que será usado, o diálogo *selecionar cor* mostrará quando você clica para escolher uma cor - ou a borda ou cor de preenchimento. Este diálogo tem quatro guias diferentes que permitem que você selecionar cores por `mIconColorBox` **!:** **sup: 'Rampa de cores', `mIconColorWheel`** Roda de cores, 

Amostra de cor  OU  Obter cor

Seja qual for o método utilizado, a cor selecionada é sempre descrita através de seletores de cores para valores HSV (Hue, Saturation, Value) e RGB (Red, Green, Blue). Há também uma deslizante `:guilabel: opacidade` para definir o nível de transparência. Na parte inferior esquerda da janela, você pode ver uma comparação entre o *Atual* e o *Nova cor* onde está atualmente a seleção e na parte inferior direita, você tem a opção de adicionar a cor que você acabou formar com o botão para o compartimento de cores.

Com  Rampa de cor ou com  Roda de cor, você pode navegar em todas as possíveis combinações de cores. Embora existem outras possibilidades. Usando *Amostra de cor*  você pode escolher a partir de uma lista pré-selecionada. Esta lista selecionada é preenchida com um dos três métodos: *Cores recentes*, `:guilabel:'Cores padrão` ou *Cores do Projeto*

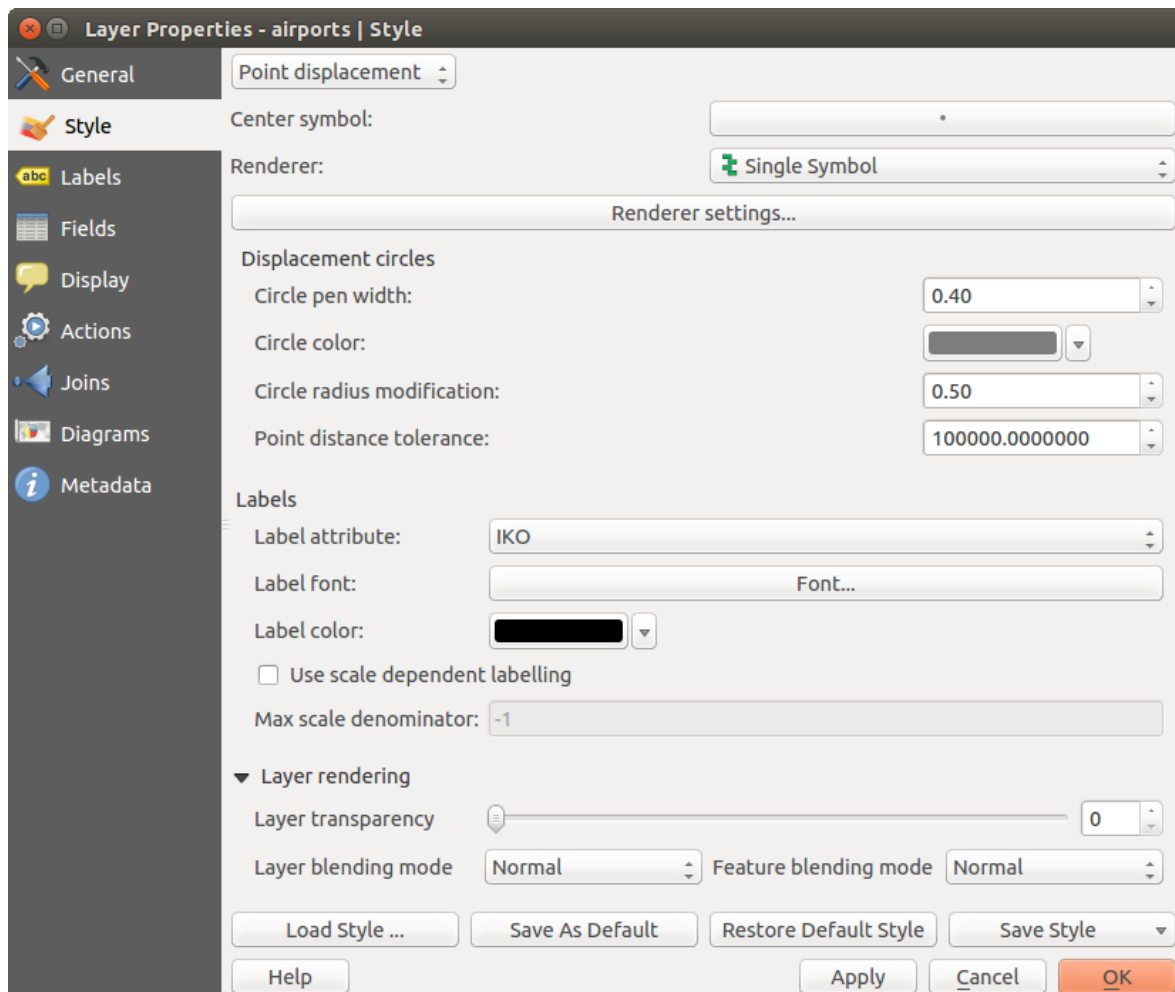



Figura 12.15: Diálogo deslocamento dos pontos 

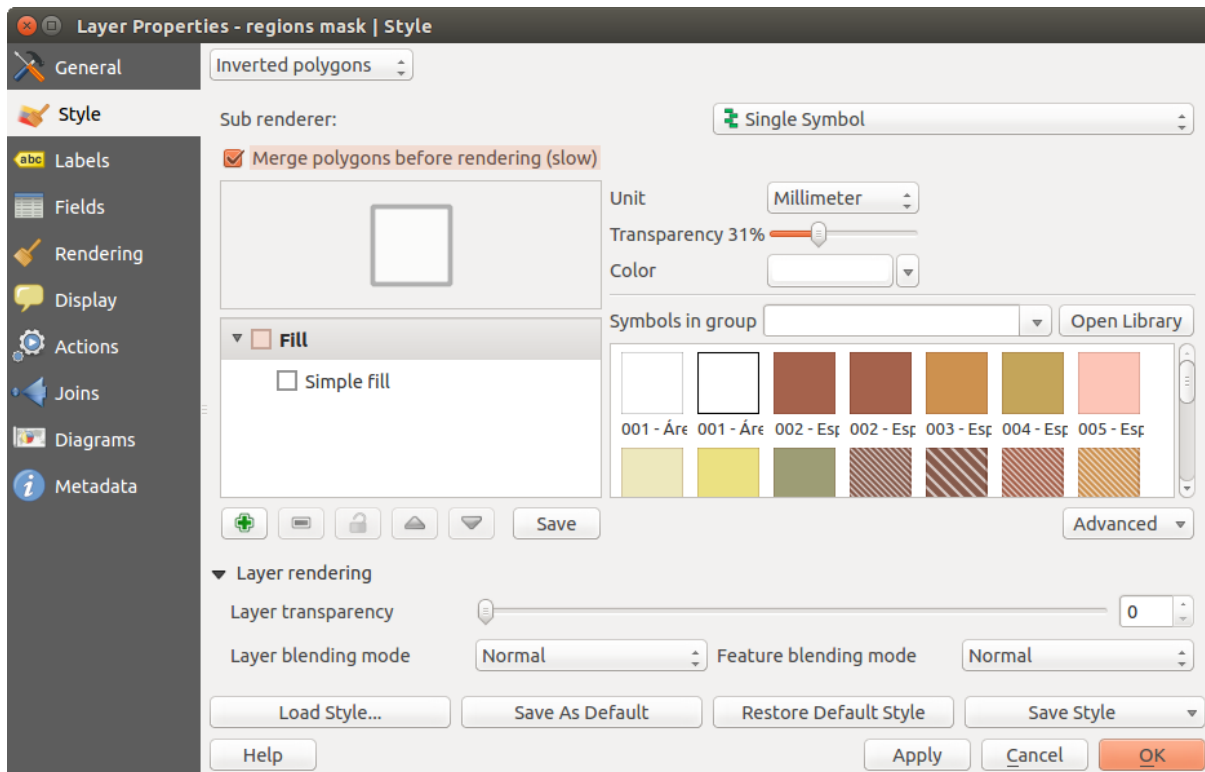


Figura 12.16: Diálogo Inverter Polígono 🐧

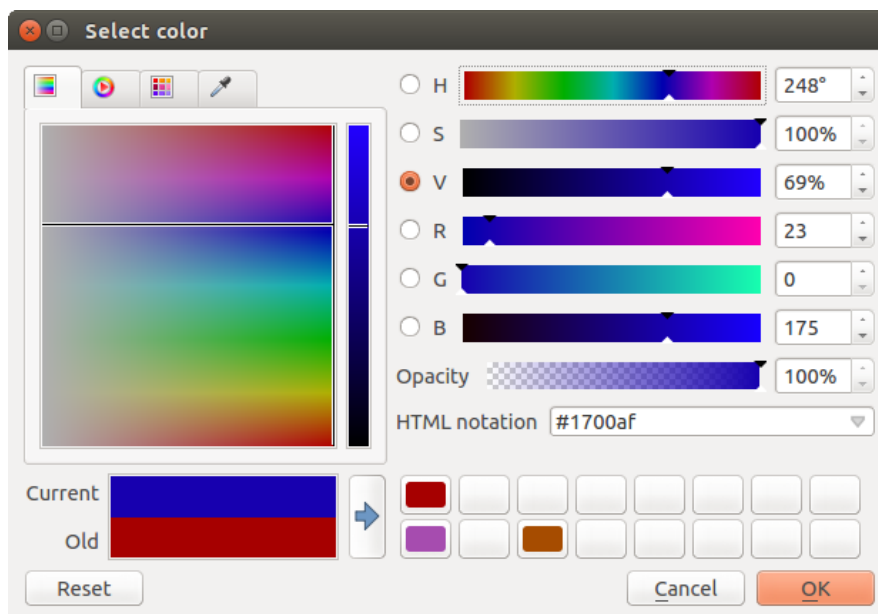


Figura 12.17: Guia Rampa de seleção de cores 🐧

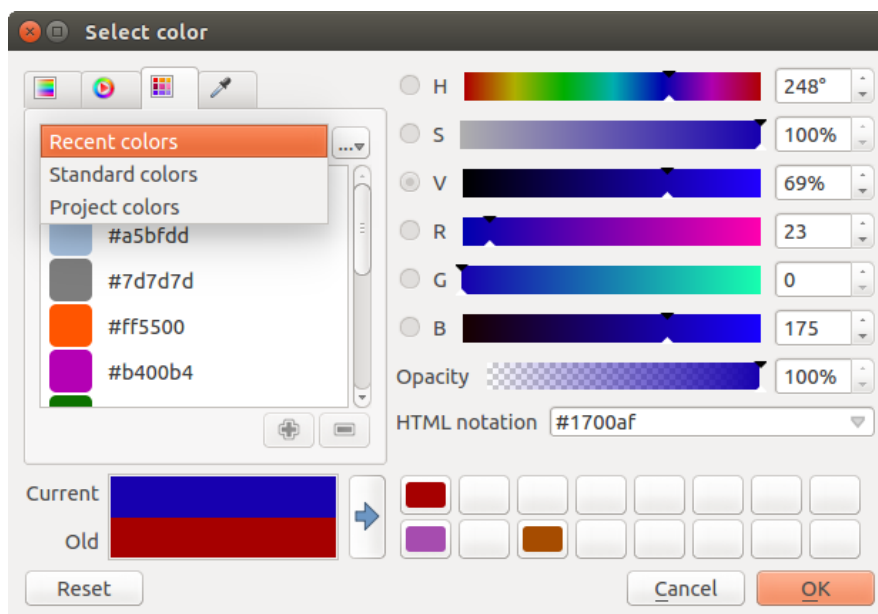


Figura 12.18: Guia swatcher de seleção de cores 🐧

Outra opção é usar o `qmlColorPicker`: `sup: seletor de cor` que lhe permite experimentar uma cor sob o ponteiro do mouse em qualquer parte do QGIS ou até mesmo de outro aplicativo, pressionando a barra de espaço. Por favor, note que o seletor de cores é dependente de sistema operacional e não é suportada atualmente pelo OSX.

Dica: Seletor de Cor rápido + copia / cola cores

Você pode rapidamente escolher pelas *cores recentes*, a partir das *Cores padrão* ou simplesmente *copiar* ou *colar* uma cor clicando na seta arrastar para baixo, que segue uma cor atual na caixa.

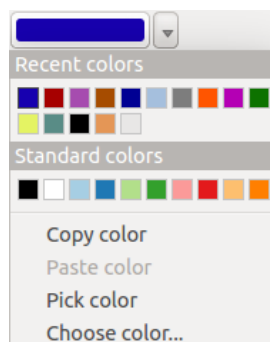


Figura 12.19: Menu seletor rápido de cores 🐧

Renderização da camada

- `guiabel: Transparência da camada` | `deslizante` `!:` Você pode fazer a camada subjacente na tela do mapa visível com esta ferramenta. Use o controle deslizante para se adaptar a visibilidade de sua camada de vetor para suas necessidades. Você também pode fazer uma definição precisa do percentual de visibilidade no menu ao lado do controle deslizante.
- `guiabel: Modo de mesclagem da camada` e `guiabel: Modo de mesclagem de recursos`: você pode conseguir efeitos especiais de renderização com essas ferramentas que você fazia antes só com programas gráficos. Os pixels da sua sobreposição e camadas sobpostas são misturados através das configurações descritas abaixo.

- Normal: Este é o modo de mistura padrão, que usa o canal alfa do pixel superior para se misturar com o pixel abaixo dela. As cores não se misturam.
- Clarear: Seleciona o máximo de cada um dos componentes do primeiro plano e pixels de fundo. Esteja ciente de que os resultados tendem a ser irregulares e duros.
- Tela: Pixels claros da fonte são pintados sobre o destino, enquanto pixels escuros não. Este modo é muito útil para misturar a textura de uma camada com outra (por exemplo, você pode usar um monte sombreado para textura de outra camada).
- Subexposição: A subexposição irá clarear e saturar os pixels subjacentes baseados na luminosidade do pixel superior. A maior claridade do pixel superior causa o aumento da saturação e brilho dos pixels subjacentes. Isto funciona melhor nos pixels superiores que não brilham muito, caso contrário o efeito é muito extremo.
- Adição: Este modo de renderização simplesmente adiciona os valores dos pixels de uma camada noutra. Nos casos que os valores são acima de 1 (no caso do RGB), o branco é exibido. Este modo é adequado para destacar elementos.
- Escurecer: Isso cria um pixel resultante que mantém os menores componentes de primeiro plano e dos pixels de fundo. Como clarear, os resultados tendem a ser irregulares e permanentes.
- Multiplicar: Aqui, os números para cada pixel da camada superior são multiplicados com os números para o pixel correspondente da camada inferior. Os resultados são imagens mais escuras.
- Queimar: As cores escuras da camada superior tornam mais escuro as camadas subjacentes. Pode ser usado para ajustar e colorizar camadas subjacentes.
- Sobreposição: Este modo combina os modos múltiplos e tela de mistura. Na imagem resultante, peças leves tornam-se mais leve e partes escuras ficam mais escuras.
- Luz suave: Este é muito semelhante ao sobrepor, mas em vez de usar multiplicar / tela que usa a cor queimar / esquivar. Este modo deve emular brilhar uma luz suave em uma imagem.
- Muita luz: Este modo é muito semelhante ao modo de sobreposição. É suposto simular a projecção de uma luz muito intensa numa imagem.
- Diferença: Diferença subtrai o pixel superior a partir do pixel inferior, ou o contrário, para obter sempre um valor positivo. A mesclagem com preto não produz alterações, como a diferença com todas as cores é zero.
- Subtração: Este modo de renderização simplesmente subtrair os valores do pixel de uma camada à outra. Em caso de valores negativos, o preto é exibido.

12.3.2 Menu Rótulos

A `mActionLabeling` (sup: *Camadas*) aplicação do núcleo fornece rotulagem inteligente para o ponto do vetor, linha e camadas de polígonos, e requer apenas alguns parâmetros. Esta nova aplicação também suporta on-the-fly camadas transformadas. As funções principais da aplicação foram redesenhadas. Em `l qg l`, há um número de outras características que melhoram a rotulagem. Os seguintes menus foram criados para identificar as camadas de vetores:

- Texto
- Formatação
- Buffer
- Pano de fundo
- Sombra
- Localização
- Desenhando

Vamos ver como os novos menus podem ser usados nas várias camadas vetoriais. **Rotulando camadas de pontos**

Iniciar | **qg** | e coloque uma camada de ponto vetor. Ative a camada na legenda e clique no | **mActionLabeling** | sup: ícone *Opções Camada Rotulagem* no | **qg** | menu de barra de ferramentas.

O primeiro passo é ativar o | **checkbox** |: **guilabel**: ‘Rotular esta camada com ‘checkbox e selecione uma coluna de atributo a ser usado para a rotulagem. Clique | **mActionmIconExpressionEditorOpen** | se você quiser definir etiquetas com base em expressões - Veja [labeling_with_expressions](#).

Os passos seguintes descrevem uma rotulagem simples sem usar o: **guilabel**: funções *Substituição de Dados Definido*, que estão situados ao lado dos menus drop-down.

Você pode definir o estilo de texto no: **guilabel**: Menu *Text* (ver [Figure_labels_1](#)). Use o: **guilabel**: opção *Caso Tipo* de influenciar o processamento de texto. Você tem a possibilidade de tornar o texto ‘Todo Maiúscula’, ‘Todo Minúsculas’ ou ‘Maiúscula a primeira letra’. Utilize os modos de mesclagem para criar efeitos conhecidos de programas gráficos (ver [blend_modes](#)).

No: **guilabel**: Menu Formatação “”, você pode definir um personagem para uma quebra de linha nos rótulos com a função ‘Caráter envolto”. Use o | **checkbox** |: **guilabel**: *Números Formatados* opção para formatar os números em uma tabela de atributos. Aqui, decimais pode ser inseridos. Se você ativar esta opção, três casas decimais são inicialmente definidas por padrão.

Para criar uma zona de amortecimento, apenas ative a | **checkbox** |: **guilabel**: ‘Desenhe zona de amortecimento de texto ‘checkbox no: **guilabel**: Menu ‘buffer’. A cor da zona de amortecimento é variável. Aqui você também pode usar modos de mistura (ver [blend_modes](#)).

Se a caixa de seleção **lcaixal***Cor de preenchimento da borda* for ativada, ela irá interagir com o texto parcialmente transparente e dar resultados mistos de transparência da cor. Desligar o preenchimento de borda corrige essa questão (exceto quando o aspecto interior do traçado da borda cruza com o preenchimento do texto) e também permite que você faça o texto esboçado.

No: **guilabel**: Menu *Fundo*, você pode definir com: **guilabel**: *Tamanho X* e : **guilabel**: ‘Tamanho Y a forma do seu fundo. Use: **guilabel**: ‘Tamanho Tipo para inserir um ‘buffer’ adicional em seu fundo. O tamanho do buffer é definido por padrão aqui. A imagem de fundo então consiste na imagem mais a fundo em: **guilabel**: *Tamanho X* e : **guilabel**: ‘Tamanho Y. Você pode definir um: **guilabel**: ‘Rotação’, onde você pode escolher entre ‘Sync com etiqueta’, ‘Compensação de etiqueta’ e ‘fixo’. Usando ‘Compensação de etiqueta’ e ‘fixo’, você pode girar o plano de fundo. Definir um: **guilabel**: ‘X offset, Y com valores X e Y, e o fundo será deslocado. Ao aplicar: **guilabel**: ‘Raio X, Y ‘, o fundo fica com cantos arredondados. Novamente, é possível misturar o fundo com as camadas subjacentes na tela do mapa usando o: **guilabel**: ‘Blend mode ‘(ver [blend_modes](#)).

Use o: **guilabel**: menu *sombra* para uma: **guilabel**: definida pelo usuário *Drop Shadow*. O desenho do fundo é muito variável. Escolha entre o “componente mais baixo da etiqueta ‘, ‘Texto ‘, ‘Buffer ‘e” Fundo”. A: **guilabel**: *ângulo offset* depende da orientação da etiqueta. Se você escolher a | **caixa** |: **guilabel**: **checkbox** *Utilize mundial sombra*, logo o ponto do ângulo zero é sempre orientada para o norte e não depende da orientação do rótulo. Você pode influenciar o aparecimento da sombra com a: **guilabel**: *borrão raio*. Quanto maior for o número, mais as sombras. A aparência da sombra também pode ser alterada pela escolha de um modo de mistura (ver [blend_modes](#)).

Escolha o: **guilabel**: Menu *Colocação* para a colocação da etiqueta e a prioridade rotulagem. Usando o | **radiobutonon** |: **guilabel**: ‘Deslocamento a partir do ponto da configuração , agora você tem a opção de usar: **guilabel**: ‘Quadrantes’ para colocar a sua marca. Além disso, você pode alterar o ângulo da colocação de etiqueta com o: **guilabel**: ‘ Definição de Rotação’. Assim, um posicionamento em um determinado quadrante, com uma certa rotação é possível.

No: **guilabel**: Menu *Rendering*, você pode definir opções de etiqueta e de recursos. Under: **guilabel**: ‘ Opções de Etiqueta’, você encontra a visibilidade baseada em escala definindo agora. Você pode impedir | **qg** | de prestar apenas rótulos selecionados com o | **checkbox** |: **guilabel**: ‘Mostrar todos os rótulos para esta camada (incluindo colidindo rótulos) *checkbox*. Under: **guilabel**: ‘Opções de Funções , você pode definir se cada parte de uma característica de várias partes devem ser rotulados. É possível definir se o número de recursos a ser rotulado é limitada e | **checkbox** |: **guilabel**: *Desencorajar rótulos de cobertura características*.

Rotulando camadas de linhas

O primeiro passo é ativar o | **checkbox** |: **guilabel**: ‘Rotular esta camada com ‘checkbox e selecione uma coluna de atributo a ser usado para a rotulagem. Clique | **mActionmIconExpressionEditorOpen** | se você quiser definir

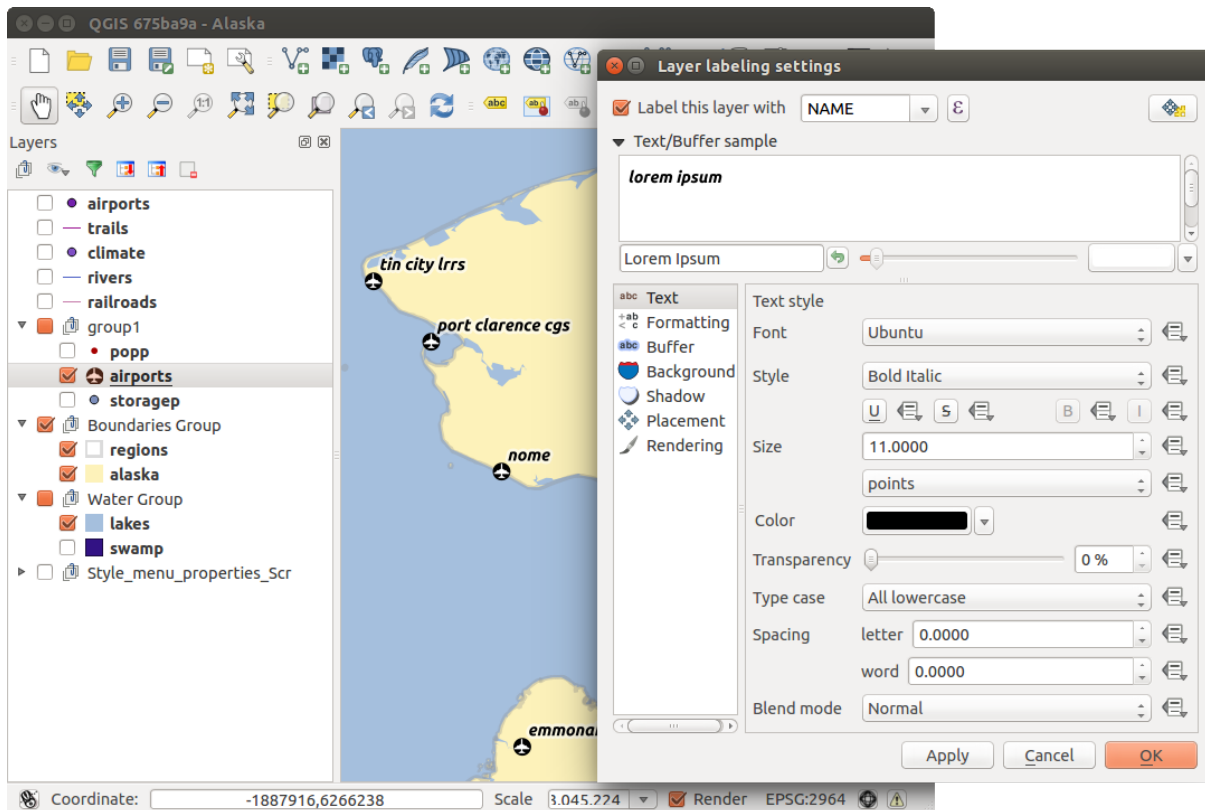



Figura 12.20: Rotulagem inteligente de camadas de pontos vetoral 

etiquetas com base em expressões - Veja [labeling_with_expressions](#).

Depois disso, você pode definir o estilo de texto no: `guiabel: Menu Text`. Aqui, você pode usar as mesmas configurações de camadas de pontos.

Além disso, no Menu `guiabel: Formatação`, as mesmas configurações para as camadas de pontos são possíveis.

O Menu `Buffer` tem as mesmas funções conforme descrito na seção [labeling_point_layers](#).

O Menu `Pano de Fundo` tem as mesmas funções conforme descrito na seção [labeling_point_layers](#).

Além disso, o menu `Sombra` tem as mesmas funções conforme descrito na seção [labeling_point_layers](#).

No: `guiabel: Menu Adaptação`, você encontrará configurações especiais para as camadas de linha. O rótulo pode ser colocado | `radiobuttonon !: guiabel: Paralelo`, | `radiobuttonoff !: guiabel: Curvo` | `radiobuttonoff !: guiabel: Horizontal`. Com a | `radiobuttonon !: guiabel: Paralela` e | `radiobuttonoff !: guiabel: opção Curvo`, você pode definir a posição | `caixa !: guiabel: 'Acima de linha`, | `caixa !: guiabel: On line` e | `caixa !: guiabel: 'Abaixo da linha`'. É possível selecionar várias opções ao mesmo tempo. Nesse caso, | `qg` | irá procurar a posição ideal do rótulo. Lembre-se que aqui você também pode usar a linha de orientação para a posição da rótulo. Além disso, você pode definir um: `guiabel: 'Ângulo máximo entre caracteres curvas` 'ao selecionar o | `radiobuttonoff !: guiabel: opção Curvo` (ver [Figure_labels_2](#)).

Você pode configurar uma distância mínima para repetição de etiquetas. A distância pode ser em mm ou em unidades do mapa.

Alguns configuração posicionamento irá exibir mais opções, por exemplo, localização `Curvada` e `Paralela` irá permitir que o usuário configure a posição da etiqueta (acima, belw ou na linha),: `guiabel: distancia` a partir da linha e para: `guiabel: 'Curvado`, o usuário também pode configurar dentro / fora ângulo máximo entre rótulo curvo.

O: `guiabel: Menu Rendering` tem quase as mesmas entradas como para as camadas de pontos. No: `guiabel: ''` opções de funções, você pode agora: `guiabel: rotulagem Repressão de recursos menores do que o`.

Rotulando camadas de polígonos

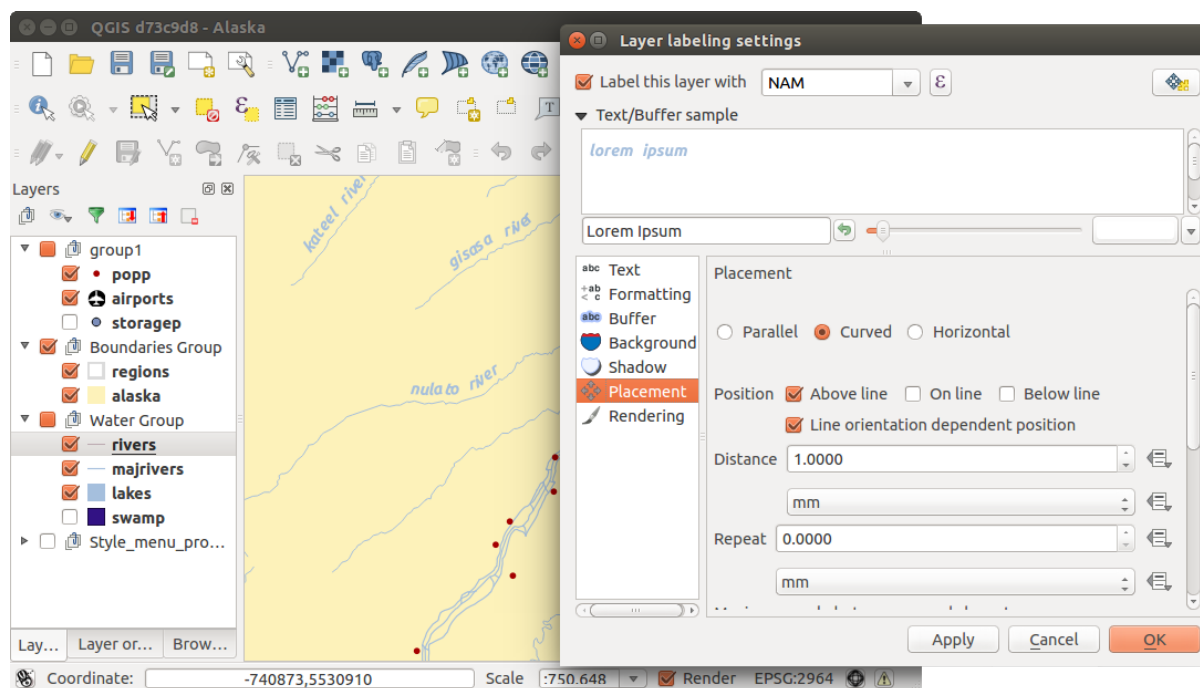


Figura 12.21: Rotulagem inteligente de camadas de linha vetorial 

O primeiro passo é ativar o : guilabel: 'Rotular esta camada com e selecione uma coluna de atributo a ser usado para a rotulagem. Clique se você quiser definir etiquetas com base em expressões - Veja [labeling_with_expressions](#).

No menu *Texto*, define o estilo de texto. A entrada pode ser com camadas de ponto e linha.

O menu *Formatação* permite formatar várias linhas, também semelhantes aos casos de camadas de ponto e linha.

Tal como acontece com as camadas ponto e linha, você pode criar um buffer de texto no Menu *buffer*.

Use o menu *Pano de Fundo* para criar um fundo definido pelo usuário complexo para a camada de polígono. Você pode usar o menu também como com as camadas de ponto e de linha.

As entradas no menu *Sombra* são as mesmas que para as camadas de ponto e linha.

No: guilabel: Menu *Adaptação*, você encontrar configurações especiais para as camadas de polígonos (ver [Figure_Labels_3](#)). : guilabel: 'Deslocamento da centróide', : guilabel: Horizontal (lento), : guilabel: Por volta centroid, : guilabel: e gratuito : guilabel: 'Usando perímetro' são possíveis.

No : guilabel: configuração *Compensação a partir centróide*, você pode especificar se o centróide é da : guilabel: *polígono visível* or : guilabel: *todo polígono*. Isso significa que ou o centróide é usado para o polígono que você pode ver no mapa, ou o centróide é determinada para todo o polígono, não importa se você pode ver toda a funcionalidade no mapa. Você pode colocar a sua marca com os quadrantes aqui, e definir a compensação e rotação. A : guilabel: configuração 'Por volta centróide' torna possível colocar a etiqueta em torno do centróide com uma certa distância. Mais uma vez, você pode definir : guilabel: " ou polígono visível : guilabel: todo polígono " para o centróide. Com a : guilabel: configurações *Usando perímetro*, você pode definir uma posição e uma distância para o rótulo. Para a posição, : guilabel: 'Acima de linha', : guilabel: On line, : guilabel: 'Abaixo da linha' e : guilabel: orientação *Linha posição dependente* são possíveis.

Relacionado com a escolha de local da etiqueta, várias opções aparecerão. Quanto ao ponto de colocação, você pode escolher a distância para o contorno do polígono, repita o rótulo ao redor do perímetro do polígono.

As entradas no menu *Desenhando* são as mesmas que para as camadas de linha. Você também pode usar *Suprimir rotulagem de feição menor do que* no *Opções de feições*. **Definir rótulos baseados nas expressões**

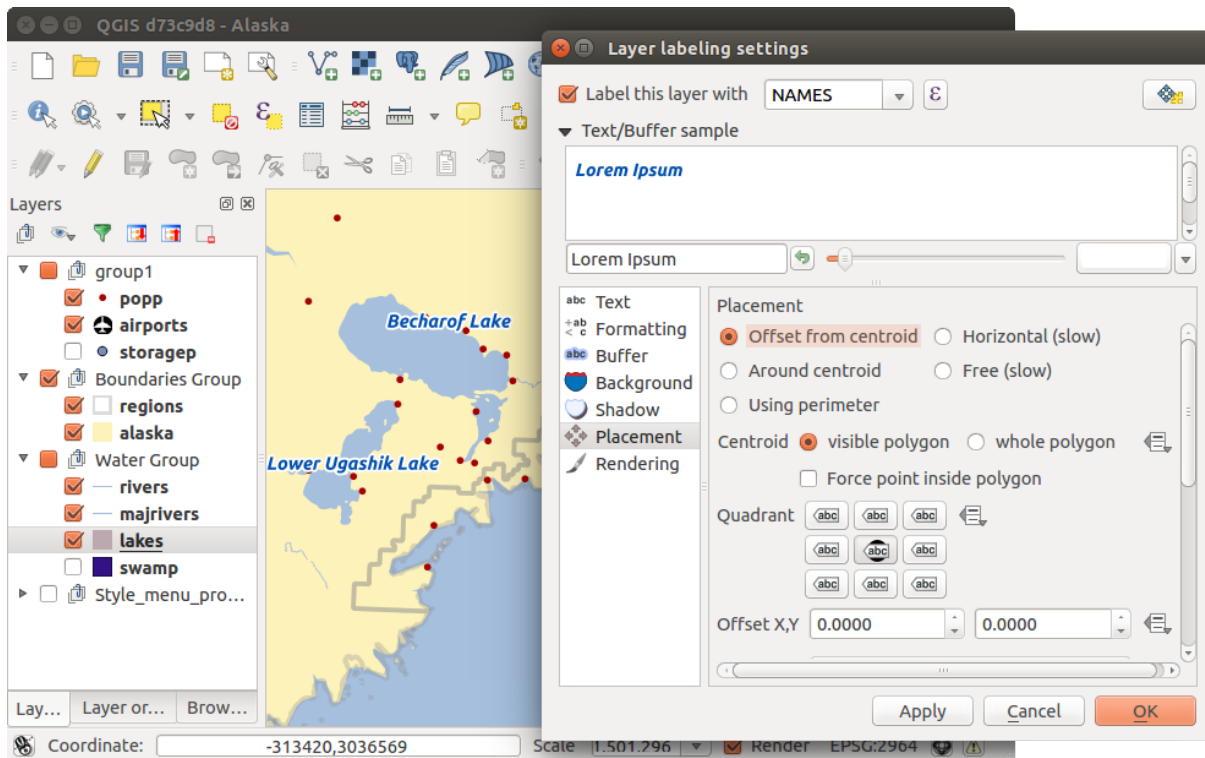



Figura 12.22: Rotulagem inteligente de camadas de polígono vetorial 

QGIS permite usar expressões características do rótulo. Basta clicar no ícone no `mActionLabeling`: sup: Menu ‘Rótulos’ da caixa de diálogo propriedades. Em [figure_labels_4](#) você vê a expressão de exemplo para rotular as regiões do Alasca com nome e tamanho da área, com base no campo ‘NAME_2’, um texto descritivo e a função ‘\$ area () “em combinação com” format_number ()’ para torná-la mais agradável.

Expressão baseado rotulagem é fácil de trabalhar com ele. Tudo que você tem que tomar cuidado é que você precisa combinar todos os elementos (string, campos e funções) com um sinal de concatenação ‘||’ e que os campos um escrito em “aspas duplas” e strings em ‘aspas simples’. Vamos dar uma olhada em alguns exemplos:

```
# label based on two fields 'name' and 'place' with a comma as separator
"name" || ', ' || "place"

-> John Smith, Paris

# label based on two fields 'name' and 'place' separated by comma
'My name is ' || "name" || 'and I live in ' || "place"

-> My name is John Smith and I live in Paris

# label based on two fields 'name' and 'place' with a descriptive text
# and a line break (\n)
'My name is ' || "name" || '\nI live in ' || "place"

-> My name is John Smith
    I live in Paris

# create a multi-line label based on a field and the $area function
# to show the place name and its area size based on unit meter.
'The area of ' || "place" || 'has a size of ' || $area || 'm²'

-> The area of Paris has a size of 105000000 m²
```

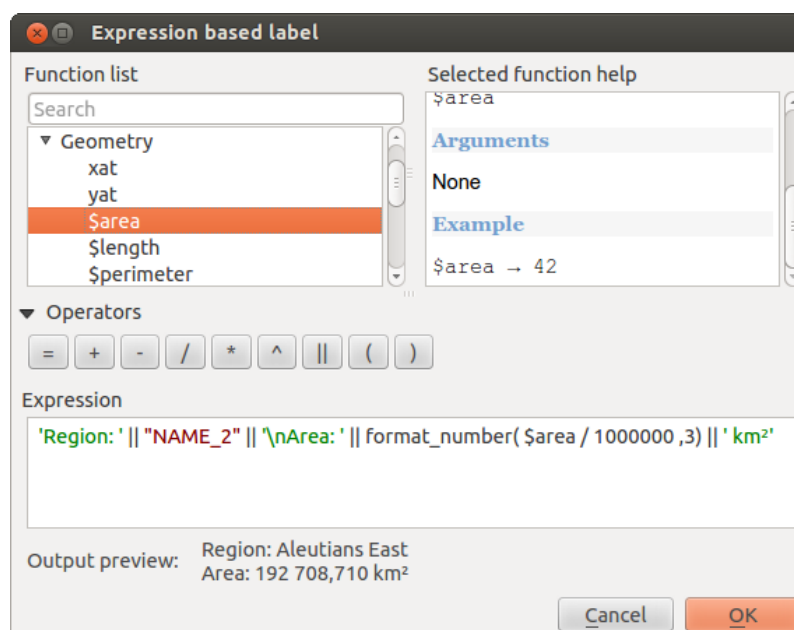


Figura 12.23: Usando expressões para rotulagem 

```
# create a CASE ELSE condition. If the population value in field
# population is <= 50000 it is a town, otherwise a city.
'<code>This place is a ' || CASE WHEN "population <= 50000" THEN 'town' ELSE 'city' END
-> This place is a town
```

Como você pode ver no construtor de expressão, você tem centenas, se as funções disponíveis para criar expressões simples e muito complexas para rotular os seus dados no QGIS. Veja: ref: *capítulo vector_expressions* para mais informações e exemplo em expressões.

Usando substituição definida em dados para rotulagem

Com as funções de substituição definido em dados, as configurações para a rotulagem são substituídas pelas entradas na tabela de atributo. Você pode ativar e desativar a função com o botão direito do mouse. Passe o mouse sobre o símbolo e você verá as informações sobre a substituição definida por dados, incluindo o campo de definição atual. Vamos agora descrever um exemplo usando a função de substituição definido em dados para o `mActionMoveLabel`: sup: function *Mover rótulo* (ver *figure_labels_5*).

1. Importar `lakes.shp` para os dados de amostra QGIS.
2. Clique duas vezes na camada para abrir as Propriedades da Camada. Clique em: `guilabel: ""` e `Labels: guilabel: ""` colocação. Selecione `mIconDataDefine`: `guilabel: 'Deslocamento da centróide'`.
3. Procure o: `guilabel: Os dados definidos` entradas. Clique no `mIconDataDefine` ícone para definir o tipo de campo para o: `guilabel: Coordenar`. Escolha 'xlabel' para X e 'ylabel' Y. Os ícones agora estão destacados em amarelo.
4. Zoom em um lago
5. Vá para a barra de ferramentas e clique no rótulo `mActionMoveLabel` ícone. Agora você pode mudar o rótulo manualmente para outra posição (ver *figure_labels_6*). A nova posição do rótulo é guardado na 'xlabel' e colunas 'ylabel' da tabela de atributos.

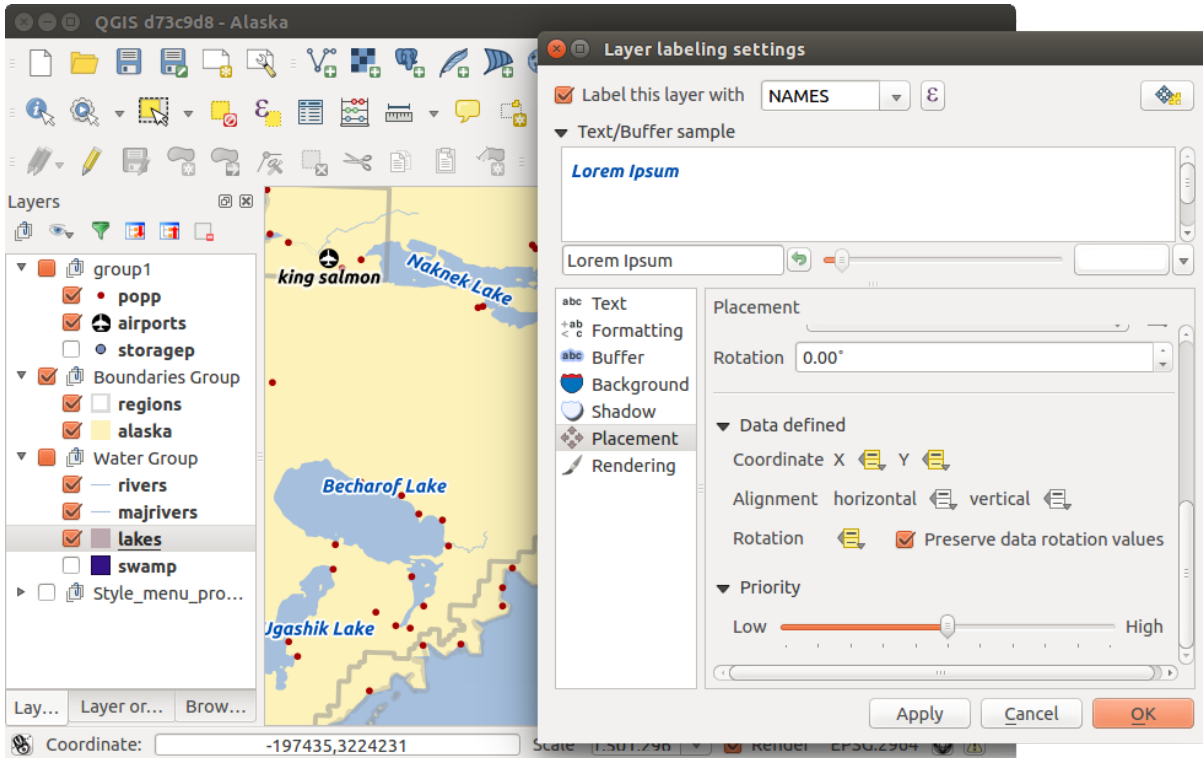


Figura 12.24: Rotulagem de camadas vetorial de polígono com substituição de dados definida 🐧

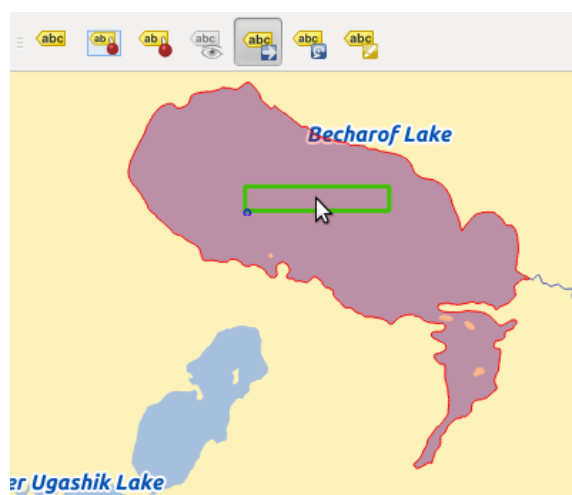






Figura 12.25: Mover rótulos 🐧

12.3.3 Menu campos

 Within the *Fields* menu, the field attributes of the selected dataset can be manipulated. The buttons  and  can be used when the dataset is in  Editing mode.

Edição de Widget

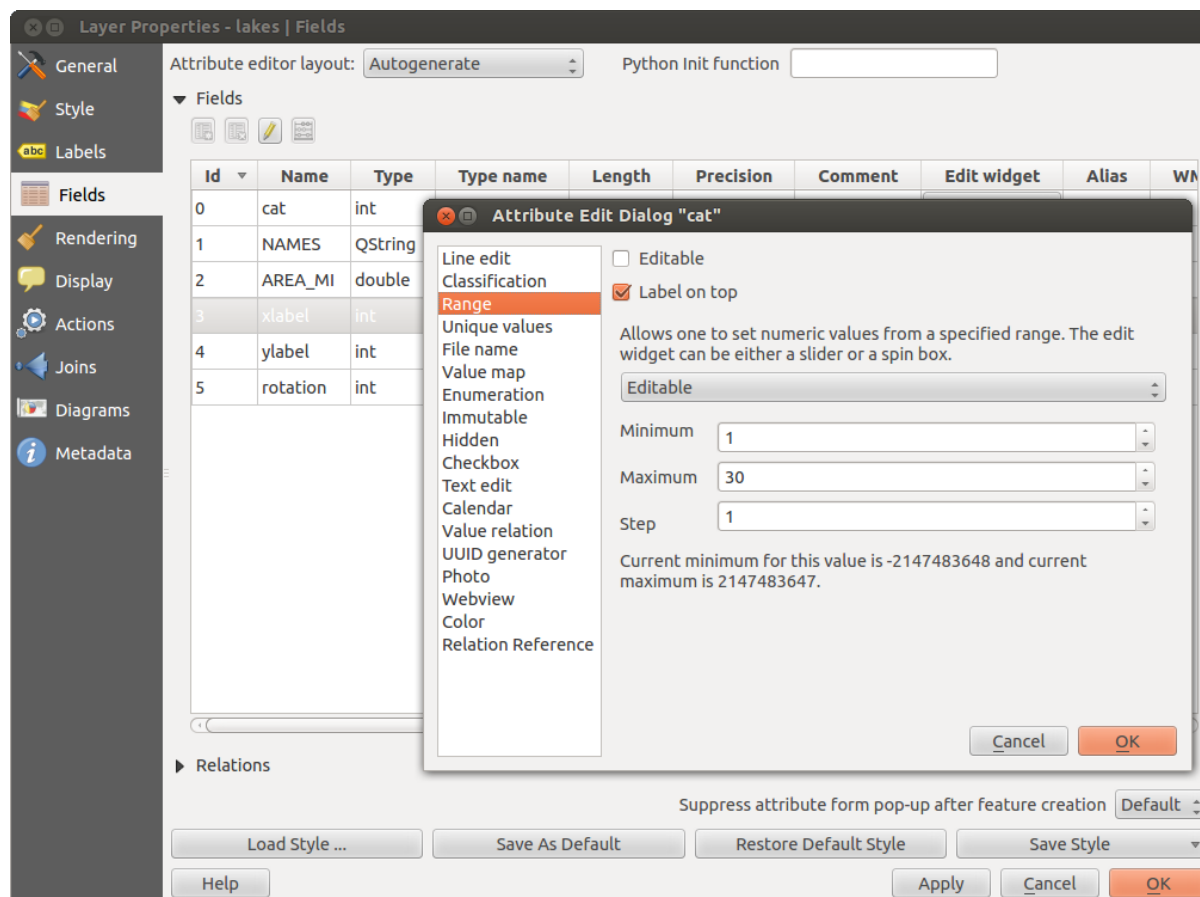



Figura 12.26: De diálogo para selecionar um widget de edição para um atributo de coluna 

No: guilabel: Menu *Campos*, você também encontrará uma coluna ****** ferramenta de edição ******. Esta coluna pode ser utilizada para definir os valores ou variedade de valores que são permitidos para serem adicionados à coluna da tabela de atributos específicos. Se você clicar no **** [edit widget] **** botão, uma janela se abre, onde você pode definir diferentes elementos. Estes elementos são:

- **** Caixa de seleção ****: Exibe uma caixa de seleção e você pode definir qual atributo é adicionado na coluna quando a caixa de seleção é ativada ou não.
- **** Classificação ****: Exibe uma caixa de combinação com os valores utilizados para a classificação, se você tiver escolhido “valor único” como o tipo de legenda no: guilabel: *Estilo* menu da caixa de diálogo propriedade
- **Cor**: Exibe um botão de cor permitindo que o usuário escolha uma cor na janela de diálogo de cor.
- **Data / Hora**: Exibe um campo de linha que pode se abre um widget de calendário para introduzir uma data, uma hora ou ambos. Tipo de coluna deve ser um texto. Você pode selecionar um formato personalizado, pop-up de um calendário, etc.
- **** Enumeração ****: Abre uma caixa de combinação com valores que podem ser usados dentro no tipo de colunas. Atualmente é suportado apenas por provedores de PostgreSQL.
- **** Nome do Arquivo ****: Simplifica a seleção, adicionando uma janela de seleção de arquivos.

- **** Oculto ****: Uma coluna de atributo oculto é invisível. O usuário não é capaz de ver o seu conteúdo.
- **Foto**: Campo que contém o nome do arquivo para a imagem. A largura e altura do campo podem ser definidos.
- **Tamanho**: Permite definir valores numéricos de um tamanho específico. A edição do Widget pode ser um controle deslizante ou uma caixa de rotação.
- ****Relação Referência ****: Este elemento permite que você incorpore a forma característica da camada de referência sobre a forma característica da camada atual. Ver: ref: *vector_relations*.
- **** Edição de texto **** (padrão): Isto abre um campo de edição de texto que permite que o texto simples ou múltiplas linhas a sejam utilizados. Se você escolher várias linhas você também pode escolher o conteúdo html.
- **** Valores únicos ****: O usuário pode selecionar um dos valores já utilizados na tabela de atributos. Se editável é ativado, uma linha de edição é mostrada com suporte para autocompletar, caso contrário, uma caixa de combinação é utilizada.
- **** Gerador UUID ****: Gera um campo UUID (identificador exclusivo universal) somente para leitura, se vazio.
- **** Mapa de Valor ****: uma caixa de combinação com itens pré-definidos. O valor é armazenado no atributo, a descrição é mostrada na caixa de combinação. Você pode definir valores manualmente ou carregá-los a partir de uma camada ou um arquivo CSV.
- **** Relação de Valor ****: Oferece valores de uma tabela relacionada em um combobox. Você pode selecionar camada, coluna de chave e na coluna valor.
- **Webview**: Campo que contém o nome do arquivo para a imagem. A largura e altura do campo podem ser definidos.

Com o **** Attribute editor layout****, agora você pode definir embutido formulários para trabalhos de entrada de dados (ver [figure_fields_2](#)). Escolha 'Arrastar e soltar design' e uma coluna de atributo. Use o ícone | mActionSignPlus | para criar uma categoria que você quer serão mostradas durante a sessão de digitalização (ver [figure_fields_3](#)). O próximo passo será o de atribuir os campos relevantes para a categoria com o | mActionArrowRight | ícone. Você pode criar mais categorias e utilizar os mesmos campos novamente. Ao criar uma nova categoria, |qg | irá inserir uma nova guia para a categoria no built-in formulário.

Outras opções na caixa de diálogo são "Autogenerate" e "Fornecer arquivo ui". "Autogenerate" apenas cria editores para todos os campos e tabula-os. A opção "Fornecer arquivo ui" permite que você use os diálogos complexas feitas com o Qt Designer. Usando um arquivo de interface do usuário permite uma grande liberdade na criação de um diálogo. Para obter informações detalhadas, consulte <http://nathanw.net/2011/09/05/qgis-tips-custom-feature-forms-with-python-logic/>.

QGIS diálogos podem ter uma função Python que é chamado quando a caixa de diálogo é aberta. Use esta função para adicionar lógica extra para seus diálogos. Um exemplo é (no módulo MyForms.py):

```
def open(dialog, layer, feature):
    geom = feature.geometry()
    control = dialog.findChild(QWidget, "My line edit")
```

Referência na Função Python Init como: MyForms.open

MyForms.py deve existir em PYTHONPATH, em .qgis2/python, ou dentro da pasta do projeto.

12.3.4 Menu Geral



Use this menu to make general settings for the vector layer. There are several options available:

Informação da Camada

- Alterar o nome de exibição da camada em: guilabel: *apresentado como*
- Define o: guilabel: *source Camada* da camada de vetor

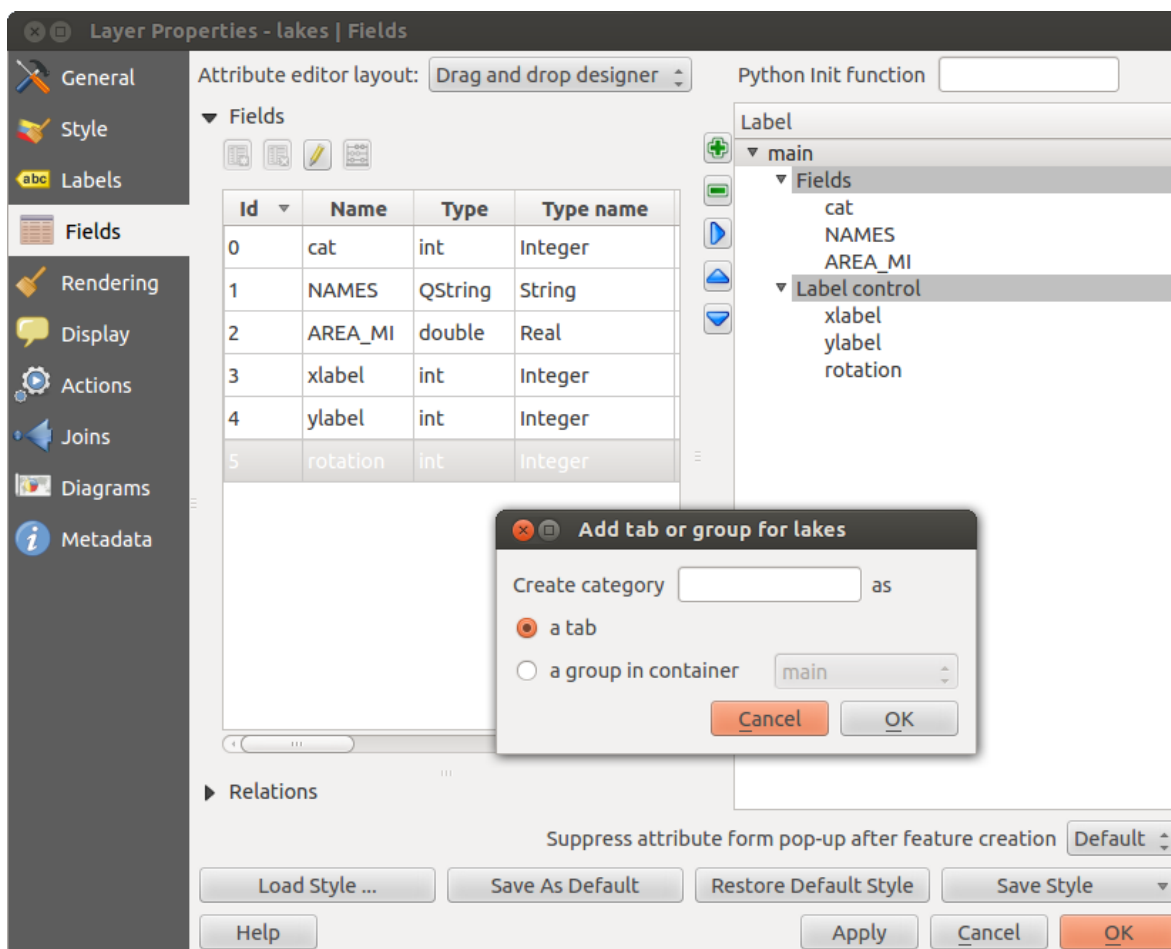


Figura 12.27: Diálogo para criar categorias com o **** |Editor Disposição Atributo ****

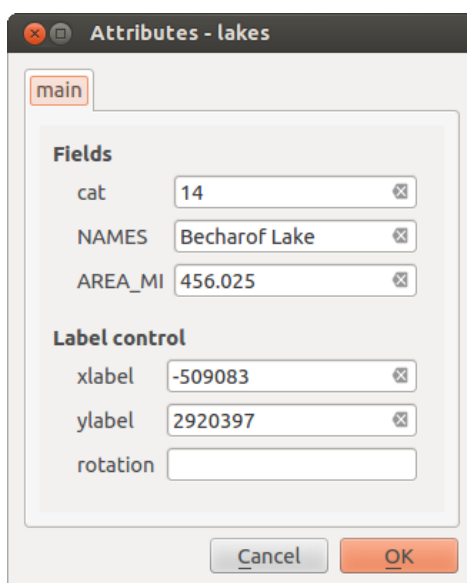


Figura 12.28: Resultando formulário embutido em uma sessão de entrada de dados

- Defina o: guilabel: *codificação de fonte de dados* para definir opções específicas do provedor e ser capaz de ler o arquivo

Sistema de Referência de Coordenadas

- : guilabel: *Especificar* o sistema de referência de coordenadas. Aqui, você pode visualizar ou alterar a projeção da camada vetorial específica.
- Crie um índice espacial (somente para formatos OGR suportados)
- : guilabel: *Atualizar Extensões* para uma camada
- Exibir ou alterar a projeção da camada de vetor específico, clicando em: guilabel: *Especifique ...*

caixa de seleção: guilabel: *Escala dependente visibilidade*

- Você pode definir o: guilabel: *máximo (inclusivo)* e: guilabel: *mínima (exclusivo)* escala. A escala também pode ser definido pelas ** [Atuais] ** botões.

Características subconjuntos

- Com o botão [**Construtor de consulta**], você pode criar um subconjunto dos feições na camada que será visualizada (consulte também a seção: ref: *vector_query_builder*).

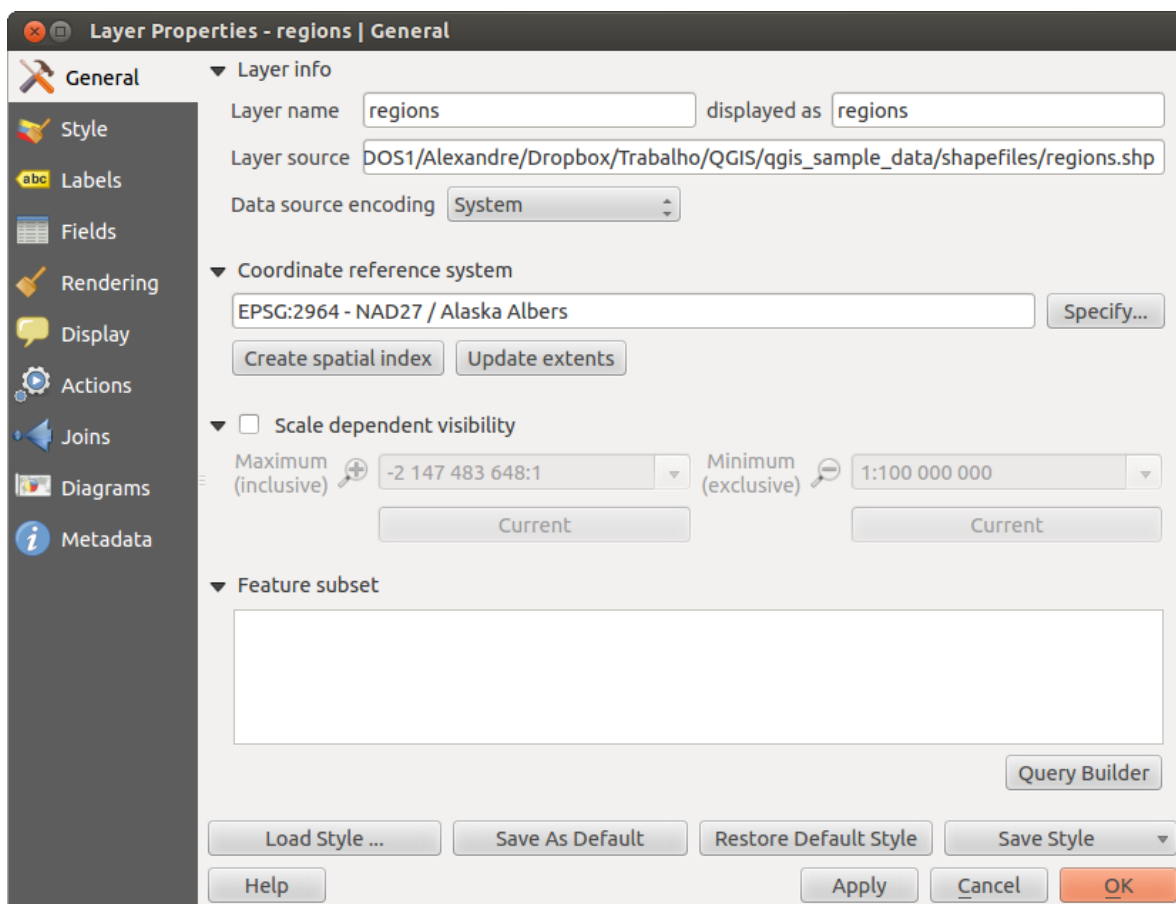



Figura 12.29: Menu geral no vetor camadas propriedades diálogo | nix |

12.3.5 Menu Renderizador

QGIS 2.2 introduz suporte para generalização de feição on-the-fly. Isso pode melhorar o tempo de edição ao desenhar muitas unidades complexas em pequenas escalas. Esta feição pode ser ativada ou desativada nas configurações de camada usando a opção **caixal** :guilabel: *Simplifique geometria*. Há também uma nova configuração global que permite a generalização, por padrão, para as camadas recém-adicionadas (ver seção :ref: *gui_options*). **Nota:**

generalização de feições pode introduzir resquícios em sua saída editada em alguns casos. Isto podem incluir lascas entre polígonos e prestação imprecisa quando usando camadas de símbolos baseadas em deslocamento.

12.3.6 Menu Visualização

 This menu is specifically created for Map Tips. It includes a new feature: Map Tip display text in HTML. While you can still choose a *Field* to be displayed when hovering over a feature on the map, it is now possible to insert HTML code that creates a complex display when hovering over a feature. To activate Map Tips, select the menu option *View* → *MapTips*. Figure Display 1 shows an example of HTML code.

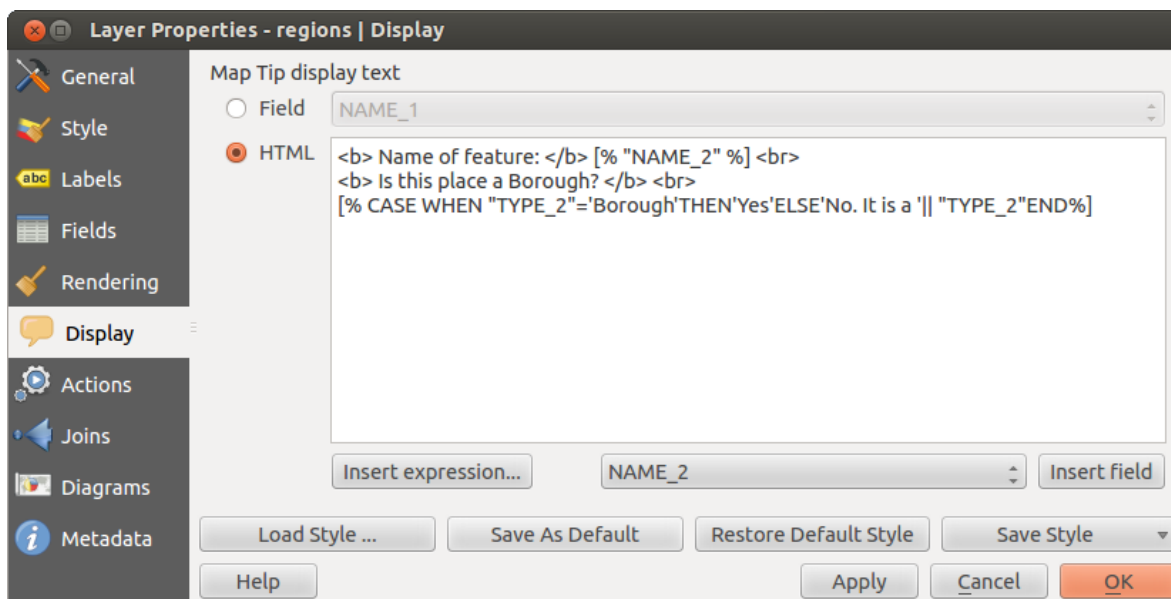


Figura 12.30: Código HTML para dica de mapa 

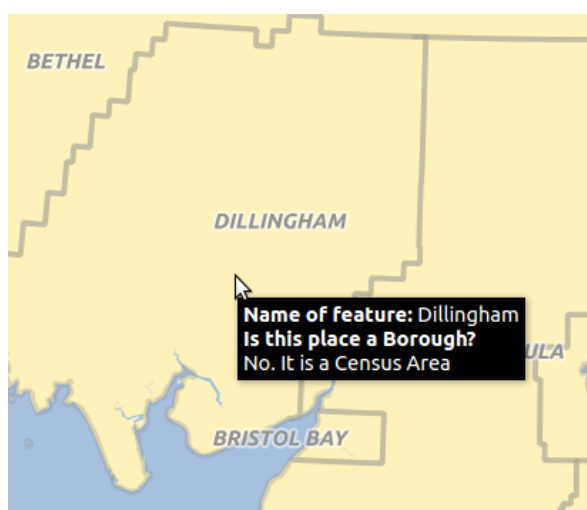



Figura 12.31: Dica de mapa feito com código HTML 

12.3.7 Menu de Ações



QGIS fornece a capacidade de executar uma ação com base nos atributos de uma feição. Isso pode ser usado para executar qualquer número de ações, por exemplo, a execução de um programa com argumentos construídos a partir dos atributos de uma feição ou passar parâmetros a uma ferramenta de comunicação web.

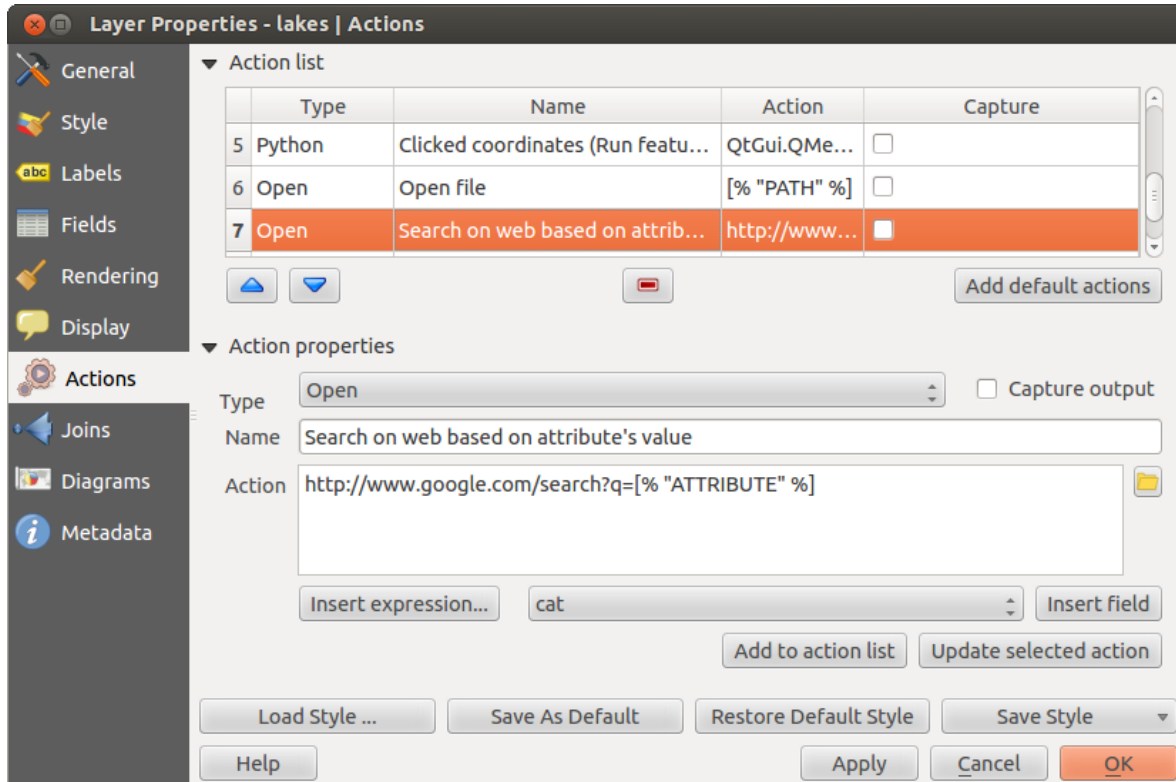


Figura 12.32: Visão geral do diálogo de ação com algumas ações de amostra | nix |

Ações são úteis quando você frequentemente deseja executar uma aplicação externa ou visualizar uma página web com base em um ou mais valores em sua camada de vetor. Elas são divididas em seis tipos e pode ser usadas assim:

- Iniciar ações genéricas, Mac, Windows e Unix em um processo externo.
- Executar ações Python uma expressão Python.
- Ações genéricas e Python são visíveis em toda parte.
- Ações Mac, Windows e Unix são visíveis apenas na respectiva plataforma (ou seja, você pode definir ações de três “Editar” para abrir um editor e os usuários só podem ver e executar a ação “Editar” na sua plataforma para executar o editor) .

Existem vários exemplos incluídos na caixa de diálogo. Você pode carregá-los clicando no **[Adicionar ações padrão]**. Um exemplo é a realização de uma pesquisa com base em um valor de atributo. Este conceito é usado na discussão a seguir.

**** Definição de Ações ****

Atributo ações são definidas a partir do vetor: guilabel: diálogo ‘Propriedades da Camada . Para: índice: ‘Definir uma Ação, abra o vetor: guilabel: ‘Propriedades da Camada ‘ de diálogo e clique no: guilabel: Menu’ Ações . Vá para o: guilabel: ‘propriedades Ação. Selecione ‘genérico’ como tipo e fornecer um nome descritivo para a ação. A própria ação deve conter o nome do aplicativo que será executado quando a ação é invocado. Você pode adicionar um ou mais valores de campos de atributos como argumentos para a aplicação. Quando a ação é chamada, qualquer conjunto de caracteres que começam com um “% “, seguido do nome de um campo será substituído pelo valor desse campo. Os caracteres especiais: índice: %% será substituído pelo valor do campo

que foi selecionado da tabela identificam resultados ou atributo (ver [using_actions](#) abaixo). Aspas duplas podem ser usadas ao texto do grupo em um único argumento para o programa, script ou comando. Aspas duplas serão ignoradas se precedido por uma barra invertida.

Se você tem os nomes de campo que são substrings de outros nomes de campo (por exemplo, " coll" e coll10), você deve indicar, cercado o nome do campo (e do %caráter) entre colchetes (por exemplo, [%coll10]). Isso impedirá que o campo com nome %Coll10 seja confundido com o campo de nome %coll1 com um 0 no final. Os suportes serão removidos por QGIS quando se substitui no valor do campo. Se você deseja que o campo substituído seja cercado por colchetes, utilizar um segundo conjunto da seguinte maneira: [[% Coll10]].

Usando o: guilabel: ferramenta *Identificar as Características*, você pode abrir o: guilabel: diálogo *Identificar resultados* . Ele inclui um item * (Derivado) * que contém informações relevantes para o tipo de camada. Os valores deste item pode ser acessados de maneira semelhante a outros campos fazendo preceder o nome do campo derivado com " (Derivado). ". Por exemplo, uma camada de pontos tem um campo "X " e "Y " , e os valores destes campos pode ser utilizado no processo com "% (derivado).X " e "% (Derived). Y ". Os atributos derivados só estão disponíveis a partir do: guilabel: *Identificar Resultados caixa de diálogo*, e não o: guilabel: caixa de diálogo 'Attribute Table ' .

Dois: índice: *ações exemplo* são mostradas abaixo:

- konqueror http://www.google.com/search?q=%nam
- konqueror http://www.google.com/search?q=%%

No primeiro exemplo, o navegador Konqueror é consultado e passou uma URL para abrir. A URL executa uma pesquisa no Google sobre o valor do campo " nam" da nossa camada vetorial. Note-se que a aplicação ou script chamado pela ação devem estar no caminho, ou você deve fornecer o caminho completo. Para ter certeza, poderíamos reescrever o primeiro exemplo como: " /opt/kde3/bin/konqueror http://www.google.com/search?q=%nam ". Isto irá assegurar que a aplicação konqueror será executado quando a ação é consultada.

O segundo exemplo usa a notação % % , que não depende de um campo específico para o seu valor. Quando a ação é consultado, o % % será substituído pelo valor do campo selecionado na tabela de identificar resultados ou atributo. ** Utilizando Ações **

As ações podem ser chamados a partir de qualquer: guilabel: " Identificar resultados de diálogo, uma: guilabel: diálogo *Tabela de Atributos* ' ou a partir de: guilabel: 'Execute Ação de Recurso (lembre-se que esses diálogos podem ser abertos clicando | mActionIdentify | : sup : *identificar as características* ou | mActionOpenTable | : sup : *Open Table Atributo* or | mAction | : sup : *Ação recurso Execute* o). Para chamar uma ação, clique direito sobre o disco e escolha a ação a partir do menu pop-up. As ações são listadas no menu pop-up com o nome que você atribuiu ao definir a ação. Clique na ação que você deseja executar.

Se você está chamando uma ação que usa o "% % " notação, clique com o botão direito sobre o valor do campo na: guilabel: " Identificar resultados de diálogo ou o: guilabel: 'Tabela de atributos' diálogo que você deseja passar para a aplicação ou script.

Aqui está outro exemplo que puxa os dados de uma camada vetorial e os insere em um arquivo usando o bash e o comando "eco" (por isso só vai funcionar nlnixl ou talvez nolosxl). A camada em questão tem campos para um nome da espécie `taxon_name` , latitude `lat` e longitude `long` . Eu gostaria de ser capaz de fazer uma seleção espacial das localidades e exportar esses campos de valores em um arquivo de texto para o registro selecionado (mostrado em amarelo na área do mapa do QGIS). Aqui é a ação para atingir este objectivo:

```
bash -c "echo \"%taxon_name %lat %long\" >> /tmp/species_localities.txt"
```

Depois de selecionar algumas localidades e executar a ação em cada um, a abertura do arquivo de saída irá mostrar algo como isto:

```
Acacia mearnsii -34.0800000000 150.0800000000
Acacia mearnsii -34.9000000000 150.1200000000
Acacia mearnsii -35.2200000000 149.9300000000
Acacia mearnsii -32.2700000000 150.4100000000
```

Como um exercício, criamos uma ação que faz uma pesquisa no Google sobre o a camada `lagos`. Primeiro, precisamos determinar o URL necessário para realizar uma pesquisa sobre a palavra-chave. Isso póde ser realizado facilmente indo ao Google e realizando uma pesquisa simples, em seguida, pegar a URL na barra de endereços do

seu navegador. A partir deste pequeno esforço, vemos que o formato é: <http://google.com/search?q=qgis> , onde QGIS é o termo de pesquisa. Armado com esta informação, podemos prosseguir:

1. Verifique se a camada `lago`s está carregada.
2. Abra a caixa de diálogo *Propriedades da camada* clicando duas vezes sobre o nome do shapefile na legenda ou com o botão direito do mouse e escolha *Propriedades* a partir do menu pop-up.
3. Clique em `:guilabel:` menu 'Ações'.
4. Digite um nome para a ação, por exemplo " Google Search ".
5. Para a ação, precisamos fornecer o nome do programa externo a ser executado. Neste caso, podemos usar o Firefox. Se o programa não estiver em seu caminho, você precisará fornecer o caminho completo.
6. Seguindo o nome da aplicação externa, adicione a URL usada para fazer uma pesquisa no Google, mas não inclua o termo de busca: " <http://google.com/search?q=> "
7. O texto no campo de ação será mostrado como este: " firefox <http://google.com/search?q=> "
8. Clique na caixa oculta que contém os nomes dos campos para camada " lago" s". Ele está localizado logo ao lado esquerdo do botão **[Inserir Campo]** .
9. A partir da caixa drop-down, selecione 'NOMES' e clique ** [Inserir campo]. **
10. Seu texto ação agora se parecerá como este:

```
firefox http://google.com/search?q=%NAMES
```
11. Para finalizar a ação, clique no botão **[Adicionar à lista de ações]**.

Isso completa a ação, e ele está pronto para usar. O texto final da ação deve ser semelhante a este:

```
firefox http://google.com/search?q=%NAMES
```

Agora podemos utilizar a ação. Feche a: aba de diálogo: 'Propriedades da camada ' e aplique um zoom em uma área de interesse. Certifique-se que a camada `lago`s está ativa e identifique um lago. Na caixa o resultado que você poderá visualizar que a nossa ação está visível:

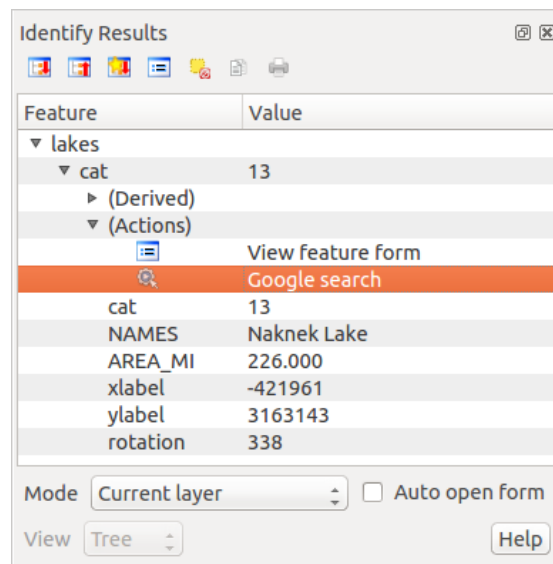


Figura 12.33: Selecione a característica e escolha ação 

Quando clicamos sobre a ação, ela faz o Firefox e navegar para o URL <http://www.google.com/search?q=Tustumena>. Também é possível adicionar outros campos de atributo para essa ação. Para isso basta um + ao final do texto de ação, selecionar outro campo e clicar em **[Inserir Campo]**. Nesse exemplo não temos outro campo disponível para efetuarmos a busca.

Você pode definir várias ações para uma camada, e cada um vai aparecer no diálogo *Identificar resultados*.

Há todos os tipos de usos para as ações. Por exemplo, se você tem uma camada de pontos que contém as localizações de imagens ou fotos, juntamente com um nome de arquivo, você pode criar uma ação para lançar um visualizador para exibir a imagem. Você também pode usar ações para lançar relatórios baseados na web para um campo de atributo ou combinação de campos, especificando-as da mesma forma que fizemos no nosso exemplo de busca Google.

Nós também podemos fazer exemplos mais complexos, por exemplo, usando ações **Python**.

Normalmente, quando nós criamos uma ação para abrir um arquivo com uma aplicação externa, podemos usar caminhos absolutos, ou, eventualmente, caminhos relativos. No segundo caso, o caminho é relativo ao local do arquivo executável do programa externo. Mas o que dizer se precisamos usar caminhos relativos, em relação à camada selecionada (um baseado em arquivo, como um shapefile ou SpatiaLite)? O código a seguir irá fazer o truque:

```
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 ] );
```

Nós apenas temos que lembrar que a ação é um do tipo variáveis *Python* e o *comando* e *imagerelpath* devem ser alteradas para atender as necessidades.

Mas o que dizer se o caminho relativo deve ser relativo ao arquivo de projeto (salvo)? O código da ação Python seria:

```
command="firefox";
imagerelpath="images/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 ] );
```


Outro exemplo de ação Python é a que nos permite adicionar novas camadas ao projeto. Por exemplo, os exemplos a seguir irão adicionar ao projeto, respectivamente, um vetor e raster. Os nomes dos arquivos a serem adicionados ao projeto e os nomes a serem indicados para as camadas são conduzidos de dados (*nome do arquivo* e *nome da camada* são nomes da tabela de atributos do vetor, onde a ação foi criada coluna):

```
qgis.utils.iface.addVectorLayer('/yourpath/[% "filename" %].shp', '[% "layername" %]',
    'ogr')
```

Para adicionar um raster (uma imagem TIF neste exemplo), ele se torna:

```
qgis.utils.iface.addRasterLayer('/yourpath/[% "filename" %].tif', '[% "layername" %]
    ')
```

12.3.8 Menu Juntar

ljuntar! O menu *Juntar* permite juntar uma tabela de atributos a uma camada de vetor carregada. Depois de clicar , aparecerá o diálogo *Juntar ao vetor adicionado*. Como colunas de chave, você tem que definir um juntar camada que você deseja se conectar com a camada de vetor alvo. Então, você tem que especificar o campo de associação que é comum a ambos as camada de aderir e a camada de destino. Agora você também pode especificar um subconjunto de campos da camada se juntou com base na caixa **lcaixa!** *Escolha quais campos serão juntados*. Como um resultado da junção, todas as informações da camada de junção e a camada de destino são apresentados

na tabela de atributos da camada de informações de destino como se juntou. Se você especificou um subconjunto de campos somente esses campos são exibidos na tabela de atributos da camada de destino.

QGIS atualmente tem suporte para juntar formatos de tabela não-espaciais suportados pela OGR (por exemplo, CSV, DBF e Excel), texto delimitado e o provedor do PostgreSQL (veja [figure_joins_1](#)).

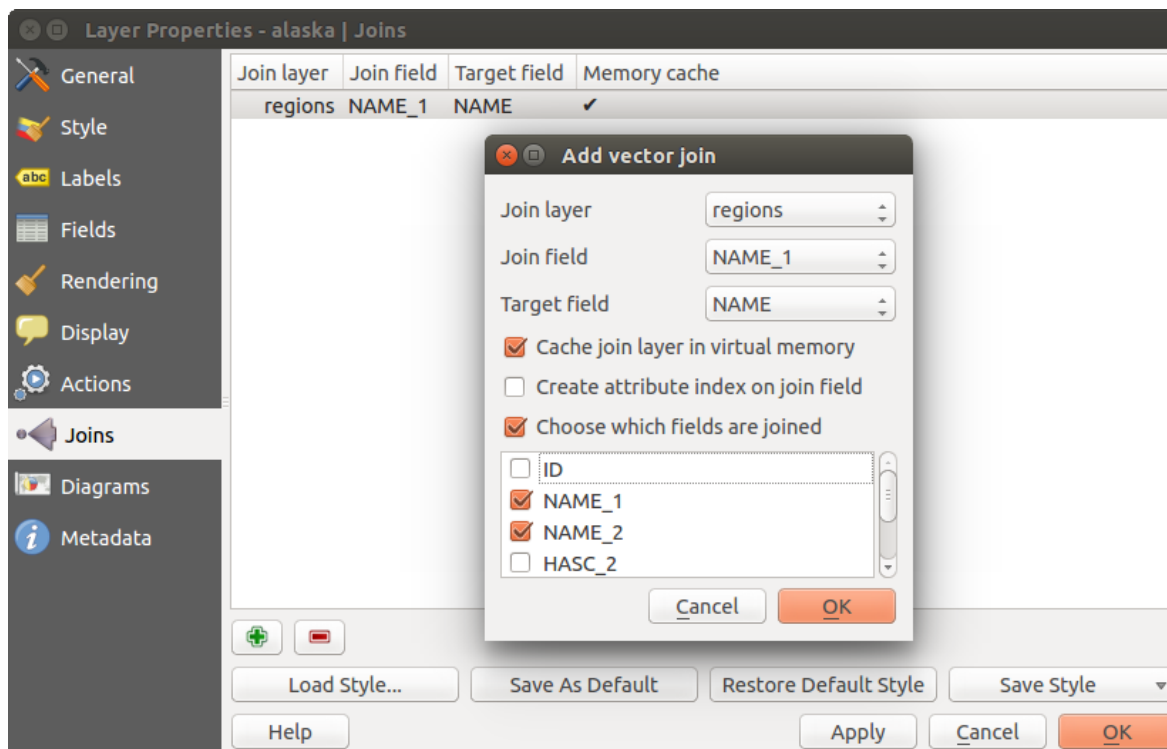


Figura 12.34: Junte uma tabela de atributos de uma camada de vetor existente | nix |

Além disso, o diálogo juntar vetor adicionado permite:

- *Cache join layer in virtual memory*
- *Create attribute index on the join field*

12.3.9 Menu Diagramas



The *Diagrams* menu allows you to add a graphic overlay to a vector layer (see [figure_diagrams_1](#)).

A implementação atual do núcleo de diagramas fornece suporte para gráficos de pizza, diagramas de texto e histogramas.

O menu é dividido em quatro abas: *Aspecto*, *Tamanho*, *Posição* e *Opções*.

Nos casos de o diagrama de texto e gráfico de pizza, valores de texto de diferentes colunas de dados são exibidos um abaixo do outro, com um círculo ou uma caixa e divisores. Na guia *Tamanho*, tamanho diagrama é baseado em um tamanho fixo ou em escala linear de acordo com um atributo de classificação. A colocação dos diagramas, o que é feito na guia *Posição*, interage com a nova rotulagem, assim conflitos de posição entre os diagramas e os rótulos são detectados e resolvidos. Além disso, as posições de gráfico pode ser corrigidas manualmente.

Vamos demonstrar um exemplo e sobrepor sobre a camada limite Alaska um diagrama de texto mostrando dados de temperatura a partir de uma camada vetorial climática. Ambas as camadas de vetores fazem parte do conjunto de dados da amostra do QGIS (ver seção [:ref: label_sampledata](#)).

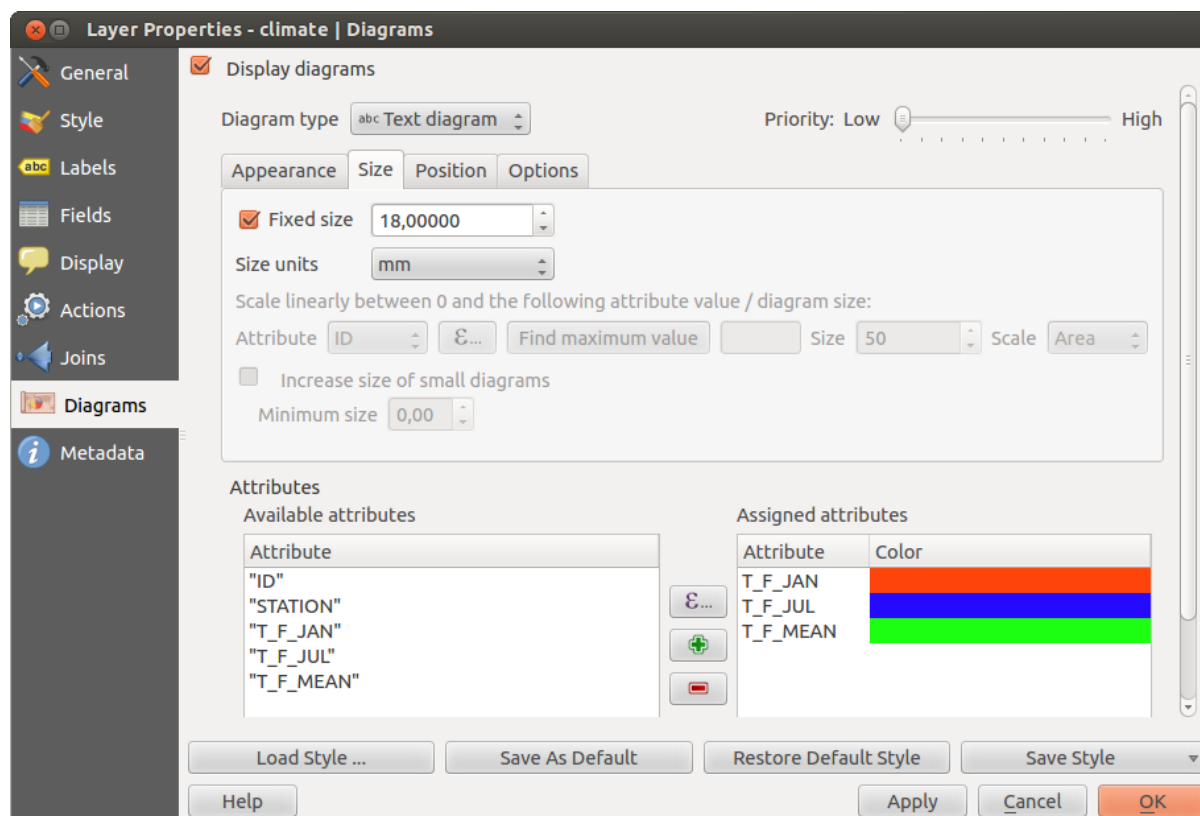





Figura 12.35: Diálogo propriedades Vetoriais com menu diagrama 

1. Primeiro, clique no ícone  Carregar Vetor, procure a pasta conjunto de dados da amostra do QGIS, e carregue as duas camadas shape vetorial :file: *alaska.shp* e :file: *climate.shp*.
2. Dê um clique duplo na camada *clima* na legenda do mapa para abrir a aba de diálogo ‘Propriedades da camada’.
3. Clique no menu *Diagramas*, ative **Exibição de diagramas**, e o tipo *Diagrama* caixa de combinação , selecione ‘diagrama de Texto’.
4. Na guia *Aparência*, escolhemos um azul claro como cor de fundo, e na guia *Tamanho*, montamos um tamanho fixo de 18 mm.
5. Na guia *Posição*, a colocação pode ser definida como ‘em torno do ponto’.
6. Iremos mostrar os valores das três colunas *T_F_JAN*, *T_F_JUL* e *T_F_MEAN*. Primeiro, selecione *T_F_JAN* como atributos e clique no botão , e de seguida *T_F_JUL* e finalmente *T_F_MEAN*.
7. Agora clique **[Aplicar]** para exibir o diagrama no *qgis* janela principal.
8. Você pode adaptar o tamanho do gráfico no: guilabel: guia ‘Tamanho’. Desative a guilabel: *Tamanho Fixo* e defina o tamanho dos diagramas com base em um atributo com o **[Encontrar valor máximo]** botão e o: guilabel: menu *Tamanho*. Se os diagramas aparecerem muito pequeno na tela, você pode ativar o guilabel: ‘Aumentar o tamanho de pequenos diagramas’ e definir o tamanho mínimo dos diagramas.
9. Mude as cores de atributos clicando duas vezes sobre os valores de cor no campo *atributos atribuídos*. [Figure_diagrams_2](#) dá uma idéia do resultado.
10. Finalmente, clique **[Ok]**.

Lembre-se que no: guilabel: guia ‘Posição’, a guilabel: *Dados posição definida* dos diagramas é possível. Aqui você pode usar atributos para definir a posição do diagrama. Você também pode definir a visibilidade dependente escala no: guilabel: guia ‘Aparência’.

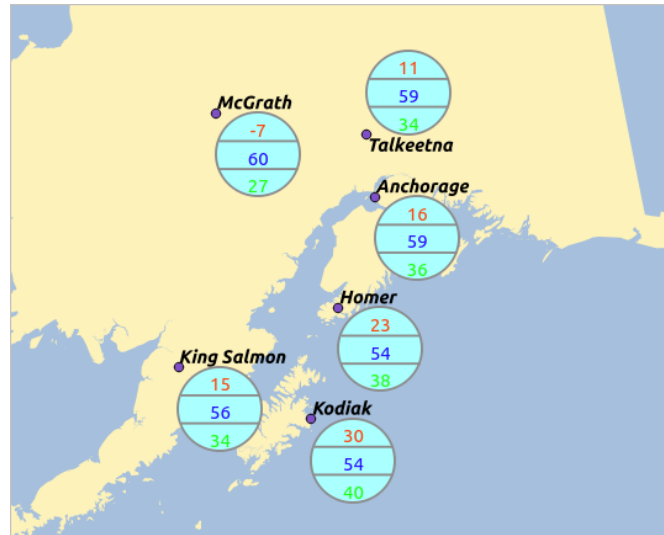



Figura 12.36: Diagrama de dados de temperatura sobrepostos no mapa | nix |

O tamanho e os atributos também pode ser uma expressão. Use o botão  para adicionar uma expressão. Veja: ref: *capítulo vector_expressions* para obter mais informações e exemplo.

12.3.10 Menu Metadados



The *Metadata* menu consists of *Description*, *Attribution*, *MetadataURL* and *Properties* sections.

No: guilabel: seção *Propriedades*, você obter informações gerais sobre a camada, incluindo especificações sobre o tipo e localização, número de recursos, tipo de recursos e capacidades de edição. A: guilabel: tabela *Extensões* fornece informações camada de extensão e a: guilabel: 'Camada Sistema de Referência Espacial', é a informação sobre o CRS da camada. Esta é uma forma rápida de obter informações sobre a camada.

Além disso, você pode adicionar ou editar um título e resumo para a camada no: guilabel: seção *Descrição*. Também é possível definir um: guilabel: *lista de palavras-chave*. Estas listas de palavras-chave podem ser utilizada em um catálogo de metadados. Se você quiser usar um título de um arquivo de metadados XML, você tem que preencher um link no: guilabel: campo *DataURL*. Use: guilabel: *Atribuição* para obter dados de atributos de um catálogo de metadados XML. In: guilabel: *MetadataUrl*, você pode definir o caminho geral para o catálogo de metadados XML. Esta informação será salva no | qg | arquivo de projeto para as sessões subsequentes e será utilizado para | qg | servidor.

12.4 Expressões

Os recursos **Expressões** estão disponíveis através da calculadora de campo ou no botão adicionar uma nova coluna na tabela de atributo ou a guia de campo nas propriedades da camada; através do graduado, categorizados e renderização baseada em regra na guia Estilo de as propriedades da camada; através da rotulagem baseado na expressão | `browsebutton` | na aplicação do núcleo | `mActionLabeling` | sup: *Rotulagem*; através da seleção de feições e através da guia diagrama das propriedades da camada.

Há uma maneira poderosa para manipular valor do atributo a fim de mudar dinamicamente o valor final, a fim de mudar o estilo de geometria, o conteúdo do rótulo, o valor para o diagrama, selecione algumas feições ou criar um campo virtual.

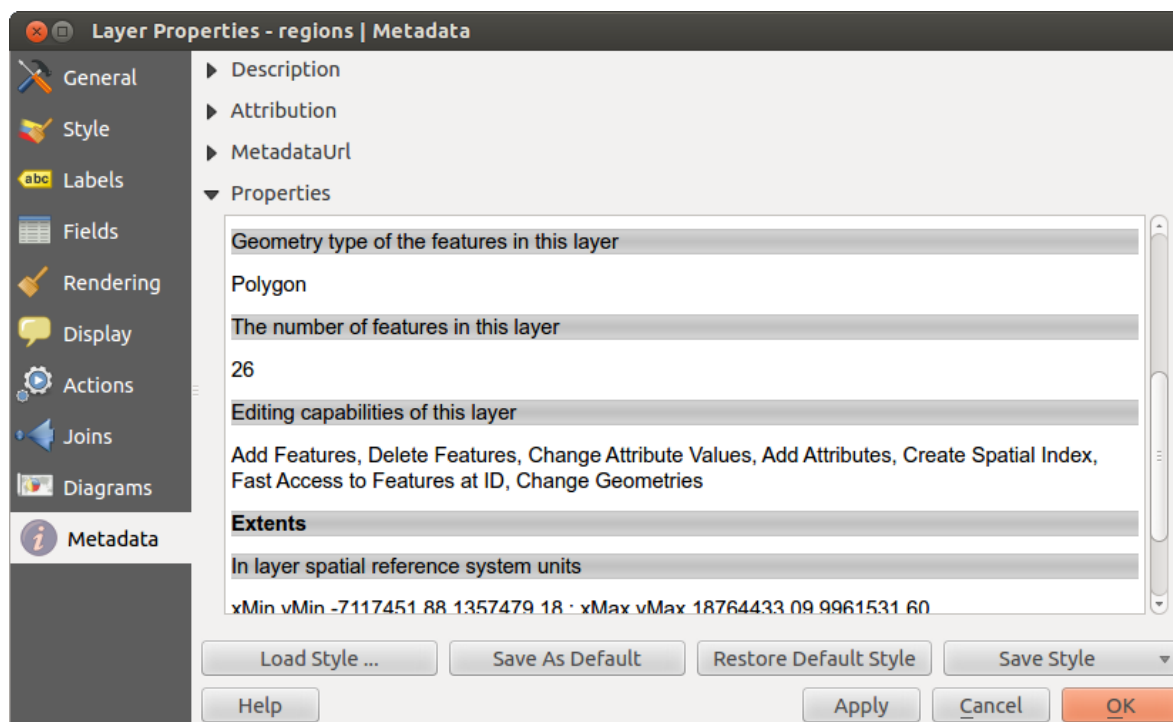


Figura 12.37: Menu Metadata no diálogo das propriedades das camadas de vetor | nix |

12.4.1 Lista de funções

A **Lista de funções** contém funções bem como os campos e valores. Veja a ajuda da função em **Ajuda da Função Seleccionada**. Em **Expressão** voce verá a expressão para calculo que voce criou com a **Lista de Funções**. Para ver as operações mais comuns usadas, veja **Operadores**.

No **Lista de funções**, clique em: guilabel: *Campos e Valores* para ver todos os atributos da tabela de atributos a serem pesquisados. Para adicionar um atributo para a calculadora campo no campo **Expressão**, clique duas vezes em seu nome na lista *Campos e Valores*. Geralmente, você pode usar nos campos diversos, valores e funções para construir a expressão de cálculo, ou você pode simplesmente digitá-la na caixa. Para exibir os valores de um campo, você apenas clique direito sobre o campo apropriado. Você pode escolher entre: guilabel:*carregar os 10 valores únicos de topo* *carregar todos valores únicos*. No lado direito, a lista dos **valores dos campos** abrirá com os valores exclusivos. Para adicionar um valor na calculadora de campo na caixa **Expressão**, clique duas vezes em seu nome na lista **Valores dos Campos**.

Os grupos :guilabel: *Operadores, Matemática, Conversões, String, Geometria e Registro* fornecem várias funções. Em *Operadores*, você encontra operadores matemáticos. Procura em *Matemática* as funções matemáticas. O grupo *Conversões* contém funções que convertem um tipo de dados para outro. O grupo *String* fornece funções para cadeias de dados. No grupo *Geometria*, você encontra funções para objetos de geometria. Com as funções do grupo *Registro*, você pode adicionar uma numeração para o seu conjunto de dados. Para adicionar uma função na calculadora de campo na caixa **Expression**, clique no > e, em seguida, clicar duas vezes sobre a função.

Operadores

O grupo operadores contem (ex.: +, -, *).

a + b	a plus b
a - b	a minus b
a * b	a multiplied by b
a / b	a divided by b
a % b	a modulo b (for example, 7 % 2 = 1, or 2 fits into 7 three times with remainder 1)
a ^ b	a power b (for example, 2^2=4 or 2^3=8)

a = b	a and b are equal
a > b	a is larger than b
a < b	a is smaller than b
a <> b	a and b are not equal
a != b	a and b are not equal
a <= b	a is less than or equal to b
a >= b	a is larger than or equal to b
a ~ b	a matches the regular expression b
+ a	positive sign
- a	negative value of a
	joins two values together into a string 'Hello' ' world'
LIKE	returns 1 if the string matches the supplied pattern
ILIKE	returns 1 if the string matches case-insensitive the supplied pattern (ILIKE can be used instead of LIKE to make the match case-insensitive)
IS	returns 1 if a is the same as b
OR	returns 1 when condition a or b is true
AND	returns 1 when condition a and b are true
NOT	returns 1 if a is not the same as b
column name "column name"	value of the field column name, take care to not be confused with simple quote, see below
'string'	a string value, take care to not be confused with double quote, see above
NULL	null value
a IS NULL	a has no value
a IS NOT NULL	a has a value
a IN (value[,value])	a is below the values listed
a NOT IN (value[,value])	a is not below the values listed

Alguns exemplos:

- Juntando um texto e um valor a partir de um nome de coluna:

```
'My feature's id is: ' || "gid"
```

- Teste se a “descrição” campo de atributo começa com o texto a ‘Olá’ do valor (observe a posição do caractere %):

```
"description" LIKE 'Hello%'
```

Condicionais

Este grupo contém funções para lidar com verificações condicionais em expressões.

CASE	evaluates multiple expressions and returns a result
CASE ELSE	evaluates multiple expressions and returns a result
coalesce	returns the first non-NULL value from the expression list
regexp_match	returns true if any part of a string matches the supplied regular expression

Alguns exemplos:

- Enviar de volta um valor se a primeira condição é verdade, outra coisa manda outro valor:

```
CASE WHEN "software" LIKE '%QGIS%' THEN 'QGIS' ELSE 'Other'
```

Funções Matemáticas

Este grupo contém funções matemáticas (por exemplo, raiz quadrada, sen e cos).

<code>sqrt(a)</code>	square root of a
<code>abs</code>	returns the absolute value of a number
<code>sin(a)</code>	sine of a
<code>cos(a)</code>	cosine of a
<code>tan(a)</code>	tangent of a
<code>asin(a)</code>	arcsin of a
<code>acos(a)</code>	arccos of a
<code>atan(a)</code>	arctan of a
<code>atan2(y,x)</code>	arctan of y/x using the signs of the two arguments to determine the quadrant of the result
<code>exp</code>	exponential of a value
<code>ln</code>	value of the natural logarithm of the passed expression
<code>log10</code>	value of the base 10 logarithm of the passed expression
<code>log</code>	value of the logarithm of the passed value and base
<code>round</code>	round to number of decimal places
<code>rand</code>	random integer within the range specified by the minimum and maximum argument (inclusive)
<code>randf</code>	random float within the range specified by the minimum and maximum argument (inclusive)
<code>max</code>	largest value in a set of values
<code>min</code>	smallest value in a set of values
<code>clamp</code>	restricts an input value to a specified range
<code>scale_linear</code>	transforms a given value from an input domain to an output range using linear interpolation
<code>scale_exp</code>	transforms a given value from an input domain to an output range using an exponential curve
<code>floor</code>	rounds a number downwards
<code>ceil</code>	rounds a number upwards
<code>\$pi</code>	pi as value for calculations

Conversões

Este grupo contém funções para converter um tipo de dado para outro (por exemplo, texto para inteiro, inteiro para texto).

<code>toint</code>	converts a string to integer number
<code>toreal</code>	converts a string to real number
<code>tostring</code>	converts number to string
<code>todatetime</code>	converts a string into Qt data time type
<code>todate</code>	converts a string into Qt data type
<code>totime</code>	converts a string into Qt time type
<code>tointerval</code>	converts a string to an interval type (can be used to take days, hours, months, etc. off a date)

Funções Data e hora

Este grupo contém funções para trabalhar datas e hora data.

\$now	current date and time
age	difference between two dates
year	extract the year part from a date, or the number of years from an interval
month	extract the month part from a date, or the number of months from an interval
week	extract the week number from a date, or the number of weeks from an interval
day	extract the day from a date, or the number of days from an interval
hour	extract the hour from a datetime or time, or the number of hours from an interval
minute	extract the minute from a datetime or time, or the number of minutes from an interval
second	extract the second from a datetime or time, or the number of minutes from an interval

Alguns exemplos:

- Obter o mês e o ano de hoje no formato “02/2015”

```
month($now) || '/' || year($now)
```

Funções String

Este grupo contém funções que operam em textos (por exemplo, quer substituir, converter para maiúsculas).

lower	convert string a to lower case
upper	convert string a to upper case
title	converts all words of a string to title case (all words lower case with leading capital letter)
trim	removes all leading and trailing white space (spaces, tabs, etc.) from a string
wordwrap	returns a string wrapped to a maximum/minimum number of characters
length	length of string a
replace	returns a string with the supplied string replaced
regexp_replace(a,this,that)	returns a string with the supplied regular expression replaced
regexp_substr	returns the portion of a string which matches a supplied regular expression
substr(*a*,from,len)	returns a part of a string
concat	concatenates several strings to one
strpos	returns the index of a regular expression in a string
left	returns a substring that contains the n leftmost characters of the string
right	returns a substring that contains the n rightmost characters of the string
rpadd	returns a string with supplied width padded using the fill character
lpadd	returns a string with supplied width padded using the fill character
format	formats a string using supplied arguments
format_number	returns a number formatted with the locale separator for thousands (also truncates the number to the number of supplied places)
format_date	formats a date type or string into a custom string format

Funções Cor

Este grupo contém funções para manipulação de cores.

color_rgb	returns a string representation of a color based on its red, green, and blue components
color_rgba	returns a string representation of a color based on its red, green, blue, and alpha (transparency) components
ramp_color	returns a string representing a color from a color ramp
color_hsl	returns a string representation of a color based on its hue, saturation, and lightness attributes
color_hsla	returns a string representation of a color based on its hue, saturation, lightness and alpha (transparency) attributes
color_hsv	returns a string representation of a color based on its hue, saturation, and value attributes
color_hsva	returns a string representation of a color based on its hue, saturation, value and alpha (transparency) attributes
color_cmyk	returns a string representation of a color based on its cyan, magenta, yellow and black components
color_cmyka	returns a string representation of a color based on its cyan, magenta, yellow, black and alpha (transparency) components

Funções Geometria

Este grupo contém funções que operam na geometria dos objetos (por exemplo, comprimento, área).

\$geometry	returns the geometry of the current feature (can be used for processing with other functions)
\$area	returns the area size of the current feature
\$length	returns the length size of the current feature
\$perimeter	returns the perimeter length of the current feature
\$x	returns the x coordinate of the current feature
\$y	returns the y coordinate of the current feature
xat	retrieves the nth x coordinate of the current feature. n given as a parameter of the function
yat	retrieves the nth y coordinate of the current feature. n given as a parameter of the function
xmin	returns the minimum x coordinate of a geometry. Calculations are in the Spatial Reference System of this Geometry
xmax	returns the maximum x coordinate of a geometry. Calculations are in the Spatial Reference System of this Geometry
ymin	returns the minimum y coordinate of a geometry. Calculations are in the Spatial Reference System of this Geometry
ymax	returns the maximum y coordinate of a geometry. Calculations are in the Spatial Reference System of this Geometry
geomFromWKT	returns a geometry created from a well-known text (WKT) representation
geomFromGML	returns a geometry from a GML representation of geometry
bbox	
disjoint	returns 1 if the geometries do not share any space together
intersects	returns 1 if the geometries spatially intersect (share any portion of space) and 0 if they don't
touches	returns 1 if the geometries have at least one point in common, but their interiors do not intersect
crosses	returns 1 if the supplied geometries have some, but not

	all, interior points in common
contains	returns true if and only if no points of b lie in the exterior of a, and at least one point of the interior of b lies in the interior of a
overlaps	returns 1 if the geometries share space, are of the same dimension, but are not completely contained by each other
within	returns 1 if geometry a is completely inside geometry b
buffer	returns a geometry that represents all points whose distance from this geometry is less than or equal to distance
centroid	returns the geometric center of a geometry
bounds	returns a geometry which represents the bounding box of an input geometry. Calculations are in the Spatial Reference System of this Geometry.
bounds_width	returns the width of the bounding box of a geometry. Calculations are in the Spatial Reference System of this Geometry.
bounds_height	returns the height of the bounding box of a geometry. Calculations are in the Spatial Reference System of this Geometry.
convexHull	returns the convex hull of a geometry (this represents the minimum convex geometry that encloses all geometries within the set)
difference	returns a geometry that represents that part of geometry a that does not intersect with geometry b
distance	returns the minimum distance (based on spatial ref) between two geometries in projected units
intersection	returns a geometry that represents the shared portion of geometry a and geometry b
symDifference	returns a geometry that represents the portions of a and b that do not intersect
combine	returns the combination of geometry a and geometry b
union	returns a geometry that represents the point set union of the geometries
geomToWKT	returns the well-known text (WKT) representation of the geometry without SRID metadata

Funções Registro

Este grupo contém funções que operam na identificação da registros.

\$rownum	returns the number of the current row
\$id	returns the feature id of the current row
\$currentfeature	returns the current feature being evaluated. This can be used with the 'attribute' function to evaluate attribute values from the current feature.
\$scale	returns the current scale of the map canvas
\$uuid	generates a Universally Unique Identifier (UUID) for each row. Each UUID is 38 characters long.
getFeature	returns the first feature of a layer matching a given attribute value.
attribute	returns the value of a specified attribute from a feature.
\$map	returns the id of the current map item if the map is being drawn in a composition, or "canvas" if the map is being drawn within the main QGIS window.

Campo e Valores

Contém uma lista de campos para a camada. Valores simples podem ser acessados via clique direito.

Selecione o nome do campo da lista, em seguida, clique com o botão direito para acessar um menu de contexto com opções para carregar os valores da amostra do campo selecionado.

Nome campos devem estar entre aspas. Valores ou texto deverá estar cotado-simples.

.

12.5 Editando

QGIS suporta vários recursos para tabelas e camadas vetoriais :**índice:‘Edição’** OGR, SpatiaLite, PostGIS, Espacial MSSQL e Oracle.

Nota: O procedimento para editar camadas do GRASS é diferente - consulte a seção *Digitalizando e editando uma camada vetorial GRASS* para mais detalhes.

Dica: Edições Simultâneas




Esta versão do QGIS não controla se alguém mais está editando uma feição ao mesmo tempo que você. A última pessoa a salvar suas edições ganha.

12.5.1 Configurando a Tolerância de Atracção e Raio de Pesquisa

Antes de podermos editar os vértices, necessitamos de configurar a tolerância e o raio de pesquisa para um valor que nos permita uma edição ideal das geometrias da camada vetorial.

Tolerância de Atração

Tolerância de Atração é a distância que o QGIS usa para *pesquisar* por vértice mais próximo e / ou segmento que você está tentando conectar-se quando você define um novo vértice ou move um vértice existente. Se você não estiver dentro da tolerância de atração, o QGIS deixará o vértice onde você soltar o botão do mouse, em vez de agarrar-lo a um vértice e / ou segmento existente. A configuração da tolerância de atração afeta todas as ferramentas que trabalham com tolerância.

1. Em geral, a tolerância de encaixe pode ser definida escolhendo *Definições* →  *Opções*. No Mac, vá para *QIS* →  *Preferencias...*. No Linux: *Editar* →  *Opções*. Na aba *Digitalização*, você pode escolher entre ‘ao vértice’, ‘ao segmento’ ou ‘ao vértice e segmento’ como modo de encaixe padrão. Você também pode definir uma tolerância de encaixe padrão e um raio de buscas de edições de vértices. A tolerância pode ser definida em unidades do mapa ou em pixels. A vantagem de escolher pixels é que a tolerância de encaixe não precisa ser mudada depois da operação de aproximação. No nosso menor projeto digitalizado (trabalhando com o conjunto de dados do Alaska), nós definimos as unidades de encaixe em pé. Seus resultados podem variar, mas algo na ordem de 300 pés em uma escala de 1:10000 deve ser uma configuração razoável.
2. Uma tolerância de encaixe com base em camadas pode ser escolhido pela escolha *Definições*→ (or *Arquivo*→) *Opções de encaixe...* para permitir e ajustar o modo de encaixe e tolerância em uma base de camada (ver [figure_edit_1](#)).

Note que estes encaixes com base em camadas substitui a opção de encaixe global na aba Digitalização. Então, se você precisa editar uma camada e encaixar seus vértices para outra camada, em seguida habilitando apenas encaixe na camada “encaixar para”, depois diminuir o encaixe global para um valor menor. Além disso, encaixar nunca ocorrerá para uma camada que não está marcada a opção encaixar, independentemente da tolerância de encaixe global. Então tenha certeza de marcar a caixa de seleção para estas camadas que você precisa encaixar.

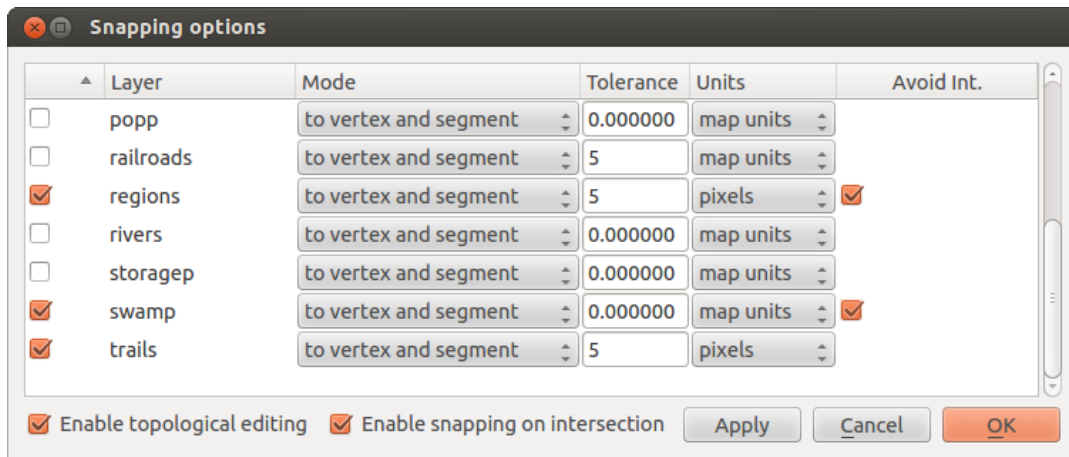



Figura 12.38: Edição das opções de atração numa camada base 


Raio de pesquisa

Pesquisa raio é a distância que o QGIS usa para buscar para o vértice mais próximo que você está tentando mover quando você clicar no mapa. Se você não estiver dentro do raio de busca, o QGIS não vai encontrar e selecionar nenhum vértice para edição, e vai aparecer um aviso irritante para o efeito. Acerte a tolerância e o raio de busca é definido em unidades do mapa ou pixels, para que você possa encontrar o que você precisa experimente definir direito para obtê-los. Se você especificar uma tolerância muito grande, QGIS pode agarrar o vértice errado, especialmente se você está lidando com um grande número de vértices nas proximidades. Defina o raio de busca muito pequeno, e não vai encontrar outro para confundir.


O raio de busca de edições de vértices em unidades da camada pode ser definido no *guia Digitizando* em :menuselection: 'Configurações -> 'lmActionOptions! :menuselection: opções. Este é o mesmo lugar onde você define o geral, projeto- atração tolerância.

12.5.2 Ampliando e Movendo

Antes de editar uma camada, deve fazer uma ampliação à zona da área de interesse. Isto evita que espere enquanto os marcadores dos vértices são renderizados em toda a camada.

Além de usar os ícones  mover mapa e  aproximar /  afastar na barra de ferramentas com o mouse, a navegação pode também ser feita com a roda do mouse, espaço e as teclas de direção.

Ampliando e movendo com a roda do mouse

Enquanto a digitalização, você pode pressionar a roda do mouse para se deslocar dentro da janela principal, e você pode rolar a roda do mouse para ampliar e reduzir o mapa. Para zoom, coloque o cursor do mouse dentro da área do mapa e rodá-lo para a frente (longe de você) para aumentar o zoom; e para trás (na sua direção) para diminuir o zoom. A posição do cursor do mouse será o centro da área ampliada de interesse. Você pode personalizar o comportamento do zoom da roda do mouse usando o :guilabel: aba *Ferramentas do Mapa* sob o :menuselection: 'Configurações -> '  :menuselection: menu de opções.

Movendo a direção com as setas do teclado

Panorâmica no mapa durante a digitalização é possível com as teclas de seta. Coloque o cursor do mouse dentro da área do mapa, e clique na seta para a direita para se deslocar a leste, seta para a esquerda para mover oeste, seta para cima para deslocar para o norte, e para baixo chave de seta para se deslocar para o sul.

Você também pode usar a barra de espaço para fazer com que temporariamente os movimentos do mouse para percorrer o mapa. A: kbd: *PgUp* e: kbd: ‘chaves *PgDown*’ do seu teclado fará a apresentação do mapa para ampliar ou reduzir, sem interromper a sessão de digitalização.

12.5.3 Edição Topológica

Além de opções de ajuste com base em camadas, você também pode definir funcionalidades topológicas no :guilabel: menu de opções *Tirando ...* diálogo no :menuselection: *Configurações (Arquivo ou:: menuselection)*. Aqui, você pode definir **lcaixal** :guilabel: ‘Ativar edição’ topológica, e / ou para as camadas de polígonos, você pode ativar a coluna **lcaixal** :guilabel: *Evite Int.*, o que evita intersecção de novos polígonos.

Ativar edição topológica

A opção **lcaixal** :guilabel: ‘Ativar edição’ topológica é para edição e manutenção de limites comuns em mosaicos poligonais. O QGIS ‘detecta’ uma fronteira compartilhada em um mosaico de polígonos, assim você só tem que mover o vértice uma vez, e o QGIS cuidará de atualizar o outro limite.

Evitar intersecções de novos polígonos

A segunda opção topológica na **lcaixal** :guilabel: *Evite coluna Int.*, chamado :guilabel: ‘Evite intersecções de novo polígonos’, evita sobreposições em mosaicos poligonais. É por digitalização mais rápida de polígonos adjacentes. Se você já tem um polígono, é possível com esta opção para digitalizar o segundo de tal forma que ambos se cruzam, e o QGIS corta então o segundo polígono da fronteira comum. A vantagem é que você não tem que digitalizar todos os vértices da fronteira comum.













Activar atracção nas intersecções

Outra opção é usar **lcaixal** :guilabel: *Habilitar encaixe em intersecção*. Ele permite que você encaixe em uma intersecção de camadas de fundo, mesmo se não há nenhum vértice na intersecção.


12.5.4 Digitalizar uma camada existente

Por padrão, QGIS carrega camadas de leitura. Esta é uma salvaguarda para evitar que acidentalmente a edição de uma camada se houver um deslize do mouse. No entanto, você pode optar por editar qualquer camada, enquanto o provedor de dados suporta, e a fonte de dados subjacente é gravável (ou seja, seus arquivos não são somente leitura).

Em geral, ferramentas de edição de camadas vetoriais estão divididas em uma barra de digitalização ou digitalização avançada, descrita na seção


Ícone	Finalidade	Ícone	Finalidade
	Edições actuais		Alternar edição
	Adicionando Elementos: Capturar Ponto		Adicionando Elementos: Capturar Linha
	Adicionando Elementos: Capturar Polígono		Mover Elemento
	Ferramenta de Nós		Apagar Seleccionados
	Cortar Elementos		Copiar Elementos
	Colar Elementos		Salvar edições da camada

Edição da Tabela: Edição básica da camada vectorial pela barra de ferramentas




Todas as sessões de edição começa escolhendo o  :sup: opção *edição* alternância. Isto pode ser encontrado no menu de contexto depois botão direito sobre a entrada de legenda para uma determinada camada.

Alternativamente, você pode usar a opção `! mActionToggleEditing` para alternância na barra de ferramentas de digitalização para iniciar ou parar o modo de edição. Uma vez que a camada está em modo de edição, marcadores aparecerá nos vértices, e botões de ferramentas adicionais na barra de ferramentas de edição ficará disponível.

Dica: Salvar Regularmente

Lembre-se de  Salvar Edições da Camada regularmente. Isto irá também verificar que a sua fonte de dados aceita todas as alterações.

Adicionando Elementos

Pode usar os ícones ,  Adicionar Elemento ou  Adicionar Elemento da barra de ferramentas para por o cursor QGIS no modo de digitalização.

Para cada elemento, primeiro digitaliza a geometria, e de seguida introduz os atributos. Para digitalizar a geometria, clique com o botão direito do mouse na área do mapa para criar o primeiro ponto do seu novo elemento.

Para linhas e polígonos, mantenha o clique com o botão direito do mouse para cada ponto adicional que pretende capturar. Quando acabar de adicionar os pontos, clique com o direito do rato em qualquer sítio da área do mapa para confirmar a finalização da introdução da geometria desse elemento.

A janela de atributo aparecerá, permitindo que você insira as informações para a nova feição. [Figure_edit_2](#) mostra definir atributos para um novo rio fictício no Alasca. No `! menu de Digitalização` sob o `! menuselection: 'Configurações -> opções'` menu, você também pode ativar `! caixa !: guilabel: 'Repressão atributos janelas pop-up após cada feição'` criado e `! caixa !: guilabel: 'Reuse último entrou atributo Valor-`.

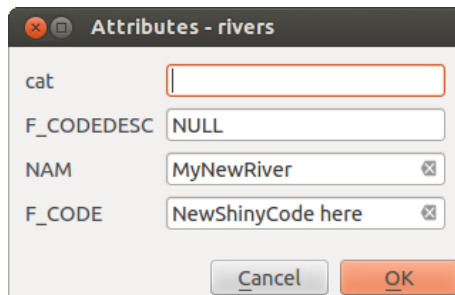






Figura 12.39: Introduza os Valores dos atributos na Janela após a digitalização do novo elemento vetorial 

Com o ícone  `! sup: Mover Feição (ões)` na barra de ferramentas, você pode mover as feições existentes.

Dica: Tipos de Valores de Atributo


Para a edição, os tipos de atributos são validados durante a entrada. Devido a isso, não é possível inserir um número em uma coluna de texto na caixa de diálogo: `! guilabel: Digite Atributo Valor-ou vice-versa. Se você precisar fazer isso, você deve editar os atributos em uma segunda etapa dentro de: ! guilabel: 'diálogo table atributo.`

Edições Atuais

Esta nova funcionalidade permite a digitalização de múltiplas camadas. Escolha  `! guilabel: 'Salve para Camada Seleccionada'` para salvar todas as alterações feitas em várias camadas. Você também tem a oportunidade de  `! guilabel: Reversão para camada seleccionada`, de forma que a digitalização pode ser retirada para todas as camadas seleccionadas. Se você quiser parar de editar as camadas seleccionadas,  `! guilabel: Cancelar para camada seleccionada (s)` é uma maneira fácil.


As mesmas funções estão disponíveis para a edição de todas as camadas do projeto.

Ferramenta de Nós


Para camadas baseadas em shapefile, bem como tabelas SpatialLite, PostgreSQL / PostGIS, MSSQL Spatial e Oracle Spatial, o  :sup: *Ferramenta de Nós* fornece capacidades de manipulação de vértices de feições semelhantes aos programas de CAD. É possível simplesmente selecionar vários vértices de uma vez e para se mover, adicionar ou excluí-los por completo. A ferramenta nó também trabalha com ‘on the fly’ projeção ligado, e ele suporta a edição topológica de feições. Esta ferramenta é, ao contrário de outras ferramentas no QGIS, persistente, por isso, quando alguma operação for feita, a seleção permanece ativo por essa ferramenta de feição. Se a ferramenta de nós não é capaz de encontrar todas as feições, um aviso será exibido.



É importante definir a propriedade: `menuselection: 'Configurações -> |mActionOptions|: menuselection: 'Opções -> Digitalização -> :guilabel:' Pesquisa Raio:` para um número maior que zero (isto é, 10). Caso contrário, QGIS não será capaz de dizer qual vértice está sendo editado.

Dica: Marcadores de Vértice

A versão atual do QGIS suporta três tipos de marcadores de vértice: ‘círculo semi-transparente’, ‘Cruz’ e ‘Nenhum’. Para alterar o estilo de marcador, escolha  :menuselection: *opções* do menu de :menuselection: *Configurações*, clique na guia :guilabel: *Digitalização* e selecione a entrada apropriada.


Operações Básicas

Comece por ativar a  Ferramenta de Nós e selecione um elemento clicando em cima de um, As caixas vermelhas irão aparecer em cada vértice deste elemento.

- **Selecionando vértices:** Você pode selecionar vértices, clicando sobre eles um de cada vez, clicando em uma borda para selecionar os vértices em ambas as extremidades, ou clicando e arrastando um retângulo em torno de alguns vértices. Quando um vértice é selecionado, sua cor muda para azul. Para adicionar mais vértices para a seleção atual, mantenha pressionada a :kbd: *tecla Ctrl* enquanto clica. Mantenha pressionado :kbd: *Ctrl* ou *Shift* ao clicar para alternar o estado de seleção de vértices (vértices que estão atualmente desmarcada serão selecionados como de costume, mas também vértices que já estão selecionados será desmarcada).
- **Adicionando vértices:** Para adicionar um vértice, basta clicar duas vezes perto de uma borda e um novo vértice aparecerá na borda perto do cursor. Note-se que o vértice aparecerá na borda, não na posição do cursor; por conseguinte, deve ser transferida, se necessário.
- **Eliminar vértices:** Depois de selecionar os vértices para eliminar, clique na tecla *Delete*. Note-se que não pode usar a  *Node Tool* para eliminar um elemento completo; QGIS vai assegurar que mantém o número mínimo de vértices necessários para o tipo de elemento que está a trabalhar. Para eliminar um elemento completo, utilize a ferramenta  *Delete Selected*.
- **Mover vértices:** Selecione todos os vértices que pretende mover. Clique num vértice selecionado ou na extremidade e arraste na direção que pretende mover. Todos os vértices selecionados serão movidos em conjunto. Se a ferramenta snapping está ativa, toda a seleção pode saltar para o vértice ou linha mais próximos.

Cada alteração feita com a ferramenta de nó é armazenado como uma entrada separada no diálogo Desfazer. Lembre-se que todas as operações de suporte à edição topológica quando este for ligado. Projeção on-the-fly também é suportada, e a ferramenta de nó fornece dicas para identificar um vértice ao passar o ponteiro do mouse sobre ele.




Cortando, Copiando e Colando Elementos

Feições selecionadas podem ser recortadas, copiadas e coladas entre camadas do mesmo projeto do QGIS contanto que a camada de destino estejam selecionadas como  :sup: Alternar edição previamente.

As feições também podem ser coladas para aplicações externas como texto. Ou seja, as feições são representadas no formato CSV, com os dados de geometria que aparece no formato de texto bem delimitado OGC (WKT).

No entanto, nesta versão do QGIS, feições de texto de fora do QGIS não pode ser colado a uma camada dentro de QGIS. Em que a função copiar e colar vir a calhar? Bem, acontece que você pode editar mais de uma camada de cada vez e funcionalidades copiar / colar entre as camadas. Por que iríamos querer fazer isso? Diga o que precisamos fazer algum trabalho em uma nova camada, mas só precisa de um ou dois lagos, e não a 5000 em nosso *camada big_lakes*. Podemos criar uma nova camada e use copiar / colar para plop os lagos necessários para ele.

Como exemplo, vamos copiar alguns lagos para uma nova camada:



1. Carregue a camada que quer copiar a partir (camada de origem)
2. Carregue ou crie a camada que quer copiar para (camada de destino)
3. Começar a editar a camada de destino
4. Ative a camada de origem clicando nela na legenda
5. Use a ferramenta  Seleccione Elemento Único para selecionar os elemento(s) na camada fonte
6. Clique na ferramenta  Copiar Elementos
7. Ative a camada de destino clicando na legenda
8. Clique na ferramenta  Colar Elementos
9. Parar a edição e salvar as alterações



O que acontece se as camadas de origem e de destino têm esquemas diferentes (nomes de campo e tipos não são a mesma coisa)? QGIS preenche o que corresponde e ignora o resto. Se você não se importa com os atributos que está sendo copiado para a camada de destino, não importa como você projeta os campos e tipos de dados. Se você quer ter certeza de tudo - a feição e seus atributos - é copiada, verifique se o jogo esquemas.

Dica: Congruência dos Elementos Colados


Se as camadas de fonte e de destino usar a mesma projecção, em seguida, as feições coladas terão geometria idêntica à da camada de fonte. No entanto, se a camada de destino é uma projeção diferente, então QGIS não pode garantir a geometria é idêntica. Isto é simplesmente porque existem erros pequenos arredondamentos envolvidos na conversão entre projeções.


Apagando os Elementos Selecionados

Se quisermos apagar o polígono todo, podemos faze-lo seleccionando o polígono usando a ferramenta  Seleccione Elemento Único. Pode selecionar múltiplos elementos para apagar. Uma vez feita a seleção do conjunto, use a ferramenta  Apagar Selecionados para apagar os elementos.

A ferramenta  :sup: *Cortar feições* na barra de ferramentas de digitalização também pode ser usado para excluir recursos. Isso exclui efetivamente a feição, mas também coloca-lo em uma “área de transferência espacial”. Então, cortamos a feição para excluir. Poderíamos, então, usar a ferramenta  :sup: *Colar Feições* para colocá-lo de volta, dando-nos a capacidade de desfazer de um nível. Recortar, copiar e colar o trabalho sobre as feições selecionadas no momento, o que significa que pode operar em mais de um de cada vez.

Salvando as Camadas Editadas

Quando uma camada está no modo de edição, as alterações permanecem na memória de QGIS. Portanto, eles não estão comprometidos / salvos imediatamente à fonte de dados ou disco. Se você quiser salvar as edições da camada atual, mas quero continuar editando sem sair do modo de edição, você pode clicar no botão  :sup:

Salvar Camadas Editadas. Quando você ativar o modo de edição off com  :sup: *editando Alternar* (ou sair QGIS para que o assunto), você também perguntado se você deseja salvar as alterações ou descartá-las.

Se as alterações não podem ser salvas (por exemplo, disco cheio, ou os atributos têm valores que estão fora de alcance), o estado em memória do QGIS é preservado. Isso permite que você ajuste as suas edições e tente novamente.

Dica: Integridade dos dados

É sempre boa ideia fazer cópias de segurança da sua fonte de dados antes de começar a editar. Enquanto os autores do QGIS fizeram todo o esforço para preservar a integridade dos seus dados, nós não oferecemos garantia neste sentido.

12.5.5 Digitalização Avançada



















Ícone	Finalidade	Ícone	Finalidade
	Retroceder		Retomar
	Rodar Elemento(s)		Simplificar elemento
	Adicionar Anel		Adicionar Parte
	Preenchimento Anel		Apagar Anel
	Apagar Parte		Refazer elementos
	Curva de Afastamento		Dividir Elementos
	Dividindo partes		Juntar Elementos Selecionados
	Juntar Atributos dos Elementos Selecionados		Rodar Símbolos de Pontos

Tabela de edição avançada: Barra de ferramentas de edição avançada de camadas vetoriais

Retroceder e Retomar

As ferramentas  :sup: *Desfazer* e  *Refazer* permitem desfazer ou refazer operações de edição de vetores. Há também uma tela acoplável, que mostra toda o histórico de todas operações desfazer / refazer (veja [Figure_edit_3](#)). Este elemento não é exibido por padrão; ele pode ser exibido com um clique direito na barra de ferramentas e ativando a opção Desfazer / Refazer. Desfazer / Refazer é no entanto ativo, mesmo que a tela não esteja sendo exibida.

Quando Desfazer é atingido, o estado de todas as feições e atributos são revertidos para o estado antes da reversão da operação acontecer. Outras mudanças para além das operações normais de edição vetorial (por exemplo, alterações feitas por um complemento), podem ou não podem serem revertidas, dependendo de como as mudanças foram realizadas.

Para usar a tela histórico de desfazer / refazer, basta clicar para selecionar uma operação na lista de histórico. Todas as feições serão revertidas para o estado em que estavam depois da operação selecionada.

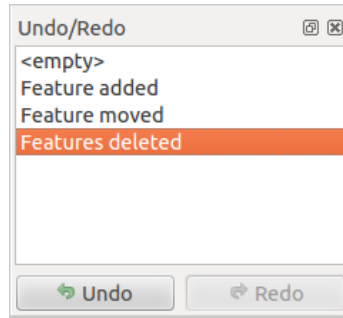






Figura 12.40: Refazer e desfazer passos de digitalização 🐧

Rodar Elemento(s)


Use  Rotação de feição(s) para rodar uma ou várias feições selecionadas na tela do mapa. Você primeiro precisa selecionar as feições e, em seguida, pressione o ícone  Rotação de feição(s). O centro de gravidade da feição(s) aparece e será o ponto de ancoragem da rotação. Se você tiver selecionado várias feições, o ponto de ancoragem da rotação será o centro comum das feições. Pressione e arraste o botão esquerdo do mouse na direção desejada para girar as feições selecionadas.

Também é possível criar um ponto de ancoragem de rotação definido pelo usuário em torno do qual a feição selecionada irá rodar. Selecione as feições para girar e ativar a ferramenta  :sup: Rotação de feição(s). Pressione e segure o botão :kbd: Ctrl e mova o ponteiro do mouse (sem pressionar o botão do mouse) para o local onde deseja que a âncora de rotação seja movida. Solte o botão Ctrl no ponto de ancoragem de rotação desejada. Agora, pressione e arraste o botão esquerdo do mouse na direção desejada para girar a função selecionada (s).

Simplificar elemento

A ferramenta  Simplificar feição permite reduzir o número de vértices de uma feição, desde que a geometria não mude. Primeiro, selecione uma feição. Ela será realçada por um elástico vermelho e uma barra aparecerá. Movendo o controle deslizante, a banda de borracha vermelha vai mudar a sua forma de mostrar como a feição está sendo simplificada. Clique [OK], para armazenar a nova geometria simplificada. Se uma feição não pode ser simplificada (por exemplo, multi-polígonos), uma mensagem aparecerá.



Adicionar Anel

Você pode criar **:índice: polígonos anéis** usando o ícone  Adicionar Anel na barra de ferramentas. Isto significa que no interior de uma área existente, é possível digitalizar outros polígonos que ocorrem como um “orifício”, de modo que apenas a zona entre as fronteiras dos polígonos exterior e interior continua sendo um anel poligonal.

Adicionar Parte


Você pode  adicionar parte no polígonos selecionado **:índice: multipolígono**. A nova parte do polígono deve ser digitadas fora do multi-polígono selecionado.

Preenchimento Anel


Você pode usar a função  Preenchimento Anel para adicionar um toque a um polígono e adicionar uma nova feição para a camada ao mesmo tempo. Assim, você não precisa usar primeiro o ícone  Adicionar Anel e então a função

a mais  Adicionar feição


Apagar Anel

A ferramenta | mActionDeleteRing : sup: *Excluir Anel* permite excluir polígonos anel dentro de uma área existente. Esta ferramenta só funciona com camadas de polígonos. Isso não muda nada quando ele é usado no anel externo do polígono. Esta ferramenta pode ser utilizada em polígonos e multi-funções poligonais. Antes de selecionar os vértices de um anel, ajuste a tolerância na edição do vértice.

Apagar Parte

A ferramenta | mActionDeletePart : sup: *Excluir Parte* permite apagar partes de multifeições (por exemplo, para excluir os polígonos de uma feição de multi-polígono). Não vai apagar a última parte da feição; esta última parte vai ficar intocada. Esta ferramenta funciona com todas as geometrias multi-parte: ponto, linha e polígono. Antes de selecionar os vértices de um papel, ajustar a tolerância de edição do vértice.




Refazer elementos





Você pode alterar feições de linha e polígono usando o ícone | mActionReshape : sup: *Remodelar Feições* na barra de ferramentas. Ele substitui a linha ou polígono parte da primeira para a última intersecção com a linha original. Com polígonos, às vezes isso pode levar a resultados indesejados. É útil principalmente para substituir partes menores de um polígono, não para grandes reparações, e a linha de remodelar não é permitido atravessar vários anéis de polígonos, pois isso geraria um polígono inválido.

Por exemplo, pode editar o limite de um polígono com esta ferramenta. Primeiro, clique na área interna do polígono junto do ponto onde pretende adicionar o novo vértice. Depois, atravesse o limite e adicione os vértices no exterior do polígono. Para finalizar, clique com o botão direito na área interna do polígono. A ferramenta vai adicionar um nó automaticamente no ponto onde a linha atravessa o limite. Também é possível remover parte da área do polígono, começa-se a nova linha no exterior do polígono, adicionam-se vértices no interior e termina-se a linha no exterior do polígono com um clique no botão direito.

Nota: A ferramenta de mudança de forma pode alterar a posição inicial de um anel ou de uma linha poligonal fechada. Assim, o ponto que está representada 'duas vezes' não será o mesmo mais. Isto pode não ser um problema para a maioria das aplicações, mas é algo a considerar.


Curvas de Afastamento

A ferramenta | mActionOffsetCurve : sup: *Deslocamento de curvas* cria cópias paralelas de camadas de linha. O ferramenta pode ser aplicada à camada editado (as geometrias são modificadas) ou também para as camadas de fundo (no caso em que se cria cópias das linhas / anéis e adiciona-los para a camada do editado). É, assim, ideal para a criação de camadas de linhas equidistantes. O deslocamento é mostrado no canto inferior esquerdo da barra de tarefas. Para criar uma mudança de uma camada de linha, você deve primeiro entrar em modo de edição e, em seguida, selecione a função. Você pode usar a ferramenta | mActionOffsetCurve : sup: 'Deslocamento de curvas' ativa e arraste a cruz para a distância desejada. As alterações podem ser salvas com a ferramenta | mActionSaveEdits : sup: *Salvar Edições da Camada*.


Para criar um deslocamento, de uma camada de linha, você precisa primeiro entrar no modo de edição desta, então selecionar a feição desejada. Com a ferramenta  Deslocamento de Curva  ativa clique sobre a linha desejada segure o botão, mova a cruz sobre o local desejado e solte o botão do mouse. Para que as mudanças sejam salvas clique na ferramenta  Salvar Edição da Camada .

QGIS opções diálogo (guia Digitalização então **Ferramentas de compensar Curva** seção) permite configurar alguns parâmetros como **Juntar estilo Segmentos no Quadrante**, **Limite Mitra**.


Dividir Elementos

Pode dividir elementos usando o ícone  Dividir Elementos da barra de ferramentas. Apenas desenhe uma linha ao longo do elemento que quer dividir.


Dividindo partes

No QGIS 2.0, é possível dividir as partes de uma feição de multi parte de modo a que o número de peças é aumentado. Basta desenhar uma linha em toda a parte que você quer dividir com o ícone  Dividir Partes.


Juntar elementos selecionados

A ferramenta  Mesclar Feições Selecionadas permite a você mesclar feições que tem bordas em comum. Uma nova janela de diálogo permitirá a você escolher quais características serão mantidas para a nova feição criada com base nas demais mescladas, ou poderá ser usada uma função de (Mínimo, Máximo, Média, Soma, Pular Atributo), usada em cada coluna da nova feição com base nas feições de origem.

Juntar os atributos dos elementos selecionados

A ferramenta  :sup: *Mesclar Atributos das Feições* escolhidas permite **:índice:‘mesclar atributos de feições‘** com fronteiras comuns fundir e atributos sem a fusão de suas fronteiras. Primeiro, selecione várias feições ao mesmo tempo. Em seguida, pressione o botão `mActionMergeFeatureAttributes` | **l: sup: ‘Mesclar Atributos de feições selecionadas‘**. **Agora lqgl** pede-lhe os atributos que devem ser aplicados a todos os objetos selecionados. Como resultado, todos os objetos selecionados têm as mesmas entradas de atributos.

Rodar Símbolos de Pontos

 Rotate Point Symbols allows you to change the rotation of point symbols in the map canvas. You must first define a rotation column from the attribute table of the point layer in the *Advanced* menu of the *Style* menu of the *Layer Properties*. Also, you will need to go into the ‘SVG marker’ and choose *Data defined properties* Activate *Angle* and choose ‘rotation’ as field. Without these settings, the tool is inactive.

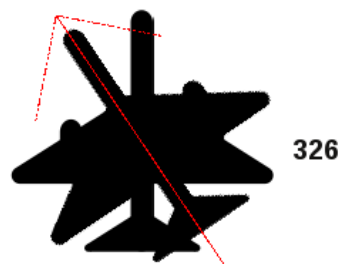



Figura 12.41: Rodar Símbolos de Pontos 

Para alterar a rotação, selecione um recurso ponto na tela do mapa e gire-o, segurando o botão esquerdo do mouse pressionado. A seta vermelha com o valor de rotação será visualizado (ver [Figura_editada_4](#)). Quando você soltar o botão esquerdo do mouse novamente, o valor será atualizado na tabela de atributos.

Nota: Se segurar a tecla `Ctrl` premido, a rotação irá ser feita em passos de 15 graus.

12.5.6 Criando novas camadas Vetoriais

QGIS allows you to create new shapefile layers, new SpatialLite layers, and new GPX layers. Creation of a new GRASS layer is supported within the GRASS plugin. Please refer to section [Criando uma nova camada vetorial GRASS](#) for more information on creating GRASS vector layers.

Criando uma nova camada Shapefile

Para criar uma nova camada de forma para edição, escolha: seleção de menu: *Novo* -> | mActionNewVectorLayer |: seleção de menu *Novo Shapefile Camada ...* do: menu de seleção: *menu de Camada*. A: guilabel: *diálogo Novo Vetor de Camada* será exibido como mostrado na [Figure_editada_5](#). Escolha o tipo de camada (ponto, linha ou polígono) e CRS (sistema de coordenadas de referência).

Note-se que o QGIS ainda não suporta a criação de feições 2.5D (ou seja, com características X, Y, Z).

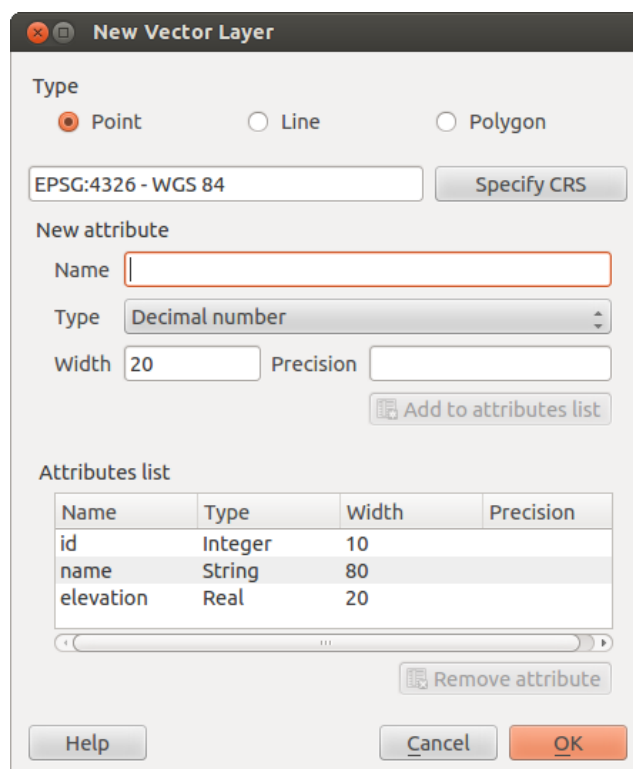



Figura 12.42: Janela de criação de uma nova camada Shapefile 🐧

Para concluir a criação da nova camada arquivo shape, adicione os atributos desejados clicando no botão **[Adicionar à lista de atributos]** e especificar um nome e tipo para o atributo. A primeira coluna ‘código’ é adicionado como padrão, mas pode ser removido, se não quiser. Apenas: guilabel: *Tipo: real* | String |,; guilabel: ‘*Tipo: inteiro*’ | String |,; guilabel: *Tipo: string* | String | e; guilabel: ‘*Tipo: data*’ | String | atributos são suportados. Além disso e de acordo com o tipo de atributo, você também pode definir a largura e a precisão da nova coluna de atributo. Quando estiver satisfeito com os atributos, clique **[OK]** e forneça um nome para o arquivo shape. | QG | irá adicionar automaticamente um: arquivo: *.shp* extensão para o nome que você especificar. Uma vez que a camada foi criado, ele será adicionado ao mapa, e você pode editá-lo da mesma forma como descrito na seção: ref: [sec_edit_existing_layer](#) acima.

Criando uma nova camada SpatialLite

Para criar uma nova camada SpatialLite para edição, escolha *Novo* →  *Nova Camada SpatialLite...* do menu *Camada*. O diálogo *Nova Camada SpatialLite* irá aparecer como é mostrado na [Figure_edit_6](#).

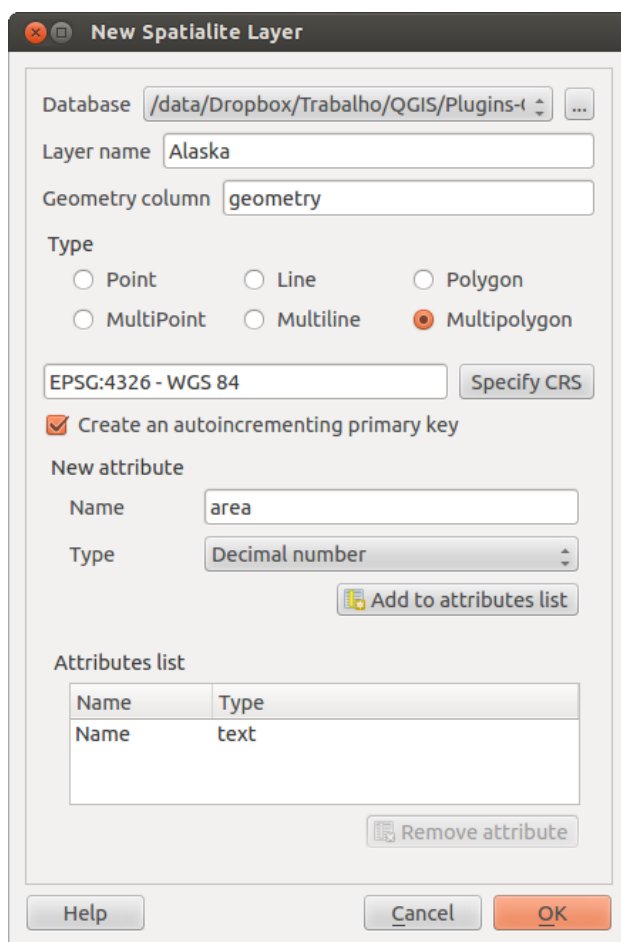



Figura 12.43: Janela de criação de uma nova camada SpatialLite 

A primeira etapa consiste em selecionar um banco de dados Spatialite existente ou criar uma nova base de dados Spatialite. Isso pode ser feito com o botão de navegação | browsebutton | à direita do campo de banco de dados. Em seguida, adicione um nome para a nova camada, definir o tipo de camada, e especificar o sistema de referência de coordenadas com ** [Especificar CRS] **. Se desejar, você pode selecionar | caixa | : guilabel: 'Criar uma chave' primária auto incremental.

Para definir uma tabela de atributos para a nova camada Spatialite, adicione os nomes das colunas de atributos que você deseja criar com o tipo de coluna correspondente e clique no botão ** [Adicionar à lista de atributos] **. Quando estiver satisfeito com os atributos, clique ** [OK] **. | QG | adiciona automaticamente a nova camada com a legenda, e você pode editá-lo da mesma forma como descrito na seção: ref: *sec_edit_existing_layer* acima.

O tratamento adicional das camadas Spatialite pode ser feito com o Gerenciador BD. Veja *Complemento Gerenciador BD*.

Criando uma nova camada GPX

Para criar um novo arquivo GPX, você precisa carregar o plugin GPS em primeiro lugar. : menu de seleção: *Plugins* -> | mActionShowPluginManager | : menu de seleção: 'Gerenciador de Plugin ...' abre o Gerenciador de diálogo de Plugin. Ative a | caixa | : guilabel: 'Ferramentas GPS'.

Quando este plugin é carregado, escolha: menu de seleção: *Novo* -> | icon_newgpx | : menu de seleção: 'Criar nova camada GPX ... do: menu de seleção: menu de Camada'. No: guilabel: 'Salvar como novo arquivo GPX', você pode escolher onde deseja salvar a nova camada GPX.

12.5.7 Trabalhando com a Tabela de Atributos

O: índice: *atributo da tabela* exibe características de uma camada selecionada. Cada linha na tabela representa um recurso de mapa, e cada coluna contém uma determinada parte das informações sobre o recurso. Recursos na tabela pode ser pesquisado, selecionado, movido ou mesmo editadas.

Para abrir a tabela de atributos de uma camada do vetor, fazer a camada ativa, clicando sobre ela na área do mapa legenda. Então, a partir do principal: menu de seleção: *menu de Camada*, escolha | mActionOpenTable | : menu de seleção: 'Abrir Tabela de Atributos'. Também é possível fazer um clique direto sobre a camada e escolha | mActionOpenTable | : menu de seleção: 'Abrir Tabela de Atributos' a partir do menu suspenso, e clique no | mActionOpenTable | : guilabel: *botão Abrir Tabela de Atributos* na barra de ferramentas.

Isto irá abrir uma nova janela que exibe os atributos do recurso para a camada (*figura_atributos_1*). O número de recursos e o número de recursos selecionados são mostrados no título da tabela de atributos.

	ID	NAME_1	NAME_2	HASC_2	TYPE_2
0	1	Alaska	Aleutians E...	US.AK.AE	Borough
1	2	Alaska	Aleutians ...	US.AK.AW	Census Area
2	3	Alaska	Anchorage	US.AK.AN	Municipality
3	4	Alaska	Bethel	US.AK.BE	Census Area
4	5	Alaska	Bristol Bay	US.AK.BR	Borough
5	6	Alaska	Denali	US.AK.DE	Borough
6	7	Alaska	Dillingham	US.AK.DI	Census Area
7	8	Alaska	Fairbanks N...	US.AK.FA	Borough
8	9	Alaska	Haines	US.AK.HA	Borough

Figura 12.44: Tabela de Atributos para a camada regiões 🐧

Selecionando elementos na tabela de atributos

Cada linha selecionada na tabela de atributos representa os atributos de um determinado elemento da camada. Se o conjunto de elementos selecionados na janela principal é alterado, a seleção também é atualizada na tabela de atributos. Da mesma forma, se um conjunto de linhas selecionadas na tabela de atributos é modificada, o conjunto de elementos selecionado na janela principal será atualizado.

As linhas podem ser selecionadas ao clicar no número da linha, do lado esquerdo desta. Podem ser marcadas **Linhas múltiplas** manter a tecla `Ctrl` primida. Pode ser feita uma **seleção contínua** se manter primida a tecla `Shift` e clicar na entrada de várias linha, do lado esquerdo. Todas as linhas entre a posição atual do cursor e a linha clicada serão selecionados. Ao mover a posição do cursor na tabela de atributos, ao clicar um célula na tabela, não tem influência na seleção de linhas. Modificar a seleção na tela principal não altera a posição do cursor na tabela de atributos.

A tabela pode ser ordenada por qualquer coluna, clicando no cabeçalho da coluna. Uma pequena seta indica a forma de ordenação (apontar para baixo significa valores descendentes do topo da linha, apontar para cima significa valores ascendentes do topo da linha).

Para uma **simples busca por atributos** em apenas uma coluna, escolha o: menu de seleção: *filtrar Coluna* -> a partir do menu no canto inferior esquerdo. Selecione o campo (coluna) em que a pesquisa deve ser realizada a partir do menu suspenso, e acertar o botão [**Aplicar**]. Então, apenas os recursos relacionados são mostrados na tabela de atributos.

Para fazer uma seleção, você tem que usar o `mIconExpressionSelect` `! sup: SELECIONAR recursos usando o ícone Expressão` no topo da tabela de atributos. `mIconExpressionSelect` `! sup: SELECIONAR recursos usando uma Expressão` permite definir um subconjunto de uma tabela usando um: `guiabel: Lista de Funções` como no `mActionCalculateField` `! sup: Campo Calculadora` (veja: ref: *vetor_campo_calculadora*). Os resultados da consulta podem ser salvos como uma nova camada vetorial. Por exemplo, se você quiser encontrar regiões que são bairros de: arquivo: *regiões.shp* do `QG` | dados de exemplo, você tem que abrir o: `guiabel: Campos e Valor-menu` e escolher o campo que você deseja consultar. Clique duas vezes no campo 'TIPO_2' e também `** [Carregar todos os valores únicos] **`. A partir da lista, selecione e clique duas vezes em Cidade. No: `guiabel: campo Expressão`, a consulta a seguir será exibida








```
"TYPE_2" = 'Borough'
```






Aqui você também pode usar o: menu de seleção: *lista de Funções* -> *recente (Seleção)* para fazer uma seleção que você usou antes. O construtor de expressão se lembra dos últimos 20 expressões usadas.


As linhas correspondentes serão selecionados, e o número total de linhas correspondentes será exibida na barra de título da tabela de atributos, bem como na barra de status da janela principal. Para pesquisas que apresentam características só selecionadas no mapa, use o Construtor de Consultas descrito na seção: ref: *vetor_consultas_construtor*.

Para mostrar apenas os registros selecionados, use *Mostrar feições selecionadas* a partir do menu na parte inferior esquerda.

Os outros botões na parte superior da janela de tabela de atributos fornecem as seguintes funcionalidades:

-  Alternar o modo de edição para editar valores individuais e permitir funcionalidades descritas abaixo (também com `Ctrl+E`)
-  Salvar Editados (também com `Ctrl+S`)
-  Deselecionar todos! (também com `Ctrl+U`)
-  Mover seleção para o topo (também com `Ctrl+T`)
-  Inverter seleção (também com `Ctrl+R`)
-  Copiar linhas selecionadas para área de transferência (também com `Ctrl+C`)
-  Zoom no mapa para linhas selecionadas (também com `Ctrl+J`)

-  Pan no mapa para linhas selecionadas (também com `Ctrl+P`)
-  Deletar feição selecionada (também com `Ctrl+D`)
-  Nova Coluna para camadas PostGIS e para camadas OGR com versão GDAL \geq 1.6 (também com `Ctrl+W`)
-  Deletar Coluna para camadas PostGIS e para camadas OGR com versão \geq 1.9 (também com `Ctrl+L`)
-  Abrir calculadora de campo (também com `Ctrl+I`)

Abaixo destes botões está a barra Calculadora de Campo, que permite que cálculos sejam aplicados rapidamente a atributos visíveis na tabela. Esta barra utiliza as mesmas expressões da  Calculadora de Campo (veja *Calculadora de Campos em Vetores*).

Dica: Ignorar geometria WKT

Se você quiser usar dados de atributos em programas externos (como Excel), use o `mActionCopySelected` **I**: *sup: Copiar as linhas selecionadas para o botão clique. Você pode copiar as informações sem geometrias vetoriais se você desativar: menu de seleção: 'Configurações -> Opções -> Fontes de dados do menu | caixa I: guilabel: Copiar geometria na representação WKT de atributo tabela.*

Salvar elementos selecionados como nova camada


As feições selecionadas podem ser salvas em qualquer formato vetorial OGR-suportado e também transformado em um outro sistema de referência de coordenadas (SRC). Basta abrir o menu do botão direito do mouse da camada e clique em *Salvar como* para definir o nome do arquivo de saída, seu formato e SRC (ver seção *Legenda do Mapa*). Para salvar a seleção garantir que a **caixa** *salvar apenas feição selecionada* esteja selecionada. Também é possível especificar opções de criação de OGR no diálogo.

Colar dentro de nova camada

Recursos que estão na área de transferência pode ser colado em uma nova camada. Para fazer isso, primeiro faça uma camada editável. Selecione algumas características, copiá-los para a área de transferência e colá-los em uma nova camada usando: menu de seleção: *Editar -> Colar Características* e escolher: menu de seleção: 'Nova Camada' no vetor ou: menu de seleção: *Nova Camada* na memória .

Isso se aplica a recursos selecionados e copiados dentro | QG | e também para recursos de outra fonte definida de acordo com o texto conhecido (WKT).

Trabalhando com tabelas de atributos não espaciais

QGIS allows you also to load non-spatial tables. This currently includes tables supported by OGR and delimited text, as well as the PostgreSQL, MSSQL and Oracle provider. The tables can be used for field lookups or just generally browsed and edited using the table view. When you load the table, you will see it in the legend field. It can be opened with the  Open Attribute Table tool and is then editable like any other layer attribute table.

Como exemplo, você pode usar colunas da tabelas não espaciais para definir valores de atributos, ou um intervalo de valores que são permitidos, para ser adicionado a uma camada de vetor específico durante a digitalização. Dê uma olhada mais de perto o widget de edição na seção: ref: *vetor_atributos_menu* para saber mais.

12.5.8 Criando uma ou mais relações

As relações são uma técnica frequentemente utilizada em bancos de dados. O conceito é, que as características (linhas) de diferentes camadas (tabelas) podem pertencer a um ao outro.

Como um exemplo, você tem uma camada com todas as regiões do Alasca (polígono), que fornece alguns atributos sobre o seu nome e tipo de região e um código único (que atua como chave primária).

Chaves Externas

Então você começa uma outra camada de ponto ou tabela com informações sobre aeroportos que estão localizados nas regiões e também querem manter o controle deles. Se você deseja adicioná-los à camada região, você precisa criar uma relação de um para muitos usando chaves estrangeiras, porque existem vários aeroportos na maioria das regiões.



Figura 12.45: Região Alasca com aeroportos 🐧

Além dos atributos já existentes na tabela de atributos de airport outro campo `fk_region` que atua como uma chave estrangeira (se você tiver um banco de dados, você provavelmente vai querer definir uma restrição sobre ele).

Este campo `fk_região` sempre conterá um código de uma região. Pode ser visto como um indicador para a região a que pertence. E você pode criar um formulário de edição personalizada para a edição e QGIS se preocupa com a configuração. Ele trabalha com diversos fornecedores (para que você também pode usá-lo com os arquivos de vários formatos e csv) e tudo o que você tem a fazer é as relações entre as tabelas QGIS.

Camadas

QGIS não faz diferença entre uma tabela e uma camada vetorial. Basicamente, uma camada de vetor é uma tabela com uma geometria. Então, pode adicionar sua tabela como uma camada vetorial. Para demonstrar você pode carregar o arquivo shape “região” (com geometrias) e a tabela ‘aeroporto’ csv (sem geometrias) e uma chave estrangeira (`fk_região`) para a região da camada. Isto significa que cada aeroporto pertence a exatamente uma região, enquanto cada região pode ter qualquer número de aeroportos (a típica relação de um para muitos).

Definição (Gerenciador de relação)

A primeira coisa que vamos fazer é deixar QGIS saber sobre as relações entre a camada. Isso é feito em: menu de seleção: ‘Configurações -> : *menu de seleção*: Projeto Propriedades’. Abra o: guilabel: *menu de Relações* e clique em: guilabel: ‘Adicionar’.

- **nome** vai ser utilizado como um título. Deve ser um texto legível, descrevendo, para que a relação é utilizada. Vamos apenas dizer chamada “Airports” neste caso.
- **camada de referência** é o único com o campo de chave estrangeira sobre ele. No nosso caso, esta é a camada airports

- **campo referência** dirá, quais os pontos de campo para a outra camada de modo que este será `fk_region` neste caso
- **camada referenciada** é a única com a chave primária, apontada, então aqui está a camada de regiões
- **campo referenciado** é a chave primária da camada referenciada por isso é ID
- ****código**** será usado para fins internos e tem de ser único. Você pode precisar dele para criar formulários personalizados uma vez que este é suportado. Se você deixá-lo vazio, será gerada para você, mas você pode atribuir um para obter que é mais fácil de manusear.

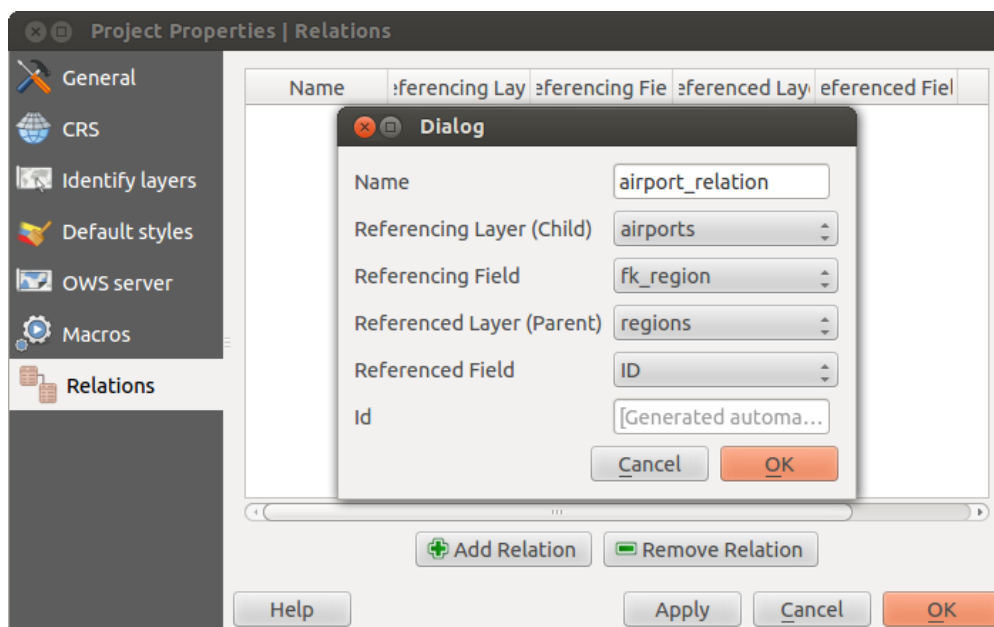



Figura 12.46: Gerenciador de relações 🐧

Formas

Agora que QGIS conhece a relação, ele vai ser usado para melhorar as formas que são gerados. Como nós não alteramos o método de formulário padrão (gerada automaticamente) ele vai apenas adicionar um novo widget em nosso formulário. Então, vamos selecionar a região de camada na legenda e use a ferramenta para identificar. Dependendo das configurações, o formulário pode ser aberto diretamente ou você terá que optar por abri-lo na janela de identificação no âmbito de ações.

Como você pode ver, os aeroportos atribuídos a esta região em particular são todos mostrados em uma tabela. E há também alguns botões disponíveis. Vamos revê-los em breve

- O botão `mActionToggleEditing` serve para alternar o modo de edição. Esteja ciente de que ele alterna o modo de edição da camada de aeroporto, apesar de estarmos na forma de uma característica da camada região. Mas o quadro está representando características da camada aeroporto.
- O botão `mActionSignPlus` irá adicionar um novo recurso para a camada de aeroporto. E vai atribuir um novo aeroporto para a região atual por padrão.
- O botão  apagará o aeroporto selecionado permanentemente.
- O símbolo `mActionLink` irá abrir uma nova janela onde você pode selecionar qualquer aeroporto existente, que será, então, atribuído à região atual. Isto pode ser útil se você criou o aeroporto na região errada por acidente.
- O símbolo `mActionUnlink` desvincula o aeroporto selecionado da região atual, deixando-os não atribuído (a chave estrangeira é definida como NULO) de forma eficaz.

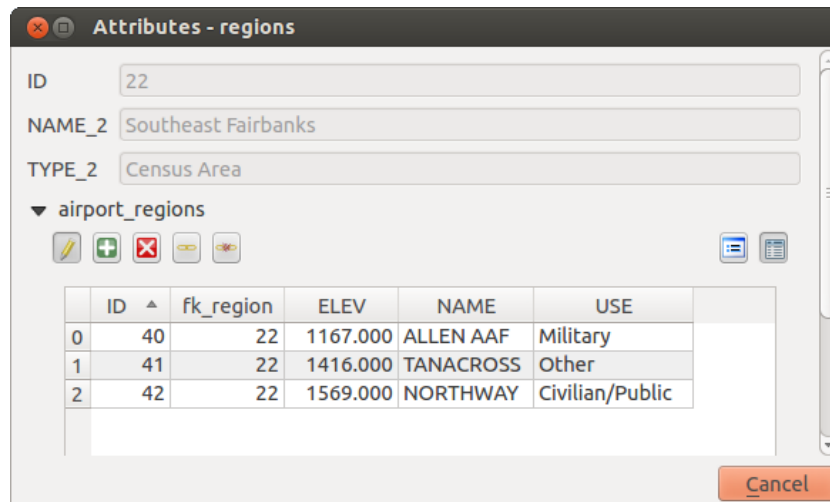


Figura 12.47: Diálogo identificação regions com relação a airports 🐧

- Os dois botões para a chave de visualização da tabela e formulário, onde o deixou-nos mais tarde, você vê todos os aeroportos no seu respectivo formulário.

Se você trabalha na tabela aeroporto, um novo tipo de widget está disponível que permite que você incorpore a forma característica da região referenciada na forma característica dos aeroportos. Ele pode ser usado quando você abre as propriedades da camada da tabela de aeroportos, mude para o: menu de seleção: *menu de campos* e mudar o tipo de widget do campo de chave estrangeira 'fk_região' a relação de referência.

Se você olhar para a caixa de diálogo agora, você vai ver, que a forma de a região estar inserida dentro do formulário aeroportos e até mesmo ter um caixa de combinações, que permite atribuir o atual aeroporto para outra região.

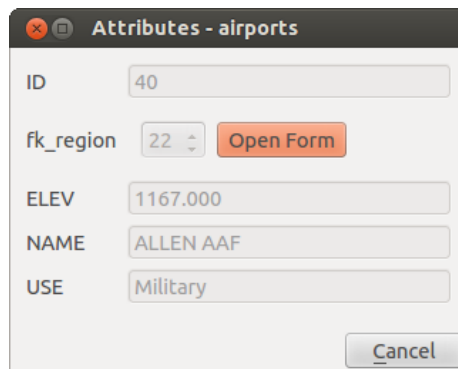


Figura 12.48: Diálogo identificação aeroportos com relação a regiões 🐧

12.6 Ferramenta de Consulta

A ferramenta de consulta permite que você defina um subconjunto de uma tabela utilizando uma condição SQL como a cláusula WHERE e mostrar o resultado na tela principal. O resultado da consulta pode ser salvo como uma nova camada vetorial.

12.6.1 Consulta

Abra a **Ferramenta de Consulta** abrindo as propriedades da camada e no menu escolha a guia *Geral*. Abaixo da guia [subdivisão de feição], clique no botão **Ferramenta de Consulta** para abrir a janela *Ferramenta de Consulta*. Por exemplo, se você tem uma camada de regiões com um campo TYPE_2, você pode selecionar apenas as regiões que são bairros na caixa *Fornecedor de expressão de filtragem específica* da Ferramenta de Consulta. A *Figure_attributes_2* mostra um exemplo da construção da Ferramenta de Consulta com a camada *regions.shp* do dados de exemplo do QGIS. As seções dos campos, valores e Operadores ajudam você a contruir uma consulta SQL.

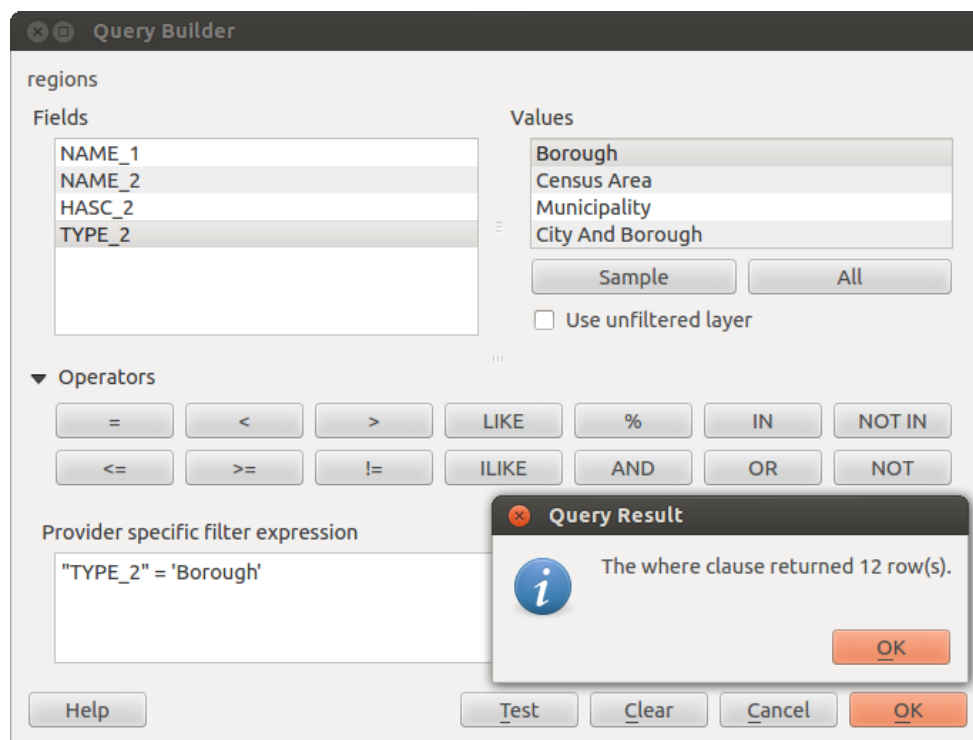


Figura 12.49: Ferramenta de Consulta

A **Lista campos** contém todas as colunas de atributo da tabela de atributos a serem pesquisadas. Para adicionar uma coluna de atributo para o campo cláusula do SQL WHERE, clique duas vezes em seu nome na lista Campos. Geralmente, você pode usar os vários campos, valores e operadores para construir a consulta, ou você pode simplesmente digitá-lo na caixa de SQL.

A **Lista valores** lista os valores de uma tabela de atributos. Para listar todos os valores possíveis de um atributo, selecione o atributo na lista Campos e clique no botão **[todos]**. Para listar os primeiros 25 valores exclusivos de uma coluna de atributo, selecione a coluna atributo na lista Campos e clique no botão **[Simples]**. Para adicionar um valor para o campo SQL WHERE cláusula, clique duas vezes em seu nome na lista de valores.


A **Seção Operadores** contém todos os operadores utilizáveis. Para adicionar um operador para o campo cláusula SQL WHERE, clique no botão apropriado. Os operadores relacionais (=, >, ...), operador de comparação de texto (como), e os operadores lógicos (e, ou, ...) estão disponíveis.

O botão **[Teste]** mostra uma caixa de mensagem com o número de feições que satisfazem a consulta atual, que é útil no processo de construção de consulta. O botão **[Limpar]** limpa o texto no campo de texto da cláusula SQL WHERE. O botão **[OK]** fecha a janela e seleciona as feições que satisfazem a consulta. O botão **[Cancelar]** fecha a janela sem alterar a seleção atual.

QGIS trata os atos de subconjunto resultantes como se onde toda a camada. Por exemplo, se você aplicou o filtro acima para 'Bairro', você não pode exibir, consultar, salvar ou editar o Município de Anorage, porque isso é um 'Município' e, portanto, não faz parte do subconjunto.

A única exceção é que, a menos que sua camada faz parte de um banco de dados, utilizando um subconjunto irá evitar que edite a camada.

12.7 Calculadora de Campo

O botão  Calculadora de Campo na tabela de atributos permite-lhe executar cálculos com base em valores de atributos ou funções existentes definidas, por exemplo, para calcular o comprimento ou área da feição geométrica. Os resultados podem ser colocados em uma nova coluna para o atributo, um campo virtual, ou serem usados para atualizar os valores de uma coluna já existente.

Dica: Campo Virtual

- Campos virtuais não são permanentes e não são salvos.
- Para fazer com que um campo virtual que será feito quando o campo é feito.

A calculadora de campo já está disponível em qualquer camada que suporta edição. Quando você clica no ícone da calculadora de campo o diálogo é aberto (ver [figure_attributes_3](#)). Se a camada não estiver em modo de edição, será exibido um aviso e usando a calculadora de campo fará com que a camada seja colocada em modo de edição antes do cálculo ser feito.




A barra da calculadora de campo rápida no topo da tabela de atributo é visível somente se a camada estiver editável.

Rápidamente na barra da calculadora de campo, você primeiro seleciona o nome do campo existente, em seguida, abre o diálogo de expressão para criar a sua expressão ou escrevê-la diretamente no campo, em seguida, clique no botão **Atualizar tudo**.

Na caixa de diálogo da calculadora de campo, primeiro você deve selecionar se deseja apenas atualizar feições selecionadas, crie um novo campo de atributo onde serão adicionados os resultados do cálculo ou atualize um campo existente.

Se optar por adicionar um novo campo, você precisa digitar um nome do campo, o tipo do campo (inteiro, real ou texto), a largura total do campo, e a precisão do campo (ver [figure_attributes_3](#)). Por exemplo, se você escolher uma largura de campo de 10 e uma precisão de 3, significa que você tem 6 números antes do ponto, depois o ponto em si e por último mais 3 dígitos indicando a precisão.

Um breve exemplo ilustra como a calculadora de campo funciona. Nós queremos calcular o comprimento em km na camada `railroads` do conjunto de dados amostra do QGIS:

1. Carregue a Shapefile `railroads.shp` no QGIS e pressione  Abrir Tabela de Atributos .
2. Clique no  Alternar o modo de edição e abra a janela da  Calculadora de Campo .
3. Seleciona a caixa de verificação *Criar novo campo* para salvar os cálculos em um novo campo.
4. Adicione `comprimento` como campo de saída do nome, `real` como o tipo de campo de saída e defina o comprimento do campo de saída de 10 e Precisão 3.
5. agora de duplo clique na função `$length` no grupo *Geometria* para adiciona-lo à caixa de expressões da Calculadora de campo.
6. Complete a expressão introduzindo `"/ 1000"` na caixa de expressões da Calculadora de campo e clique [OK].
7. Agora você pode encontrar um novo campo `length` na tabela de atributos.

As funções disponíveis são listadas no capítulo [Expressões](#).

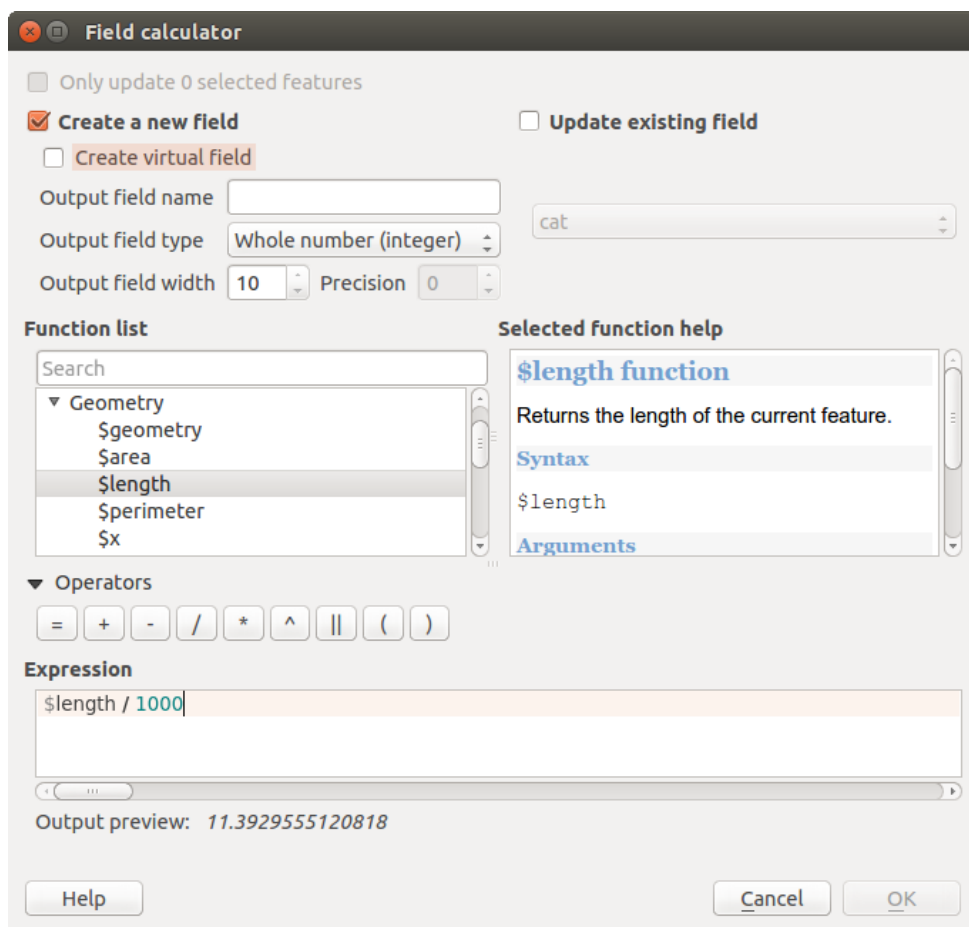


Figura 12.50: Calculadora de Campo 🐧

Trabalhando com Dados Raster

13.1 Trabalhando com dados raster

Esta seção descreve como visualizar e definir as propriedades da camada raster. QGIS usa a biblioteca GDAL para ler e gravar os formatos de dados raster, incluindo Grid ArcInfo Binary, Grid ArcInfo ASCII, GeoTIFF, ERDAS IMAGINE, e muitos mais. Apoio raster GRASS é fornecido por um complemento nativo do provedor de dados do QGIS. Os dados raster também podem ser carregados no modo de leitura de arquivos ZIP e GZIP no QGIS.

A partir da data deste documento, mais de 100 formatos raster são suportados pela biblioteca GDAL (ver GDAL-SOFTWARE-SUITE em *Referências Bibliográficas e Web*). A lista completa está disponível em http://www.gdal.org/formats_list.html.

Nota: Nem todos os formatos mencionados podem trabalhar no QGIS por várias razões. Por exemplo, alguns exigem bibliotecas comerciais externas, ou a instalação de seu sistema operacional GDAL pode não ter sido construída para suportar o formato que você deseja usar. Apenas os formatos que foram bem testados irão aparecer na lista de tipos de arquivos ao carregar um raster no QGIS. Outros formatos não testados podem ser carregados, selecionando o filtro [GDAL] Todos os arquivos (*).

Trabalhando com dados raster GRASS é descrita na seção *Integração com SIG GRASS*.

13.1.1 O que são dados raster?

Dados raster em SIG são matrizes de células distintas que representam feições sobre, acima ou abaixo da superfície da terra. Cada célula da grade raster é do mesmo tamanho, e as células são geralmente retangular (no QGIS elas vão sempre ser retangulares). Conjuntos de dados raster típicos incluem dados de sensoriamento remoto, como fotografias aéreas ou imagens de satélite e dados modelados, como uma matriz de elevação.

Ao contrário dos dados vetoriais, dados raster, normalmente, não têm um registro de banco de dados associado a cada célula. Eles estão georreferenciados por pixels de resolução e coordenadas x/y de um pixel de canto da camada raster. Isso permite o QGIS posicionar os dados corretamente na tela do mapa.

QGIS faz uso de informações de georreferenciamento dentro da camada raster (por exemplo,: índice: *GeoTiff*) ou em um arquivo mundo apropriado para exibir corretamente os dados.

13.1.2 Carregando dados raster no QGIS

Camadas raster são carregadas ou clicando no ícone  Adicionar Camada Raster ou selecionando a opção de menu *Camada* →  Adicionar Camada Raster. Mais de uma camada podem ser carregadas ao mesmo tempo mantendo

a tecla `Ctrl` ou `Shift` pressionada e clicando em vários itens na janela *Abrir uma fonte de dados Raster GDAL suportada*.

Uma vez que uma camada raster é carregada na legenda do mapa, você pode clicar sobre o nome da camada com o botão direito do mouse para selecionar e ativar feições específicas da camada ou para abrir um diálogo para definir as propriedades raster para a camada.

Menus do Botão direito do mouse para camadas raster

- *Zoom para extensão da camada*
- *Zoom para melhor escala (100%)*
- *Estenda usando a extensão atual*
- *Mostrar em visão geral*
- *Remover*
- *Duplicado*
- *Definir SRC da camada*
- *Definir SRC do projeto para camada*
- *Salvar como ...*
- *Propriedades*
- *Renomear*
- *Copiar Estilo*
- *Adicionar novo Grupo*
- *Expandir tudo*
- *Fechar tudo*
- *Atualização da ordem do desenho*

!atualizadireitos!

13.2 Diálogo de propriedades do Raster

Para visualizar e definir as propriedades da camada de um layer, dê um duplo clique no nome da camada na legenda do mapa, ou clique com botão direito no nome da camada e escolha: *Propriedades* a partir do menu de contexto. Isto vai abrir o diálogo `:guilabel: 'Propriedades da camada Raster '` (ver [figura_raster_1](#)).

Existem vários menus na janela de dialogo:

- *Geral*
- *Estilo*
- *Transparência*
- *Pirâmides*
- *Histograma*
- *Metadados*

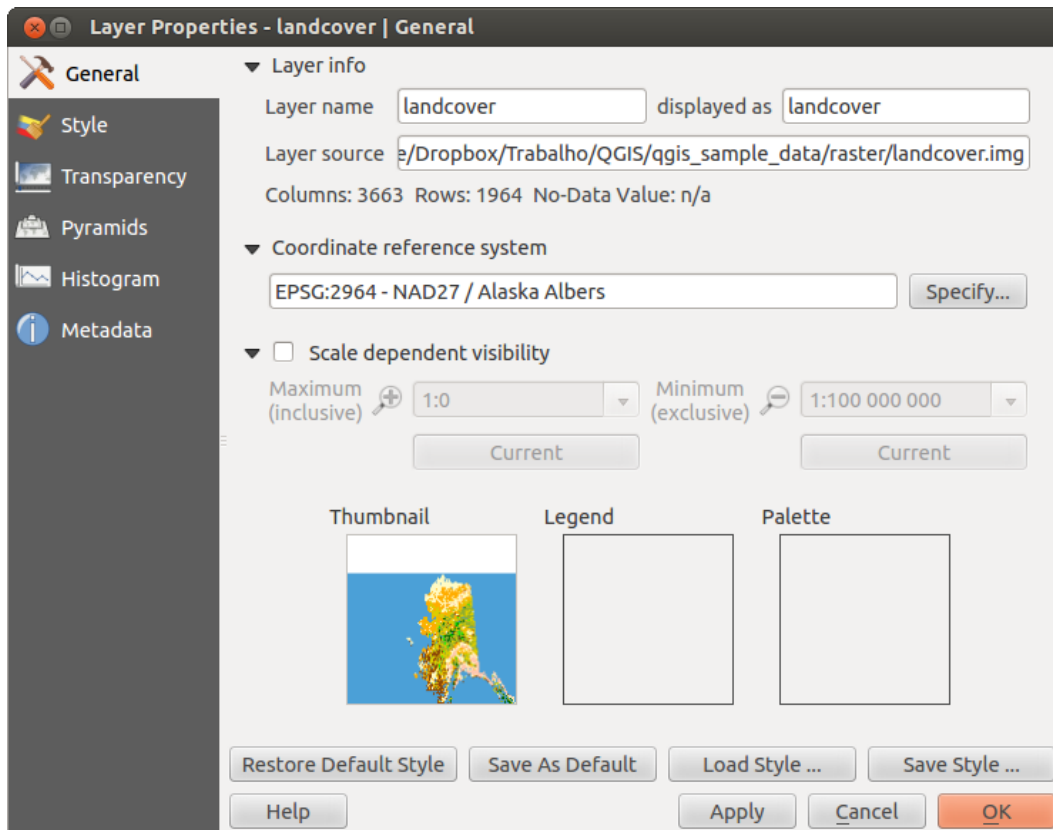


Figura 13.1: Diálogo de Propriedades das Camadas do Raster

13.2.1 Menu Geral

Informação da camada

O menu *Geral* apresenta informação básica do raster selecionado, incluindo o caminho da camada origem, o nome que aparece na legenda (que pode ser modificado), e o número de colunas, linhas e valores nulos do raster.

Sistema de Referência de Coordenadas

Aqui encontra-se a informação do Sistema de Referência de Coordenadas (SRC), impressa na linha PROJ.4. Se esta definição não é a correta, pode-se modificar, clicando no botão **[Especifique]**

Visibilidade dependente da escala

Também nesta aba pode acertar a visibilidade dependente da escala. Será necessária clicar a caixa de checagem e colocar uma escala apropriada, na qual seus dados serão visualizados na tela do mapa.

Na parte inferior, pode-se ver uma miniatura da camada, a simbologia da legenda e o mapa de cores.

13.2.2 Menu de Estilos

Representar a banda

QGIS permite quatro diferentes *Tipos de Representação*. A representação escolhida depende do tipo de dados.

1. Color multibanda - se o arquivo vem como multibanda, com várias bandas (por exemplo, usado para imagens de satélite com várias bandas)

2. Mapa de Cores - se um arquivo de banda única vem com um mapa de cores indexado (por exemplo, usado para mapas topográficos digitais)
3. Banda única cinza - (uma banda apenas) a imagem será representada como cinza; QGIS escolherá o método de representação, se o arquivo não é multibandas, não tem um mapa de cores indexado ou não tem um mapa de cores contínuos (por exemplo, usado para mapa de relevo sombreado)
4. Banda única Falsa Cor - este método de representação é usado em arquivos com mapa de cores contínuos ou com mapa de cores (por exemplo, para mapa de elevações)

Multibanda Colorida

Para representar em color multibanda, selecione três bandas da imagem que vai representar, cada banda representa respectivamente, a componente vermelha, verde e azul, que serão usadas para criar a cor da imagem. Podem-se escolher vários métodos para *Melhora do contraste* : ‘Sem melhora’, ‘Estique para MinMax’, ‘Estique e corte no MinMax’ e ‘Corte no min max’.

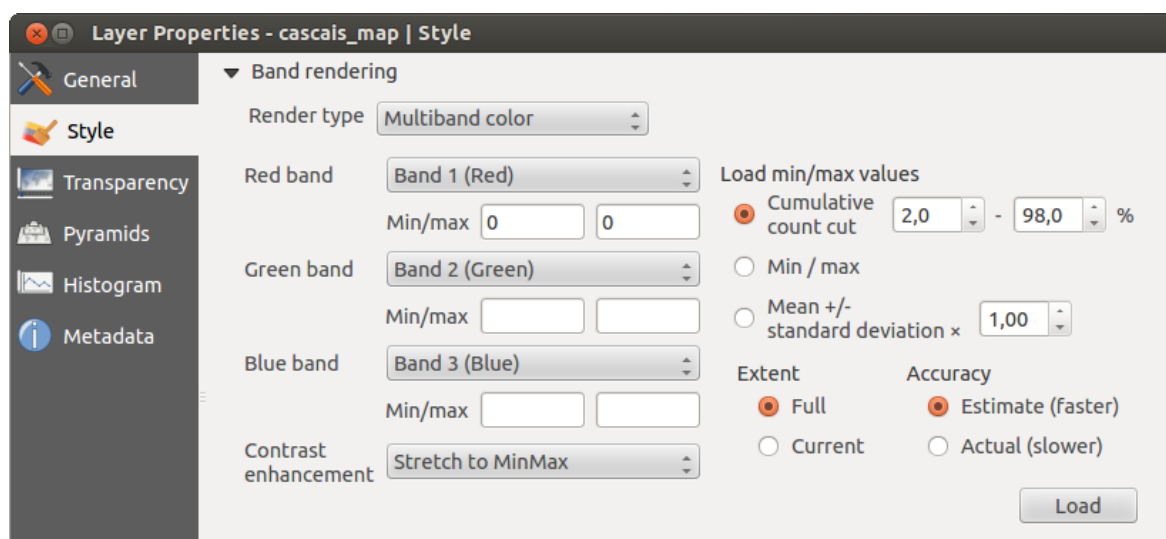


Figura 13.2: Representação do Raster - Multibanda Colorida

Esta seleção lhe oferece uma ampla variedade de opções para modificar a aparência da sua camada raster. Primeiramente, deve-se ter a amplitude dos dados da sua imagem. Isto pode ser visto escolhendo *Extensão* e clicando **[Carregar]**. QGIS pode *Estimar (mais rápido)* os valores *Mín* e *Máx* das bandas ou use o *Real (mais lento)* *Precisão*.

Agora pode-se criar uma escala de cores com ajuda da seção *Carregar valores mín/máx* ‘. *Muitas imagens tem valores muito baixos ou muito altos. Estes valores discrepantes, ou outliers, podem ser eliminados usando o radiobutton* *:guilabel: ‘Corte na Contagem acumulativa* para definir a eliminação. A amplitude padrão está fixada entre 2% a 98% dos valores dos dados e pode ser modificada manualmente. Com esta definição, o tipo cinza da imagem pode desaparecer. Com a opção de escala *Mín/máx*, QGIS cria uma tabela de cores com todos os dados incluídos na imagem original (por exemplo, QGIS cria uma tabela de cor com 256 valores, se temos uma imagem com banda de 8 bits). Pode-se calcular também uma tabela de cores usando *:guilabel: ‘Média +/- x desvio padrão* ‘ . Assim, apenas valores dentro do desvio padrão o dentro de múltiplos desvios padrão serão considerados na tabela de cores. Isto é útil quando se tem uma ou duas células com valores sumamente altos em uma gride raster que tem um impacto negativo na representação da imagem raster

Todos os cálculos pode ser feitos também através da extensão *Atual*

Dica: Visualizando uma única banda do Raster Multibanda

Se deseja ver uma única banda de uma imagem multibanda (por exemplo apenas a Vermelha), pode-se colocar as bandas Verde e Azul como “Não definidas”, mas isto não é a forma correta. Para mostrar apenas a banda Vermelha, coloque o tipo da imagem como ‘Banda única Cinza’, depois selecione o Vermelho como a banda para usar no Cinza.

Mapa de Cores

Esta é a opção padrão de representação para arquivos de uma banda que incluem uma tabela de cores, onde a cada valor de pixel é atribuída uma determinada cor. Nesse caso, o mapa de cores é gerado automaticamente. SE deseja cambiar as cores atribuídas a determinados valores, apenas de clique duplo no color e a janela de diálogo *Selecione cor* vai aparecer. Também em QGIS 2.2, agora é possível atribuir um rótulo aos valores das cores. Assim o rótulo aparece na legenda da camada raster.

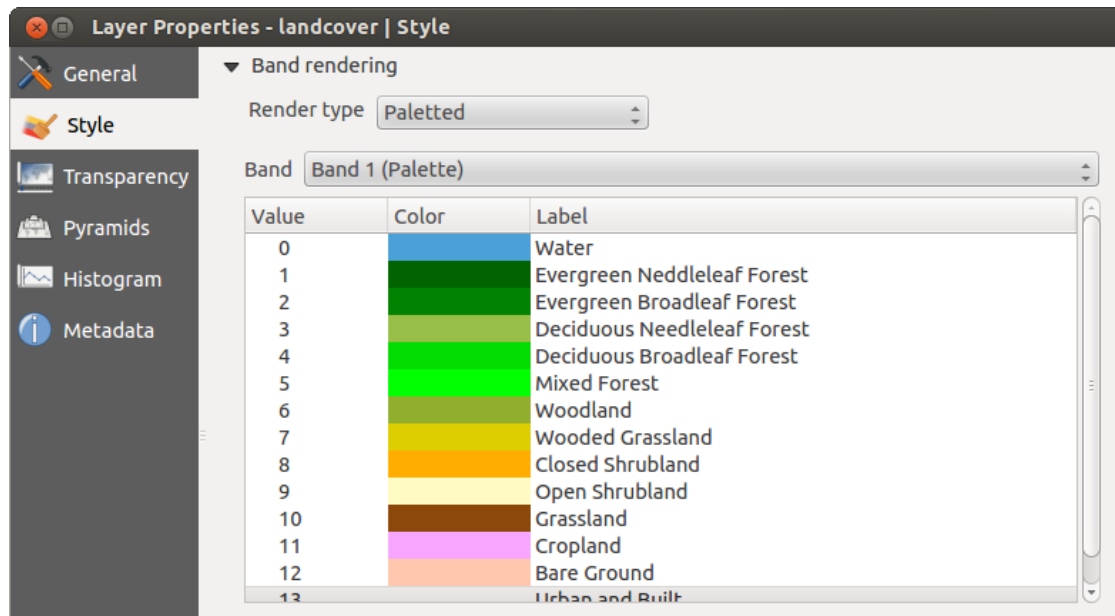


Figura 13.3: Representação do Raster - Mapa de cores 🐧

Melhora do contraste

Nota: Quando adicionamos camadas GRASS, a opção *Melhorar o Contraste* será automaticamente definida como *esticar ao mín máx*, independentemente de estar definido outro valor na QGIS opções gerais.

Banda única cinza

Esse método permite representar uma camada de uma banda através de *Gradiente de Cores: Preto para Branco* ou *Branco para Preto*. Pode definir os valores *Mín* e *Máx* escolhendo primeiro a *Extensão* e depois clicando em **[Carregar]**. QGIS pode *Estimar (mais rápido)* os valores *Mín* e *Máx* das bandas ou usar *Real (mais lento) Precisão*.

Pode-se criar uma escala de cores com ajuda da seção *Carregar valores mín/máx*. Os valores discrepantes podem ser eliminados usando o :guilabel:Com esta definição, os valores cinza da imagem podem desaparecer. Outras definições podem ser determinadas com o *Mín/máx* e *Média +/- x desvio padrão* . Enquanto a primeira opção cria uma tabela de cores com todos os dados incluídos na imagem original, a segunda cria uma tabela de cores que considera apenas dentro do desvio padrão ou dentro de vários desvios pad' Corte na Contagem acumulativa' para definir a eliminação. A amplitude padrão está fixada entre 2% a 98% dos valores dos dados e pode ser modificada manualmente. rão. Isto é útil quando se tem uma ou duas células com valores anormalmente altos numa gride raster que provocam um impacto negativo na representação da imagem raster.

Singleband pseudocolor

Este é um opção de representação para arquivos de uma banda, que incluem um mapa de cores contínuo. Pode-se também aqui criar mapas de cores individuais para uma banda. Existem três tipos de interpolação de cores:

1. Método Discreto
2. Método Linear

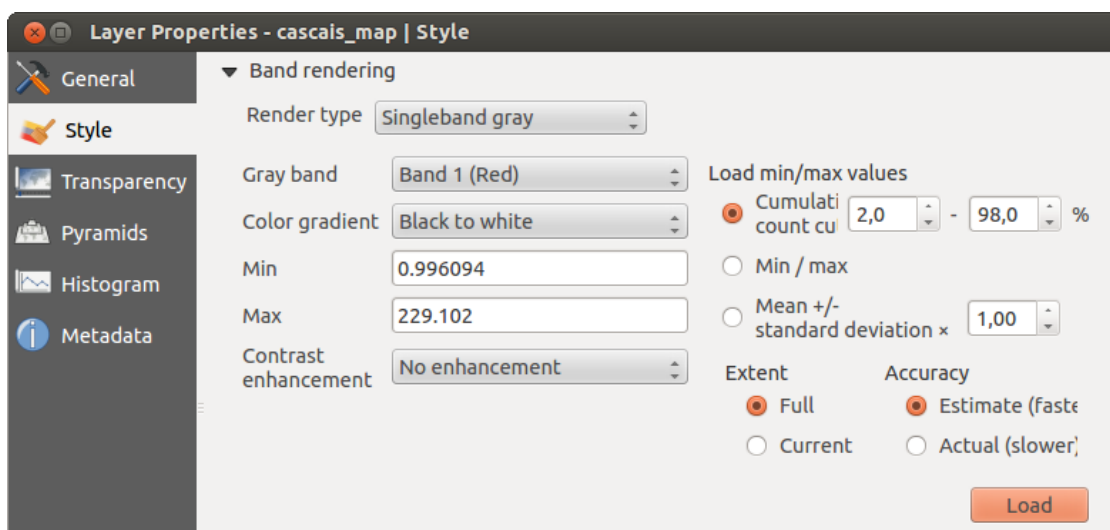


Figura 13.4: Representação do Raster - Banda única cinza 🐧

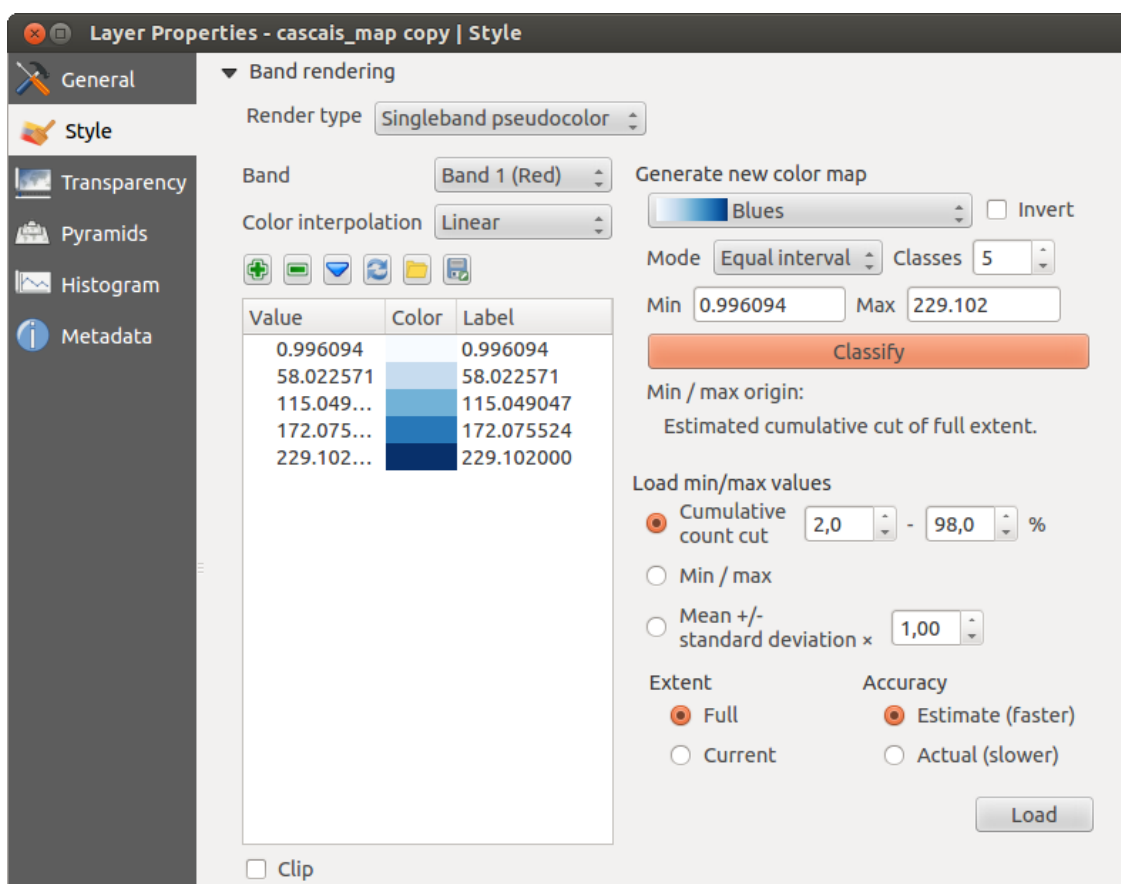














Figura 13.5: Representação do Raster - Banda única Falsa Cor

3. Método Exato

No bloco da esquerda, o botão  Adicionar valores manualmente, adiciona um valor individual na tabela de cores. O botão , apaga um valor individual da tabela e o botão , ordena a tabela de acordo ao valor de pixel da coluna valor. Também é possível adicionar rótulos para cada cor, mas esse valor não será mostrado quando se use o identificador feição ferramenta. Também pode clicar no botão , o qual tenta carregar uma tabela de cores a partir da banda (se tem alguma). Pode-se usar também os botões  Carregar mapa de cor de arquivo e , para carregar uma tabela de cor já existente ou salvar uma tabela de cor para uso futuro.

No bloco da direita, *Gerar novo mapa de cor*, permite criar novos mapas de cor categorizados. Em *Modo Classificação*  'Intervalos iguais', apenas necessita selecionar o *Número de classes* e clicar o botão *Classificar*. Pode-se inverter as cores do mapa de cores clicando na caixa de seleção  *Invertir*. Em caso de *Modo*  'Contínuo', QGIS cria as classes automaticamente, dependendo dos valores *Mín* and *Máx*. A definição dos valores *Mín/Máx*, pode ser feita com ajuda da seção *Carregar valores mín/máx*. Muitas imagens tem uns poucos valores com dados muito altos e baixos. Esses valores discrepantes ou outliers, podem ser eliminados usando o botão  :guilabel: "Corte na Contagem acumulativa" para definir a eliminação. A amplitude padrão está fixada entre 2% a 98% dos valores dos dados e pode ser modificada manualmente. Com esta definição, os valores cinzas na imagem podem desaparecer. Com a opção de escala , QGIS cria uma tabela de cores com todos os dados incluídos na imagem original (por exemplo, QGIS cria uma tabela de cores com 256 valores, dado o fato de termos uma imagem com banda de 8 bit). Pode-se calcular também uma tabela de cores usando o botão  *Média +/- x desvio padrão* . Assim, são considerados para a tabela de cores, apenas os valores dentro do desvio padrão ou dentro de vários desvios padrão.

Representação das cores

Em cada *Representação da banda*, é possível encontrar uma *Representação da cor*

Podem-se fazer efeitos especiais de representação para seus arquivo(s) raster, usando um dos modos de combinação (veja *Janela de Propriedades de Vetor*).


Definições adicionais podem ser estabelecidas modificando em *Brilho*, *Saturação* e *Contraste*. Pode usar também a opção *Escala de cinzas*, onde pode escolher entre 'Por claridade', 'Por luminosidade' e 'Por média'. Por um determinado matiz na tabela de cores, você pode modificar a 'Força'.

Reamostragem

A opção *Reamostragem*, faz a representação da imagem quando se dá mais ou menos zoom nela. Os modos de reamostragem podem melhorar a aparência do mapa. Eles calculam um novo valor de cinza através de uma transformação geométrica.

Quando aplicamos o método 'Vizinho mais próximo', o mapa pode ter uma estrutura tipo pixelada, quando damos mais zoom. Essa aparência pode ser melhorada usando os métodos 'Bilinear' ou 'Cúbico', o qual causa que as feições mais afiadas, se suavizem.

13.2.3 Menu de transparência

QGIS mostra cada camada raster com um nível diferente de transparência. Use a barra deslizante de transparência  para indicar a que nível as camadas de baixo (se houver alguma) devem ser visíveis através da camada em uso. Isto é muito útil se deseja sobrepor mais de uma camada (por exemplo um mapa de relevo sombreado sobreposto a um mapa raster com classificação). Assim o mapa terá um aspecto mais tridimensional.

Além disso pode-se colocar um valor de pixel que será considerado como *SEMDADOS* no menu *Valor adicional sem dados*

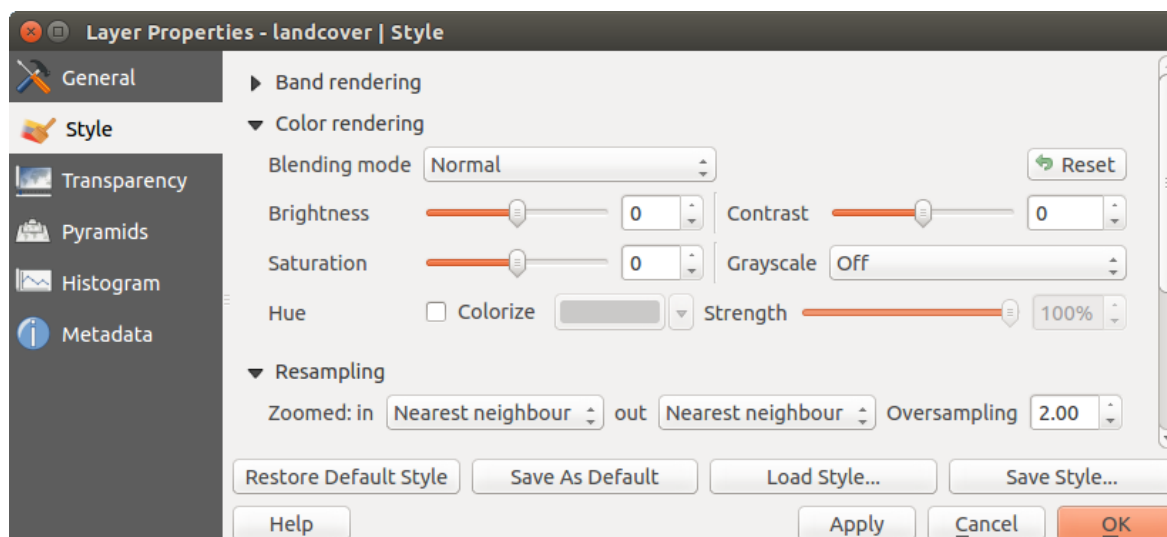





Figura 13.6: Representação Raster - Reamostragem 

Uma maneira ainda mais flexível de modificar a banda de transparência poder ser feita no :guilabel: *Modificações das opções de transparência*. Aqui podemos definir a transparência de cada pixel.

Como exemplo, desejamos colocar a água de nosso arquivo raster de exemplo **:Arquivo:'landcover.tif'**, com transparência de 20%. Os seguintes passos são necessários:

1. Carregar o arquivo raster: Arquivo:*landcover.tif*.
2. Abra o diálogo *Propriedades* fazendo clique duplo no nome do raster na legenda, o clicando com botão e selecionando:*Propriedades* do menu pop-up.
3. Selecione o menu *Transparência*
4. No menu *Transparência da banda*, escolher 'Nenhum'.
5. Clique no botão  Adicionar valores manualmente uma nova linha vai aparecer na lista de pixels.
6. Entre o valor raster na coluna 'De' e 'Até' (usamos 0 aqui), e ajuste a transparência a 20%.
7. Pressione o botão **[Aplicar]** e visualize no mapa as modificações feitas.

Podemos repetir os passos 5 e 6 para definir mais valores com a transparência desejada.

Como podemos ver, é fácil definir uma transparência desejada, mas requer um grande esforço. Aliás, pode-se usar o botão  Exportar para arquivo para gravar sua lista de transparências num arquivo. O botão  Importar do arquivo carrega sua definição de transparências e a aplica à camada do raster em uso.

13.2.4 Menu de Pirâmides

Grandes camadas raster com grande resolução, podem tornar a navegação lenta em QGIS. Criando cópias em baixa resolução dos dados (pirâmides), o desempenho pode ser melhorado consideravelmente, já que QGIS, seleciona a resolução mais apropriada dependendo do nível do zoom.

Você deve ter direito de gravação no diretório onde os dados originais são armazenados para construir pirâmides.

Vários métodos de reamostragem podem ser usados para calcular as pirâmides.

- Vizinho mais próximo
- Média
- Gauss

- Cúbico
- Modo
- Nenhum

Se seleciona 'Interno (se possível)' do menu *Formato Overview*, QGIS tenta calcular as pirâmides internamente. Pode-se selecionar também 'Externo' e 'Erdas Externo (Formato Erdas Imagine).

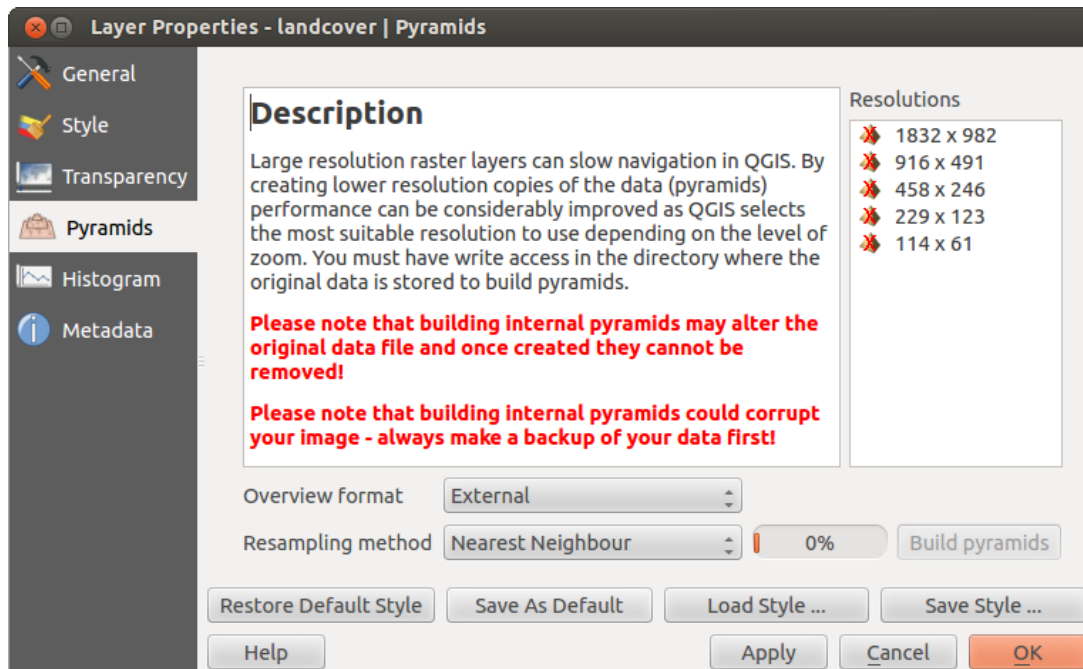



Figura 13.7: The Pyramids Menu 

Note que o cálculo de pirâmides pode modificar o arquivo original de dados, e uma vez criado, não pode ser apagado. Se deseja preservar uma versão 'sem pirâmides' de seu raster, faça uma cópia de segurança antes do cálculo das pirâmides.

13.2.5 Menu Histograma

O menu *The Histograma* permite ver a distribuição das bandas ou cores no raster. O histograma é gerado automaticamente quando se abre o menu *Histograma*. Todas as bandas presentes são apresentadas juntas. Pode-se salvar o histograma como imagem com o botão **ImAçãoSalvarArquivol**. Com a opção *With the Visibilidade* do menu  *Prefs/Ações*, podem se mostrar histogramas individuais de bandas. Para isto, selecione a opção *Mostrar banda selecionada*. O *Opções Mín/máx* permite 'Sempre mostrar marcadores mín/máx', a 'Zoom para mín/máx' e 'Atualizar estilo para mín/máx'. Depois de escolher *Opções Mín/máx* usando a opção *Ações* pode-se 'Redefinir' e 'Recalcular o histograma'.

13.2.6 Menu Metadados

O menu *Metadados*, mostra o estado da informação da camada do raster, incluindo estatísticas de cada banda na camada do raster em uso. A partir deste menu, podem ser definidas entradas na guia *Descrição*, *Atribuição*, *MetadadosUrl* e *Propriedades*. Na guia: `guilabel:Propriedades`, são geradas estatísticas na base de 'é preciso saber' ou seja é possível que uma determinada e específica estatística da camada, no tenha sido ainda coletada.

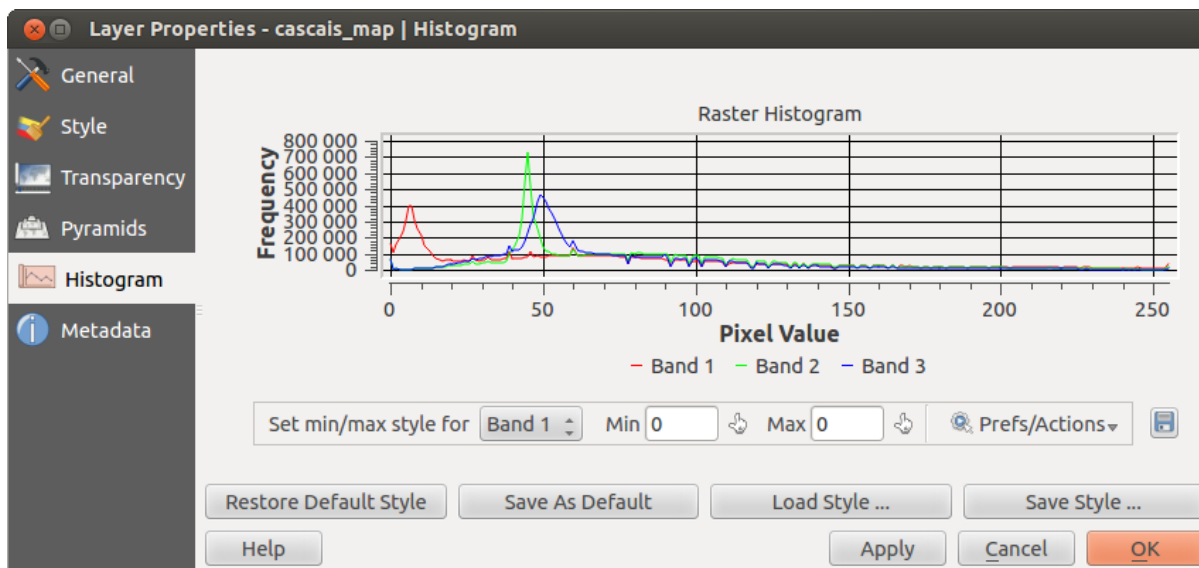


Figura 13.8: Histograma do raster 🐧

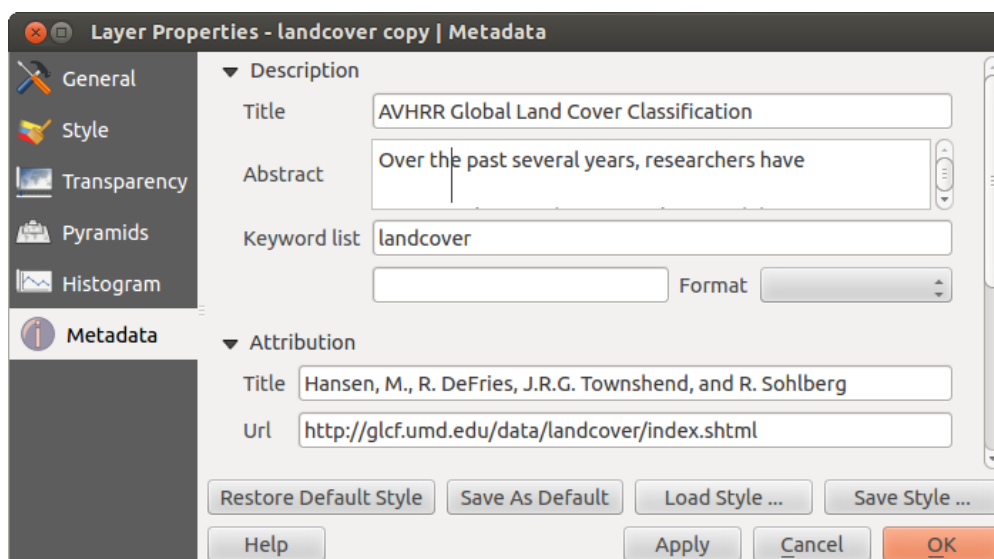


Figura 13.9: Metadados do raster 🐧

13.3 Calculadora Raster

A *Calculadora Raster* no menu *Raster* permite realizar cálculos com base em valores de pixel raster existentes (ver `figure_raster_10`). Os resultados são gravados em uma nova camada com um formato suportado-GDAL.

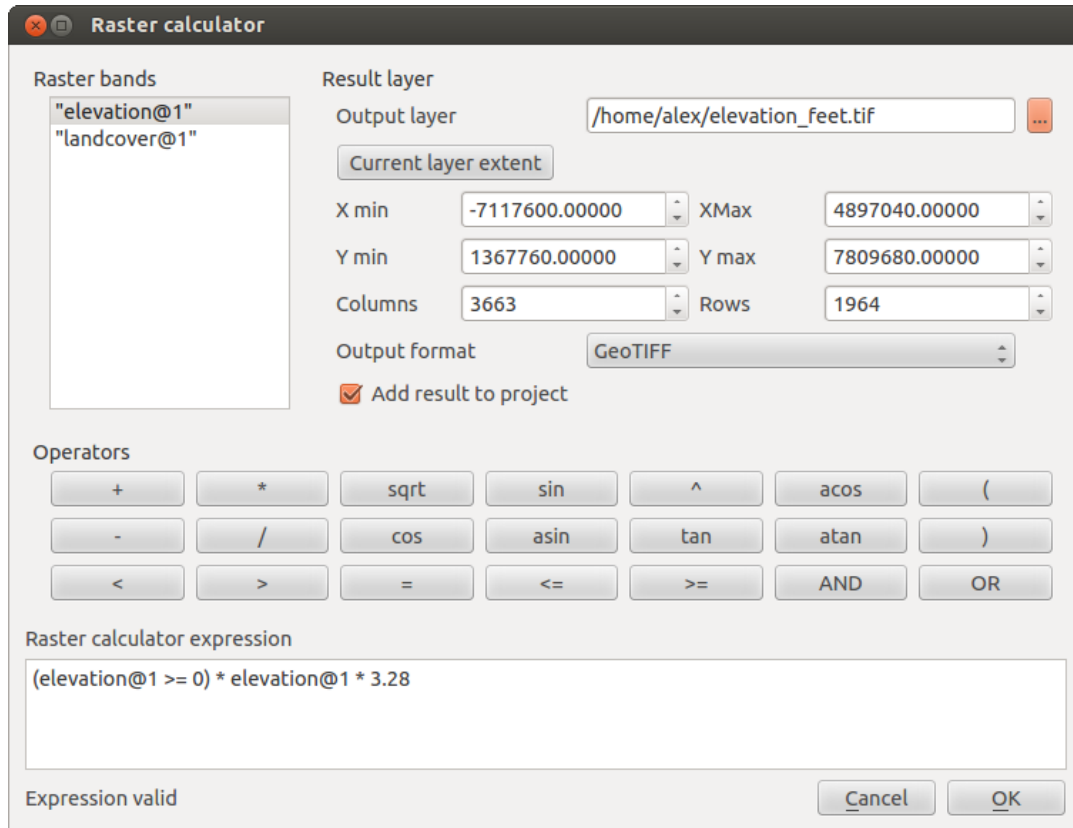



Figura 13.10: Calculador Raster 

As lista **Bandas raster** contém todas camadas raster carregadas e que podem ser usadas. Você pode então usar os operadores para construir expressões de cálculo, ou você pode simplesmente digitá-las na caixa.

Na seção **camada resultado**, você precisará definir uma camada de saída. Você pode, então, definir a extensão da área de cálculo baseado em uma camada raster de entrada, ou com base nas coordenadas X, Y e em colunas e linhas, para definir a resolução da camada de saída. Se a camada de entrada tem uma resolução diferente, os valores serão redefinidos com o algoritmo vizinho mais próximo.

A seção **operadores** contém todos os operadores disponíveis. Para adicionar um operador de caixa de expressão na calculadora raster, clique no botão apropriado. Cálculos Matemáticos (‘+’, ‘-’, ‘*’, ‘/’, ‘^’, ‘sqrt’, ‘sin’, ‘cos’, ‘atan’, ‘asin’, ‘tan’, ‘acos’) e funções trigonométricas (‘sen’, ‘cos’, ‘tan’, ...) estão disponíveis. Fique ligado para mais operadores que poderão surgir!

Com a caixa *Adicionar resultado para projetar*, a camada resultado será adicionada automaticamente à área de legenda e poderá ser visualizada.

13.3.1 Exemplos

Converter valores de elevação de metros para pés

Criar um raster de elevação em pés de uma varredura em metros, você precisa usar o fator de conversão de metros para pés: 3.28. A expressão é:

```
"elevation@1" * 3.28
```

Usando uma máscara

Se você quer mascarar partes de um raster – digamos, por exemplo, porque você está interessado apenas em altitudes superiores a 0 metros – você pode usar a seguinte expressão para criar uma máscara e aplicar o resultado de uma varredura em uma única etapa.

```
("elevation@1" >= 0) * "elevation@1"
```

Em outras palavras, para cada célula maior ou igual a 0, defina seu valor para 1. Caso contrário, defina-o como 0. Isso cria a máscara sobreposta.

Se você quiser classificar um raster - digamos, por exemplo, em duas classes de elevação, você pode usar a seguinte expressão para criar um raster com dois valores 1 e 2 em uma única etapa.

```
("elevation@1" < 50) * 1 + ("elevation@1" >= 50) * 2
```

Em outras palavras, para cada célula menor que 50 defina seu valor como 1. Para cada célula maior ou igual a 50 defina seu valor como 2.

Trabalhando com dados OGC

14.1 QGIS como Cliente de Dados OGC

O Consórcio Geoespacial Aberto (OGC), é uma organização internacional como mais de 300 organizações em todo mundo do tipo comerciais, sem fins lucrativos e de investigação. Os membros desenvolvem e implementam padrões para os conteúdos e serviços geoespaciais, processamento e troca de dados SIG.

Descrevendo um modelo básico de dados para elementos geográficos e um número crescente de especificações estão desenvolvidos para servir necessidades específicas para localização interoperável e tecnologia geoespacial, incluindo o SIG. Mais informação é encontrada em <http://www.opengeospatial.org/>.

As especificações OGC importantes suportadas pelo QGIS são:

- **WMS** — Serviço de Mapas Web (*Cliente WMS/WMTS*)
- **WMTS** — Serviço de Mosaicos de Mapa Web (*Cliente WMS/WMTS*)
- **WFS** — Serviços de Elementos Web (*WFS e WFS-T Cliente*)
- **WFS-T** — Serviços de Elementos Web - Transacionais (*WFS e WFS-T Cliente*)
- **WCS** — Serviços de Cobertura Web (*WCS Cliente*)
- **SFS** — Elementos Simples para SQL (*Camadas PostGIS*)
- **GML** — Linguagem de Marcadores Geográfico

Os serviços OGC estão a ser crescentemente usados para troca de dados geoespaciais de diferentes implementações SIG e armazenamento de dados. O QGIS consegue lidar com as especificações em baixo como cliente, sendo **SFS** (através do suporte do fornecedor PostgreSQL/ PostGIS, veja Seção *Camadas PostGIS*).

14.1.1 Cliente WMS/WMTS

Visão Global do Suporte WMS

O QGIS atualmente pode funcionar como cliente WMS que entende servidores WMS 1.1, 1.1.1 e 1.3. Foi particularmente testado contra serviços públicos de acesso como os DEMIS.

Os servidores WMS funcionam através de pedidos pelo cliente (ex.: QGIS) para mapas raster com uma dada extensão, conjunto de camadas, estilos de simbolização, e transparência. O servidor WMS de seguida consulta as suas fontes de dados locais, matricializa num mapa e manda de volta ao cliente em formato raster. Para o QGIS isto tipicamente vem em JPEG ou PNG.

WMS é um serviço REST genérico (Estado de Transferência Representativo) mais que um Serviço Web completamente fundido. Como tal, pode realmente obter os URLs gerados pelo QGIS e usá-los num navegador da web para recuperar as mesmas imagens que o QGIS usa internamente. Isto pode ser útil para resolver problemas, uma

vez que existem várias marcas de servidores WMS no mercado e todos eles têm a sua própria interpretação da norma WMS.

As camadas WMS podem ser adicionadas facilmente, desde que conheça o acesso URL para o servidor WMS, tenha uma ligação de serviço a esse servidor, e o servidor compreenda HTTP como um mecanismos de transporte de dados.

Visão Global do Suporte WMTS

O QGIS pode também agir como um cliente WMTS. O WMTS é um padrão OGC para a distribuição de conjuntos de mosaicos de dados geoespaciais. Isto é uma forma mais rápida e mais eficiente para distribuir dados que o WMS porque com o WMTS os conjuntos de mosaicos são pré-gerados e o cliente apenas faz pedidos de transmissão dos mosaicos e não os produz. Um pedido típico do WMS envolve a geração e transmissão de dados. Um exemplo conhecido de um padrão não-OGC para a visualização de mosaicos de dados geoespaciais é o Google Maps.

De maneira a exibir os dados a várias escalas perto do que o utilizador queira ver, os conjuntos de mosaicos WMTS são produzidos em vários diferentes níveis de escala e são disponibilizados para o cliente SIG a seu pedido.

Este diagrama ilustra o conceito dos conjuntos de mosaicos

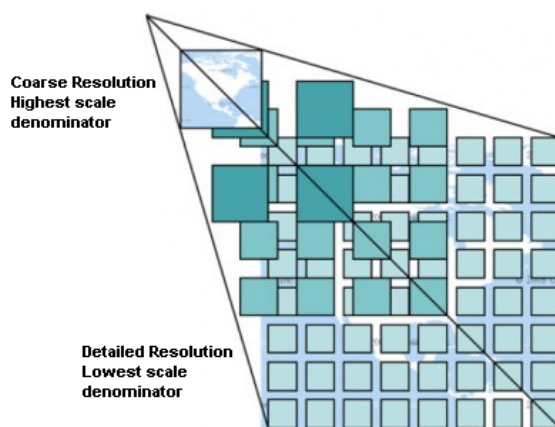


Figura 14.1: Conceito dos conjuntos de mosaicos do WMTS

Os dois tipos de interfaces WMTS que o QGIS suporta são via Key-Value-Pairs (KVP) e RESTful. Estas duas interfaces são diferentes e necessita especificá-los ao QGIS de forma diferente.

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"
```

Um exemplo deste tipo de endereço é

```
http://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&request=GetCapabilities
```

A camada topo2 funciona lindamente para testar neste WMTS. Adicionando esta linha e texto indica que é um serviço web WMTS que deverá ser usado em vez de o serviço WMS.

2. O serviço **WMTS RESTful** torna-se uma forma diferente, é um URL simples, a forma recomendada pela OGC é:

```
{WMTSBaseURL}/1.0.0/WMTSCapabilities.xml
```


Este formato ajuda-a a reconhecer que isto é um endereço RESTful. O WMTS RESTful é acedido no QGIS adicionando simplesmente o endereço na configuração do WMS no campo do URL da forma. Como exemplo para um mapa base Austríaco deste tipo de endereço é <http://maps.wien.gv.at/basemap/1.0.0/WMTSCapabilities.xml>.

Nota: Pode encontrar alguns serviços antigos chamados de WMS-C. Esses serviços são muito semelhantes ao WMTS com a mesma finalidade mas trabalham ligeiramente diferente). Pode gerir-los da mesma forma que faz nos serviços WMTS. Apenas adicione `?tiled=true` no final do url. Veja http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification para mais informações sobre esta especificação.

Quando lê o WMTS pode muitas vezes pensar em WMS-C.

Selecionando os Servidores WMS/WMTS


Na primeira vez que usa o elemento WMS no QGIS, não existem servidores definidos.

Comece clicando no botão  Adicionar camada WMS na barra de ferramentas, ou através do menu *Camada* → *Adicionar Camada WMS*

A janela *Adicionar Camada(s) do Servidor* para adicionar camadas dos servidores WMS aparecem. Pode adicionar alguns servidores para brincar clicando no botão **[Adicionar servidores padrões]** . Isto irá adicionar dois servidores WMS demonstração para que possa usar, os servidores WMS do DM Solutions Group e Lizardtech. Para definir uma novo servidor WMS no separador *Camadas*, selecione o botão **[Novo]** . De seguida introduza os parâmetros de ligação que deseja para o Servidor WMS, como está listado na *table_OGC_1*:

Nome	Um nome para esta ligação. Este nome será usado para a lista de Ligações do Servidor para que possa distingui-la de outros Servidores WMS.
URL	URL do servidor que fornece os dados. Isto deverá ser um nome de alojamento válido – o mesmo formato que irá usar para abrir a ligação telnet ou o ping a um alojamento.
Usuário	Nome de Utilizador para aceder a um Servidor WMS protegido. Este parâmetro é opcional.
Senha	Senha para autenticação básica no Servidor WMS. Este parâmetro é opcional.
Ignorar GetMap URI	<input checked="" type="checkbox"/> <i>Ignorar GetMap URI reportado nas capacidades.</i> Use um URI dado em cima para o campo URL.
Ignorar GetFeatureInfo URI	<input checked="" type="checkbox"/> <i>Ignorar GetMap URI reportado nas capacidades,</i> use um URI dado em cima para o campo URL.

Tabela 1 OGC : Parâmetros de Ligação do WMS

Se precisar de configurar um servidor proxy para receber serviços WMS da internet, pode adicionar o seu servidor proxy nas opções. Escolha o menu *Configurações* → *Opções* e clique no separador *Rede & Proxy*. De seguida pode adicionar as suas configurações de proxy e ativá-las configurando o *Use o proxy para acesso web*. Certifique-se que selecionou o tipo proxy correto da lista de menu *Tipo de Proxy*  .

Uma vez a nova ligação do Servidor WMS for criada, será preservada para sessões futuras do QGIS.

Dica: Ligar URLs dos Servidores WMS

Certifique-se, que quando introduzir o URL do servidor WMS, seja o URL base. Por exemplo, não deve ter fragmentos como `request=GetCapabilities` ou `version=1.0.0` no seu URL.

Carregando as camadas WMS/WMTS

Uma vez preenchido os parâmetros com sucesso pode usar o botão **[Ligar]** para responder às capacidades do servidor selecionado. Isto inclui a codificação da imagem, Camadas, Estilos de Camadas, e Projeções. Uma vez que isto é uma operação de rede, a velocidade de resposta depende da qualidade da ligação da sua rede ao servidor WMS. Enquanto faz a transferência de dados do servidor WMS, o progresso da transferência é visualizada no canto inferior esquerdo da janela do WMS.

O seu ecrã irá ficar parecido um pouco como *figure_OGR_1*, que mostra a resposta fornecida pelo servidor WMS do DM Solutions Group.

Codificação da Imagem

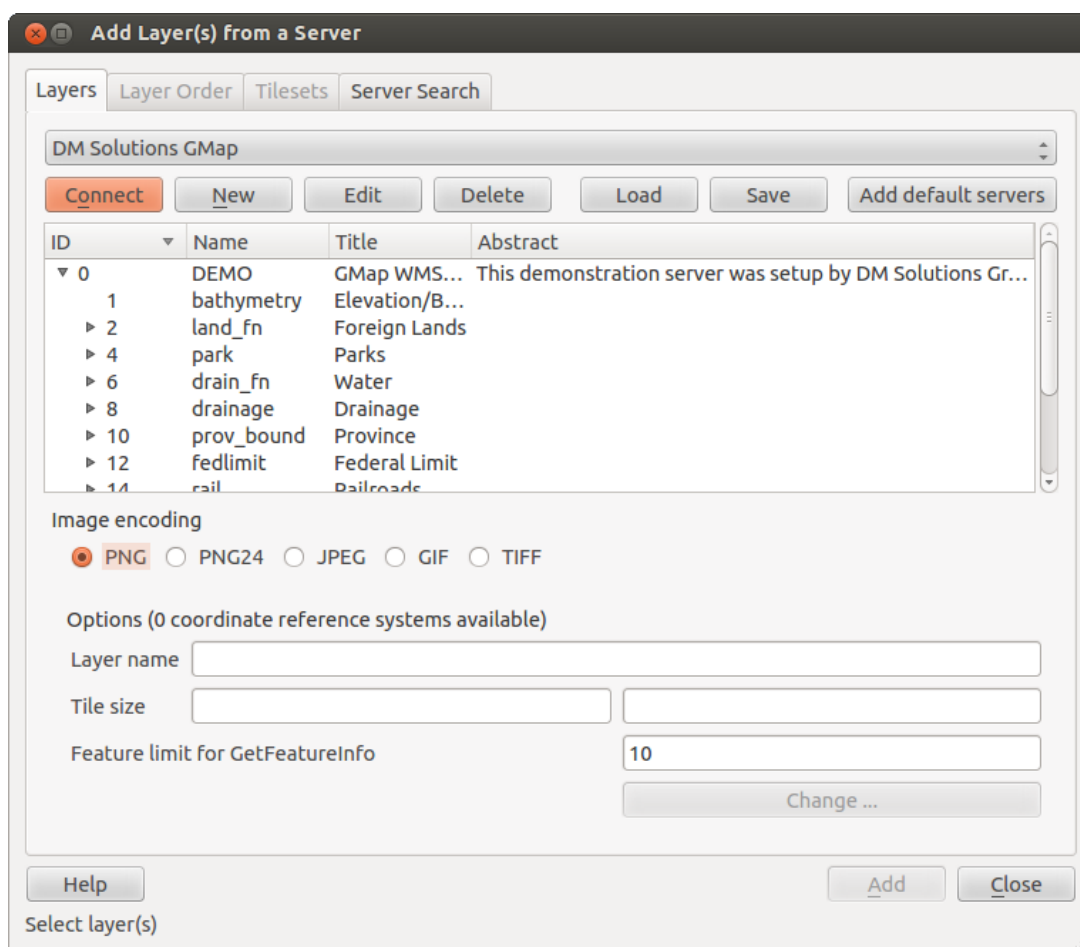


Figura 14.2: Janela para adicionar servidores WMS, mostra as camadas disponíveis 🐧

A seção *Codificação de Imagem* agora apresenta uma lista de formatos suportados pelo cliente e servidor. Escolha de acordo com as exigências de precisão da sua imagem.

Dica: Codificação da Imagem

Tipicamente irá encontrar servidores WMS que oferecem a escolha da codificação de imagem JPEG ou PNG. O JPEG é um formato de perda de compressão, enquanto que o PNG reprodução fielmente os dados raster brutos.

Use JPEG se espera dados WMS para fotografia na natureza e/ou não se importa da perda de alguma qualidade na imagem. Este típico trade-off reduz em 5 vezes o requerimento de transferência comparado ao PNG.

Use o PNG se necessitar representações precisas dos dados originais, e não se importa dos requerimentos de transferência de dados.

Opções

O campo Opções fornece um campo de texto onde pode adicionar *Nome da Camada* para a camada WMS. Este nome irá ser apresentado na legenda após carregamento da camada.

Em baixo do nome da camada pode definir *Tamanho do mosaico*, se quiser configurar os tamanhos do mosaico (ex.: 256x256) para dividir o pedido do WMS em pedidos múltiplos.

O *Limite de elementos para GetFeatureInfo* define quais os elementos do servidor para consulta.

Se selecionou um WMS da lista, um campo com a projeção padrão, fornecida pelo mapserver, aparece. Se o botão **[Change...]** está ativo, pode clicar e mudar para a projeção padrão do WMS para outro SRC fornecido pelo Servidor WMS.

Ordem das Camadas

O separador *Ordem de Camada* tem uma lista das camadas selecionadas disponíveis do WMS actual ligado. Poderá reparar que algumas camadas expansíveis, isto significa que a camada pode ser exibida na escolha dos estilos de imagem.

Pode seleccionar várias camadas de uma só vez, mas apenas um estilo de imagem por camada. Quando várias camadas são seleccionadas, elas serão combinadas no Servidor WMS e transmitido ao QGIS de uma só vez.

Dica: Ordenação das Camadas WMS

As camadas WMS renderizadas por um servidor são sobrepostas na ordem da lista da seção de Camadas, de cima para baixo da lista. Se quiser alterar a ordem de sobreposição, pode usar o separador *Ordem de camadas*.

Transparência

Nesta versão do QGIS, a configuração *Transparência Global* da *Propriedades da Camada* está codificado para estar sempre ligado, quando disponível.

Dica: Transparência da Camada WMS

Disponibilidade da transparência da imagem do WMS depende da codificação da imagem usada: PNG e GIF suportam transparência enquanto que o JPEG deixa como não suportado.

Sistema de Referência de Coordenadas

O Sistema de Coordenadas Referência (SRC) é a terminologia OGC para uma projeção QGIS.

Cada Camada WMS pode ser apresentada em múltiplos SRC, dependendo da capacidade do servidor WMS.

Para escolher o SRC selecione **[Alterar...]** e a janela semelhante à da Figura Projeção 3 na *Trabalhando com Projeções* aparecerá. A diferença principal com a versão do WMS do ecrã é que neste apenas os SRC suportados apareceram no Servidor WMS.

Pesquisa de servidor

Dentro do QGIS pode pesquisar por servidores WMS. *Figure_OGC_2* mostra o separador *Pesquisar Servidor* com a janela *Adicionar Camada(s) do Servidor*.

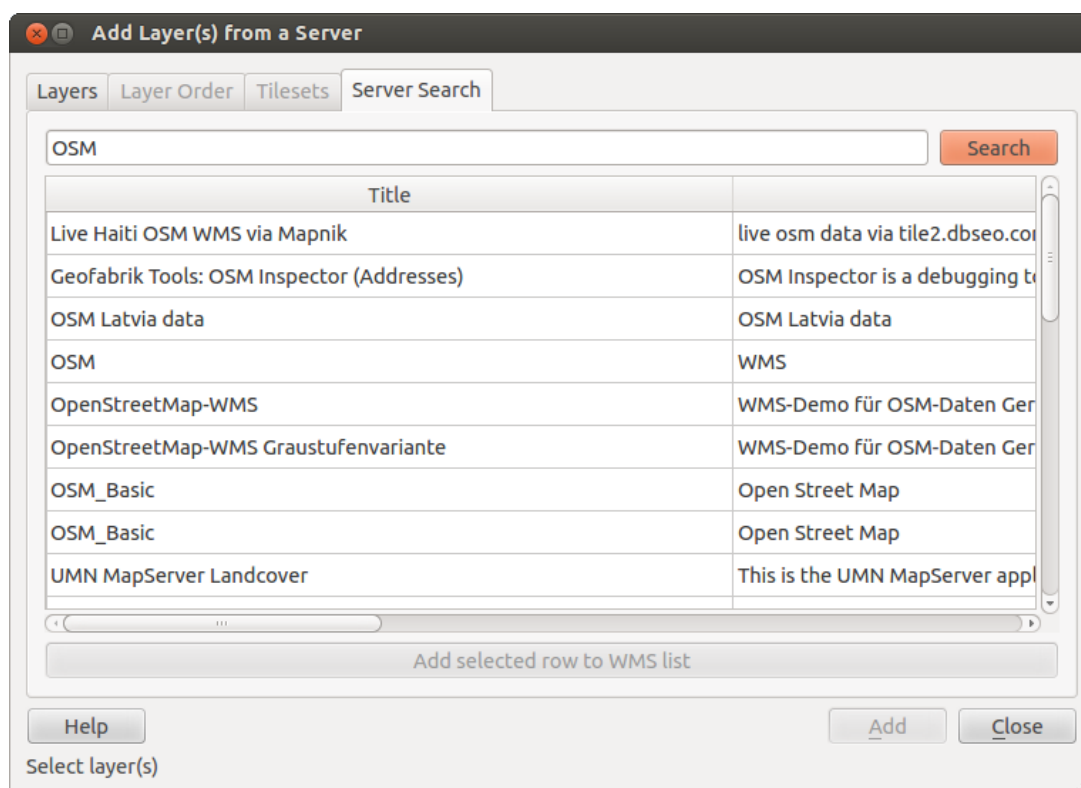


Figura 14.3: Janela de pesquisa de servidores WMS depois de algumas palavras-chave 🐧

Como pode ver é possível introduzir uma cadeia de texto de pesquisa no campo de texto e pressionar o botão **[Pesquisar]**. Após pouco tempo o resultado da pesquisa será preenchido na lista debaixo do campo de texto. Procure a lista de resultados dentro da tabela. Para visualizar os resultados, selecione uma entrada da tabela, prima o botão **[Adicionar a linha selecionada à lista WMS]** e mude para o separador *Camadas*. O QGIS automaticamente atualiza a sua lista de servidores e o resultado selecionado da pesquisa está disponível na lista de servidores WMS salvados no separador *Camadas*. Apenas precisará de pedir a lista de camadas clicando no botão **[Ligar]**. Esta opção é especialmente útil quando quer pesquisar mapas por palavras-chave específicas.

Basicamente esta opção é um dianteira para a API do <http://geopole.org>.


Mosaicos

Ao usar Serviços WMTS (WMS em cache) como


```
http://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?
service=WMTS&request=GetCapabilities
```

pode procurar através do separador *Conjuntos de Mosaicos* dados pelo servidor. Informação adicional como o tamanho do mosaico, formatos e SRC suportados estão na lista desta tabela. Em combinação com esta feição pode usar a escala da quadrícula do *Configurações* → *Painéis* (KDE e Windows) ou *Ver* → *Painéis* (Gnome e MacOSX) e de seguida escolha *Escala da quadrícula*, que dá escalas disponíveis do servidor de mosaicos com uma barra de deslizamento acoplada.

Usando a Ferramenta Identificar

Uma vez adicionado o servidor WMS, e se qualquer camada do servidor WMS é consultável, pode usar a ferramenta  **Identificar** para selecionar o pixel do enquadramento do mapa. A consulta é feita ao servidor WMS para cada seleção feita. Os resultados da consulta vêm na forma de um texto plano. A formatação desse texto depende do servidor WMS particular usado. **Seleção do formato**

Se múltiplos formatos são suportados pelo servidor, uma caixa de combinação com os formatos suportados é automaticamente adicionado aos resultados da janela de identificação e o formato selecionado irá armazenar no projeto para a camada. **Suporte do formato GML**

A ferramenta  **Identificar** suporta respostas do Servidor WMS (GetFeatureInfo) no formato GML (é chamado de Elemento no GUI do QGIS neste contexto). Se o formato “Elemento” for suportado pelo servidor e selecionado, os resultados da ferramenta Identificar são elementos vetoriais como normais camadas vetoriais. Quando um elemento é selecionado na árvore, é destacada no mapa e pode ser copiada para a área de transferência e colada em outra camada vetorial. Veja o exemplo da instalação em baixo do UMN Mapserver para suportar o formato GML GetFeatureInfo.

```
# 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 send 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"
```

Propriedades de Visualização

Uma vez ter adicionado o servidor WMS, pode ver as suas propriedades clicando com o direito do mouse na legenda, e selecionando *Propriedades*. **Separador de Metadados**

O separador *Metadados* exibem a riqueza de informação sobre o servidor WMS, geralmente recolhidos a partir da declaração de de Capacidades devolvidos a partir desse servidor. Muitas das definições podem ser removidas através da leitura dos padrões WMS (veja OPEN-GEOSPATIAL-CONSORTIUM *Referências Bibliográficas e Web*), mas aqui estão algumas definições úteis:

- **Propriedades do Servidor**

- **Versão do WMS** — Versão do WMS suportada pelo servidor.
- **Formatos de Imagem** — A lista de MIME-types que o servidor pode responder com o desenho do mapa. O QGIS formata qualquer que seja as bibliotecas Qt subjacentes que foram construídas, que é pelo menos tipicamente a `image/png` e `image/jpeg`.
- **Formatos de Identidade** — A lista dos MIME-types do servidor pode responder quando usa a ferramenta identificar. Atualmente o QGIS suporta o tipo `text-plain`.

- **Propriedades da Camada**

- **Selecionado** — Querendo ou não esta camada selecionado quando o seu servidor foi adicionado a este projeto.
- **Visível** — Seja ou não essa camada esta é selecionada como visível na legenda. (Ainda não é usado nesta versão do QGIS.)
- **Pode Identificar** — Camada que retorna ou não resultados através do uso da ferramenta Identificar.
- **Pode ser Transparente** — Quando uma camada pode ou não pode ser editada com transparência. Esta versão do QGIS irá sempre usar a transparência se esta é `Sim` e a codificação da imagem suportar transparência
- **Permite Aproximar** — Quando uma camada pode ou não pode ser ampliada neste servidor. Esta versão do QGIS assume que todas as camadas WMS têm esta definida como `Sim`. Camadas deficientes podem ser renderizadas de forma estranha.
- **Contagem em cascata** — Os servidores WMS pode funcionar como proxy para outros servidores WMS para obter dados raster para uma camada. Esta entrada mostra quantas vezes o pedido para esta camada é avançada por peer nos servidores WMS para o resultado.
- **Largura Fixa, Altura Fixa** — Quando uma camada pode ou não pode ter dimensões de pixels da fonte fixadas. Esta versão QGIS assume que todas as camadas WMS têm esta definição como nada. Camadas deficientes podem ser renderizadas de forma estranha.
- **Caixa de Enquadramento WGS 84** — A caixa de enquadramento desta camada, nas coordenadas WGS 84. Alguns servidores WMS não configuram esta situação corretamente (ex.: usam as coordenadas UTM). Se for esse o caso, então a vista inicial desta camada pode ser editada com uma aparência ‘afastada’ pelo QGIS. O webmaster do WMS deve informar deste erro, que pode ser conhecido como elementos WMS XML `LatLonBoundingBox`, `EX_GeographicBoundingBox` ou `CRS:84 BoundingBox`.
- **SRC disponíveis** — As projeções desta camada podem ser renderizadas pelo servidor WMS. Os mesmos estão listados num formato nativo WMS.
- **Disponível em estilo** — Os estilos da imagem desta camada podem ser renderizadas pelo o servidor WMS.

Mostra a legenda gráfica do WMS na tabela de conteúdos e compositor

O provedor de dados WMS QGIS é capaz de exibir um gráfico legenda na tabela da lista de camadas conteúdos e no compositor mapa. A legenda WMS será exibida somente se o servidor WMS suportar `GetLegendGraphic` e a camada tiver `getCapability url` especificado, para que você também tenha como escolher um estilo para a camada.

Se uma `legendGraphic` está disponível, será mostrada abaixo da camada. É pequena, e você tem que clicar sobre ela para abrir na dimensão real (devido à limitação arquitetônica `QgsLegendInterface`). Clicando sobre a legenda da camada abrirá um quadro com a legenda em resolução máxima.


No compositor de impressão, a legenda será integrada a ele (baixando) dimensão originária. Resolução do gráfico da legenda pode ser definido nas propriedades do item sob `Legenda -> WMS LegendGraphic` para atender às suas necessidades de impressão

A legenda irá exibir informações contextuais com base em sua escala atual. A legenda WMS será exibida somente se o servidor WMS tiver capacidade `GetLegendGraphic` e a camada `getCapability url` for especificada, então você tem que selecionar um estilo.

Limitações do Cliente WMS

Nem todas as funcionalidades possíveis do cliente WMS foram incluídas nesta versão do QGIS. Seguem-se algumas das mais notáveis exceções .

Editando as Configurações da Camada WMS

Uma vez completo o procedimento  Adicionar camada WMS, não existe forma para alterar as configurações. Uma forma alternativa é apagar a camada completamente e começar de novo.

Servidores WMS que Requerem Autenticação

Atualmente o acesso público e serviços WMS protegidos são suportados. Os servidores WMS protegidos podem ser acedidos pela autenticação pública. Pode adicionar credenciais (opcionais) quando adiciona o servidor WMS. Veja a seção *Selecionando os Servidores WMS/WMTS* para detalhes.


Dica: Acedendo a camadas protegidas OGC

Se necessitar de proteger camadas com outros métodos seguros além da autenticação básica, pode usar o InteProxy como proxy transparente, que suporta vários métodos de suporte. Mais informação pode ser encontrada no manual InteProxy no sítio na internet <http://inteproxy.wald.intevation.org>.

Dica: QGIS WMS Mapserver

Desde a Versão 1.7, o QGIS tem uma implementação própria do Mapserver WMS 1.3.0. Leia mais sobre este assunto no capítulo *QGIS com Servidor de Dados OGC*.

14.1.2 WCS Cliente

 O Serviço Web de Cobertura (WCS) fornecem acesso aos dados matriciais de forma a serem úteis à renderização cliente, como entrada a modelos científicos, e para outros clientes. O WCS pode ser comprado ao WFS e ao WMS. Como as instâncias WMS e WFS, o WCS permite aos clientes escolher porções de informação salvadas de servidores baseados condicionantes espaciais e outros critérios de consulta.

O QGIS tem um fornecedor WCS nativo e suporta a versão 1.0 e 1.1 (que são significativamente diferentes), mas atualmente se dá preferência ao 1.0, porque o 1.1 tem vários problemas (ex.: cada servidor implementa de formas diferentes com várias particularidades).

O fornecedor WCS nativo gere todos pedidos de rede e usa as configurações padrões de rede do QGIS (especialmente o proxy). É também possível selecionar o modo de cache ('sempre cache', 'preferir cache', 'preferir rede', 'sempre rede') e fornece também suporte à seleção de posição do tempo se o domínio temporal é oferecido pelo servidor.



14.1.3 WFS e WFS-T Cliente

No QGIS, a camada WFS comporta-se de forma semelhante a uma camada vetorial. Pode identificar e selecionar elementos e ver a tabela de atributos. Desde o QGIS 1.6 a edição WFS-T é também suportada.

De uma forma geral adicionar uma camada WFS é muito semelhante ao procedimento usado com o WMS. A diferença é que não existe servidores padrões definidos, portanto nós teremos de adicionar o nosso.

Carregando uma camada WFS

Como exemplo nós usaremos o servidor WFS DM Solutions e exibiremos uma camada, O URL é: http://www2.dmsolutions.ca/cgi-bin/mswfs_gmap

1. Clique na ferramenta  Adicionar Camada WFS na barra de ferramentas Camadas, a janela *Adicionar Camada WFS do Servidor* aparecerá.
2. Clique em [Novo].
3. Introduza 'DM Solutions' como nome.
4. Introduza o URL (veja em cima).
5. Clique [OK].
6. Escolha 'DM Solutions' da lista *Ligações do Servidor* .

7. Clique [**Ligar**].
8. Espere pela lista de camadas a ser preenchida.
9. Selecione a camada *Parques* da lista.
10. Clique [**Aplicar**] para adicionar a camada ao mapa.

Repare que as configurações proxy que definiu nas suas preferências são também reconhecidas.

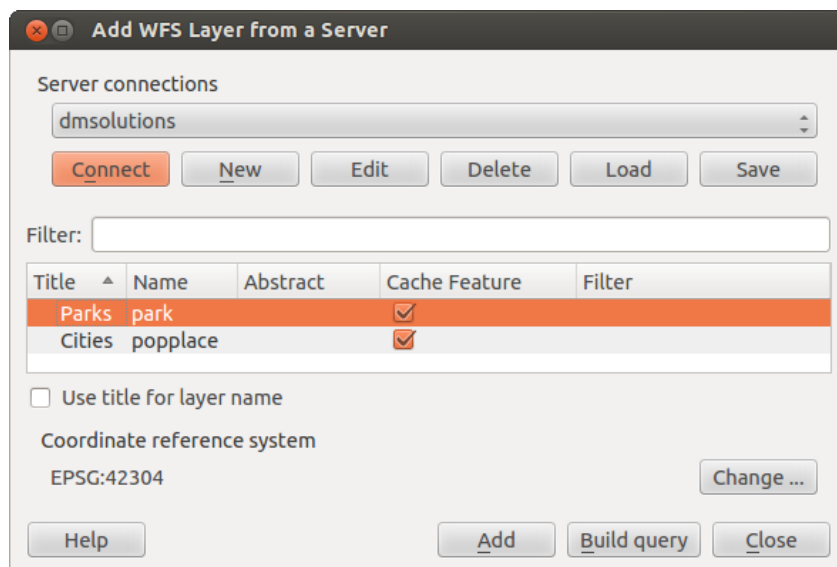


Figura 14.4: Adicionando uma camada WFS

Você notará que o progresso do download é visualizado no canto inferior esquerdo da janela principal QGIS. Uma vez que a camada é carregada, você pode identificar e selecionar uma província ou duas e vê-la na tabela de atributos.

Apenas o WFS 1.0.0 é suportado. Até este momento não houve muitos testes contra versões WFS implementadas em outros servidores WFS. Se encontrar problemas com outros servidores WFS, por favor não hesite em contatar a equipe de desenvolvimento. Por favor dirija-se à Seção *Ajuda e Suporte* para mais informação sobre as listas de discussões.

Dica: Encontrando Servidores WFS

Pode encontrar servidores WFS adicionais usando o Google ou outro motor de busca favorito. Existe um número de listas com URLs públicos, alguns têm manutenção outros não.

14.2 QGIS com Servidor de Dados OGC

O Servidor QGIS é uma implementação WMS 1.3, WFS 1.0.0 e WCS 1.1.1 de código aberto que, em adição, implementam características avançadas de cartografia para mapas temáticos. O Servidor QGIS é uma aplicação FastCGI/CGI (Common Gateway Interface) escrita em C++ que funciona juntamente com um Servidor Web (ex.: Apache, Lighttpd). Foi fundado pelos Projetos EU Orchestra, Sany e a cidade de Uster na Suíça.

O servidor QGIS usa o QGIS como fundo para a lógica SIG e renderização de mapas. Além disso, a biblioteca Qt é usada para gráficos e para a programação independente C++. Em contraste com outros softwares WMS, o Servidor QGIS usa regras cartográficas como linguagem de configuração, quer na configuração do servidor quer nas regras cartográficas definidas pelo utilizador.

Uns desktop QGIS e Servidores QGIS usam as mesmas bibliotecas de visualização, os mapas que são publicados na web na mesma aparência como no SIG do desktop.

Num dos seguintes manuais iremos fornecer um exemplo de configuração para definir um Servidor | qg | . Mas, por agora, nós recomendamos que leia um dos seguintes URLs para obter mais informações:

- http://karlinapp.ethz.ch/qgis_wms/
- http://hub.qgis.org/projects/quantum-gis/wiki/QGIS_Server_Tutorial
- <http://linfiniti.com/2010/08/qgis-mapserver-a-wms-server-for-the-masses/>

14.2.1 Instalação simples no Debian Squeeze

A este ponto nós daremos um exemplo simples e curto de Como fazer uma instalação para o Debian Squeeze. Os outros SO também fornecem pacotes para o Servidor QGIS. Se tiver de construir tudo da fonte, por favor consulte os URLs em baixo.

Além do QGIS e Servidor QGIS necessita de um Servidor web, no nosso caso o apache2. Você pode instalar todos os pacotes aptitude ou apt-get install juntos com outros pacotes dependentes necessários. Depois da instalação, você deve testar se o servidor web e o Servidor QGIS funcionam como esperado. Certifique-se que o servidor apache está iniciando em `/etc/init.d/apache2 start`. Abra o browser da internet e digite o URL: `http://localhost`. Se o apache estiver funcionando, você deverá ver a mensagem 'It works!'.

Agora nós testamos a instalação do QGIS Servidor. O `qgis_mapserv.fcgi` está disponível em `/usr/lib/cgi-bin/qgis_mapserv.fcgi` e fornece um wms padrão que mostra os limites do estado do Alaska. Adicione o WMS com o URL `http://localhost/cgi-bin/qgis_mapserv.fcgi` como está descrito no *Selecionando os Servidores WMS/WMTS*.

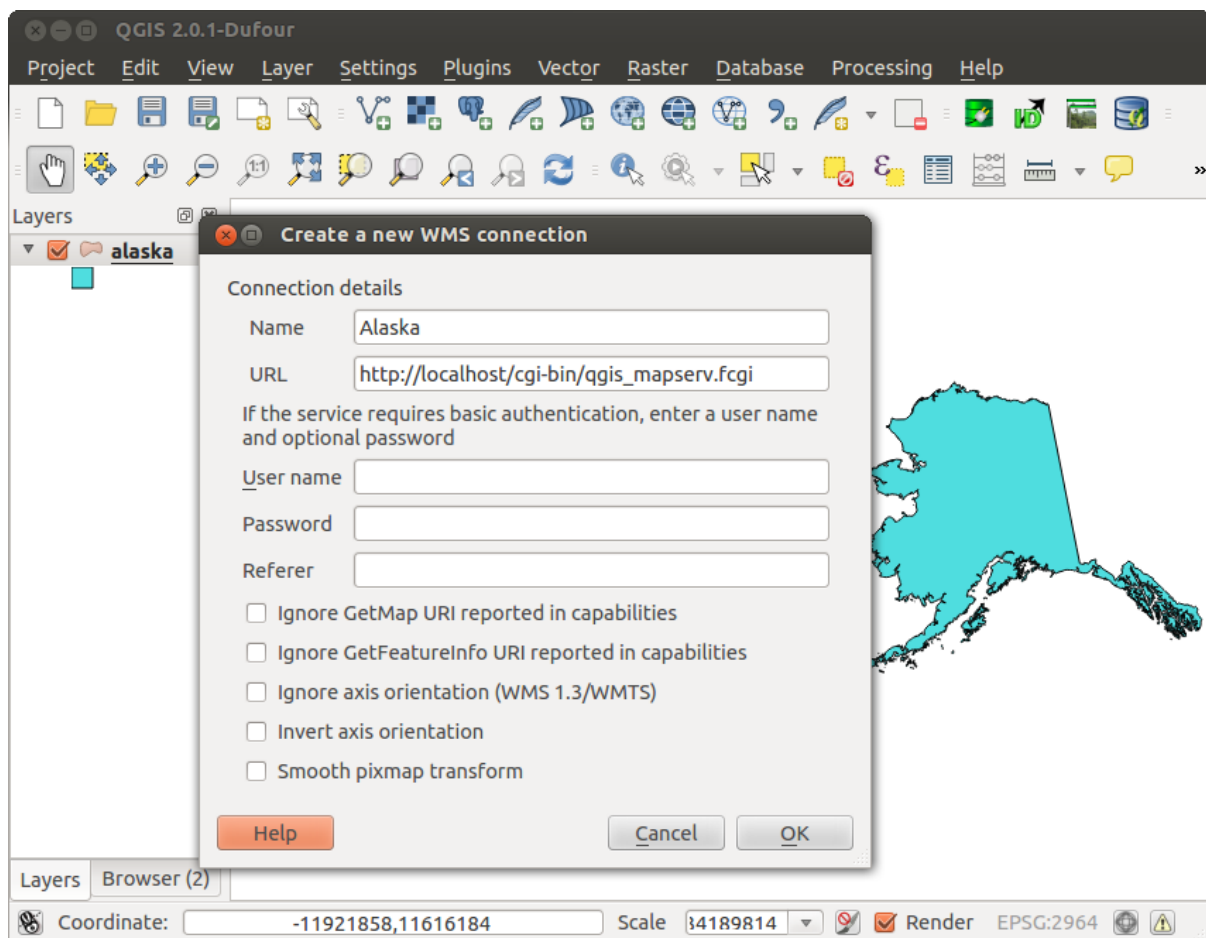


Figura 14.5: WMS padrão com os limites dos EUA incluído no Servidor QGIS (KDE) 🐧

14.2.2 Criando um WMS/WFS a partir de um projeto QGIS

Para fornecer um novo Servidor QGIS WMS, WFS or WCS nós tivemos de criar um ficheiro de projeto QGIS com alguns dados. Aqui usamos a shapefile 'alaska' do conjunto de dados de amostra do QGIS. Defina as cores e estilo das camadas no QGIS e defina o SRC do projeto, se não estiver definido.

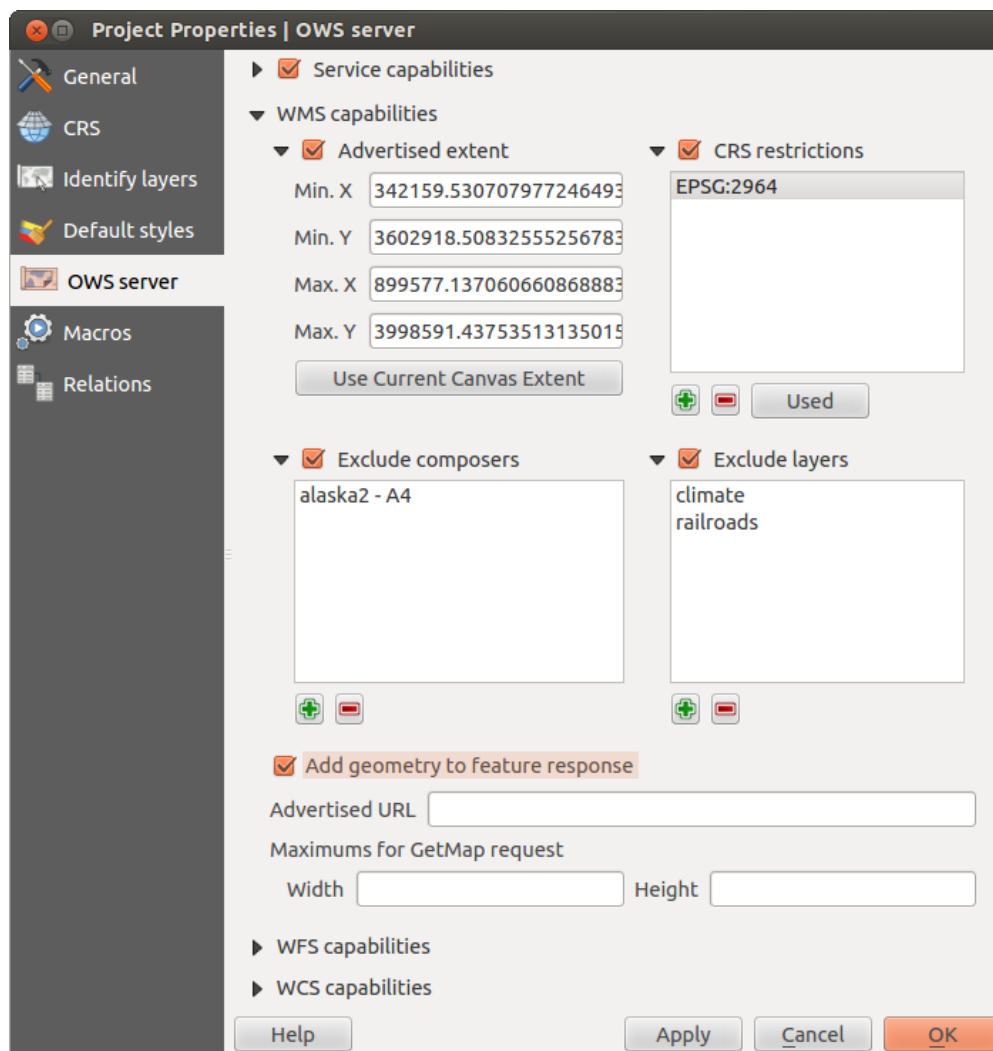




Figura 14.6: Definições para o projeto do Servidor QGIS WMS/WFS/WCS (KDE)


Em seguida vá ao menu *Servidor OWS* da janela *Projeto* → *Propriedades do Projeto* e dê alguma informação sobre o OWS nos campos da :guilabel: 'Capacidades do Serviço'. Isto irá aparecer na resposta GetCapabilities do WMS, WFS or WCS. Se você não ativar a *Capacidades do Serviço* o Servidorlqgl irá usar a informação dada no arquivo `wms_metadata.xml` localizado na pasta `cgi-bin`.

Capacidades WMS

Na seção *Capacidades WMS* você pode definir a extensão aconselhada na resposta GetCapabilities do WMS ao introduzir os valores mínimos e máximos de X e Y nos campos da *Extensão aconselhável*. Clicando *Usar Extensão atual do Mapa* define estes valores para a extensão Atualmente exibida no enquadramento do mapa QGIS. Ao ativar *Restrições SRC* pode restringir os sistemas de coordenadas referências (SRC) que o Servidor QGIS poderá oferecer para renderizar mapas. Use o botão  em baixo para selecionar os SRC a partir do Seleccionador de Sistemas de Coordenadas Referências, ou clique *Usado* para adicionar o SRC usado no projeto QGIS à lista.

Se você tiver um compositor de impressão definido no seu projeto, ele aparecerá na lista da resposta GetCapa-

bilities, e podem ser usados pelo pedido GetPrint para criar impressões, usando um dos layouts do compositor de impressão como modelo. Isto é uma extensão específica do QGIS para a especificação do WMS 1.3.0. Se quiser excluir qualquer compositor de impressão de ser publicado pelo WMS, ative *Excluir compositores* e clique no botão em baixo . Em seguida selecione o compositor de impressão da janela *Selecione o compositor de impressão* de forma a adicionar ou excluir a lista de compositores.

Se você desejar excluir uma camada ou um grupo de camadas da publicação pelo WMS, ative checkbox *Excluir Camadas* e clique no botão  abaixo. Isto abrirá a janela *Selecione camadas e grupos restritos* que irá permitir escolher as camadas e grupos que não quer que sejam publicados. Use a Shift ou Ctrl se quiser selecionar múltiplas entradas de uma vez só.

Você pode receber o GetFeatureInfo solicitado como texto simples, XML e GML. O padrão é XML, texto ou formato GML depende do formato de saída escolhido para no pedido GetFeatureInfo.

Se desejar você pode ativar o *Adicionar geometria à resposta de informação do elemento*. Isto irá incluir na resposta GetFeatureInfo das geometrias dos elementos em formato de texto. Se quiser que o Servidor QGIS avise-o dos pedidos específicos dos URLs na resposta do GetCapabilities do WMS, introduza o URL correspondente no campo *URL Anunciado*. Além disso, você pode restringir o tamanho máximo dos mapas trazido pelo pedido GetMap ao introduzir a altura e largura máxima nos campos respectivos no *Máximos para pedidos GetMap*.

Se uma das suas camadas usa a exibição Tipo do Mapa (ou seja, para mostrar expressões de texto usadas) isso vai ser listados no interior da saída GetFeatureInfo. Se a camada usa um valor Mapa para um de seus atributos, também estas informações serão mostrados na saída GetFeatureInfo.

Capacidades WFS

Na área *Capacidades WFS* você pode selecionar camadas que quer fornecer ao WFS, e especificar permissões para efetuar operações para atualizar, inserir ou apagar. Se introduzir um URL no campo *URL Anunciado* da secção *Capacidades WFS*, o Servidor QGIS irá informá-lo este URL específico na resposta GetCapabilities do WFS.

Capacidades WCS

Na área *Capacidades WCS* você pode selecionar camadas que quer fornecer ao WCS. Se introduzir um URL no campo *URL Anunciado* da secção *Capacidades WCS*, o Servidor QGIS irá informá-lo este URL específico na resposta GetCapabilities do WCS.

Agora salve a sessão em um arquivos de projeto `alaska.qgs`. Para fornecer ao projeto como um WMS/WFS, nós criamos uma nova pasta `/usr/lib/cgi-bin/project` com privilégios de administrador e adicionamos o arquivo do projeto `alaska.qgs` e a cópia do arquivo do `qgis_mapserv.fcgi` - e é tudo.

Agora testamos o nosso projeto WMS, WFS e WCS. Adicione o WMS, WFS e WCS. como está descrito em *Carregando as camadas WMS/WMTS e WFS e WFS-T Cliente* ao QGIS e carregue os dados. O URL é:

```
http://localhost/cgi-bin/project/qgis_mapserv.fcgi
```

Afinando o seu OWS

Para camadas vectoriais, o menu *Campos* da janela *Camada* → *Propriedades* permite que você defina para cada atributos se irá ser ou não publicado. Por padrão, todos os atributos são publicados pelo seu WMS ou WFS. Se quiser um atributo específico que não estiver publicado, desative a caixa correspondente na coluna *WMS* ou *WFS*.

Você pode sobrepor marcas de água sobre mapas produzidos no seu WMS adicionando anotações de texto ou anotações SVG através do arquivos de projeto. Veja *sec_annotatations* para instruções para a criação de anotações. Para anotações exibidas como marcas de água na saída do WMS, a caixa de verificação *Posição fixa do mapa* na janela *Anotação de texto* deve estar desativada. Isto pode ser acessado através de um clique duplo na anotação enquanto que uma das ferramentas de anotação está ativa. Para as anotações SVG você necessitará de configurar o projeto para salvar os diretórios absolutos (no menu *Geral* menu da janela *Projeto* → *Propriedades do Projeto*) ou modificar manualmente o diretório para a imagem SVG numa forma que represente um diretório relativo válido.

Os parâmetros extra suportados pelo pedido WMS GetMap

No pedido GetMap WMS o Servidor QGIS aceita um par de parâmetros extra em adição aos parâmetros padrão correspondendo à especificação da OGC WMS 1.3.0:

- Parâmetro **MAPA**: Semelhante ao MapServer, o parâmetro MAPA pode ser usado para especificar o caminho para o ficheiro de projecto QGIS. Pode especificar um caminho absoluto ou relativo para a localização do executável do servidor (`qgis_mapserv.fcgi`). Se não estiver especificado, o Servidor QGIS procura por ficheiros `.qgs` no directório onde o executável do servidor está localizado.

Exemplo:

```
http://localhost/cgi-bin/qgis_mapserv.fcgi?\nREQUEST=GetMap&MAP=/home/qgis/mymap.qgs&...
```

- Parâmetro **DPI**: O parâmetro DPI pode ser usado para especificar o pedido de saída da resolução.

Exemplo:

```
http://localhost/cgi-bin/qgis_mapserv.fcgi?REQUEST=GetMap&DPI=300&...
```

- Parâmetro **OPACIDADE**: A opacidade pode se definida numa camada ou ao nível do grupo. O intervalo de valores permitido vai de 0 (totalmente transparente) a 255 (totalmente opaco).

Exemplo:

```
http://localhost/cgi-bin/qgis_mapserv.fcgi?\nREQUEST=GetMap&LAYERS=mylayer1,mylayer2&OPACITIES=125,200&...
```

QGIS registro no servidor

Para registrar as solicitações de enviar ao servidor, defina as seguintes variáveis de ambiente:

- **** QGIS_SERVER_LOG_FILE ****: Especifique o caminho e o nome do arquivo. Certifique-se de que o servidor tem permissões adequadas para escrever no arquivo. Arquivo deve ser criado automaticamente, basta enviar algumas solicitações ao servidor. Se ele não estiver lá, verifique as permissões.
- **** QGIS_SERVER_LOG_LEVEL ****: Especifique o nível de log desejado. Os valores disponíveis são:
 - 0 INFO (log todos os pedidos),
 - 1 AVISO,
 - 2 CRÍTICO (log apenas erros críticos, adequado para fins de produção).

Exemplo:

```
SetEnv QGIS_SERVER_LOG_FILE /var/tmp/qgislog.txt\nSetEnv QGIS_SERVER_LOG_LEVEL 0
```

**** Nota ****

- Ao usar o módulo Fcgid usar `FcgidInitialEnv` em vez de `SetEnv`!
- Servidor de registro está habilitado também se executável é compilado no modo release.

Variáveis ambientais

- **** QGIS_OPTIONS_PATH ****: A variável especifica o caminho para o diretório com as configurações. Ele funciona da mesma forma opção `--optionspath` aplicação QGIS como. Ele está à procura de arquivo de configurações em `<QGIS_OPTIONS_PATH>/QGIS/QGIS2.ini`. Para exaple, para definir servidor QGIS em Apache para usar o arquivo de configurações `/path/to/config/QGIS/QGIS2.ini`, adicione a configuração do Apache:

```
SetEnv QGIS_OPTIONS_PATH "/path/to/config/"
```

Trabalhando com dados GPS

atualizações

15.1 Complemento GPS

15.1.1 O que é GPS?

GPS, o Sistema de Posicionamento por Satélite, é um sistema baseado em satélites que permite que qualquer pessoa com um receptor de GPS possa encontrar sua posição exata em qualquer lugar do mundo. O GPS é usado como uma ajuda nas rotas, por exemplo, em aviões, em barcos e por pedestres. O receptor de GPS utiliza os sinais de satélites para calcular a sua latitude, longitude e (às vezes) altitude. A maioria dos receptores também têm a capacidade de armazenar locais (conhecidos como “trilhas”), sequências de locais que compõem a rota prevista ou pista de circulação do pedestre ao longo do tempo. Pontos de passagem, rotas e trilhas são os três tipos de funcionalidades básicas em dados de GPS. QG mostra locais em camadas de pontos, enquanto rotas e trilhas são exibidos em cadeia de camadas de linhas.

15.1.2 Carregando dados GPS de um arquivo

Existem dezenas de diferentes formatos de arquivo para armazenar dados de GPS. O formato que QGIS usa é chamado GPX (GPS eXchange format), que é um formato padrão de intercâmbio que pode conter qualquer número de waypoints, rotas e trilhas no mesmo arquivo.

Para carregar um arquivo GPX, primeiro você precisa carregar o plugin: selecione no menu: *Plugins* -> *mActionShow PluginManager*: marque no menu: *Plugin manager*, vai abrir uma caixa de diálogo e ative *GPS Tools*. Quando este plugin é carregado, dois botões com um pequeno dispositivo GPS portátil vai aparecer na barra de ferramentas:

- Criar uma um GPX novo
- Importar GPX através da barra de ferramentas GPS

Para trabalhar com os dados do GPS, nós fornecemos um arquivo GPX exemplo disponível no QG com um conjunto de dados de amostra:: file: *qgis_sample_data / gps / national_monuments.gpx*. Consulte a seção: ref: *label_sampledata* para obter mais informações sobre os dados do exemplo.

1. Selecione no menu: *Vector* -> *GPS* -> *GPS Tools* ou clique gpx importação: no ícone GPS Tools na barra de ferramentas e abra a aba: *guilabel*: Carregue o arquivo GPX. (ver figura [GPS_1](#)).
2. Navegue até a pasta *qgis_sample_data/gps/*, selecione o arquivo GPX *national_monuments.gpx* e clique **** [Abrir]. ****

Use o botão **[Procurar...]** para selecionar o arquivo GPX, então use as caixas de seleção para escolher os tipos de recursos que você deseja carregar a partir desse arquivo GPX. Cada tipo de recurso será carregado em uma camada separada quando você clicar **[OK]**. O arquivo *national_monuments.gpx* inclui apenas waypoints.

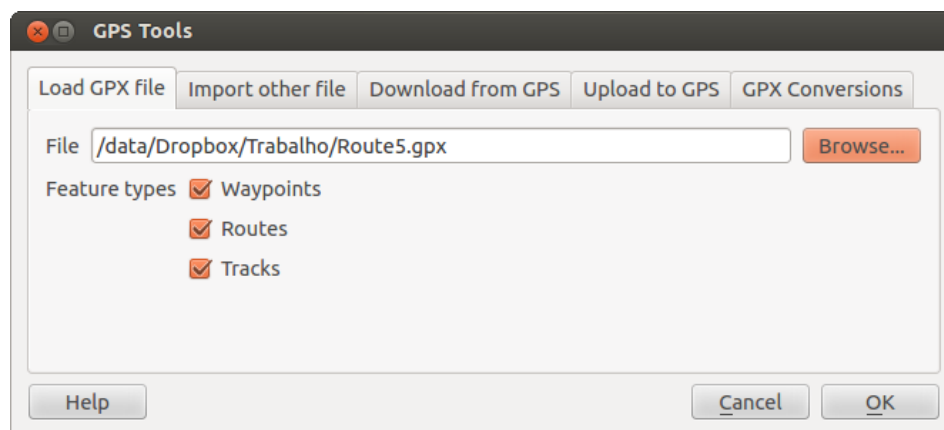


Figura 15.1: Janela de diálogo *Ferramentas GPS* 

Nota: Unidades de GPS permitem armazenar dados em diferentes sistemas de coordenadas. Ao baixar um arquivo GPX (a partir de sua unidade de GPS ou site) e, em seguida, colocá-lo no QG, certifique-se de que os dados armazenados no arquivo GPX utiliza WGS 84 (latitude / longitude). QG espera que esta é a especificação oficial do GPX. Veja em: <http://www.topografix.com/GPX/1/1/>.

15.1.3 GPSTabel

Como QG usa arquivos GPX como padrão, você precisa encontrar uma maneira de converter outros formatos de arquivos GPS para GPX. Isto pode ser feito por muitos formatos usando o programa GPSTabel que está disponível em <http://www.gpsbabel.org>. Este programa também pode transferir dados GPS entre um computador e um dispositivo GPS. QG usa GPSTabel para fazer essas coisas, por isso é recomendado que você instale. No entanto, se você quiser apenas para carregar os dados de GPS a partir de arquivos GPX você não vai precisar dele. A versão 1.2.3 do GPSTabel é conhecido por trabalhar com QG, mas você pode usar versões posteriores, sem problemas.

15.1.4 Importando dados GPS

Para importar dados de GPS de um arquivo que não é GPX, você irá usar a ferramenta: guilabel: na caixa de diálogo Ferramentas GPS. Aqui, você seleciona o arquivo que deseja importar (e o tipo de arquivo), e onde você deseja armazenar o arquivo GPX convertido e qual o nome da nova camada. Note-se que nem todos os formatos de dados GPS irá suportar todos os três tipos de recurso, portanto, para muitos formatos você só será capaz de escolher entre um ou dois tipos.

15.1.5 Descarregando dados de um receptor GPS

QG pode usar GPSTabel para baixar dados de um dispositivo GPS diretamente com novas camadas vetoriais. Para isso, use o: guilabel: -> 'Faça o download da guia GPS' na caixa de diálogo Ferramentas GPS (ver [Figure_GPS_2](#)). Aqui, selecionamos o tipo de dispositivo GPS, a porta que ele está conectado (ou USB se o seu GPS suporta isso), local onde deseja fazer o download, o arquivo GPX onde os dados devem ser armazenados e o nome da nova camada.

O tipo de dispositivo que você selecionar no menu do aparelho GPS determina como GPSTabel tenta se comunicar com o seu dispositivo GPS. Se nenhum dos tipos disponíveis funcionar com o seu dispositivo GPS, você pode criar um novo tipo (ver seção ref: definindo-novo-dispositivo).

A porta pode ser um nome de arquivo ou algum outro nome que o seu sistema operacional utiliza como referência para a porta física em seu computador que o dispositivo GPS está conectado. Também pode ser simplesmente USB, para unidades de GPS habilitados para USB.

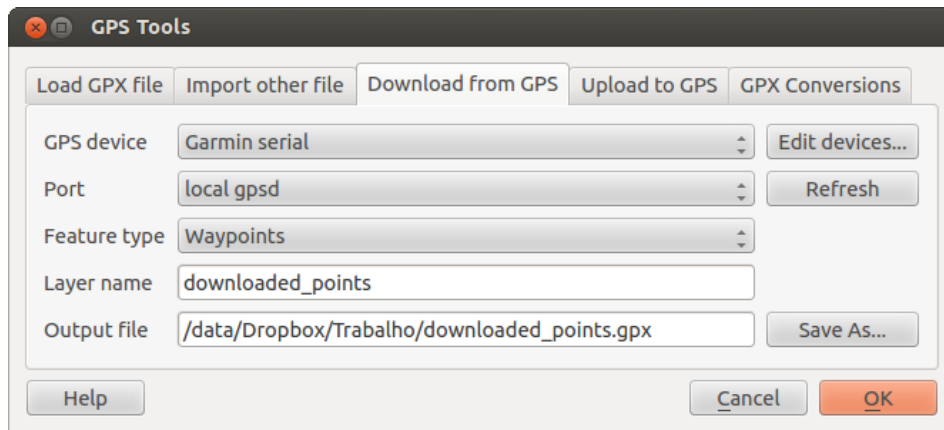


Figura 15.2: Ferramenta de download

- 🐧 On Linux, this is something like /dev/ttyS0 or /dev/ttyS1.
- 🪟 No Windows, é “COM1” ou “COM2”.

Quando você clica [OK], os dados serão transferidos a partir do dispositivo e aparece como uma camada em QG.

15.1.6 Carregando dados no receptor GPS

Você também pode enviar dados diretamente de uma camada de vetor em QGIS a um dispositivo GPS usando o guilabel: ‘Carregar o GPS: caixa de diálogo Ferramentas GPS’. Para fazer isso, basta selecionar a camada que você deseja carregar (que deve ser uma camada GPX), o tipo de dispositivo de GPS, e a porta (ou USB) que está conectada. Assim com a ferramenta de download, você pode especificar novos tipos de dispositivos caso o dispositivo não estiver na lista.

Esta ferramenta é muito útil em combinação com as capacidades de edição de vetores de QGIS. Ela permite que você carregue um mapa, crie pontos de passagem e rotas, para enviá-los e usá-los no seu dispositivo GPS.

15.1.7 Definindo novos tipos de dispositivo

Há lotes de diferentes tipos de dispositivos GPS. A QGIS desenvolvedores não pode testar todos eles, por isso, se você tem um dispositivo que não funciona com qualquer um dos tipos listados no guilabel: -> download a partir de GPS e -> guilabel: Carregar o GPS, você pode definir o seu próprio tipo de dispositivo. Você pode fazer isso usando o editor de dispositivo GPS, que você inicia clicando em: [Editar dispositivos] -> botão download ou upload.

Para definir um novo dispositivo, basta clicar no botão [Novo dispositivo], digite um nome, digite os comandos de download e upload para o seu dispositivo e clique no botão [Atualizar]. O nome será listado nos menus do aparelho nas janelas de Atualizar e Baixar - pode ser qualquer linha. O comando de download é o comando que é usado para baixar os dados do dispositivo para um arquivo GPX. Este será, provavelmente, um comando GPSBabel, mas você pode usar qualquer outro programa com linha de comando para criar um arquivo GPX. QGIS irá substituir as palavras-chave “% type”, “% in”, e “% out” quando executar o comando.

“% type” será substituído por “-w” se você estiver baixando “Locais”, “-r” para “Rotas” e “-t” para “Trilhas”. Estas são as opções de comando que o GPSBabel possui para download.

“% in” será substituído pelo nome da porta que você escolher durante o download e “% out” será substituído pelo nome que você escolher para o arquivo GPX onde os dados baixados devem ser armazenados. Então, se você criar um tipo de dispositivo com o comando “gpsbabel % -i Garmin -o gpx% em% out” (este é o comando de download para um tipo de dispositivo pré-definido ‘Garmin série’) e, em seguida, usá-lo para baixar locais de porta “/ dev / ttyS0” para o arquivo “output.gpx”. QGIS irá substituir as palavras-chave e executará o comando “gpsbabel -w -i Garmin -o gpx / dev / ttyS0 saída. gpx”.

Os comandos indicados nas linhas de ‘Carregar’ são aqueles utilizados para carregar os dados no dispositivo. As mesmas palavras-chave são utilizadas, porém %in é substituído pelo nome do arquivo GPS que é utilizado e %out é substituído pelo nome da porta.

Você pode aprender mais sobre GPSBabel e as opções de linha de comando disponíveis em <http://www.gpsbabel.org>.

Depois de ter criado um novo tipo de dispositivo, ele aparecerá na lista de dispositivos para baixar e atualizar as ferramentas.

15.1.8 Download de pontos/trilhas para GPS

Conforme descrito nas seções anteriores, o QGIS usa GPSBabel para baixar pontos / caminhos diretamente no projeto. QGIS abre a caixa com um perfil pré-definido para baixar de dispositivos Garmin. Infelizmente há um [bug #6318](#) que não permite criar outros perfis, por isso o download diretamente no QGIS usando as ferramentas de GPS está no momento limitado a unidades USB Garmin.

Garmin GPSMAP 60cs

MS Windows

Instalar os drivers USB da Garmin através do link http://www8.garmin.com/support/download_details.jsp?id=591

Ligue o aparelho de GPS, abra e digite “type = serial Garmin” e “port = usb”. Preencha os campos guilabel: ‘Nome da Camada’ e: guilabel: ‘Arquivo de Saída’. Às vezes, parece ter problemas em uma determinada pasta, use algo como “c:temp” que geralmente funciona.

Ubuntu/Mint GNU/Linux

É necessário em primeiro lugar discutir sobre as permissões do dispositivo, conforme descrito no site: https://wiki.openstreetmap.org/wiki/USB_Garmin_on_GNU/Linux. Você pode tentar criar um arquivo: ‘/ etc / udev / rules.d / 51 garmin.rules’ contendo esta regra

```
ATTRS{idVendor}=="091e", ATTRS{idProduct}=="0003", MODE="666"
```

Depois disso, é necessário ter certeza de que o módulo do kernel “garmin_gps” não está carregando.

```
rmmod garmin_gps
```

e, em seguida, você pode usar as ferramentas de GPS. Infelizmente, não parece ser um [bug#7182](#) e, geralmente, o QGIS congela várias vezes antes do bom trabalho de operação.

Registro de dados BTGP-38KM (somente Bluetooth)

MS Windows

O erro já referido não permite fazer o download dos dados a partir do QGIS, por isso é necessário utilizar o GPSBabel a partir do comando ou usando a sua interface.

```
gpsbabel -t -i skytraq,baud=9600,initbaud=9600 -f COM9 -o gpx -F C:/GPX/aaa.gpx
```

Ubuntu/Mint GNU/Linux

Use o mesmo comando (ou as configurações do GPSBabel GUI) no Windows. No Linux de alguma forma pode obter uma mensagem

```
skytraq: Too many read errors on serial port
```

seria a questão de desligar e ligar o registro de dados novamente.

Registro de dados BlueMax GPS-4044 (BT e USB).

MS Windows

Nota: Precisa instalar os drivers antes de usá-lo no Windows 7. Veja no site do fabricante qual o download adequado.

Baixando com GPSTabel, ambos com USB e BT retorna sempre um erro.

```
gpsbabel -t -i mtk -f COM12 -o gpx -F C:/temp/test.gpx
mtk_logger: Can't create temporary file data.bin
Error running gpsbabel: Process exited unsuccessfully with code 1
```

Ubuntu/Mint GNU/Linux

Com USB

Depois de ter ligado o cabo e utilizar o comando “dmesg” para a porta que está sendo usada, digite por exemplo “/dev/ttyACM3”. Então, será usado o GPSTabel para CLI ou GUI.

```
gpsbabel -t -i mtk -f /dev/ttyACM3 -o gpx -F /home/user/bluemax.gpx
```

Com Bluetooth





Use Blueman como gerenciador de dispositivos para associar o dispositivo e disponibilizá-lo através de uma porta do sistema executando o GPSTabel.

```
gpsbabel -t -i mtk -f /dev/rfcomm0 -o gpx -F /home/user/bluemax_bt.gpx
```

15.2 Rastreo GPS em tempo real

Para ativar o rastreamento GPS ao vivo QGIS, você precisa selecionar :menuselection: ‘Exibir -> Painéis’ **lcaixal** *informação do GPS*. Você receberá uma nova janela ancorada no lado esquerdo da tela.


Existem quatro telas possíveis nesta janela de monitoramento GPS

-  coordenadas de posição do GPS e uma interface para inserir manualmente vértices e feições
-  força conexões do sinal GPS via satélite
-  tela GPS mostrando número e posição polar dos satélites
-  tela de opções do GPS (ver [figure_gps_options](#))

Com um receptor GPS conectado (tem de ser compatível com o seu sistema operacional), um simples clique em **[Conectar]** liga o GPS no QGIS. Um segundo clique (agora em **[Desconectar]**) desliga o receptor GPS do seu computador. Para GNU / Linux, suporte `gpsd` é integrado para apoiar a conexão para a maioria dos receptores GPS. Portanto, você primeiro tem que configurar `gpsd` corretamente para conectar ele ao QGIS.

Aviso: Se você quer gravar a sua posição para a tela, você tem que criar uma nova camada vetorial primeiro e mudar para o estado editável para gravar o seu caminho.

15.2.1 Posição e atributos adicionais

 Se o GPS está recebendo sinais de satélites, você vai ver a sua posição em latitude, longitude e altitude, juntamente com atributos adicionais.

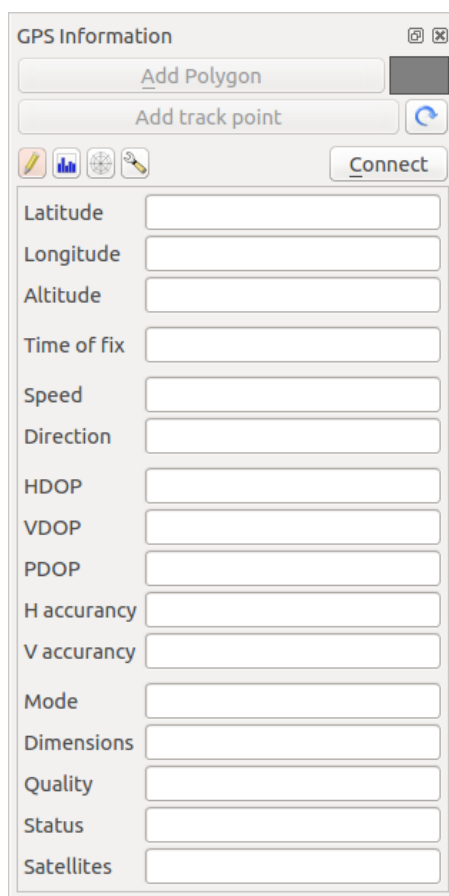



Figura 15.3: Posição GPS e atributos adicionais 🐧

15.2.2 Força do sinal GPS

 Aqui, você pode ver a intensidade do sinal dos satélites que está recebendo sinais.

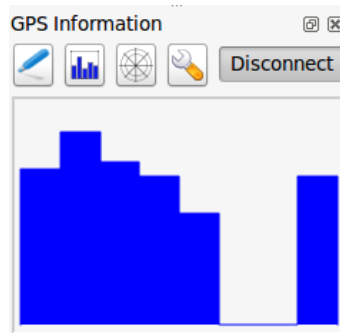




Figura 15.4: Intensidade do sinal de rastreamento GPS 

15.2.3 Posição polar GPS

 Se você quer saber onde estão todos os satélites conectados, é preciso mudar para a tela polar. Você também pode ver os números de identificação dos satélites conectados.

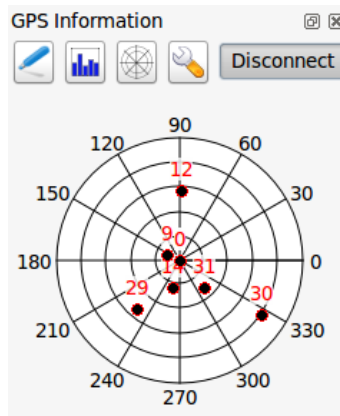




Figura 15.5: Posição polar de rastreamento GPS 

15.2.4 Opções GPS

 Em caso de problemas de conexão, você pode alternar entre:

- Autodetectar
- Interno
- Dispositivo serial
- `gpsd` (selecionando o anfitrião, porta e o dispositivo GPS que está ligado)

Um clique sobre **** [Conectar] **** reinicia a ligação ao receptor GPS.

Você pode ativar **lcaixal salvar automaticamente feições adicionadas** quando você está em modo de edição. Ou você pode ativar **lcaixal :guilabel:'Adicionar pontos automaticamente'** para a tela mapa com uma determinada largura e cor.

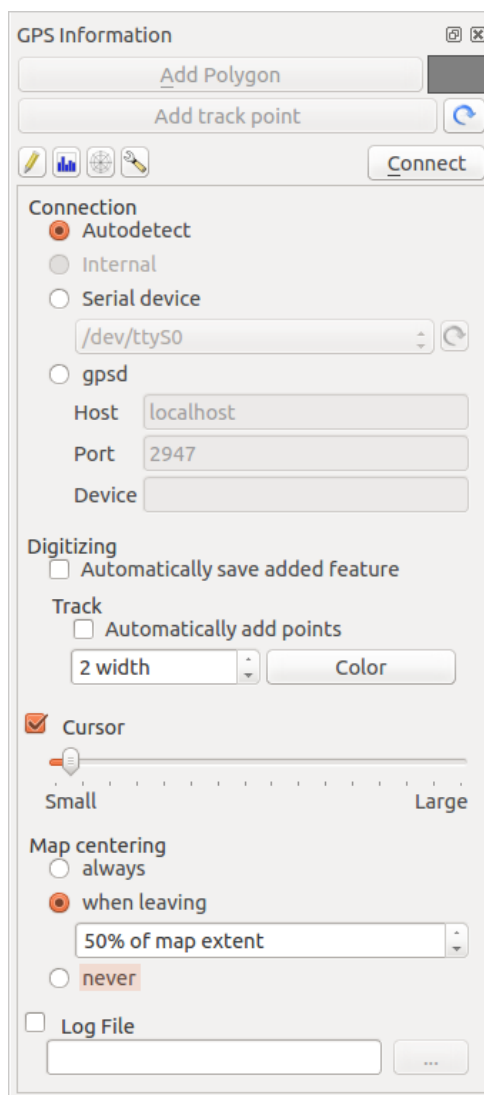




Figura 15.6: Janela de opções de rastreamento GPS 🐧

Ativando **lcaixal** *Cursor*, você pode usar um controle deslizante **lbarral** para diminuir e aumentar a posição do cursor na tela.

Ativando  *Centragem do Mapa* permite decidir de que forma a tela será atualizada. Isto inclui 'sempre', 'quando sair', se suas coordenadas registradas começar a se mover para fora da tela, ou 'nunca', para manter a extensão do mapa.

Finalmente, você pode ativar **lcaixal** *Arquivo Log* e definir um caminho e um arquivo onde as mensagens de log sobre o rastreamento GPS serão registradas.

Se você quiser definir uma feição manualmente, você tem que voltar para  Posição e clique em [**Adicionar Ponto**] ou [**Adicionar ponto de passagem**].

15.2.5 Conectar um GPS com Bluetooth para rastreamento ao vivo


Com QGIS você pode conectar um GPS via Bluetooth para a coleta de dados em campo. Para executar esta tarefa, você precisa de um dispositivo GPS com Bluetooth e um receptor Bluetooth em seu computador.

Em primeiro lugar você deve deixar seu aparelho de GPS ser reconhecido e emparelhado com o computador. Ligue o GPS, vá até o ícone Bluetooth na área de notificação e busca de um novo dispositivo.

No lado direito da máscara de Seleção do aparelho certifique-se que todos os dispositivos são selecionados para que a sua unidade de GPS provavelmente aparecerá entre os disponíveis. Na próxima etapa, um serviço de conexão serial deve estar disponível, selecione-o e clique no botão [**Configurar**].

Lembre-se o número da porta COM atribuído à conexão GPS como resultante pelas propriedades Bluetooth.

Depois que o GPS foi reconhecido, faça o emparelhamento para a conexão. Normalmente, o código de autorização são 0000.


Agora abra o painel *Informação GPS* e mudar para tela de opções de GPS . Selecione a porta COM atribuído à conexão GPS e clique em [**Conectar**]. Depois de um tempo um cursor indicando a sua posição deve aparecer.

Se QGIS não pode receber dados do GPS, então você deve reiniciar seu dispositivo GPS, espere 5-10 segundos, em seguida, tente se conectar novamente. Normalmente este trabalho soluciona. Se você receber um erro de conexão novamente certifique-se que você não tem outro receptor Bluetooth perto de você, emparelhado com a mesma unidade GPS.

15.2.6 Utilizando GPSPMAP 60cs

MS Windows

Maneira mais fácil de fazer o trabalho é a utilização de um middleware (freeware, não aberto) chamado *GPSPGate* <<http://update.gpsgate.com/install/GpsGateClient.exe>> _.

O lançamento do programa, faça a varredura para dispositivos GPS (funciona tanto para USB e os BT) e depois no QGIS basta clicar [**Conectar**] no painel de monitoramento ao vivo usando o modo  *Autodeteção*.

Ubuntu/Mint GNU/Linux

Como para Windows a maneira mais fácil é usar um servidor no meio, neste caso GPSPD, assim

```
sudo apt-get install gpsd
```

Em seguida, carregar o módulo do kernel *garmin_gps*

```
sudo modprobe garmin_gps
```

E, em seguida, ligue a unidade. Em seguida, verifique com *dmesg* o dispositivo real que está sendo utilizado bu da unidade, por exemplo */dev/ttyUSB0*. Agora você pode iniciar *gpsd*

```
gpsd /dev/ttyUSB0
```


E, finalmente, conectar-se com a ferramenta de acompanhamento ao vivo do QGIS.

15.2.7 Usando BTGP-38KM datalogger (somente Bluetooth)

Usando GPSD (no Linux) ou GPSTGate (no Windows) é fácil.

15.2.8 Usando datalogger BlueMax GPS-4044 (ambos BT e USB)

MS Windows

O acompanhamento ao vivo funciona tanto para modos BT USB e, usando GPSTGate ou mesmo sem ele, é só usar o modo  *Autodetectar*, ou apontar a ferramenta da porta direita.

Ubuntu/Mint GNU/Linux

Para USB

O acompanhamento ao vivo funciona tanto com GPSD

```
gpsd /dev/ttyACM3
```

ou sem ela, ligando a ferramenta de acompanhamento ao vivo QGIS diretamente para o dispositivo (por exemplo, /dev/ttyACM3).

Para Bluetooth

O acompanhamento ao vivo funciona tanto com GPSD

```
gpsd /dev/rfcomm0
```











ou sem ela, ligando a ferramenta de acompanhamento ao vivo QGIS diretamente para o dispositivo (por exemplo, /dev/rfcomm0).

Atualizações

Integração com SIG GRASS



O complemento GRASS oferece acesso ao SIG GRASS bases de dados e funcionalidades (ver GRASS-PROJECT *Referências Bibliográficas e Web*). Isto inclui a visualização de camadas raster GRASS e vetoriais, digitalização camadas vetoriais, atributos vetor de edição, criando novas camadas de vetores e análise de dados GRASS em 2D e 3D com mais de 400 módulos do GRASS.

Nesta seção, nós vamos apresentar as funcionalidades do complemento e dar alguns exemplos de gestão e trabalhos com dados do GRASS. As seguintes características principais são fornecidas com o menu da barra de ferramentas, quando você inicia o complemento GRASS, como descrito na seção: ref: *sec_starting_grass*:

-  Abrir mapset
-  Novo mapset
-  Fechar mapset
-  Adicionar uma camada vetorial do GRASS
-  Adicionar uma camada raster do GRASS
-  Criar um novo vetor do GRASS
-  Editar camada vetorial do GRASS
-  Abrir ferramentas GRASS
-  Mostrar a região atual do GRASS
-  Edita região atual do GRASS

16.1 Iniciando o complemento GRASS







Para usar as funcionalidades GRASS e/ou visualizar camadas vetoriais e raster GRASS no QGIS, você deve selecionar e carregar o complemento GRASS com o Gerenciador de complementos. Para isso, clique no menu:

menuselection: *Complementos* →  *Gerenciador de complementos*, selecione  :guilabel:‘GRASS →‘ e clique [OK].

Agora você pode começar a carregar camadas raster e vetoriais a partir de um GRASS existente: arquivo: *LOCALIZAÇÃO* (ver seção: ref: *sec_load_grassdata*). Ou você cria um novo GRASS: arquivo: *LOCALIZAÇÃO* com QGIS (ver seção: ref: *sec_create_loc*) e importa alguns dados raster e vetor (ver Seção: ref: *sec_import_loc_data*) para futura análise com a caixa de ferramentas GRASS (ver seção: ref: *subsec_grass_toolbox*).

16.2 Carregando camadas raster e vetorial GRASS

Com o complemento GRASS, você pode carregar camadas vetoriais ou raster utilizando o botão apropriado no menu da barra de ferramentas. Como exemplo, vamos usar o conjunto de dados Alaska do QGIS (ver seção *Amostra de Dados*). Ele inclui uma pequena amostra GRASS LOCALIZAÇÃO com três camadas vetoriais e um mapa de elevação raster.

1. Crie uma nova pasta chamada `grassdata`, baixe o banco de dados 'Alaska' do QGIS `qgis_sample_data.zip` de <http://download.osgeo.org/qgis/data/> e descompacte o arquivo em `grassdata`.
2. Inicie o QGIS.
3. Se ainda não tiver feito em uma seção anterior QGIS, carregue o complemento GRASS clicando no *Complementos* →  *Gerenciador de Complementos* e ativar **laixal** GRASS. A barra de ferramentas do GRASS aparecerá na janela principal QGIS.
4. Na barra de ferramentas GRASS, clique no  :sup: ícone *Abrir mapset* para abrir o :guilabel: assistente *MAPSET*.
5. Pelo `GISDBASE`, procure e selecione ou digite o caminho para a pasta recém-criada `grassdata`.
6. Você agora será capaz de selecionar o *LOCATION*  `alaska` e o *MAPSET*  `demo`.
7. Clique **[OK]**. Observe que algumas ferramentas anteriormente desabilitadas na barra de ferramentas GRASS agora estão habilitadas.
8. Clique em  *Adicionar camada raster GRASS*, escolha o nome do mapa: arquivo: `gtopo30` e clique **[OK]**. A camada de elevação será visualizada.
9. Clique em  *Adicionar camada vetorial GRASS*, escolher nome do mapa `alaska` e clique **[OK]**. Os limites da camada vetor Alaska será coberto na parte superior do mapa `gtopo30`. Você poderá adaptar as propriedades da camada descrita no capítulo *Janela de Propriedades de Vetor* (e.x., mudar opacidade, cor de contorno e preenchimento).
10. Também carregar as outras duas camadas vetoriais, `rios` e `aerportos`, e adaptar suas propriedades.

Como você vê, é muito simples carregar camadas raster e vetor GRASS no QGIS. Consulte as seguintes seções para a edição de dados GRASS e criar uma nova LOCALIZAÇÃO. Mais amostra GRASS locais estão disponíveis no site do GRASS em <http://grass.osgeo.org/download/sample-data/>.

Dica: Carregando dados GRASS

Se você tiver problemas ao carregar dados ou o QGIS termina de forma anormal, certifique-se de ter carregado o complemento GRASS corretamente conforme descrito na seção *Iniciando o complemento GRASS*.

16.3 Localização e MAPSET GRASS

Dados do GRASS são armazenados em um diretório chamado de GISDBASE. Este diretório, muitas vezes chamado `grassdata`, deve ser criado antes de começar a trabalhar com o complemento GRASS no QGIS. Dentro deste diretório, os dados SIG GRASS são organizados por projetos armazenados em subdiretórios chamados *LOCATIONS*. Cada *location* é definido por seu sistema de coordenadas, projeção de mapa e fronteiras geográficas. Cada *location* pode ter vários *MAPSETS* (subdiretórios do *location*) que são usados para subdividir o projeto em diferentes tópicos ou sub-regiões, ou como espaços de trabalho para os membros da equipe (ver Neteler & Mitasova de 2008, em *Referências Bibliográficas e Web*). A fim de analisar as camadas vetoriais e raster com módulos do GRASS, você deve importá-los para um GRASS *location*. (Isto não é estritamente verdadeiro - com os módulos `grass r.external` e `v.external` você pode criar somente leitura links para GDAL externo / conjuntos de dados OGR suportados, sem importar-lhes Mas porque esta não é a maneira usual para iniciantes para trabalhar com GRASS, esta funcionalidade não será descrita aqui.)

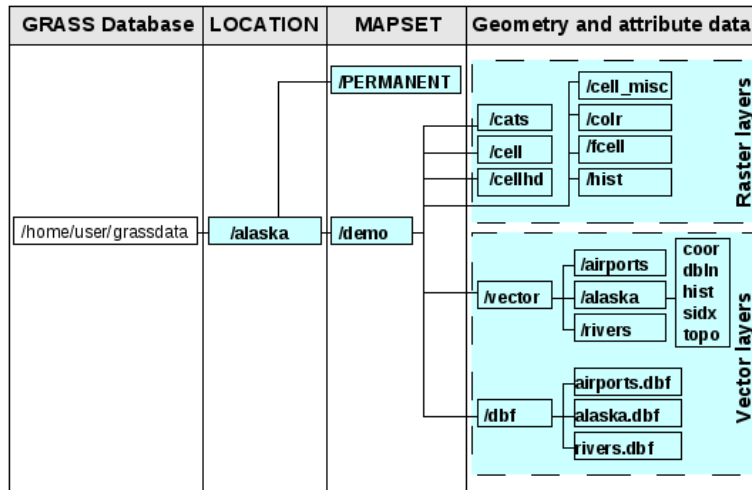




Figura 16.1: Dados GRASS em LOCATION alaska

16.3.1 Criando uma nova Localização GRASS

Como exemplo, aqui está a amostra GRASS LOCALIZAÇÃO alaska, que é projetada na projeção Albers Equal Area com unidade pés foi criada para o conjunto de dados de amostra QGIS. Esta amostra GRASS alaska local será usado para todos os exemplos e exercícios nas seções seguintes, relacionadas com GRASS. Elas são úteis para baixar e instalar o conjunto de dados em seu computador (veja *Amostra de Dados*).

1. Iniciar QGIS e certifique-se o complemento GRASS está carregado.
2. Visualize o shapefile alaska.shp (ver seção: ref: 'vector_load_shapefile') do conjunto de dados Alaska do QGIS (veja: ref: label_sampledata).
3. Na barra de ferramentas GRASS, clique no ícone  para abrir o :arquivo: Assistente 'MAPSET'.
4. Selecione uma pasta existente do banco de dados GRASS (GISDBASE) :arquivo: 'grassdata', ou criar uma para o novo :arquivo: 'local' usando um gerenciador de arquivos em seu computador. Em seguida, clique [Próximo].
5. Podemos usar este assistente para criar um novo :arquivo: MAPSET dentro de um já existente :arquivo: 'LOCATION' (ver seção :ref: 'sec_add_mapset') ou para criar um novo: arquivo: 'LOCATION'. Selecione Criar nova location (ver figure_grass_location_2).
6. Digite um nome para o :arquivo: 'local' – que usa 'alaska' - e clique [Próximo].
7. Defina a projeção, clicando no botão de rádio :guilabel: Projeção para habilitar a lista de projeções.
8. Nós estamos usando (pés) projeção Albers Equal Area Alaska. Desde que descobri que ela é representada pelo ID EPSG 2964, entramos ele na caixa de pesquisa. (Nota: Se você quiser repetir esse processo para outro location e projeção e não memorizei o ID EPSG, clique no  :sup: 'ícone status SRC' no canto inferior direito da barra de status (veja a seção :ref: label_projections)).
9. Na Filtro, insira 2964 para selecionar a projeção.
10. Clique [Próximo].
11. Para definir a região padrão, temos que entrar no arquivo: limites 'LOCALIZAÇÃO' norte, sul, leste, oeste e direções. Aqui, basta clicar no botão ** [Conjunto QG | medida] **, para aplicar a extensão da camada carregado: arquivo: alaska.shp com a extensão padrão GRASS.
12. Clique [Próximo].
13. Também precisamos definir um: arquivo: 'CONJUNTO DE MAPAS' dentro do novo: arquivo: local (isto é necessário quando criar um novo: arquivo: LOCAL). Você pode nomeá-lo o que quiser - foi utilizado 'demonstração'. GRASS cria automaticamente um arquivo: CONJUNTO DE MAPAS chamado: arquivo

‘PERMANENTE’, projetado para armazenar os dados básicos para o projeto, a sua extensão padrão e coordenar as definições do sistema (ver Neteler & Mitasova 2008 em: ref: ‘ literature_and_web’).

14. Confira o resumo para ter certeza que está correto e clique [**Fim**].
15. O novo: arquivo ‘LOCALIZAÇÃO’: ‘Alaska’, e dois: arquivos: *CONJUNTO DE MAPAS*, demonstração e” permanente ”, são criados. O conjunto de trabalho aberto no momento é ‘demonstração’, como você definiu.
16. Note que algumas das ferramentas na barra de ferramentas GRASS que foram desativadas estão agora habilitadas.

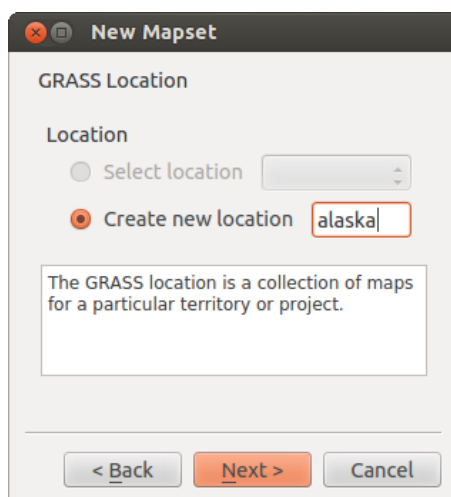



Figura 16.2: Criando um novo LOCATION GRASS ou um novo MAPSET no QGIS

Se isso parecia uma série de etapas, não é realmente tão ruim assim é uma maneira muito rápida de criar um: arquivo: *local*. O arquivo: *local* ‘Alaska’ está pronto para importação de dados (ver seção: ref: ‘ sec_import_loc_data’). Você também pode usar os dados já existentes vetoriais e amostra quadrícula no: arquivo: *local* ‘Alaska’, incluídas no | QG | base de ados ‘Alaska’: ref: dados, e ir para a seção: ref: ‘modelo vetorial’

16.3.2 Adicionando um novo MAPSET

Um usuário tem acesso apenas ao GRASS: arquivo: ‘CONJUNTO DE MAPAS’ que criou. Isso significa que, além do acesso ao seu próprio: arquivo: ‘CONJUNTO DE MAPAS’, você pode ler mapas de outros usuários de arquivo ‘CONJUNTO DE MAPAS’, mas você pode modificar ou remover apenas os mapas em seu próprio: arquivo ‘CONJUNTO DE MAPAS’.


Todos: arquivos: ‘CONJUNTO DE MAPAS’ inclui um arquivo ‘WIND’ que armazena as coordenadas dos valores e a resolução selecionada (ver Neteler & Mitasova de 2008, em: ref: ‘ literatura web’ e seção: ref: *sec_regiao_grass*).

1. Iniciar QGIS e certifique-se o complemento GRASS está carregado.
2. Na barra de ferramentas GRASS, clique no ícone  **Abrir mapset** para abrir o :arquivo: Assistente ‘MAPSET’.
3. Selecione a pasta do banco de dados GRASS (GISDBASE): arquivo *grassdata* com o arquivo: ‘ local’ ‘Alaska’, onde queremos adicionar mais um: arquivo: *CONJUNTO DE MAPAS* chamado ‘teste’.
4. Clique [**Próximo**].
5. Podemos usar este assistente para criar um novo **:arquivo: ‘MAPSET’** dentro de um já existente **:arquivo: ‘LOCATION’** ou para criar um completamente novo **:arquivo: ‘LOCATION’**. Clique no botão de rádio *Selecionar a localização* (ver *figure_grass_location_2*) e clique [**Próximo**].

6. Digite o nome: arquivo: *texte* para o novo: arquivo: 'CONJUNTO DE MAPAS'. Abaixo no assistente, você verá uma lista existente arquivo: *CONJUNTO DE MAPAS* e seus proprietários
7. Clique [Próximo], confira o resumo para se certificar de que está tudo correto e clique [Fim].

16.4 Importando dados para uma localização GRASS

Esta seção fornece um exemplo de como importar dados e vetores para o 'Alaska' GRASS: arquivo: 'local' fornecido pelo | QG | dados 'Alaska'. Por isso, usamos o mapa: arquivo: *landcover.img* e o arquivo-vetor GML : arquivo: 'lakes.gml' do | QG | dados 'Alaska' (veja: ref: *label_sampledata*).

1. Iniciar QGIS e certifique-se o complemento GRASS está carregado.
2. Na barra de ferramentas GRASS, clique no ícone  **Abriu MAPSET** para abrir o :gulabel: assistente *MAPSET*.
3. Escolha como banco de dados GRASS a pasta:arquivo: *grassdata* no QGIS base de dados Alasca, como: arquivo: *local* 'Alaska', como: arquivo: 'CONJUNTO DE MAPAS' 'demonstração' e clique ** [OK] **.
4. Agora, clique no ícone | *grass_tools* | sup: 'Abrir ferramentas GRASS . A caixa de ferramentas GRASS (ver seção: ref: '*subsec_grass_toolbox*') aparece o diálogo.
5. Para importar o mapa raster: arquivo: *landcover.img*, clique no módulo: arquivo: 'r.in.gdal' no: guilabel: *guia módulos em árvore*. Este módulo GRASS permite importar arquivos raster GDAL que suporta GRASS: arquivo: 'local'. O módulo para: arquivo: *aparece r.in.gdal*.
6. Navegue até a pasta: arquivo: *raster* no QGIS base de dados 'Alaska' e selecione o arquivo: 'landcover.img'.
7. Como nome de saída raster, definir: arquivo: *landcover_grass* e clique ** [Rodar] **. Na: guilabel: *guia Saída*, você vê o comando GRASS atualmente em execução 'r.in.gdal -o input=/path/to/landcover.img output= landcover_grass
8. Quando se diz ** Com sucesso**, clique ** [saída] **. O arquivo: camada *landcover_grass* é agora importada para a GRASS e vai ser visualizado no QGIS.
9. Para importar o arquivo vetorial GML: arquivo *lakes.gml*, clique no módulo: arquivo: 'v.in.ogr' no: guilabel: *módulos em árvores*. Este módulo GRASS permite importar arquivos vetoriais OGR apoiados na GRASS: arquivo: *local*. A caixa de diálogo para o arquivo *aparece v.in.ogr*.
10. Navegue até a pasta: arquivo: *gml* no QGIS base de dados 'Alaska' e selecione o arquivo: 'lakes.gml' como arquivo OGR.
11. Como o nome de saída do vetor, defina: arquivo: *lakes_grass* e clique ** [Ir] **. Você não tem de se preocupar com as outras opções neste exemplo. No: guilabel: *saídas* você vê o comando GRASS atualmente em execução 'v.in.ogr -o dsn = / path / to / lakes.gml output = *lacos_grass*'.
12. Quando se diz **finalizado com sucesso** , clique ** [saída] **. O arquivo: *vetorial lakes_grass* agora é importado para o GRASS e será visualizado no QGIS.

16.5 O modelo de dados vetorial do GRASS

É importante compreender o modelo de dados vetor GRASS antes da digitalização.

Em geral, GRASS utiliza um modelo de vetor topológico.

Isto significa que as áreas não são representadas como polígonos fechados, mas por um ou mais limites. A fronteira entre duas áreas adjacentes é digitalizada apenas uma vez, e é compartilhada por ambas as áreas. Limites precisam estar conectados e fechados sem lacunas. Uma área é identificada (e rotulada) pelo ** centróide** da área.

Além de limites e centróides, um mapa vetor também pode conter pontos e linhas. Todos estes elementos de geometria podem ser misturados em um vetor e será representado em diferentes 'camadas' dentro de um GRASS vetor. Então, no GRASS, uma camada não é um vetor ou raster, mas um nível dentro de uma camada vetorial. Isso

é importante para distinguir cuidadosamente. (Embora seja possível misturar elementos de geometria, é incomum e, mesmo no GRASS, utilizado somente em casos especiais, tais como rede de análise vetorial. Normalmente, você deve preferir armazenar diferentes elementos geométricos em diferentes camadas.)

É possível armazenar várias “camadas” em um conjunto de dados vetoriais. Por exemplo, campos, florestas e lagos pode ser armazenado em um vetor. Uma floresta adjacente e um lago podem compartilhar o mesmo limite, mas eles têm tabelas de atributos separados. Também é possível fixar atributos limites. Um exemplo pode ser o caso em que a fronteira entre um lago e uma floresta é um caminho, para que ele possa ter uma tabela de atributos diferentes.

A “camada” do recurso é definido pela “camada” dentro do GRASS. ‘Camada’ é o número que define, se houver mais do que uma camada interior do conjunto de dados (por exemplo, se a geometria for uma floresta ou lago). Por enquanto, ele pode ser apenas um número. No futuro, o GRASS também apoiará nomes como campos na interface do usuário.

Os atributos podem ser armazenados dentro do GRASS: arquivo: ‘local’: como dBase ou SQLite3 ou em tabelas de banco de dados externos, por exemplo, PostgreSQL, MySQL, Oracle, etc


Atributos em tabelas de base de dados estão ligados a elementos geométricos usando um valor ‘categoria’.

‘Categoria’ (chave, ID) é um número inteiro ligado a geometrias primitivas, e que é usada como ligação a uma coluna chave na tabela de base de dados.

Dica: Aprendendo um Modelo Vetor GRASS

A melhor maneira de aprender os modelos de vetores GRASS, e as suas capacidades é baixando um dos muitos tutoriais GRASS onde o modelo de vetor é descrito mais profundamente. Veja <http://grass.osgeo.org/documentation/manuals/> Para mais informações, livros e tutoriais em várias línguas.

16.6 Criando uma nova camada vetorial GRASS

Para criar uma nova camada GRASS vetor com o plugin GRASS, clique no ícone  Criar novo vetor GRASS na barra de ferramentas. Digite um nome na caixa de texto, e você pode começar a digitalização das geometrias de ponto, linha ou polígono seguindo o procedimento descrito na seção: ref: ‘digitalização_grass’.

No GRASS, é possível organizar todos os tipos de tipos de geometria (ponto, linha e área) em uma camada, porque GRASS usa um modelo de vetor topológico, assim você não precisa selecionar o tipo de geometria ao criar um novo vetor no GRASS. Isso é diferente da criação de arquivos modelos com | QG | porque arquivos modelos usam o modelo de vetor (consulte a seção: ref: ‘Criação de Vetor’).

Dica: Criando uma tabela de atributos para uma nova camada vetorial GRASS

Se você deseja atribuir atributos a seus recursos de geometria digitalizados, certifique-se de criar uma tabela de atributos com colunas antes de começar a digitalização (ver [figure_grass_digitizing_5](#)).

16.7 Digitalizando e editando uma camada vetorial GRASS

As ferramentas de digitalização para as camadas vetor GRASS são acessadas usando o | grass_edit | sup:ícone *Editar camada vetorial GRASS* na barra de ferramentas. Certifique-se de que você tenha carregado um vetor GRASS e é a camada selecionada na legenda antes de clicar na ferramenta de edição. Figura [figure_grass_digitizing_2](#) mostra o diálogo de edição GRASS que é exibido quando você clica na ferramenta de edição. As ferramentas e as configurações serão discutidas nas seções seguintes.

Dica: Digitalizando polígonos no GRASS

Se você quiser criar um polígono no GRASS, primeiro você digitaliza o limite do polígono, definindo o modo como ‘Sem categoria’. Em seguida, você adiciona um centróide (ponto de etiqueta) no limite fechado, definindo

o modo para a ‘próximo não usado’. A razão para isto é que um modelo vetorial topológico liga a informação de atributos de um polígono para o centro e não para o limite.

Barra de Ferramentas

Na figura [grass_digitalização_1](#), você vê os ícones de digitalização GRASS fornecidas pelo complemento GRASS. Tabela [grass_digitalização_1](#) explica as funcionalidades disponíveis.

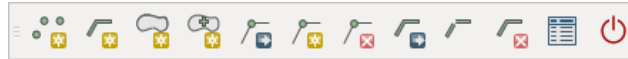


Figura 16.3: Barra de ferramentas de Digitalização GRASS

Ícone	Ferramenta	Propósito
	Novo Ponto	Digitaliza novo ponto
	Nova Linha	Digitaliza nova linha
	Nova Fronteira	Digitaliza novo limite (acaba ao selecionar nova ferramenta)
	Novo Centróide	Digitaliza novo centróide (etiqueta da área existente)
	Mova vértice	Mova um vértice da linha ou limite existente e identifique nova posição
	Adiciona vértice	Adiciona novo vértice à uma linha existente
	Exclua vértice	Exclua vértice da linha existente (confirme vértice selecionado por outro clique)
	Mova elemento	Mova limite, linha, ponto ou centróide selecionado e clique em nova posição
	Dividir linha	Dividir uma linha existente em duas partes
	Exclua elemento	Exclua limite existente, linha, ponto ou centróide (confirme elemento selecionado por outro clique)
	Editar atributos	Editar atributos do elemento selecionado (note que um elemento pode representar mais recursos, veja acima)
	Fechar	Fechar sessão e salvar o estado atual (reconstrói topologia depois)

GRASS Digitalização de Tabela 1: GRASS Ferramentas de Digitalização

Aba Categoria

A: guilabel: guia *Categoria* permite definir a forma em que os valores da categoria serão atribuídos à um novo elemento geométrico.

- **Modo:** O valor da categoria que será aplicado aos novos elementos de geometria.
 - Seguinte não usado - Aplicar próximo valor categoria ainda não utilizados para a geometria do elemento.
 - Entrada Manual - definir manualmente o valor da categoria para o elemento geometria no campo de entrada ‘Categoria’.
 - Sem categoria - Não aplique um valor de categoria para o elemento de geometria. Isto é usado, por exemplo, para os limites da área, pois os valores de categoria são ligados através do centróide.
- **Categoria** - O número (ID) que está ligado a cada elemento geometria digitalizado. Ele é usado para ligar cada elemento de geometria com os seus atributos.
- **** Campo (camada)**** - Cada elemento de geometria pode ser conectado a diferentes tabelas de atributos usando diferentes camadas de geometria GRASS. A camada padrão é 1.

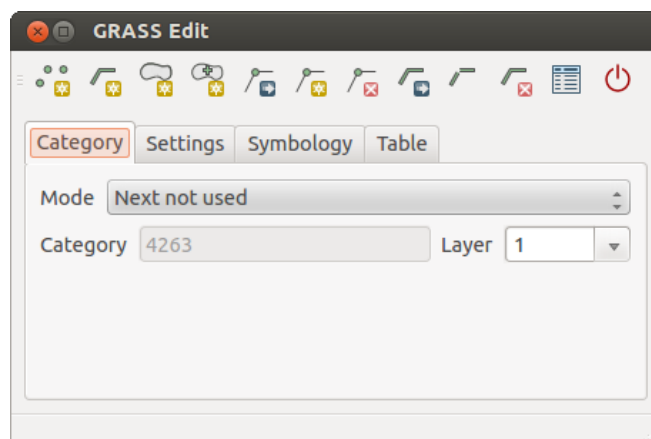


Figura 16.4: Aba Categoria Digitalização GRASS

Dica: Criando uma ‘camada’ adicional GRASS com o lqgl

Se você gostaria de adicionar mais camadas para o conjunto de dados, basta adicionar um novo número no ‘Campo (camada)’ e pressione enter. Na guia Tabela, você pode criar sua nova tabela ligada a sua nova camada.

Aba Configurações

A: guilabel: guia *Configurações* permite que você defina o encaixe em pixels da tela. O limite que define qual distância para novos pontos de linha ou extremidades são agarrados por nós existentes. Isso ajuda a evitar lacunas ou oscilações entre fronteiras. O padrão é definido como 10 pixels.

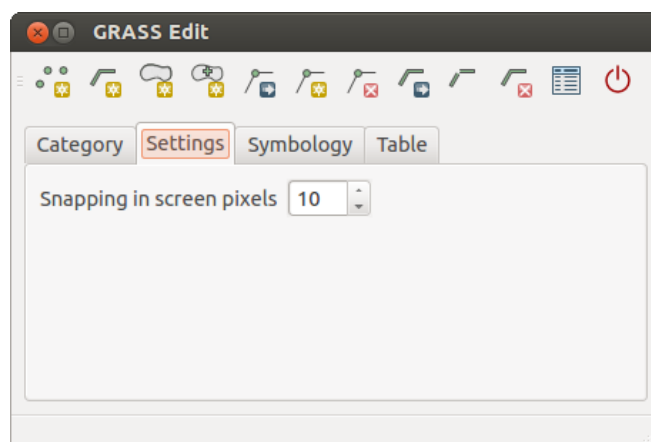


Figura 16.5: Aba Configurações de Digitalização GRASS

Aba Simbologia

A: guilabel: ‘Simbologia’ permite visualizar e definir as configurações de simbologia e as cores para vários tipos de geometria e seu estado topológico (por exemplo, abriu/fechou limite).

Aba Tabela

A: guilabel: ‘tabela’ fornece informações sobre a tabela de banco de dados para uma determinada ‘camada’. Aqui, você pode adicionar novas colunas a uma tabela de atributos já existente ou criar uma nova tabela de banco de dados para uma nova camada do vetor GRASS (ver seção: ref: ‘criar_novo_vetor_GRASS’).

Dica: GRASS Edita Permissões

Você deve ser o proprietário do GRASS: arquivo: ‘CONJUNTO DE MAPAS’ que deseja editar. É impossível editar camadas de dados em: arquivo: ‘CONJUNTO DE MAPAS’ que não é seu, mesmo que você tenha permissão

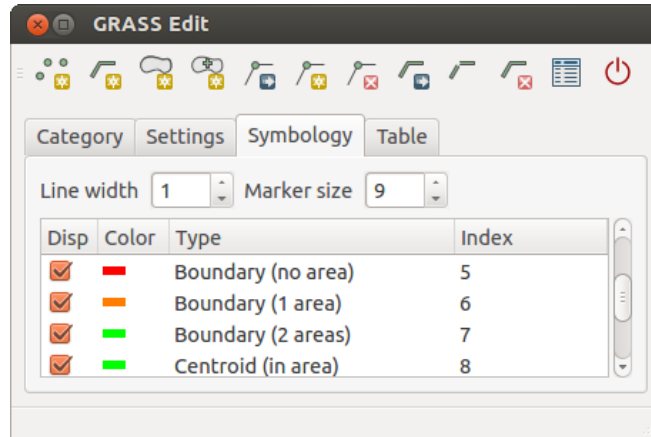


Figura 16.6: Guia Digitalizando Simbologia GRASS

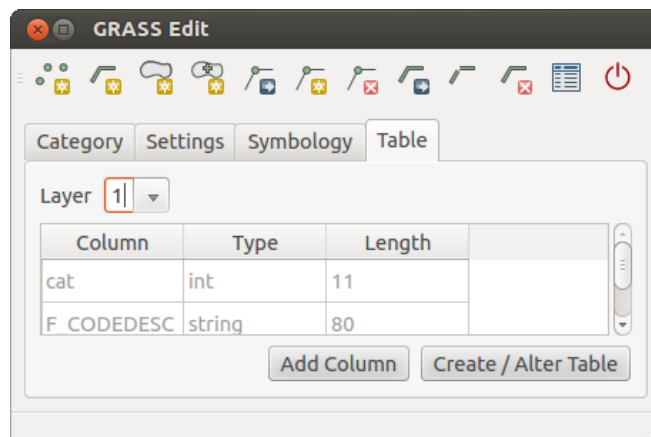


Figura 16.7: Aba Tabela de Digitalização GRASS

de gravação.

16.8 A ferramenta região GRASS

A região (definição de uma janela de trabalho espacial) no GRASS é importante para trabalhar com camadas. Análise Vetorial é, por padrão, não se limitando a quaisquer definições de região definida. Mas todos os vetores recém-criados terá a extensão espacial e a resolução da região GRASS atualmente definido, independentemente da sua extensão e da resolução original. A região GRASS atual é armazenado no: arquivo: '\$ LOCAL / \$ mapset / WIND', e define o norte, sul, leste e oeste como limites, o número de colunas e linhas, resolução espacial horizontal e vertical.

É possível ligar e desligar a visualização da região de GRASS no | QG | usando o | grass_região | : sup: *botão GRASS região* Exibição.

Com a | grass_região_editar | : sup: **'Editar ícone região GRASS'**, você pode abrir uma caixa de diálogo para alterar a região atual e a simbologia de retângulo para a região grass no |qgl|. Digite os novos limites e resolução de região e clique **** [OK] ****. A caixa de diálogo também permite selecionar uma nova região de forma interativa com o mouse sobre a tela do QGIS. Por isso, clique com o botão esquerdo do mouse no QGIS, abra um retângulo, feche-o novamente usando o botão esquerdo do mouse e clique **** [OK] ****.

O módulo GRASS: arquivo: *g.região* fornece muito mais parâmetros para definir uma medida para a região e resolução adequados para sua análise. Você pode usar esses parâmetros com as Ferramentas GRASS, descrito na seção: ref: *subseção_grass_ferramentas*.

16.9 Ferramentas GRASS

O |grass_ferramentas| :sup: 'Abrir ferramentas GRASS' fornece funcionalidades do módulo GRASS para trabalhar com dados dentro do GRASS selecionado: arquivo: 'local' e arquivo: 'CONJUNTO DE MAPAS'. Para usar as Ferramentas GRASS você precisa para abrir um: arquivo: 'local' e: arquivo: 'CONJUNTO DE DADOS' que você tem permissão para gravar (geralmente concedido, se você criou o: arquivo: 'CONJUNTO DE MAPAS'). Isso é necessário, pois novas camadas vetoriais serão criados durante a análise e precisa ser escrito para ser selecionado: arquivo: 'local' e: arquivo: 'Coleção de Mapas'.

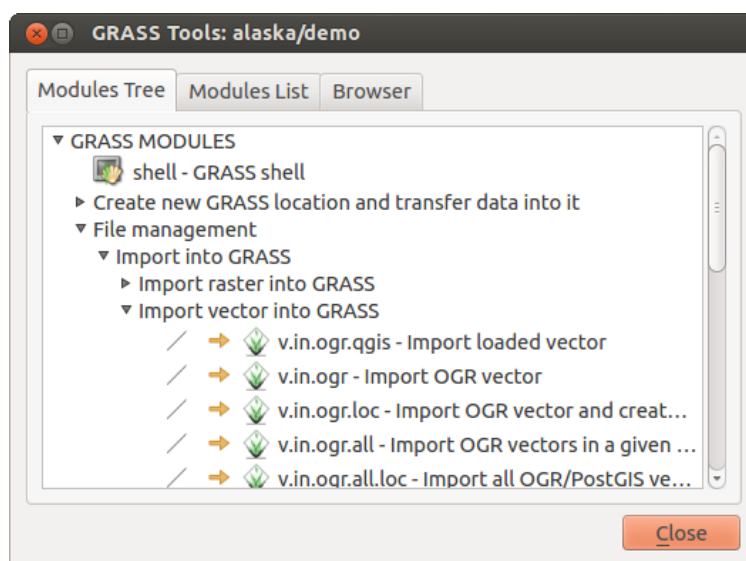


Figura 16.8: Ferramentas do GRASS e Modulo de Árvore 

16.9.1 Trabalhando com módulos GRASS

O comando GRASS dentro do GRASS Ferramentas fornece acesso a quase todos os módulos (mais de 300) grass em uma interface de linha de comando. Para oferecer um ambiente de trabalho mais amigável, cerca de 200 dos módulos GRASS disponíveis e funcionalidades também são fornecidos por gráficos dentro do plugin Ferramentas GRASS.

A lista completa dos módulos GRASS disponíveis na caixa de ferramentas gráfica em | QG | versão | ATUAL | estão disponíveis no wiki grass em http://grass.osgeo.org/wiki/GRASS-QGIS_relevant_module_list.

Também é possível personalizar o conteúdo Ferramentas GRASS. Este procedimento é descrito na seção: ref: *sec_customizacao_ferramentas*.

Como mostrado na [figura_grass_ferramentas_1](#), você pode olhar para o módulo GRASS apropriado usando o temática agrupada: `guiabel: Módulos em Árvores` ou pesquisar: `guiabel: 'Lista de Modelos'`.

Clicando em um ícone do módulo gráfico, uma nova aba será adicionada ao diálogo da caixa de ferramentas, oferecendo três novas sub-abas: `guiabel: 'Opções'`, `guiabel: 'Saída'` e `guiabel: 'Manual'`.

Opções

A: `guiabel: 'Opções'` proporciona um módulo simplificado onde normalmente você pode selecionar uma camada ou vector visualizado na tela do QGIS e entrar com outros parâmetros específicos do módulo para executá-lo.

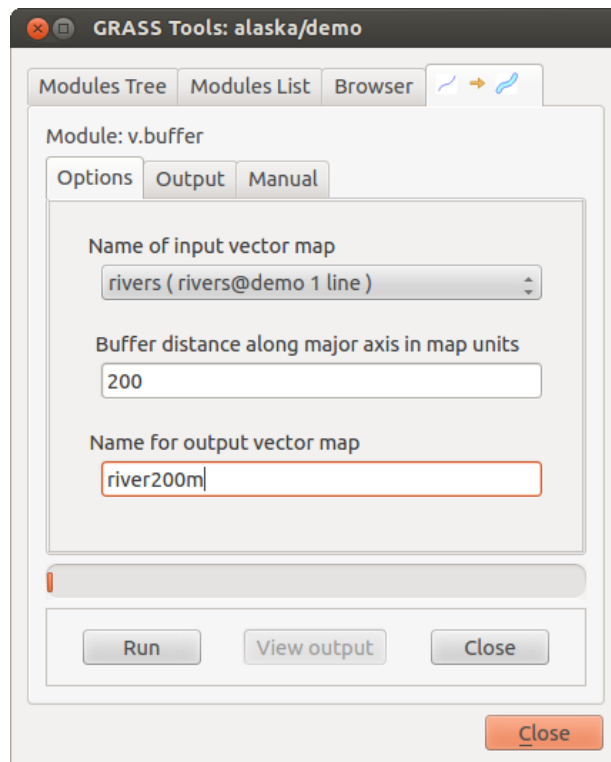


Figura 16.9: Ferramentas GRASS Opções de Módulo 🐧

Os parâmetros do módulo fornecidas muitas vezes não são completos para manter um diálogo claro. Se você quiser usar outros parâmetros do módulo e sinalizadores, você precisa iniciar o comando grass e executar o módulo na linha de comando.

Um novo recurso desde | QG | 1.8 é o suporte para a: `guiabel: 'Mostrar Opções Avançadas'` abaixo do módulo simplificado no: `guiabel: 'opções'`. No momento, só é adicionado ao módulo: arquivo: *v.in.ascii* como um exemplo, mas ele provavelmente irá fazer parte de todos os módulos no grass Ferramentas em versões futuras do | QG |. Isso permite que você use as opções de módulos GRASS completos sem a necessidade de mudar para o comando grass.

Saída

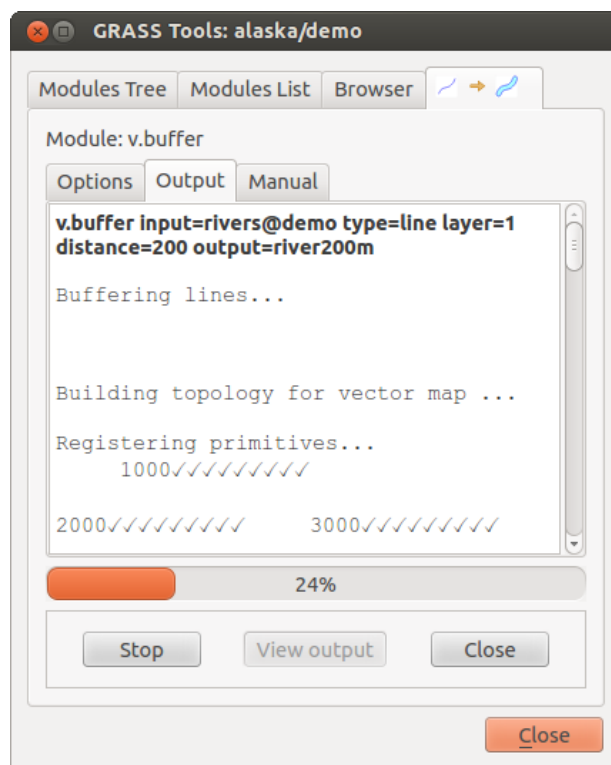


Figura 16.10: Ferramentas GRASS Módulo de Saída 🐧

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

A: guilabel: 'Manual' mostra a página de ajuda em HTML do módulo grass. Você pode usá-lo para verificar outros parâmetros do módulo e sinalizadores ou para obter um conhecimento mais profundo sobre o objetivo do módulo. No final de cada página do manual, você vê outras ligações a: arquivo: 'Ajuda', o arquivo: 'Temática' e do: arquivo: 'Completo'. Estas ligações fornecem a mesma informação que o módulo: arquivo: *g.manual*.

Dica: Mostra Resultados Imediatamente



Se você quiser exibir os resultados de cálculo imediatamente em sua tela do mapa, você pode usar o botão "Ver Saída" na parte inferior da aba módulo.

16.9.2 Exemplos de módulos GRASS

Os exemplos seguintes demonstrarão o poder de alguns dos módulos grama.

Criando linhas de contorno

O primeiro exemplo cria um mapa de contorno vetorial de elevação (DEM). Aqui, presume-se que você tem o Alasca: arquivo: 'local' configurado como explicado na seção: ref: 'sec_import_loc_data

- Primeiro, abra o local clicando no ícone : sup: 'Abrir conjunto de mapas' e escolher o local Alasca.
- Agora carregue o raster *gtopo30* elevação clicando no ícone : sup: *Adicionar camada raster GRASS* e selecionando o raster "gtopo30" a partir do local de demonstração.

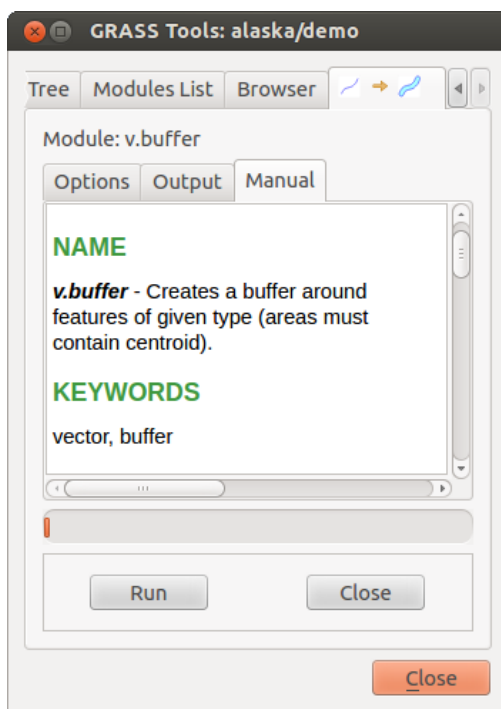


Figura 16.11: Ferramentas GRASS Modulo Manual 🐧

- Agora, abra a caixa de ferramentas com o 🗑️: sup: botão ‘Abrir ferramentas GRASS’.
- Na lista de categorias de ferramentas, clique duas vezes no: menu: ‘Raster -> Gestão de Superfície -> Gerar linhas de contorno do vetor’.
- Agora, um simples clique na ferramenta **** r.contour **** irá abrir uma janela, conforme explicado acima (veja: ref: *grass_módulos*). O “raster gtopo30” deve aparecer como o: guilabel: ‘Nome de entrada’.
- Digite no *Incremento entre os níveis de contorno* o valor 100. (Isto criará linhas de contorno em intervalos de 100 metros.)
- Digite no :guilabel: *Nome para a saída do vetor mapa* o nome `ctour_100`.
- Clique [**Rodar**] para iniciar o processo. Aguarde alguns instantes até que a mensagem `Terminado com sucesso` apareça na janela de saída. Então, clique [**Ver saída**] e [**fechar**].

Como se trata de uma grande região, vai demorar um pouco para exibir. Depois de terminar a renderização, você pode abrir a janela de propriedades da camada para alterar a cor da linha para que os contornos apareçam claramente sobre o raster elevação, como em: ref: *vector_properties_dialog*.

Em seguida, amplie uma pequena área montanhosa no centro do Alasca. Aproxima-se, você vai notar que os contornos têm cantos afiados. GRASS oferece a ferramenta **v.generalize** para alterar ligeiramente mapas vetoriais, mantendo sua forma original. A ferramenta usa diversos algoritmos diferentes, com finalidades diferentes. Alguns dos algoritmos (ie, Douglas Peuker e redução da Vertex) simplificam a linha removendo alguns dos vértices. O vetor resultante irá carregar mais rápido. Este processo é útil quando você tem um vetor altamente detalhado, mas você está criando um mapa muito pequena em escala, de modo que o detalhe é desnecessário.

Dica: A ferramenta simplificar

Note-se que o | QG | plugin Ferramentas tem um: menu: ‘Simplificar geometrias’ -> ferramenta que funciona exatamente como o GRASS **v.generalize** no algoritmo Douglas-Peuker.

No entanto, o objetivo do exemplo é diferente. As linhas de contorno criado pelo “r.contour” têm ângulos agudos que devem ser suavizadas. Entre o algoritmo **** v.generalize ****, há Chaiken que faz exatamente isso (também ranhuras de Hermite). Esteja ciente de que estes algoritmos podem **** adicionar vértices**** ao vetor, fazendo-a carregar ainda mais lentamente.

- Abra o GRASS Ferramentas e clique duas vezes na categoria: menu: ‘Vetor -> Desenvolver mapa -> Geral’, em seguida, clique no módulo **v.generalize** para abrir a janela de opções.
- Verifique se o vetor ‘ctour_100’ aparece como *Nome do vetor de entrada*.
- A partir da lista de algoritmos, escolha Chaiken. Deixe todas as outras opções em padrão, e desça até a última fila para entrar no campo: guilabel: ‘Nome de saída do vetor mapa’ ‘ctour_100_smooth’, e clique **** [Ir] ****.
- O processo leva vários minutos. Após o “Finalizado com Sucesso” aparece as janelas de saída, clique **** [Saída] **** e **** [Fechar] ****.
- É possível mudar a cor do vetor para apresentar claramente o fundo matricial e para contrastar com as linhas de contorno originais. Você notará que as novas linhas de contorno têm cantos mais suaves do que o original, enquanto permanecer fiel à forma geral de origem.

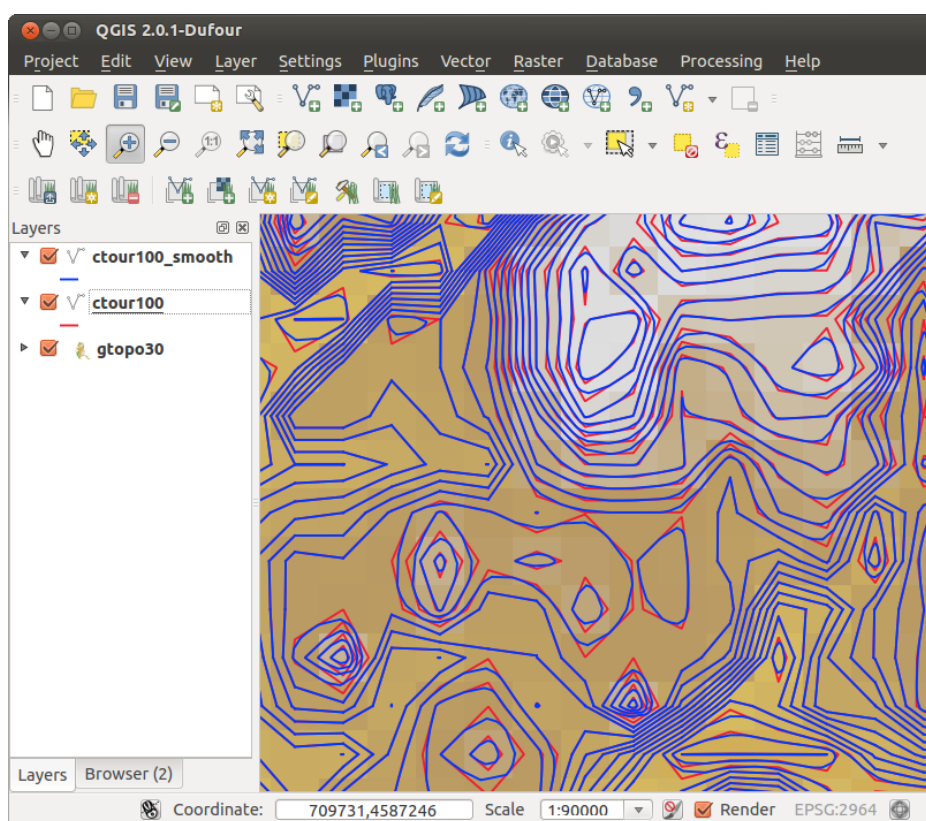



Figura 16.12: Módulo GRASS v.generalize para visualizar um mapa vetorial 

Dica: Outros usos para r.contour

O processo descrito acima pode ser utilizado em outras situações equivalentes. Se você tem um mapa de dados de precipitação, por exemplo, então o mesmo método será utilizado para criar um mapa de vetor de linhas (chuvas constantes).

Criando um efeito de sombreamento 3D

Vários métodos são usados para exibir as camadas de elevação e dar um efeito 3-D para mapas. O uso de curvas de nível, como mostrado acima, é um método popular, muitas vezes escolhido para produzir mapas topográficos. Outra forma de apresentar um efeito 3-D é de sombreamento. O efeito Sombra é criado a partir de um DEM (elevação) que calcula a inclinação e o aspecto de cada célula, em seguida, simula a posição do sol no céu dando um valor de reflexo para cada célula. Assim, você tem pistas virada para o sol; as encostas viradas longe do sol (na sombra) são escurecidas.

- Comece este exemplo carregando o ‘raster de elevação gtopo30’. Inicie o GRASS Ferramentas, e sob a categoria Raster, clique duas vezes para abrir: menu: ‘Análise espacial -> Análise do Terreno’.
- Então, clique **r.shaded.relief** para abrir o módulo.
- Altere o *ângulo azimuth* 270 a 315.
- Digite gtopo30_shade para o novo raster sombreado e clique **[rodar]**.
- Quando o processo termina, adicione o raster sombreado ao mapa. Você deve vê-lo exibido em escala de cinza.
- Para visualizar tanto o sombreado da colina e as cores do “gtopo30”, mova o mapa sombreado abaixo “mapa gtopo30” na tabela de conteúdo, em seguida, abra o: menu: ‘Propriedades de’ “GTOPO30”, mude para o: guilabel: ‘transparência’: e defina seu nível de transparência a cerca de 25%.

Agora você deve ter a elevação gtopo30 com o seu mapa de cores e configuração de transparência apresentada **acima** o tons de cinza mapa sombreado. Para ver os efeitos visuais do sombreado, desligue o mapa gtopo30_shade em seguida, ligue-o novamente.

Usando a linha de comando GRASS

O plugin GRASS em | QG | é projetado para usuários que são novos para o grass e não está familiarizado com todos os módulos e opções. Como tal, alguns módulos na caixa de ferramentas não mostram todas as opções disponíveis, e alguns módulos não aparecem. O comando GRASS (ou console) dá ao usuário acesso aos módulos adicionais GRASS que não aparecem na caixa de ferramentas, e também algumas opções adicionais aos módulos que estão na caixa de ferramentas com os parâmetros padrão mais simples. Este exemplo demonstra o uso de uma opção adicional para o módulo **** r.shaded.relief **** que foi mostrado acima.

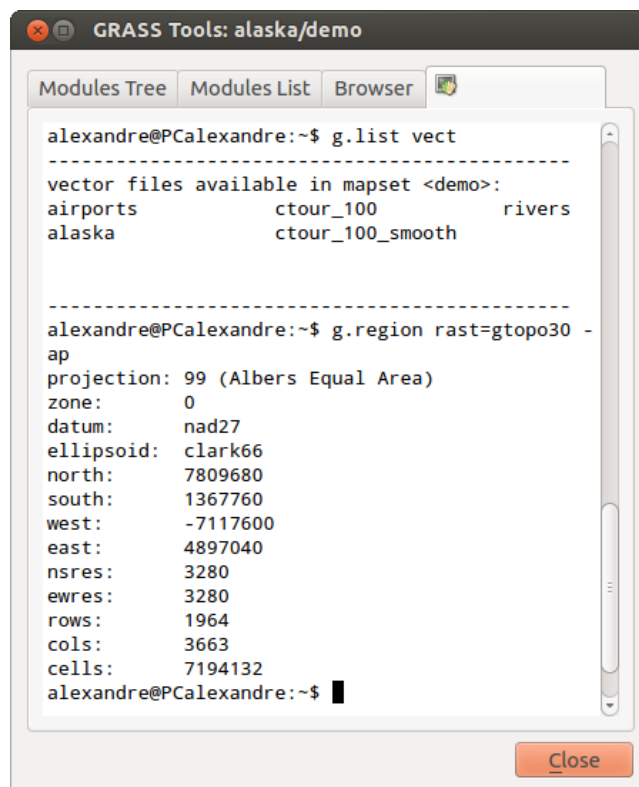


Figura 16.13: A shell GRASS, módulo r.shaded.relief 

O módulo **** r.shaded.relief **** pode ter um parâmetro “zmult”, que multiplica os valores de elevação em relação ao XY unidades de coordenadas de modo a que o efeito de sombreado é ainda a mais usada.

- Coloque o “raster elevação gtopo30”, e depois inicie o GRASS Ferramentas e clique no comando grass. Na janela de comando, digite: “mapa r.shaded.relief = sombra GTOPO30 = gtopo30_shade2 azimuth = 315 = zmult 3” e pressione **** [Entrar] ****.

- Após terminar o processo, mude para o: guilabel: ‘Navegador’ e clique duas vezes sobre o novo “raster gtopo30_shade2” para exibi-lo no | QG I.
- Como explicado acima, mova o raster com o relevo sombreado abaixo da tabela de conteúdo “raster gtopo30” em seguida, verifique a transparência da “camada gtopo30” colorida. Você deverá ver que o efeito 3-D se destaca mais fortemente em relação ao primeiro mapa com relevo sombreado.

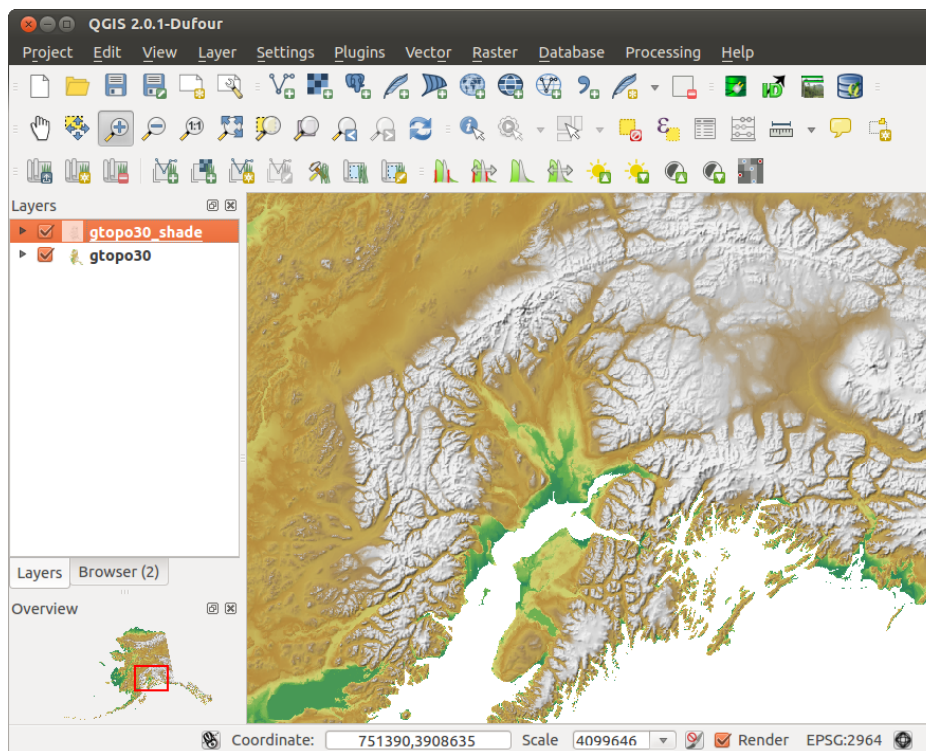


Figura 16.14: Exibindo relevo sombreado criado com o módulo GRASS r.shaded.relief 🐧

Estatísticas Raster em um mapa vetor

O próximo exemplo mostra como um módulo GRASS pode agregar dados raster e adicionar colunas de estatísticas para cada polígono em um mapa vetor.

- Novamente usando os dados do Alasca, consulte a: ref: *sec_import_loc_data* para importar o shapefile árvores do diretório *shapefiles* no GRASS.
- Agora, um passo intermediário é necessária: centroides deve ser adicionado ao mapa importando as árvores para torná-lo num vetor de área de grass completo (incluindo os limites e centróide).
- Na caixa de ferramentas, escolha: menu: *Vetor -> Gerenciar recursos, e abra o módulo ** v.centroids **.
- Digite como *mapa vetor de saída* ‘forest_areas’ e rode o módulo.
- Agora carregue o vetor “ forest_areas“ e exiba os tipos de florestas - Deciduais, verde, misturado - em cores diferentes: na camada janela :guilabel: *Propriedades*, guia *Simbologia*, escolha: guilabel: ‘Tipo de Legenda’ ‘valor único’ e definir o campo *Classificação* para ‘VEGDESC’. (Consulte a explicação da guia simbologia em: ref.: *Vector_style_menu* da seção vetor.)
- Em seguida, reabrir o GRASS Ferramentas no: menu: ‘Vetor -> Atualização do Vetor’ para outros mapas.
- Clique no módulo **v.rast.stats**. Digite “gtopo30” e “forest_areas”.
- Apenas um parâmetro adicional é necessária: Entra no: guilabel: ‘prefixo’ coluna ‘elev’ e clique ** [Ir] **. Esta é uma operação computacionalmente pesada, que será executado por um longo tempo (provavelmente até duas horas).

- Finalmente, abra o “*areas_floresta*” na tabela de atributos, e verifique que várias novas colunas foram adicionadas, incluindo ‘*elev_min*’, ‘*elev_max*’, ‘*elev_média*’, etc, para cada polígono florestal.

16.9.3 Trabalhando com o buscador LOCALIZAÇÃO GRASS

Outro recurso útil dentro do GRASS Ferramentas é o GRASS: arquivo: ‘navegador local’. Na figura_grass_modula_7, você pode ver o trabalho atual: arquivo: *local* com seu: arquivo: ‘Conjunto de Mapas’.

A esquerda das janelas do navegador , você pode navegar através de todos: arquivo: ‘Conjunto de Mapas’ dentro do atual: arquivo: ‘*local*’. A direita da janela do navegador mostra alguns meta-informação para camadas vetoriais selecionados (por exemplo, resolução, caixa, fonte de dados, tabela de atributos conectado para dados vetoriais, e um histórico de comandos delimitador).

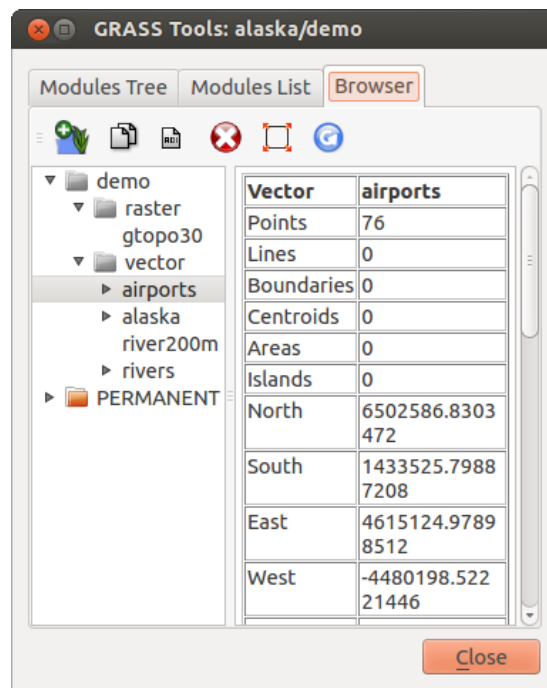










Figura 16.15: Buscador de LOCALIZAÇÃO GRASS 

A barra de ferramentas dentro do: guilabel: ‘Navegador’ oferece as seguintes ferramentas para gerenciar o selecionado: arquivo: ‘*local*’:

- : guilabel: *Adicione mapa selecionado à tela*
- : guilabel: ‘Copiar mapa selecionado’
- : guilabel: ‘Renomeia mapa selecionado’
- : guilabel: ‘Exclua mapa selecionado’
- : guilabel: ‘Defina região atual de mapa selecionado’
- : guilabel: *Atualizar janela de busca*

A guilabel: : *Renomeia mapa selecionado* e guilabel: : ‘Exclua mapa selecionado’ só trabalha com os mapas dentro do seu selecionado: arquivo: *MAPSET*. Todas as outras ferramentas também trabalham com camadas vetoriais e raster em outro: arquivo: *MAPSET*.

16.9.4 Personalizando a caixa de ferramentas GRASS

Quase todos os módulos de GRASS podem ser adicionados a caixa de ferramentas. Uma interface XML é fornecido para analisar os arquivos mais simples que configuram a aparência e os parâmetros dos módulos dentro da caixa de ferramentas.

Um exemplo de arquivo XML para gerar o módulo `v.buffer` (`v.buffer.qgm`) se parece com isto:

```
<?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>
```

O analisador lê esta definição e cria uma nova aba no interior da caixa de ferramentas quando você seleciona o módulo. Uma descrição mais detalhada para a adição de novos módulos, mudando o grupo de um módulo, etc, pode ser encontrado na | qg | wiki em http://hub.qgis.org/projects/quantum-gis/wiki/Adding_New_Tools_to_the_GRASS_Toolbox.

QGIS estrutura de processamento

17.1 Introdução

Este capítulo apresenta a estrutura de processamento QGIS, um ambiente de geoprocessamento, que pode ser usado para chamar algoritmos nativos e de terceiros a partir do QGIS, fazendo suas tarefas de análise espacial mais produtivas e fáceis de realizar.

Nas seções seguintes, vamos analisar como usar os elementos gráficos deste quadro e tirar o máximo proveito de cada um deles.

Existem quatro elementos básicos no quadro da GUI, que são usados para executar algoritmos para diferentes fins. Escolher uma ferramenta ou outra vai depender do tipo de análise que está sendo realizada e das características particulares de cada usuário e do projeto. Todos eles (exceto para a interface de processamento em lote, que é chamado a partir da caixa de ferramentas, como veremos) podem ser acessados a partir do ítem de menu :menuselection: *Processamento*. (Você vai ver mais de quatro entradas. Os restantes não são usados para executar algoritmos e será explicado mais adiante neste capítulo.)

- A caixa de ferramentas. O elemento principal do GUI, é usado para executar um algoritmo único ou rodar um processo batch baseado nesse algoritmo.

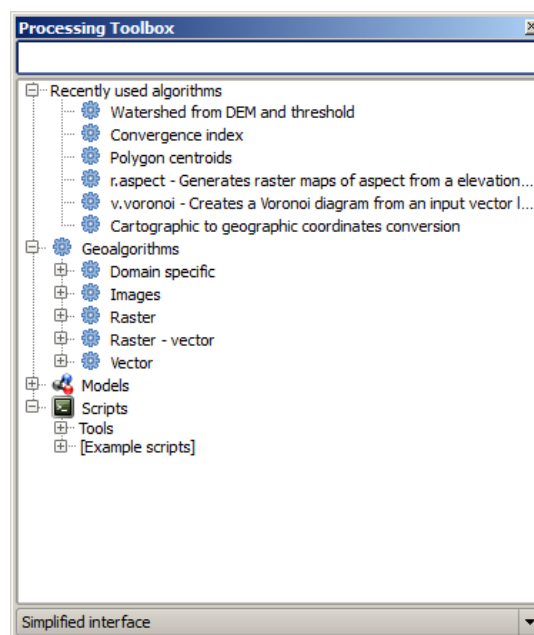



Figura 17.1: Caixa de Ferramentas Processamento 

- O modelador gráfico. Vários algoritmos podem ser combinados graficamente usando o modelador para definir um fluxo de trabalho, a criação de um único processo que envolve vários subprocessos.

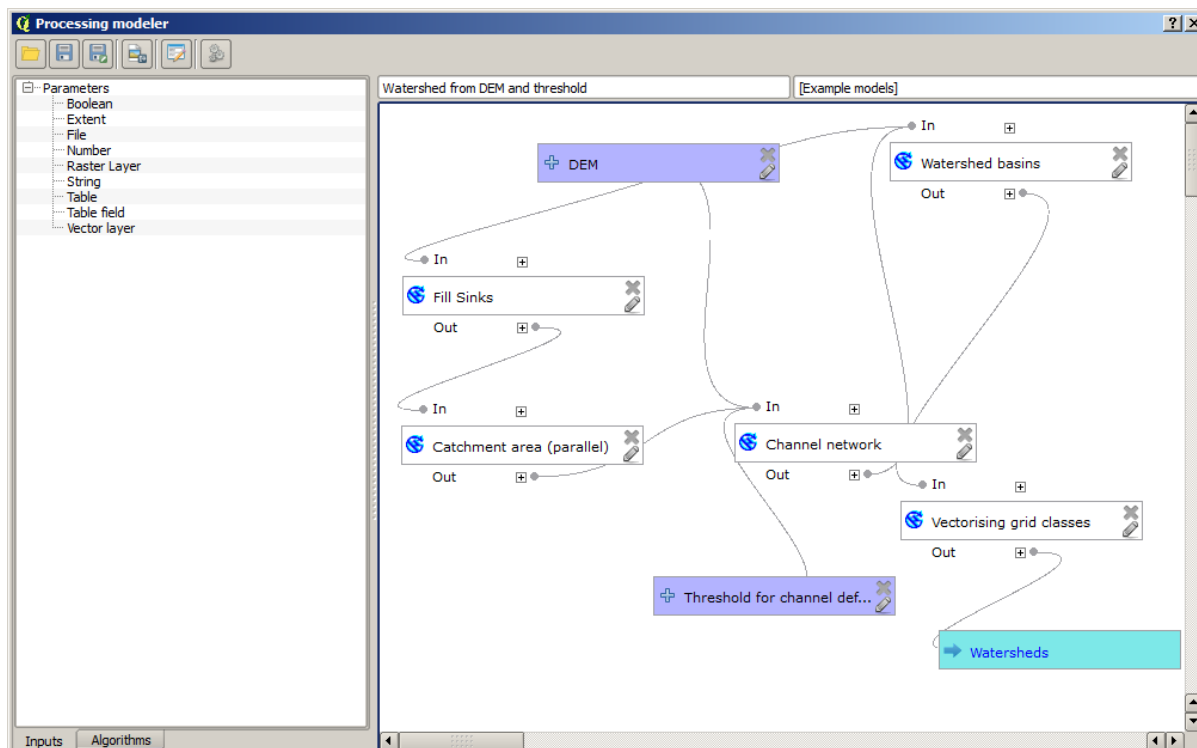


Figura 17.2: Modelador do Processamento

- O gerenciador do histórico. Todas as ações realizadas usando qualquer um dos elementos acima mencionados são armazenados em um arquivo de histórico e pode ser facilmente reproduzida mais tarde usando o gerenciador do histórico.
- A interface de processamento batch. Esta interface permite que possa executar processos batch e automaticamente a execução de um único algoritmo num múltiplo conjunto de dados.

Nas seções seguintes, vamos analisar cada um desses elementos em detalhe.

17.2 A caixa de ferramentas

A *Caixa de ferramenta* é o elemento principal da GUI de processamento, e o que você mais usará provável em seu trabalho diário. Ele mostra a lista de todos os algoritmos disponíveis agrupados em diferentes blocos, e é o ponto de acesso para executá-los, seja como um processamento único ou em lote que envolve várias execuções do mesmo algoritmo em diferentes conjuntos de entradas.

A caixa de ferramentas contém todos os algoritmos disponíveis, divididos em grupos pré-definidos. Todos esses grupos podem ser encontrados numa árvore única com uma entrada que se chama *Geoalgoritmos*

Além disso, mais duas entradas são encontradas, ou seja, *Modelos* e *Scripts*. Estes incluem algoritmos criados pelo usuário, e eles permitem que você definir seus próprios fluxos de trabalho e tarefas de processamento. Vamos dedicar uma seção completa a eles um pouco mais tarde.

Na parte superior da caixa de ferramentas, você vai encontrar uma caixa de texto. Para reduzir o número de algoritmos mostrados na caixa de ferramentas e tornar mais fácil de encontrar o que você precisa, você pode digitar qualquer palavra ou frase na caixa de texto. Observe que, enquanto você digita, o número de algoritmos na caixa de ferramentas é reduzido para apenas aqueles que contêm o texto que introduziu o nome.

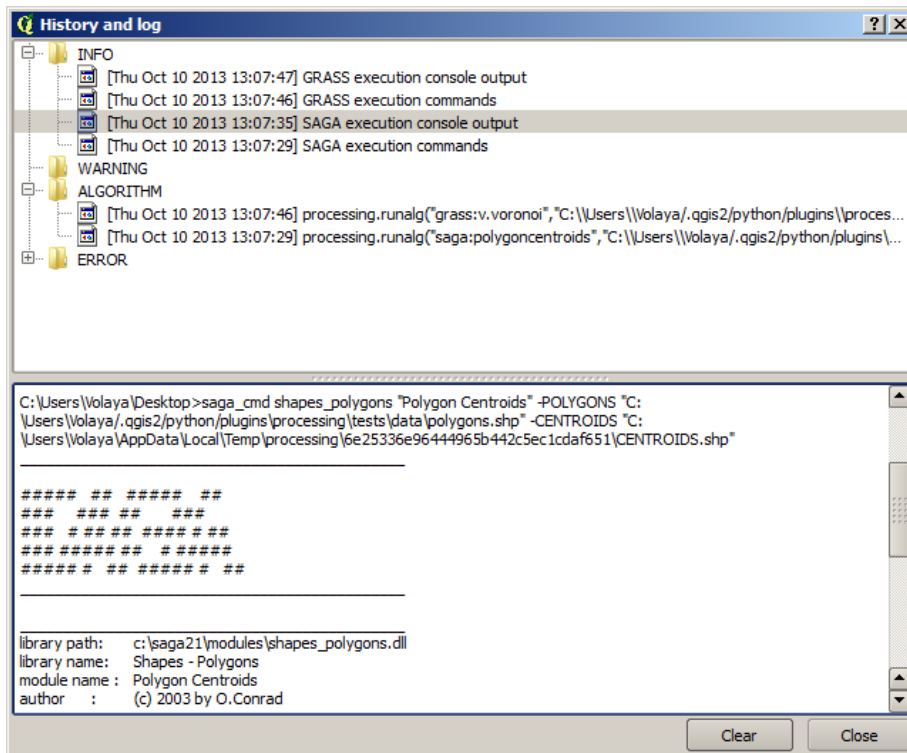


Figura 17.3: Histórico do Processamento

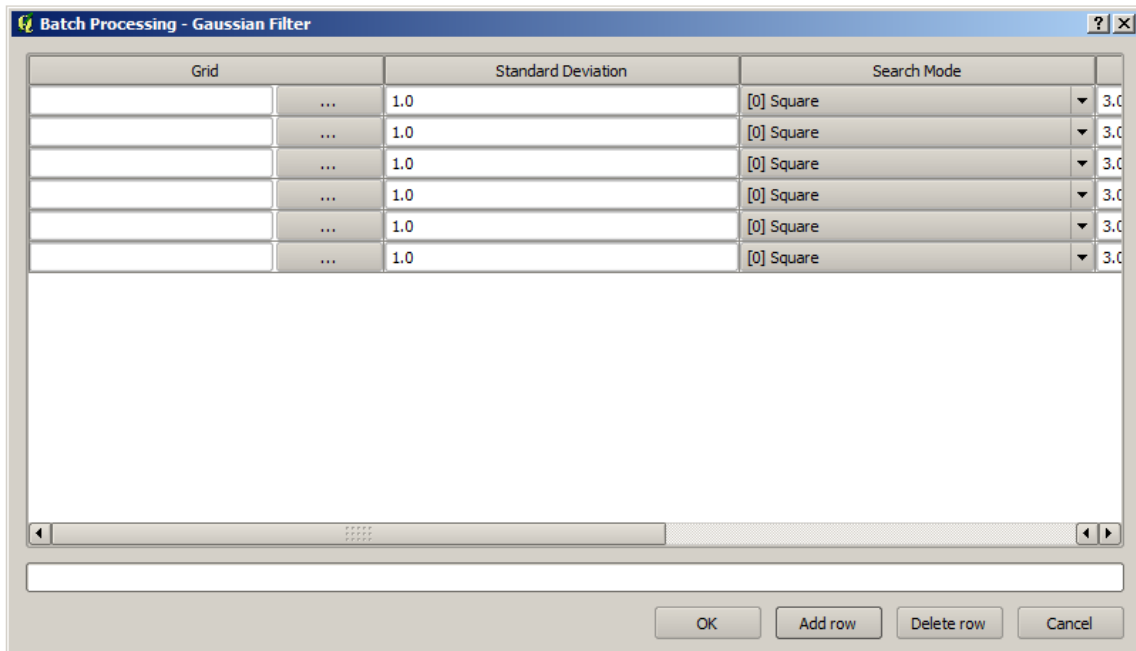


Figura 17.4: Interface de Processamento Batch

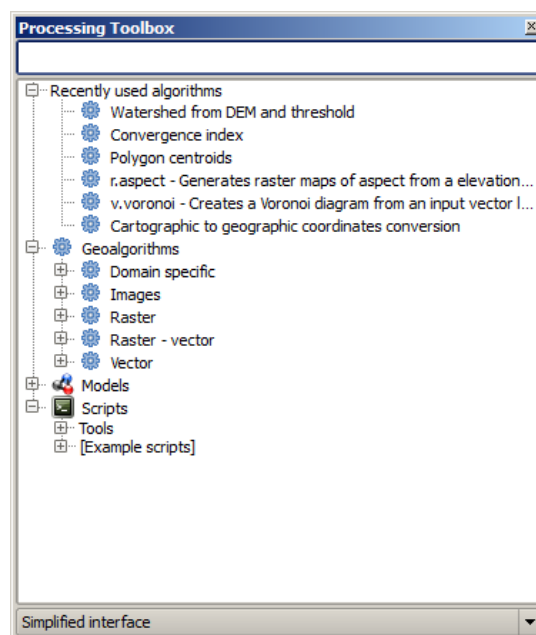


Figura 17.5: Caixa de Ferramentas Processamento 

Na parte inferior, você encontrará uma caixa que lhe permite alternar entre a lista algoritmo simplificado (aquele explicado acima) e a lista avançada. Se você mudar para o modo avançado, a caixa de ferramentas será parecida com esta:

Na exibição avançada, cada grupo representa um chamado “provedor algoritmo”, que é um conjunto de algoritmos que vêm da mesma fonte, por exemplo, a partir de um aplicativo de terceiros com recursos de geoprocessamento. Alguns destes grupos representam algoritmos de aplicativos de terceiros, como SAGA, GRASS ou R, enquanto outros contêm algoritmos diretamente codificado como parte do complemento de processamento, não dependendo de nenhum software adicional.

Esta vista é recomendado para aqueles usuários que têm um certo conhecimento das aplicações que estão apoiando os algoritmos, uma vez que eles serão mostrados com seus nomes originais e grupos.

Além disso, alguns algoritmos adicionais estão disponíveis apenas na visualização avançada, tais como ferramentas de LiDAR e scripts com base no software de computação estatística R, entre outros. Independente QGIS complementos que adicionam novos algoritmos para a caixa de ferramentas só serão mostrado na exibição avançada.

Particularmente, a vista simplificada contém algoritmo dos seguintes fornecedores:

- GRASS
- SAGA
- OTB
- Algoritmos nativos QGIS

No caso de execução QGIS com o Windows, estes algoritmos são totalmente funcionais em uma nova instalação do QGIS, e eles podem ser executados sem a necessidade de nenhuma instalação adicional. Além disso, executá-los não requer nenhum conhecimento prévio dos aplicativos externos que usam, tornando-as mais acessível para usuários de primeira viagem.

Se você quiser usar um algoritmo não fornecido por qualquer dos provedores acima, mude para o modo avançado, selecionando a opção correspondente na parte inferior da caixa de ferramentas.

Para executar um algoritmo, de duplo clique no seu nome na caixa de ferramentas.

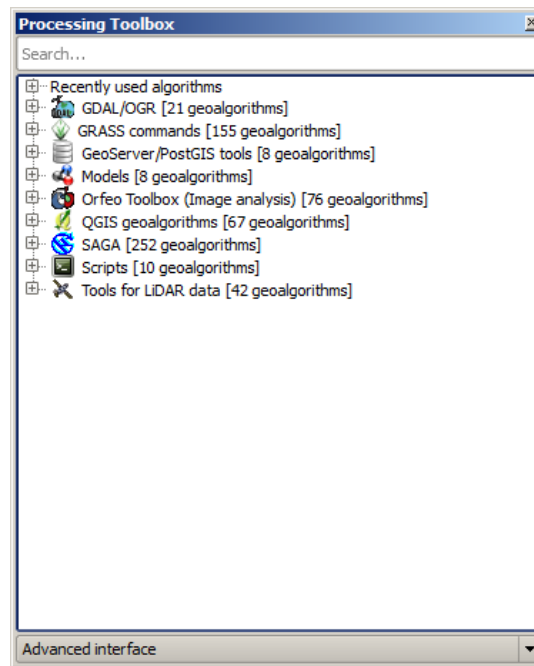


Figura 17.6: Caixa de Ferramentas Processamento (modo avançado) 

17.2.1 O diálogo do algoritmo

Uma vez que você clicar duas vezes sobre o nome do algoritmo que você deseja executar, uma caixa de diálogo semelhante à da figura abaixo é mostrado (neste caso, o diálogo corresponde ao algoritmo ‘índice de Convergência’ do SAGA).

Este diálogo é utilizado para definir os valores de entrada que o algoritmo precisa para ser executado. Ele mostra uma tabela onde os valores de entrada e os parâmetros de configuração serão definidos. É, obviamente, tem um teor diferente, dependendo dos requisitos do algoritmo a ser executado, e é criado automaticamente, com base em tais requisitos. No lado esquerdo, o nome do parâmetro é mostrado. No lado direito, o valor do parâmetro pode ser definido.

Embora o número e tipo de parâmetros dependerem das características do algoritmo, a estrutura é semelhante para todos eles. Os parâmetros encontrados na tabela podem ser de um dos seguintes tipos.

- Uma camada raster, selecione a partir de uma lista de todas as camadas disponíveis (atualmente abertas) no QGIS. O seletor contém também um botão no seu lado direito, para que você escolha nomes que representam as camadas atualmente não carregadas no QGIS.
- Uma camada de vetor, selecione a partir de uma lista de todas as camadas de vetores disponíveis no QGIS. Camadas não carregadas no QGIS podem ser selecionadas, bem como, como no caso de camadas raster, mas somente se o algoritmo não necessita de um campo da tabela selecionada a partir da tabela de atributos da camada. Nesse caso, apenas as camadas abertas podem ser selecionadas, uma vez que precisa ser aberta, de modo a recuperar a lista de nomes de campos disponíveis.

Irá ver um botão por cada selecionador de camada vetorial, como é exibido na figura em baixo.

Se o algoritmo contém vários deles, você será capaz de alternar apenas um deles. Se o botão que corresponde a um vetor de entrada é alternado, o algoritmo será executado de forma iterativa em cada uma de suas características, em vez de apenas uma vez para toda a camada, produzindo o maior número de saídas em vezes do algoritmo ser executado. Isto permite a automatização do processo quando todas as características de uma camada ter que ser tratada separadamente.

- Uma tabela, selecione a partir de uma lista de todos os disponíveis no QGIS. Tabelas não-espaciais são carregados para QGIS como camadas vetoriais, e na verdade, eles são tratados como tal pelo programa. Atualmente, a lista de tabelas disponíveis que você verá ao executar um algoritmo que precisa de um deles é

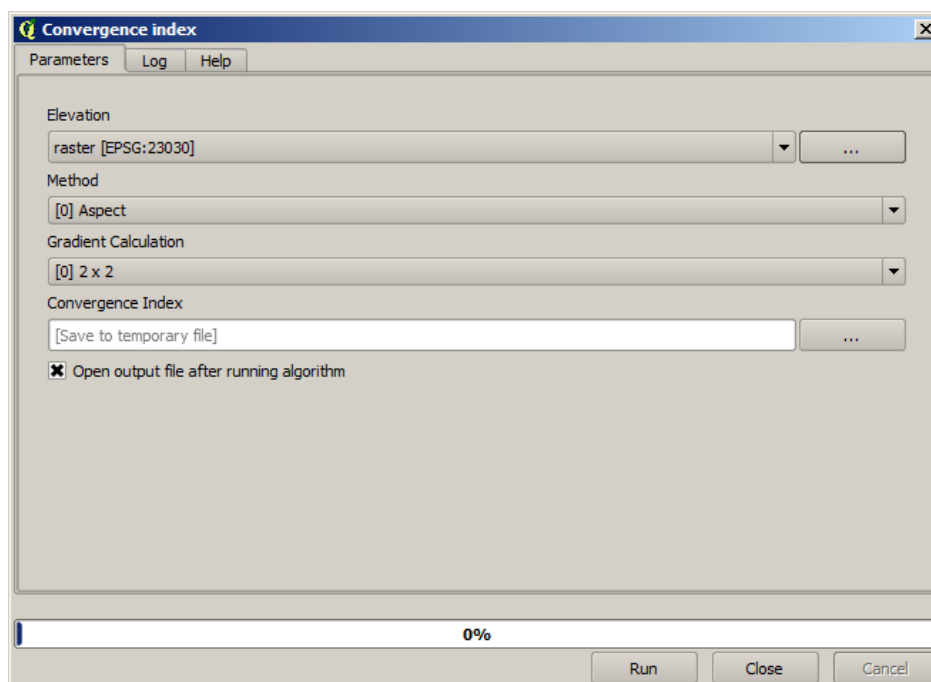


Figura 17.7: Diálogo dos Parâmetros

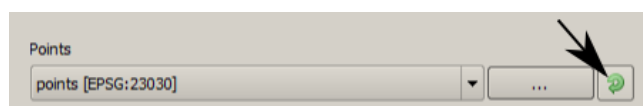


Figura 17.8: Botão de interação vetorial

restrito às tabelas provenientes de arquivos em formatos dBase (:file:. *dbf*) ou valores separados por vírgula (:file:. *csv*).

- Uma opção, para escolher de uma lista de seleção de uma lista de opções possíveis.
- Um valor numérico, a ser introduzido na caixa de texto. Você vai encontrar um botão ao seu lado. Clicando nele, você verá uma caixa de diálogo que permite que você digite uma expressão matemática, para que você possa usá-lo como uma calculadora de mão. Algumas variáveis úteis relacionados com dados carregados em QGIS pode ser adicionado à sua expressão, de modo que você pode selecionar um valor derivado de qualquer uma dessas variáveis, como o tamanho das células de uma camada ou a mais setentrional de coordenadas do outro.
- Um intervalo, com valores min e máx para serem introduzidos em duas caixas de texto.
- Uma cadeia de texto, para ser introduzida na caixa de texto.
- Um campo, para escolher a partir de uma tabela de atributos de uma camada vetorial ou uma tabela única de outro parâmetro.
- Um sistema de coordenadas de referência. Você pode digitar o código EPSG diretamente na caixa de texto, ou selecione-o na janela de seleção de SRC que aparece quando você clica com o botão do lado direito.
- Uma medida, a ser celebrada por quatro números que representam os seus limites x_{min} , x_{max} , y_{min} , y_{max} . Clicando no botão no lado direito do seletor de valor, um menu pop-up será exibido, dando-lhe duas opções: para selecionar o valor de uma camada ou a extensão da tela atual, ou defini-la arrastando diretamente na mapa de lona.

Se selecione a primeira opção, irá ver uma janela igual a próxima.

Se selecionar o segundo, os parâmetros da janela irão esconder-se, para que possa clicar e arrastar para o enquadramento. Uma vez definido o retângulo selecionado, o diálogo irá reaparecer, contendo os valores

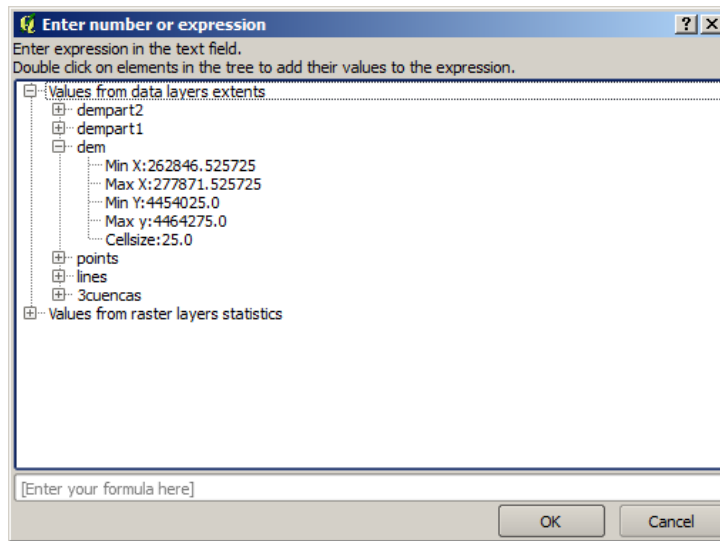


Figura 17.9: Seleccionador de Números

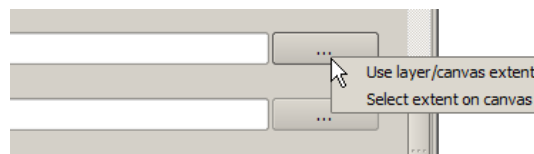


Figura 17.10: Seleccionador de Extensão

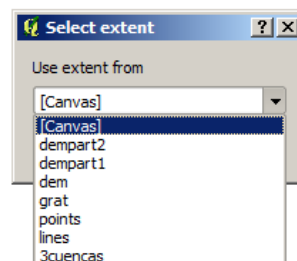


Figura 17.11: Lista de Extensão

na caixa de texto da extensão.

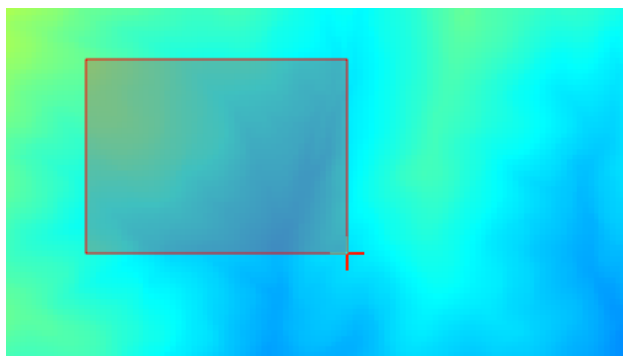


Figura 17.12: Arrastamento para Extensão 🌐

- A lista de elementos (se as camadas raster, camadas de vetor ou tabelas), para seleccionar a partir da lista de tais camadas disponíveis no QGIS. Para fazer a seleção, clique no pequeno botão do lado esquerdo da linha correspondente para ver uma janela como a seguinte.

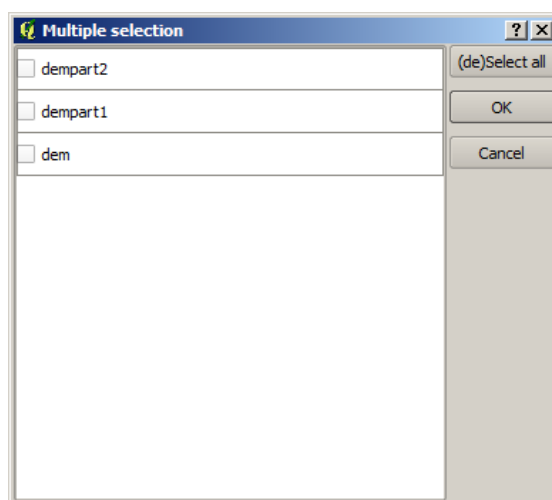


Figura 17.13: Múltipla Seleção 🌐

- Uma pequena tabela para ser editada pelo utilizador. Estes são usados para definir os parâmetros como tabelas lookup ou kernels de convolução, entre outros.

Clique no botão do lado direito para ver a tabela e editar os seus valores.

Dependendo do algoritmo, o número de linhas pode ser modificado ou não, utilizando os botões do lado direito da janela.

Você vai encontrar uma guia **[Ajuda]** na caixa de diálogo dos parâmetros. Se um arquivo de ajuda estiver disponível, ele será mostrado, dando-lhe mais informações sobre o algoritmo e descrições detalhadas do que cada parâmetro faz. Infelizmente, a maioria dos algoritmos não têm uma boa documentação, mas se você sentir vontade de contribuir com o projeto, este seria um bom lugar para começar.

Uma nota nas projeções

Algoritmos executados a partir da estrutura de processamento — isso também é verdade para a maioria das aplicações externas cujos algoritmos são expostos através dele. Não realize nenhuma reprojeção em camadas de entrada e assuma que todas elas já estão em um sistema de coordenadas comum e pronto para serem analisadas. Sempre que você usar mais de uma camada como entrada para um algoritmo, se vetor ou raster, irá perguntar a você para se certificar que elas estão todas no mesmo sistema de coordenadas.

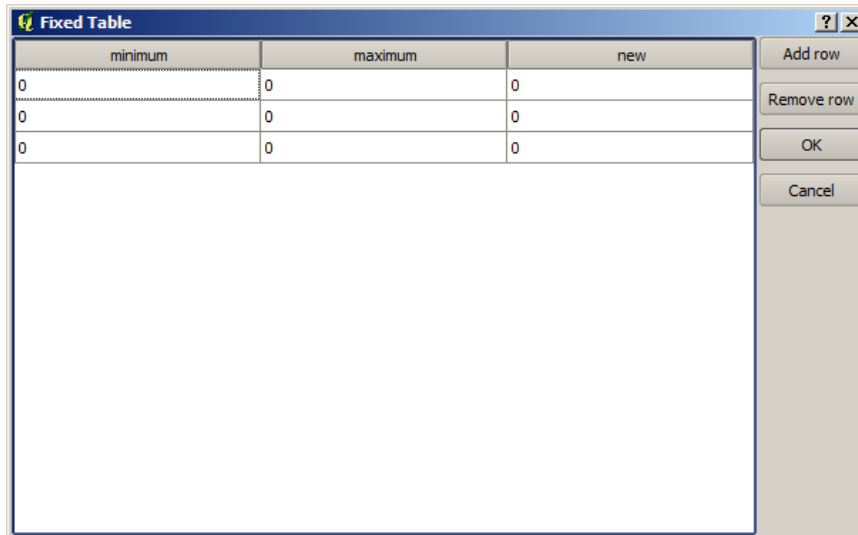


Figura 17.14: Tabela Fixa 🌐

Note-se que, devido a capacidades de reprojeção on-the-fly do QGIS, apesar de duas camadas pode parecer sobrepor e corresponder, que pode não ser verdade, se as suas coordenadas originais são usadas sem reprojeção em um sistema de coordenadas comum. Essa reprojeções deve ser feita manualmente, e, em seguida, os arquivos resultantes devem ser utilizados como entrada para o algoritmo. Também, notar que o processo reprojeções pode ser realizada com os algoritmos que estão disponíveis na própria estrutura de processamento.

Por padrão, a caixa de diálogo parâmetros mostrará uma descrição do SRC de cada camada, juntamente com o seu nome, tornando mais fácil para selecionar as camadas que compartilham o mesmo SRC para ser usado como camadas de entrada. Se você não quer ver essas informações adicionais, você pode desativar essa funcionalidade na janela de configuração de processamento, desmarcando a opção *Mostrar SRC*.

Se você tentar executar um algoritmo utilizando como entrada duas ou mais camadas com sem correspondência de SRC, um diálogo de aviso será exibido.

Pode continuar a executar o algoritmo, mas tenha atenção que na maioria dos casos irá produzir resultados errados, tais como, camadas vazias devido à falta de sobreposição das camadas usadas como arquivos de entrada.

17.2.2 Objetos de dados gerados por algoritmos

Objetos de dados gerado por um algoritmo podem ser dos seguintes tipos:

- Uma camada raster
- Uma camada vetorial
- Uma tabela
- Um arquivo HTML (usado para arquivo de saída de texto e gráficos)

Estes todos são salvos em disco, e a tabela de parâmetros irá conter uma caixa de texto correspondente a cada uma dessas saídas, onde você pode digitar o canal de saída e usar para salvá-la. Um canal de saída contém as informações necessárias para salvar o objeto resultante em algum lugar. No caso mais comum, você vai salvá-lo em um arquivo, mas a arquitetura permite a qualquer outra forma de armazenamento. Por exemplo, uma camada vetorial pode ser armazenada em um banco de dados ou até mesmo enviada para um servidor remoto usando um serviço WFS-T. Embora as soluções como estas ainda não estão implementadas, a estrutura de processamento está preparada para lidar com elas, e esperamos adicionar novos tipos de canais de saída em uma próxima feição.

Para selecionar um canal de saída, basta clicar no botão do lado direito da caixa de texto. Isso vai abrir uma caixa de diálogo salvar arquivo, onde você pode selecionar o caminho desejado para o arquivo. Extensões de arquivos suportadas são mostradas no seletor de formato de arquivo da caixa de diálogo, dependendo do tipo do produto e do algoritmo.

O formato da saída é definido pela extensão do arquivo. Os formatos suportados dependem do que é suportado pelo próprio algoritmo. Para selecionar um formato, basta selecionar a extensão do arquivo correspondente (ou em vez adicioná-la, se você estiver digitando diretamente o caminho do arquivo). Se a extensão do caminho do arquivo que você digitou não corresponde a nenhum dos formatos suportados, uma extensão do padrão (normalmente `.dbf` para tabelas, `.tif` para camadas raster e `.shp` para camadas vetoriais) será anexado ao caminho do arquivo e o formato de arquivo correspondente a essa extensão será usado para salvar a camada ou tabela.

Se você não digitar qualquer nome de arquivo, o resultado será salvo como um arquivo temporário no formato de arquivo padrão correspondente, e ele vai ser excluído depois de sair QGIS (tome cuidado com isso, no caso de você salvar seu projeto e contém camadas temporárias).

Você pode definir uma pasta padrão para objetos de dados de saída. Vá para a janela de configuração (você pode abri-la a partir do Menu :menuselection: *Processamento*), e no grupo *Geral*, você vai encontrar um parâmetro chamado *pasta de saída*. Esta pasta de saída é usada como o caminho padrão no caso de você digitar apenas um nome de arquivo sem caminho (isto é, `myfile.shp`) durante a execução de um algoritmo.

Ao executar um algoritmo que usa uma camada de vetor no modo interativo, o caminho do arquivo digitado é utilizado como o caminho de base para todos os arquivos gerados, que são nomeados com o nome de base e acrescentando-se um número que representa o índice da iteração. A extensão do arquivo (e formato) é usado para todos esses arquivos gerados.

Além de camadas raster e tabelas, algoritmos também geram gráficos e textos em arquivos HTML. Estes resultados estão apresentados no final da execução do algoritmo em uma nova caixa de diálogo. Este diálogo vai manter os resultados produzidos por qualquer algoritmo durante a seção atual, e pode ser mostrado em qualquer momento selecionando *Processamento* → *visualizador de resultados* do menu principal QGIS.

Algumas aplicações externas podem ter arquivos (sem restrições de extensão particulares) como saída, mas elas não pertencem a nenhuma das categorias acima. Os arquivos de saída não serão processados pelo QGIS (aberto ou incluído no atual projeto QGIS), uma vez que a maior parte do tempo, eles correspondem a formatos de arquivo ou elementos não suportados pelo QGIS. Este é, por exemplo, o caso com os arquivos LAS usados para dados LiDAR. Os arquivos são criados, mas você não vai ver nada de novo no sua seção de trabalho QGIS.

Para todos os outros tipos de saída, você encontrará uma caixa de verificação que você pode usar para dizer o algoritmo para saber se carregará o arquivo, uma vez que é gerado pelo algoritmo ou não. Por padrão, todos os arquivos são abertos.

Saídas opcionais não são suportadas. Ou seja, todas as saídas são criadas. No entanto, você pode desmarcar a caixa correspondente, se você não está interessado em um determinado produto, que essencialmente torna comportar-se como uma saída opcional (em outras palavras, a camada é criada de qualquer jeito, mas se você deixar a caixa de texto vazia, será salvo em um arquivo temporário e apagado quando sair QGIS).

17.2.3 Configurando a infraestrutura do processamento

Como foi mencionado, o menu de configuração dá acesso a um novo diálogo onde pode configurar a forma como o algoritmo trabalha. Os parâmetros de configuração são estruturados em blocos separados que podem ser selecionados no lado esquerdo do diálogo.

Juntamente com o que já foi mencionada da entrada *Pasta de saída*, o bloco *Geral* contém parâmetros para configuração de estilos de renderização padrão para camadas de saída (ou seja, camadas geradas pela utilização de algoritmo de qualquer componente da infraestrutura do GUI). Basta criar um estilo que quer usar no QGIS, salva-lo num arquivo, e de seguida introduzir um caminho para o arquivo nas configurações para que sejam usado pelos algoritmos. Cada vez que a camada for carregada pelo SEXTANTE e adicionada ao enquadramento do QGIS, esta será renderizada com esse estilo.

Os estilos de renderização podem ser configurados individualmente para cada algoritmo e cada um para os seus arquivos de saída. Apenas clique com o direito do mouse no nome do algoritmo na caixa de ferramentas e selecione *Editar estilos de renderização*. Irá ver um diálogo como o que é exibido a seguir.

Selecione o arquivo de estilo (`.qml`) que quer para cada arquivo de saída e pressione **[OK]**.

Os outros parâmetros de configuração no grupo *Geral* estão em baixo referidos:

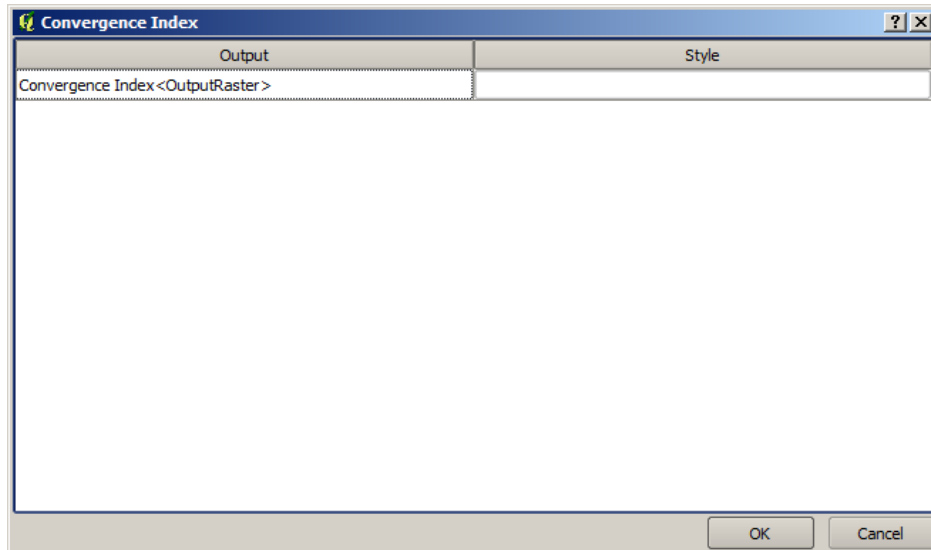


Figura 17.15: Estilos de Renderização

- *Use nome da camada como nome do arquivo.* O nome de cada camada resultante criada por um algoritmo é definido pelo próprio algoritmo. Em alguns casos, um nome fixo pode ser utilizado, o que significa que o mesmo nome de saída vai ser usado, não importa qual a camada de entrada é usada. Em outros casos, o nome pode depender do nome da camada de entrada ou de alguns dos parâmetros usados para executar o algoritmo. Se esta opção for assinalada, o nome será retirado o nome do arquivo de saída em seu lugar. Observe que, se a saída é salvo em um arquivo temporário, o nome deste arquivo temporário é geralmente uma longo e sem sentido que se destina a evitar a conflito com outros nomes de arquivos já existentes.
- *Use somente recursos selecionados.* Se essa opção for selecionada, sempre que uma camada de vetor é usada como entrada para um algoritmo, apenas suas feições selecionadas serão usadas. Se a camada não tiver feições selecionadas, todas as feições serão utilizadas.
- *arquivo de pré-execução do script e arquivo de script pós-execução.* Estes parâmetros se referem aos scripts escritos usando a funcionalidade de script de processamento e são explicados na seção cobertura scripting e do terminal.

Além de o bloco *Geral* na caixa de diálogo de configurações, você também vai encontrar um bloco para provedores de algoritmo. Cada entrada neste bloco contém um ítem *Ativar* que você pode usar para fazer algoritmos aparecer ou não na caixa de ferramentas. Além disso, alguns provedores de algoritmo têm seus próprios itens de configuração, que explicaremos mais tarde, quando cobrindo determinados provedores de algoritmo.

17.3 O modelador gráfico

O *modelador gráfico* permite que possa criar modelos complexos usando uma simples interface fácil-de-usar. Quando trabalha com um SIG, a maioria das operações de análises não são isoladas, mas sim parte das cadeias de operações. Usando um modelador gráfico, a cadeia de processos podem ser agregados num único processo, sendo mais fácil e mais conveniente a sua execução de um processo único num conjunto de diferentes dados de entrada. Não importa o número de etapas e os diferentes algoritmos envolvidos, o modelo é executado como um algoritmo único, desta forma vai poupar tempo e esforços, especialmente para modelos largos.

O modelador pode ser aberto a partir do menu processamento.

O modelador tem uma área de trabalho onde a estrutura do modelo e o sue fluxo de trabalho são representados como está exibido. Na parte esquerda da janela, um painel com dois separadores podem ser usados para adicionar novos elementos ao modelo.

A criação de um modelo envolve dois passos:

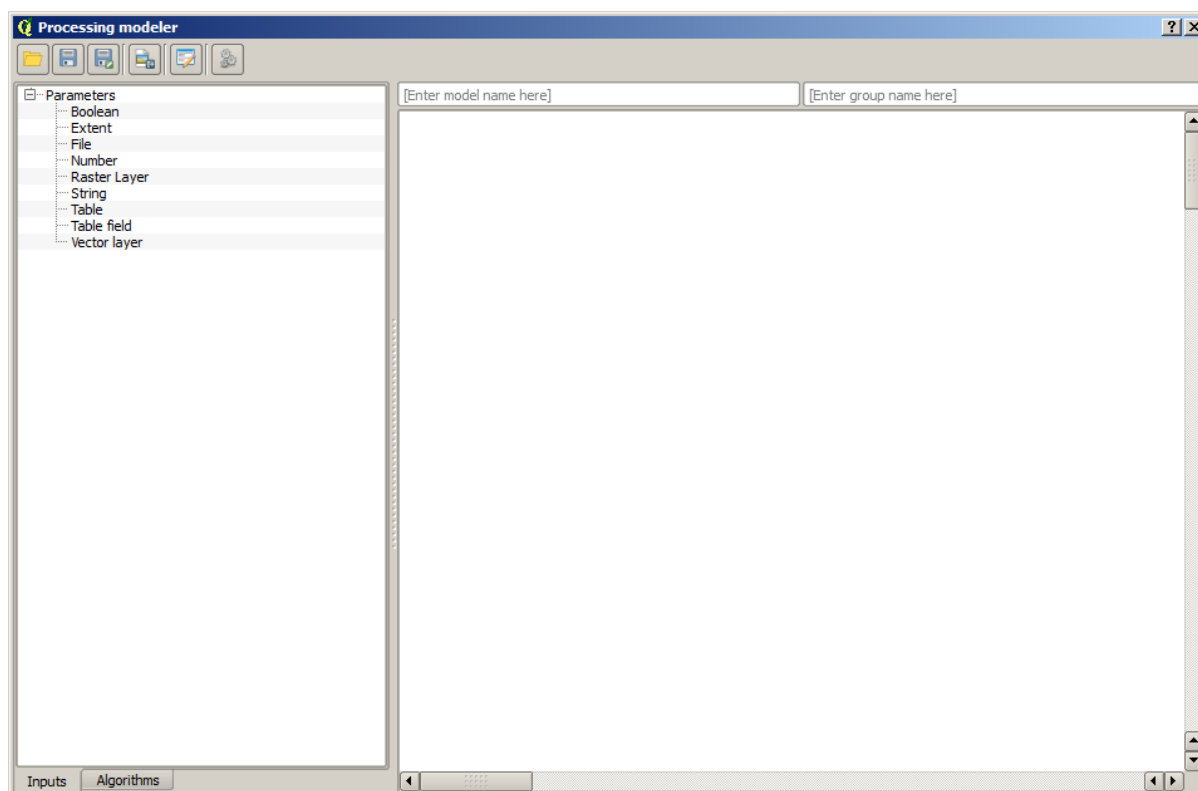



Figura 17.16: Modelador 

1. *Definição dos dados de entrada necessários.* Estes dados de entrada serão adicionados na janela de parâmetros, para que o utilizador possa configurar os seus valores quando executa o modelo. O modelo por si é um algoritmo, portanto os parâmetros da janela é gerado automaticamente como acontece em todos os algoritmos disponíveis na infraestrutura de processamento.
2. *Definição do fluxo de trabalho.* Usando os dados de entrada do modelo, o fluxo de trabalho é definido adicionando algoritmos e selecionando como vão usar esses arquivos de entrada ou de saída gerados por outros algoritmos que já existem no modelo

17.3.1 Definição das entradas

O primeiro passo para criar um modelo é definir as entradas que necessita. Os seguintes elementos são encontrados no separador *Entradas* no lado esquerdo da janela do modelador:

- Camada raster
- Camada vetorial
- Cadeia de texto
- Campo da tabela
- Tabela
- Extensão
- Número
- Booleano
- Arquivo

Faça duplo clique em qualquer um, e será exibido um diálogo para definir as suas características. Dependendo do parâmetro, o diálogo irá conter apenas um elemento básico (a descrição, que será o que o usuário irá ver quando

executar o modelo) ou outros. Por exemplo, quando quer adicionar um valor numérico, como aparece na próxima figura, além da descrição do parâmetro tem um conjunto de valor por defeito e um intervalo de valores válidos.

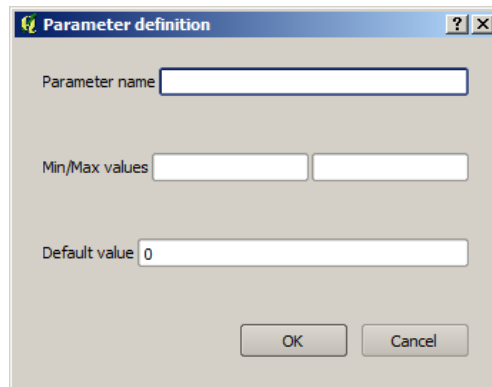


Figura 17.17: Parâmetros do Modelo

Para cada entrada adicionada, um novo elemento é adicionado ao enquadramento do modelador.



Figura 17.18: Parâmetros do Modelo

Pode também adicionar arquivos de entrada arrastando o tipo de entrada a partir de uma lista e largar no enquadramento do modelador, na posição onde quer que fique.

17.3.2 Definição do fluxo de trabalho

Uma vez as entradas definidas, é tempo de definir os algoritmos que vamos aplicar. Os algoritmos podem ser encontrados no separador *Algoritmos*, agrupados na mesma forma que estão na caixa de ferramentas.

A aparência da caixa de ferramentas tem dois modos: simplificado e avançado. Contudo, não existe nenhum elemento para trocar entre as vistas no modelador, e pode fazê-lo na caixa de ferramentas. O modo que é selecionado na caixa de ferramentas é aquele que será usado para a lista de algoritmos no modelador.

Para adicionar um algoritmo ao modelo, clique duas vezes no seu nome. Um diálogo de execução irá aparecer, com um conteúdo similar ao que é encontrado no painel de execução que é exibido quando da execução do algoritmo a partir da caixa de ferramentas. O algoritmo que é exibido corresponde ao algoritmo SAGA ‘Convergence index’, o mesmo que viu na seção dedicada na caixa de ferramentas.

Como pode ver, existem algumas diferenças. Em vez da caixa de saída do arquivo ser usada para configurar o caminho do arquivo para as camadas e tabelas de saída, é usado apenas uma caixa de texto simples. Se a camada gerada pelo algoritmo é apenas um resultado temporário, este será usado como o arquivo de entrada de outro algoritmo e não será mantido como resultado final, mas não edite essa caixa de texto. Introduzindo alguma coisa nela, significa que o resultado é final, e o texto que forneceu será a descrição para o arquivo de saída, que será aquele que o usuário irá ver quando executar o modelo.

A seleção do valor para cada parâmetro é também um pouco diferente, uma vez que existem diferenças importantes entre o contexto do modelador e a caixa de ferramentas. Vamos ver como é introduzido os valores para cada tipo de parâmetro.

- Camadas (raster e vetor) e tabelas. Estas são selecionados de uma lista, mas só em caso dos valores possíveis serem camadas ou tabelas carregadas no QGIS, a lista dos dados de entrada do modelo do tipo correspondente, ou outras camadas ou tabelas geradas por algoritmos que já tenham sido adicionadas ao modelo.

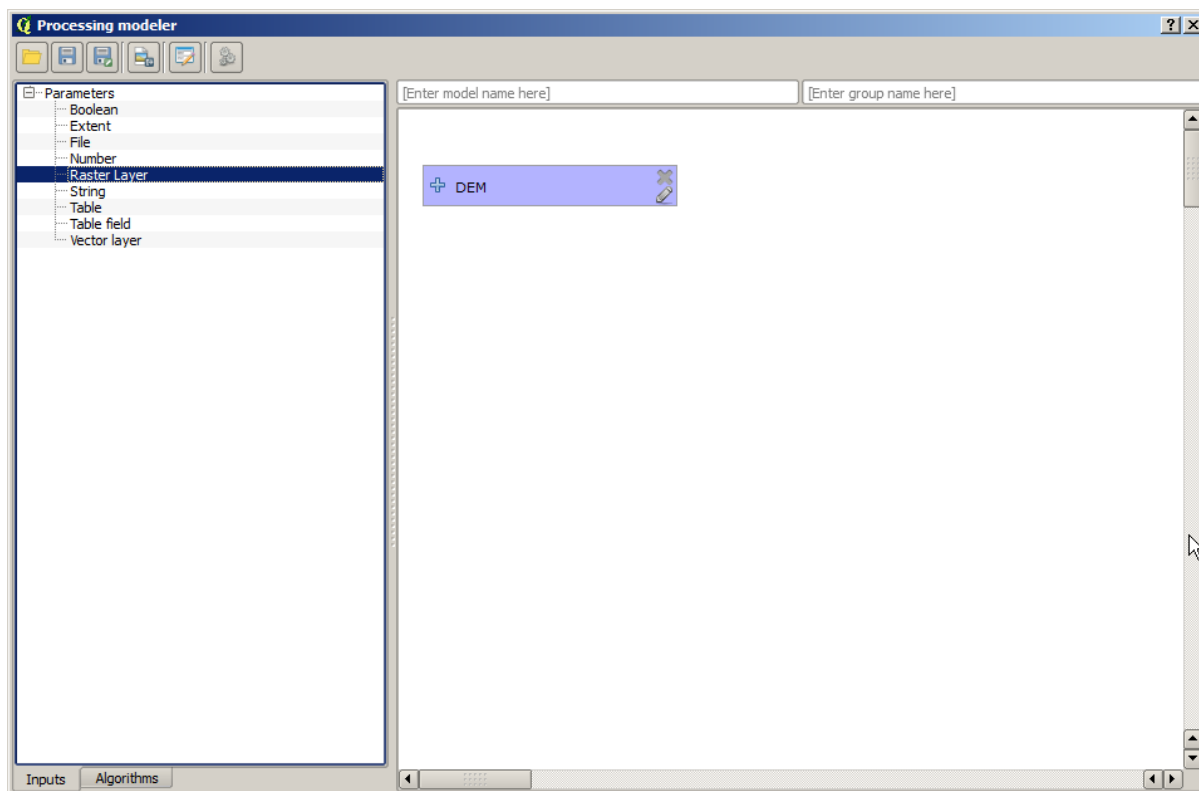


Figura 17.19: Parâmetros do Modelo

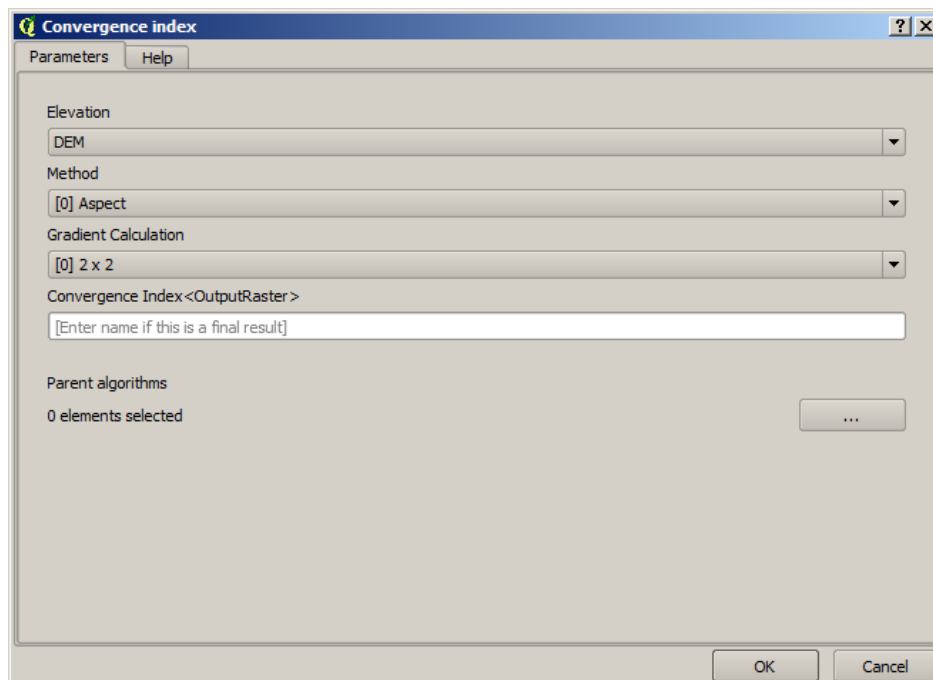


Figura 17.20: Parâmetros do Modelo

- Valores numéricos. Valores literais podem ser introduzidos diretamente na caixa de texto. Mas esta caixa de texto é também uma lista que pode ser usada para selecionar qualquer valor numérico de entrada do modelo. Nesse caso, o parâmetro irá tomar o valor introduzido pelo utilizador quando executado o modelo.
- Cadeia de texto. Como no caso dos valores numéricos, as cadeias de texto podem ser introduzidas, ou uma cadeia de texto de entrada pode ser selecionada.
- Campo da Tabela. Os campos de uma tabela origem ou camada não podem ser conhecidas na altura do seu desenho, uma vez que dependem da seleção do usuário cada vez que o modelo é executado. Para definir o valor para este parâmetro, introduza o nome do campo diretamente na caixa de texto, ou usa a lista para selecionar a entrada do campo de tabela que já está adicionada no modelo. A validade de um campo selecionado irá ser verificado no seu processo de execução.

Em todos os casos, irá encontrar um parâmetro adicional denominado de *Algoritmos Parentes* que não está disponível quando é chamado o algoritmo a partir da caixa de ferramentas. Este parâmetro permite que possa definir a ordem de como os algoritmos são executados, definindo explicitamente um algoritmo como parente do atual, que irá força-lo a executá-lo antes.

Quando usa um arquivo de saída de um algoritmo prévio como arquivo de entrada do seu algoritmo, isso implica definir o antigo como parent do atual (e posiciona a seta correspondente no enquadramento do modelador), Contudo, em alguns casos o algoritmo pode depender de outro mesmo que não use um objeto de saída a partir dele (por exemplo, um algoritmo que executa uma instância SQL numa base de dados PostGIS e outra que importa uma camada para a mesma base de dados). Nesse caso, apenas selecione no parâmetro *Algoritmos Parent* e eles irão ser executados na ordem correta.

Uma vez todos os parâmetros estiver relacionados a valores válidos, clique no botão **[OK]** e o algoritmo irá ser adicionado ao enquadramento. Será ligado a todos os outros elementos do enquadramento, sejam eles algoritmos ou arquivos de entrada, o que irá fornecer objetos que foram usados como arquivos de entrada para esse algoritmo.

Os elementos podem ser arrastados para diferentes posições dentro do enquadramento, para alterar a forma como é exibida a estrutura do módulo e tornar mais claro e intuitivo. As ligações entre os elementos são atualizadas automaticamente. Pode ampliar ou afastar usando a roda do mouse.

Pode correr o seu algoritmo em qualquer altura clicando no botão **[Executar]**. Contudo, se quiser usá-lo a partir da caixa de ferramentas, é necessário salva-lo e fechar o diálogo do modelador, para permitir que a caixa de ferramentas atualize os seus conteúdos.

17.3.3 Salvando e carregando os modelos

Use o botão **[Salvar]** para salvar o modelo atual e o **[Abrir]** para abrir qualquer modelo anteriormente salvo. Os modelos são salvos com a extensão `.model`. Se o modelo for previamente salvo da janela do modelador, não lhe irá ser solicitado por um nome de arquivo, uma vez que já existe um arquivo associado ao modelo, e será usado.

Antes de salvar o modelo, necessita de introduzir o nome e o grupo do modelo, usando as caixas de texto para parte superior da janela.

Os modelos salvos na pasta `modelos` (a pasta padrão quando é pedido o nome do arquivo para salvar o modelo) irá aparecer na caixa de ferramentas na ligação correspondente. Quando a caixa de ferramentas é chamada, ele procura na pasta `modelos` por arquivos com a extensão `.model` e carrega os modelos salvos. Uma vez que o modelo é um próprio algoritmo, este pode ser adicionado na caixa de ferramentas como qualquer outro algoritmo.

A pasta de modelos pode ser configurada a partir do diálogo de configuração do processamento, no grupo *Modelador*

Os modelos carregados a partir da pasta `modelos` aparecem na caixa de ferramentas, mas também na árvore de algoritmos do separador *Algoritmos* da janela do modelador. Isto significa que pode incorporar o modelo como parte de um modelo maior, tal como adiciona qualquer outro algoritmo.

Em alguns casos, um modelo pode não ser carregado porque nem todos os algoritmos incluídos estão disponíveis. Se tiver usado um dado algoritmo como parte do seu modelo, este deverá estar disponível (isto é, deverá aparecer na caixa de ferramentas) de forma a carregar o modelo. Desativando um fornecedor de algoritmos nas configurações do da janela do processamento poderá desativar o modelador, o que pode trazer problemas quando carrega os modelos. Tenha isso em atenção quando tiver problemas a carregar ou a executar modelos.

17.3.4 Editando um modelo

Pode editar o modelo que criou, redefinindo o fluxo de trabalho e as relações entre os algoritmos e os dados de entrada que definiu no modelo.

Se clicar com o direito do mouse num algoritmo na área que representa o modelo, irá ver o menu contexto como é mostrado a seguir:

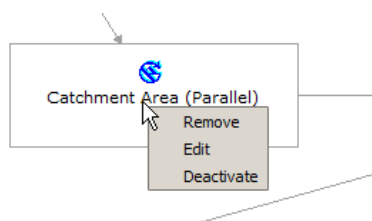


Figura 17.21: Clique direito do Modelador

Selecionando a opção *Remove* irá fazer com que o algoritmo seja removido. Um algoritmo só pode ser removido apenas se existir outros algoritmos dependentes dele. Isto é, se não estiver sendo usado num arquivo de saída de um algoritmo usado como arquivo de entrada diferente. Se tentar remover um algoritmo que tenha outros que dependam dele, uma mensagem de aviso igual à que vê aqui em baixo:

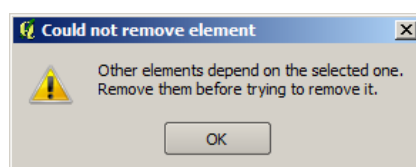


Figura 17.22: Não pode apagar o Algoritmo

Selecionando a opção *Editar* ou fazendo simplesmente duplo clique no ícone do algoritmo, irá ser exibido o diálogo de parâmetros do algoritmo, para que possa mudar os arquivos de entrada e os valores do parâmetro. Tenha atenção pois nem todos os elementos disponíveis no modelo aparecerão neste caso como arquivos de entrada disponível. Camadas ou valores gerados num passo mais avançado no fluxo de trabalho definido pelo modelo não irá estar disponível se causar dependências em ciclo.

Selecione novos valores e de seguida clique no botão **[OK]** como comum. As ligações entre os elementos do modelo irão alterar de acordo com o enquadramento do modelador.

17.3.5 Editando os arquivos de ajuda do modelo e a meta-informação

Pode documentar os seus modelos a partir do próprio modelador. Basta apenas clicar no botão **[Editar ajuda do modelo]** e o diálogo como aquele que irá aparecer.

No lado direito irá ver uma página simples HTML, criada usando a descrição dos parâmetros de entrada e de saída do algoritmo, juntamente com itens adicionais como uma descrição geral do modelo ou o seu autor. A primeira vez que abre o editor da ajuda, todas as ajudas estão vazias, mas pode editá-los usando os elementos do lado esquerdo do diálogo. Selecione um elemento na parte superior e escreva a descrição na caixa de texto em baixo.

A ajuda do Modelo é salva num arquivo na mesma pasta ao pé do seu respectivo modelo. Não tem de ser preocupar como salvá-lo, uma vez ser feito automaticamente.

17.3.6 Sobre os algoritmos disponíveis

Poderá notar que alguns algoritmos podem ser executados a partir da caixa de ferramentas mas não aparecem na lista de algoritmos disponíveis quando está a desenhar o modelo. Para ser incluído no modelo, o algoritmo deve ter uma semântica correta, assim como ligações apropriadas a outros fluxos de trabalho. Se um algoritmo não tiver

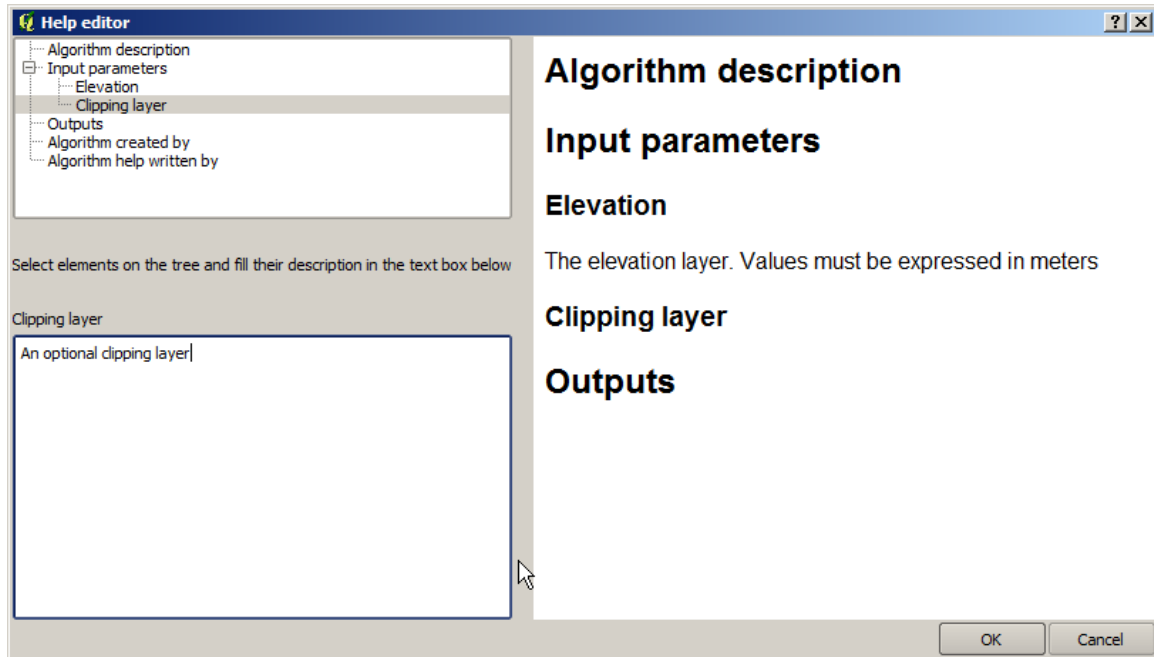


Figura 17.23: Edição da Ajuda

uma boa semântica bem definida (por exemplo, se o número de camadas de saída não são conhecidas), portanto não é possível usá-lo dentro do modelo, e não aparece na lista na janela do diálogo do modelador.

Adicionalmente, irá ver algoritmos no modelador que não se encontram na caixa de ferramentas. Esses algoritmos são para ser usados exclusivamente como parte do modelo, e eles não têm interesse em outro contexto diferente. O algoritmo 'Calculadora' é um exemplo disso. É apenas uma simples calculadora aritmética que pode usar para modificar valores numéricos (introduzidos pelo utilizador ou gerados por outro tipo de algoritmo). Estas ferramentas são realmente úteis dentro do modelo, mas fora do contexto, não fazem sentido.

17.4 A interface do processamento em lote

17.4.1 Introdução

Todos os algoritmos (incluindo modelos) podem ser executados como um processo descontínuo. Isto é, eles podem ser executados não utilizando apenas um único conjunto de entradas, mas alguns delas, a execução do algoritmo tantas vezes quanto necessário. Isto é útil quando se processa grandes quantidades de dados, uma vez que não será necessário iniciar o algoritmo várias vezes na caixa de ferramentas.

Para executar um algoritmo como um processamento em lote, clique com o direito do mouse na caixa de ferramentas e selecione a opção *Executar como processamento em lote* no balão de opções que irá aparecer.

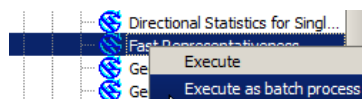


Figura 17.24: Clique no botão direito do mouse para processar em lote

17.4.2 Os parâmetros da tabela

A execução de um processamento em lote é semelhante à execução única de um algoritmo. Os valores do parâmetro têm de ser definidos, mas neste caso nós necessitamos um conjunto de parâmetros, uma para cada vez que o algoritmo é executado. Os valores são introduzidos usando uma tabela como é mostrada a seguir.

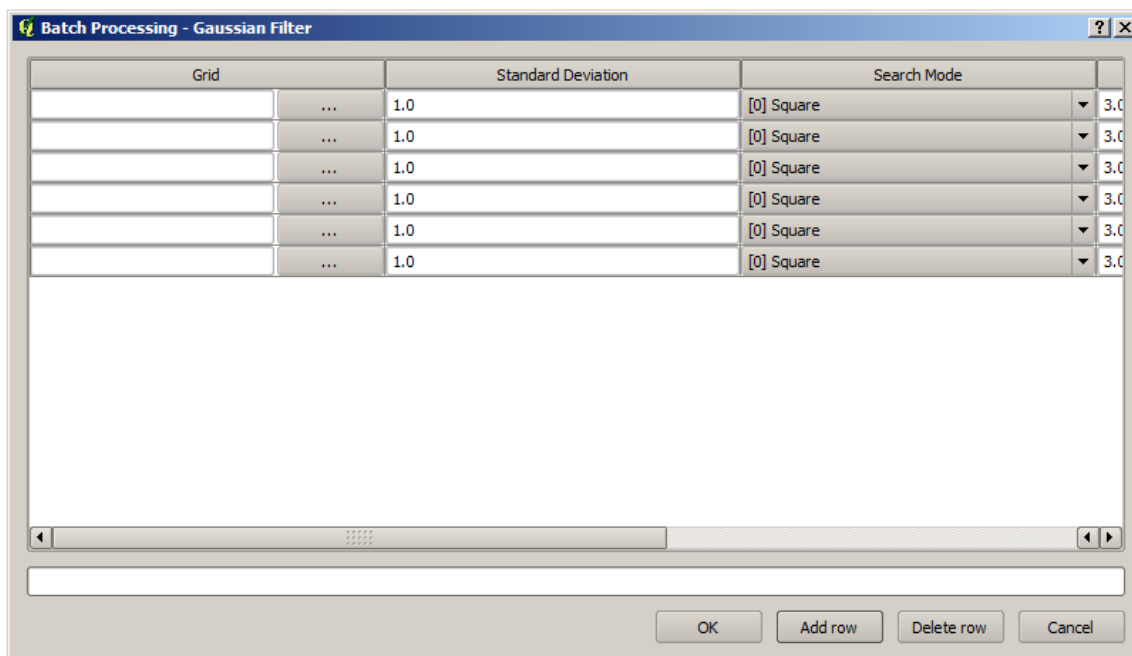


Figura 17.25: Processamento em Lote

Cada linha desta tabela representa uma única execução do algoritmo, e cada célula contém o valor de um dos parâmetros. É semelhante aos parâmetros do diálogo que vê quando executa o algoritmo a partir da caixa de ferramentas, mas com uma diferente disposição.


Por defeito, a tabela contém apenas duas linhas. Pode adicionar ou remover linhas usando os botões na parte inferior da janela.

Uma vez que o tamanho da tabela for configurado, terá de ser preenchido com os valores desejados.

17.4.3 Preenchendo os parâmetros da tabela

Para a maioria dos parâmetros, a configuração do valor é trivial. Basta digitar o valor ou seleccioná-lo na lista de opções disponíveis, dependendo do tipo de parâmetro.

As principais diferenças são encontradas para os parâmetros que representam camadas ou tabelas, e caminho de arquivo de saída. Quanto camadas de entrada e tabelas, representam um algoritmo que será executado como parte de um processo de grupo, os objetos de dados de entrada são levados diretamente a partir de arquivos, e não a partir do conjunto deles já aberto no QGIS. Por esta razão, qualquer algoritmo pode ser executado como um processo em lote, mesmo que os objetos de dados ao todo não sejam abertos e o algoritmo não possa ser executado a partir da caixa de ferramentas.

Nomes de arquivos de objetos de dados de entrada são introduzidos digitando diretamente ou, mais convenientemente, clicando no botão  do lado direito da célula, o que mostra uma janela típica de seleção de arquivos. Múltiplos arquivos podem ser seleccionados de cada vez. Se o parâmetro de entrada representa um único objeto de dados e vários arquivos são seleccionados, cada um deles será colocado em uma linha separada, acrescentando novos, se necessário. Se o parâmetro representa uma entrada múltipla, todos os arquivos seleccionados serão adicionados a uma única célula, separados por ponto e vírgula (;).

Objetos de dados de saída sempre são salvos em um arquivo e, ao contrário, quando a execução de um algoritmo a partir da caixa de ferramentas, salvar em um arquivo temporário não é permitido. Você pode digitar o nome diretamente ou usar a janela de diálogo de seleção de arquivos que aparece ao clicar no botão que o acompanha.

Um vez selecionado o arquivo, um novo diálogo será exibido para permitir que faça o auto-preenchimento das outras células na mesma coluna (o mesmo parâmetro).

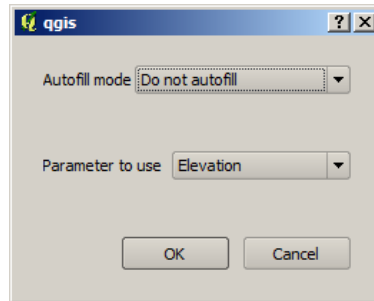


Figura 17.26: Salvar Processamento em Lote

Se o valor padrão ('Não efetuar auto-preenchimento') é selecionado, irá apenas por o nome do arquivo selecionado na célula selecionada da tabela de parâmetros. Se alguma das outras opções estiver selecionada, todas as células abaixo da selecionada irão automaticamente ser preenchidas baseando-se no critério definido. Desta forma, é mais fácil preencher a tabela, e o processamento em lote pode ser definido com menor esforço.

O preenchimento automático pode ser feito simplesmente adicionando números relativos ao caminho do arquivo selecionado, ou acrescentando o valor de outro campo na mesma linha. Isto é particularmente útil para nomear objetos de dados de saída de acordo com os de entrada.

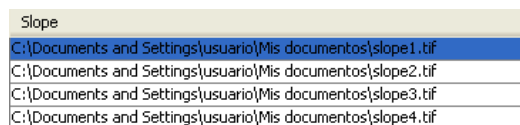


Figura 17.27: Caminho do Arquivo do Processamento em Lote

17.4.4 Executando o processo em lote

Após a introdução de todos os valores necessários pode executar o processamento em lote, apenas clicando em [OK]. O progresso global da tarefa de processamento será exibido na barra de progresso na parte inferior do diálogo.

17.5 Usando os algoritmos do processamento a partir do Terminal Python.

O terminal permite que usuários avançados aumente sua produtividade e execute operações complexas que não podem ser utilizados através dos outros elementos GUI da caixa de processamento. Modelos envolvendo vários algoritmos podem ser definidos usando a interface de linha de comando e operações adicionais como loops e sentenças condicionais podem ser adicionados para criar fluxos de trabalho mais flexíveis e poderosos.

Não há um terminal de processamento específico no QGIS, mas todo os comandos de processamentos estão disponíveis embutidos no Terminal Python em vez do QGIS. Isso significa que poderá incorporar os comandos em seu terminal de trabalho e conectar algoritmos de processamentos de todos os outros recursos disponíveis (incluindo métodos da | qg | API) a partir deste terminal.

O código que pode executar a partir do Terminal Python, mesmo quando não chama nenhum método específico do processamento, pode ser convertido num novo algoritmo que pode mais tarde chamar da caixa de ferramentas, o modelador gráfico ou qualquer outro componente, como faz para outro algoritmo. De fato, alguns algoritmos que encontra na caixa de ferramentas são scripts simples.

Nesta seção, veremos como usar algoritmos de processamento do QGIS console Python e também a forma de escrever algoritmos usando Python.

17.5.1 Chamando os algoritmos a partir do Terminal Python

A primeira coisa que tem de fazer é importar as funções do processamento com a seguinte linha:

```
>>> import processing
```

Agora, basicamente é uma (interessante) coisa que pode fazer a partir do terminal: executar um algoritmo. Isso é feito usando o método `runalg()`, que toma o nome do algoritmo a ser executado como o seu primeiro parâmetro, então o número da variável de um parâmetro adicional depende dos requisitos do algoritmo. Portanto a primeira coisa que necessita saber é o nome do algoritmo a executar. Não será o nome que vê na caixa de ferramentas, mas um único nome da linha de comando. Para encontrar o nome correto do seu algoritmo, pode usar o método `alplist()`. Introduza a seguinte linha no seu terminal:

```
>>> processing.alplist()
```

Irá ver algo como isto.

```
Accumulated Cost (Anisotropic)----->saga:accumulatedcost (anisotropic)
Accumulated Cost (Isotropic)----->saga:accumulatedcost (isotropic)
Add Coordinates to points----->saga:addcoordinatestopoints
Add Grid Values to Points----->saga:addgridvaluestopoints
Add Grid Values to Shapes----->saga:addgridvaluestoshapes
Add Polygon Attributes to Points----->saga:addpolygonattributestopoints
Aggregate----->saga:aggregate
Aggregate Point Observations----->saga:aggregatepointobservations
Aggregation Index----->saga:aggregationindex
Analytical Hierarchy Process----->saga:analyticalhierarchyprocess
Analytical Hillshading----->saga:analyticalhillshading
Average With Mask 1----->saga:averagewithmask1
Average With Mask 2----->saga:averagewithmask2
Average With Thershold 1----->saga:averagewiththershold1
Average With Thershold 2----->saga:averagewiththershold2
Average With Thershold 3----->saga:averagewiththershold3
B-Spline Approximation----->saga:b-splineapproximation
...
```

Esta é a lista de todos os algoritmos disponíveis, ordenados alfabeticamente, juntamente com os seus nomes da linha de comandos correspondentes.

É possível usar uma cadeia de texto como parâmetro para este método. Em vez de retornar a lista completa de algoritmo, ele só irá exibir aqueles que incluem esta cadeia. Se, por exemplo, está procurando um algoritmo para calcular a inclinação de um MDE, digite `alplist("slope")` para obter o seguinte resultado:

```
DTM Filter (slope-based)----->saga:dtmfilter (slope-based)
Downslope Distance Gradient----->saga:downslopedistancegradient
Relative Heights and Slope Positions----->saga:relativeheightsandslopepositions
Slope Length----->saga:slopelength
Slope, Aspect, Curvature----->saga:slopeaspectcurvature
Upslope Area----->saga:upslopearea
Vegetation Index[slope based]----->saga:vegetationindex[slopebased]
```

Este resultado pode mudar dependendo dos algoritmos que estão disponíveis.

Assim é mais fácil encontrar o algoritmo que procura e o seu nome da linha de comandos, neste caso `saga:slopeaspectcurvature`.

Uma vez conhecido o nome da linha de comandos do algoritmo, a próxima coisa a fazer é determinar a sintaxe correta para executá-lo. Isto significa saber quais os parâmetros necessários e a ordem que devem passar quando chamado pelo método `runalg()`. Existe um método para descrever um algoritmo em detalhe, o qual pode ser usado para obter uma lista de parâmetros que requer uns algoritmos e os arquivos de saída que irão gerar. Para fazer isso, pode usar o método `alghelp(name_of_the_algorithm)`. Use o nome do algoritmo na linha de comando, não o nome descritivo completo.

Chamado o método `saga:slopeaspectcurvature` como parâmetro, você terá a seguinte descrição.

```
>>> processing.alghelp("saga:slopeaspectcurvature")
ALGORITHM: Slope, Aspect, Curvature
  ELEVATION <ParameterRaster>
  METHOD <ParameterSelection>
  SLOPE <OutputRaster>
  ASPECT <OutputRaster>
  CURV <OutputRaster>
  HCURV <OutputRaster>
  VCURV <OutputRaster>
```

Agora tem tudo o que necessita para correr qualquer algoritmo. Como já tínhamos mencionado, existe apenas um comando para executar algoritmos: `runalg()`. A sua sintaxe é como está descrito a seguir:

```
>>> processing.runalg(name_of_the_algorithm, param1, param2, ..., paramN,
  Output1, Output2, ..., OutputN)
```

A lista de parâmetros e arquivos de saída para adicionar dependem do algoritmo que quer correr, e é exatamente a lista que o método `alghelp()` lhe dá, na mesma ordem que é exibido.

Dependendo do tipo de parâmetro, os valores são introduzidos diferentemente. A próxima explicação é uma rápida revisão de como introduzir valores para cada tipo de parâmetro de entrada:

- Camada Raster, Camada Vetorial e Tabela. Simplesmente usa uma cadeia de texto com nome que identifica o objeto de dados a usar (o nome está na Tabela de Conteúdos do QGIS) ou o nome do arquivo (se a camada correspondente não for aberta, mas não adicionada no enquadramento do mapa). Se tiver uma instância do objeto QGIS representado na camada, pode também passá-lo como parâmetro. Se o arquivo de entrada for opcional e não quer usá-lo como qualquer tipo de objeto de dados, use `None`.
- Seleção. Se algum algoritmo tiver um parâmetro de seleção, o valor desse parâmetro deve ser introduzido usando um valor inteiro. Para saber as opções disponíveis, pode usar o comando `algorithms()`, como é exibido no seguinte exemplo:

```
>>> processing.algorithms("saga:slopeaspectcurvature")
METHOD (Method)
  0 - [0] Maximum Slope (Travis et al. 1975)
  1 - [1] Maximum Triangle Slope (Tarboton 1997)
  2 - [2] Least Squares Fitted Plane (Horn 1981, Costa-Cabral & Burgess 1996)
  3 - [3] Fit 2.Degree Polynom (Bauer, Rohdenburg, Bork 1985)
  4 - [4] Fit 2.Degree Polynom (Heerdegen & Beran 1982)
  5 - [5] Fit 2.Degree Polynom (Zevenbergen & Thorne 1987)
  6 - [6] Fit 3.Degree Polynom (Haralick 1983)
```

Neste caso, o algoritmo tem um dos referidos parâmetros com 7 opções e sua ordenação é iniciada em zero.

- Múltiplas entradas. O valor é uma cadeia de texto com descritores de entrada separados por ponto e vírgula (;). Como no caso das camadas ou tabelas únicas, cada descritor de entrada pode ser o nome do objeto ou o caminho do arquivo.
- Campo da Tabela de XXX. Use uma cadeia de texto com o nome do campo a usar. O parâmetro é caso sensetivo.
- Tabela Fixa. Digite a lista de todos os valores da tabela separadas por vírgulas (,) e feche entre aspas ("). Os valores começam na linha superior e vai da esquerda para a direita. Pode também usar uma matriz 2D de valores que representam a tabela.
- SRC. Introduza o número do código EPSG do SRC desejado.

- Extensão. Deve usar uma cadeia de texto com `xmin`, `xmax`, `ymin` e `ymax` valores separados por vírgulas (,).

Os parâmetros booleanos, de arquivo, cadeia de texto e numéricos não necessitam de explicações adicionais.

Os parâmetros de entrada, como textos, booleanos, ou valores numéricos têm valores padrão. Para usá-los, especificar `None` na entrada do parâmetro correspondente.

Para objetos de saída de dados, digite o caminho de arquivo a ser utilizado para salvá-lo, como é feito a partir da caixa de ferramentas. Caso queira salvar o resultado em um arquivo temporário, use `None`. A extensão do arquivo determina o formato do arquivo. Se você digitar uma extensão de arquivo não suportada pelo algoritmo, será utilizado o formato de arquivo padrão para esse tipo de saída e a sua extensão correspondente anexado ao conjunto de arquivos de dados.

Contrariamente, quando o algoritmo é executado a partir da caixa de ferramentas, os arquivos de saída não são adicionados ao enquadramento do mapa se executar o mesmo algoritmo a partir do Terminal Python. Se quiser adicionar um arquivo de saída, você tem de adicioná-lo depois de correr o algoritmo, Para o fazer, pode usar os comandos do API QGIS ou, mais fácil, use um dos métodos úteis para esta tarefa.

O método `runalg` retorna um dicionário com os nomes de saída (os que são exibidos na descrição do algoritmo) como chaves ou caminhos de arquivo dessas saídas como valores. Pode carregar essas camadas passando o seu caminho de arquivo correspondente pelo método `load()`.

17.5.2 Funções adicionais para a manipulação dos dados

Além das funções usadas para chamar os algoritmos, importar o pacote `processamento` irá também importar algumas funções adicionais que facilitará o trabalho dos dados, particularmente os dados vetoriais. São funções de conveniência que envolvem alguma funcionalidade a partir da API do QGIS, usualmente com uma sintaxe menos complexa. Estas funções devem ser usadas quando são programados novos algoritmos, para tornar mais fácil a operação com o os dados de entrada.

Abaixo está uma lista de algum dos comandos. Mais informação podem ser encontradas nas classes sob o pacote `processing/tools` e também nos exemplos de script fornecidos no QGIS.

- `getObject(obj)`: Retorna um objeto QGIS (a camada ou tabela) do objeto passado, o que pode ser um nome ou o nome do objeto no Índice QGIS.
- `values(layer, fields)`: Retorna os valores da tabela de atributos de uma camada vetorial para os campos validados. Os campos podem ser validados como nomes de campos ou índices de campo iniciados em zero. Retorna um dicionário de listas com campos de identificadores validados como chaves. Considera a seleção existente.
- `features(layer)`: Retorna um iterador sobre as feições de uma camada vetorial, considerando a seleção existente.
- `UniqueValues(layer, field)`: Retorna uma lista de valores exclusivos para um determinado atributo. Os atributos podem ser passado como um nome de campo ou um índice de campo baseado em zero. Ele considera a seleção existente.

17.5.3 Criando scripts e correndo-os a partir da caixa de ferramentas

Pode criar os seus próprios algoritmos através da escrita do código Python correspondente e adicionar algumas linhas extras para fornecer informação adicional necessária para definir as semânticas do algoritmo. Pode encontrar um *Criar novo script* menu sob o `:guilabel: Tools` no grupo *Script* bloco de algoritmos da caixa de ferramentas. Faça duplo clique nele e irá abrir um diálogo de edição do script. É onde deverá digitar o código. Guarde o script na pasta `scripts` (a pasta padrão quando abre o diálogo de guardar ficheiro), com a extensão `.py`, irá automaticamente criar o algoritmo correspondente.

O nome do algoritmo (aquele que irá ver na caixa de ferramentas) é criado a partir do nome do arquivo, removendo a extensão e substituindo os hífens inferiores com espaços em branco.

Vamos ter o seguinte código, que calcula o Índice Topográfico de Umidade (ITW) diretamente do MDE.


```

##dem=raster
##twi=output
ret_slope = processing.runalg("saga:slopeaspectcurvature", dem, 0, None,
                             None, None, None, None)
ret_area = processing.runalg("saga:catchmentarea(mass-fluxmethod)", dem,
                             0, False, False, False, False, None, None, None, None, None)
processing.runalg("saga:topographicwetnessindex(twi)", ret_slope['SLOPE'],
                 ret_area['AREA'], None, 1, 0, twi)

```

Como pode ver, o cálculo envolve 3 algoritmos, todos eles provêm do SAGA. O último deles calcula o ITW, mas necessita da camada de dados do declive e outra de acumulação de escoamento. Nós não temos estas camadas, mas uma vez que temos o MDE, podemos calculá-los chamando os algoritmos SAGA correspondentes.

A parte do código onde este processamento tem lugar não é difícil de perceber se leu-o as seções anteriores deste capítulo. Contudo, as primeiras linhas, necessitam de uma explicação adicional. Eles fornecem a informação que é necessária para tornar o código num algoritmo que possa ser corrido a partir qualquer componente do GUI, como por exemplo a caixa de ferramentas ou o modelador gráfico.

Estas linhas começam com o símbolo de duplo comentário Python (##) e tem a seguinte estrutura:

```
[parameter_name]=[parameter_type] [optional_values]
```

Aqui está uma lista de todos os tipos de parâmetros que são suportados nos scripts de processamento, sua sintaxe e alguns exemplos.

- `raster`. Uma camada raster.
- `vector`. Uma camada vetorial.
- `table`. Uma tabela.
- `number`. Um valor numérico. Um valor padrão deve ser fornecido. Por exemplo, `depth=number 2.4`
- `string`. Um campo de texto. Como no caso de valores numéricos, um valor por padrão deve ser adicionado. Por exemplo, `name=string Victor`
- `boolean`. Um valor booleano. Adicione `True` ou `False` depois de definir o valor padrão. Por exemplo, `verbose=boolean True`
- múltiplos `raster`. Um conjunto de camadas rasters de entrada.
- vetores múltiplos. Um conjunto de camadas vetoriais de entrada.
- `campo`. Um campo da tabela de atributos de uma camada vetorial. O nome da camada tem de ser adicionada depois da etiqueta `campo`. Por exemplo, se declarou um arquivo de entrada vetorial com `mylayer=vector`, poderá usar `myfield=field mylayer` para adicionar o campo a partir dessa camada como parâmetro.
- `folder`. Uma pasta.
- `file`. Nome do arquivo

O nome do parâmetro é o nome que será exibido ao utilizado quando executa o algoritmo, e também o nome da variável a usar no código do script. O valor introduzido pelo usuário para esse parâmetro será atribuído à variável com esse nome.

Quando é exibido o nome do parâmetro ao usuário, o nome irá ser editado para melhorar a aparência, substituindo o hífen inferior com espaços. Portanto, por exemplo, se quer que o usuário veja o parâmetro chamado `Um valor numérico`, pode usar o nome da variável `A_numerical_value`.

Os valores das camadas e tabelas são cadeias de texto que contêm caminhos de arquivos para o objeto correspondente. Para transformá-los em objetoslqgl, pode usar a função `processing.getObjectFromUri()`. Os múltiplos dados de entrada também têm um valor de cadeia de texto, que contêm caminhos de arquivos para todos os objetos selecionados, separados por ponto e vírgulas (;).

Os arquivos de saída são definidos numa maneira semelhante, usando as seguintes etiquetas:

- `raster de saída`

- vetor de saída
- tabela de saída
- html de saída
- arquivo de saída
- número de saída
- cadeia de texto de saída

O valor atribuído às variáveis de saída sempre são conjuntos de caracteres que representam o caminho dos arquivos. Corresponderá a um caminho de arquivos temporários, caso o usuário não inserir um nome de arquivo de saída.

Quando declara um arquivo de saída, o algoritmo irá tentar adicioná-lo no QGIS uma vez finalizado. É por isso que, embora o método `runalg()` não carrega camadas quando as produz, a camada TWI final será carregada (usando o caso do nosso exemplo anterior) uma vez que é salvo no arquivo digitado pelo usuário, que é o valor do arquivo de saída correspondente.

Não use a o método `load()` nos algoritmos, mas apenas quando se trabalha com a linha do terminal. Se a camada é criada como saída de um algoritmo, deve ser declarado como tal. Caso contrário, você não será capaz de usar adequadamente o algoritmo no modelador, desde sua sintaxe (como definido pelas tags, explicado acima) não coincidirá com o que o algoritmo realmente produz.

Saídas ocultas (números e textos) não têm um valor. Em vez disso, o usuário tem que atribuir um valor a eles. Para isso, basta definir o valor de uma variável com um nome para declarar a saída. Por exemplo, se você usou esta declaração,

```
##average=output number
```

a linha seguinte irá configurar o valor de saída para 5:

```
average = 5
```

Em adição às etiquetas para os parâmetros e arquivos de saída, pode também definir o grupo onde o algoritmo será exibido, usando a etiqueta `group`.

Se o algoritmo leva um longo tempo para processar, é uma boa ideia informar ao usuário. Você tem uma chamada global `progress` disponível, com dois métodos: `setText(text)` e `setPercentage(percent)` para modificar o texto de progresso e a barra de progresso.

Vários exemplos são apresentados. Por favor, verifique-os para ver exemplos reais de como criar algoritmos usando as classes do quadro de processamento. Pode clicar com o botão direito do mouse em qualquer script do algoritmo e selecionar *Editar script* para editar o seu código ou apenas para vê-lo.

17.5.4 Documentando os seus scripts

Como no caso dos modelos, você pode criar documentação adicional para o seu script, para explicar o que fazem e como são usados. No diálogo de edição do script irá encontrar um botão [**Editar ajuda do script**]. Clique nele e irá levá-lo para o diálogo de edição da ajuda. Verifique o capítulo sobre o modelador gráfico para saber mais sobre este diálogo e como usá-lo.

Os arquivos de ajuda são salvos na mesma pasta como o próprio script, adicionando a extensão `.help` ao nome do ficheiro. Note que, você pode editar a ajuda do seu script antes de guardá-lo pela primeira vez. Se fechar mais tarde o diálogo de ajuda do script sem salvar o script (ex.: descartá-lo) o conteúdo da ajuda que escreveu será perdido. Se o seu script já estiver salvo e associado ao nome do arquivo, o conteúdo de ajuda será salvo automaticamente.

17.5.5 Pré- e pós-execução de encaixe da script

Os scripts podem também ser usados para definir um encaixe de pré- e pós-execução que correm antes e depois do algoritmo correr. Isto pode ser usado para automatizar tarefas que serão executados quando qualquer algoritmo é

executado.

A sintaxe é idêntica à sintaxe explicada em cima, mas uma variável global `alg` está disponível, representando o algoritmo que foi (ou está prestes a ser) executado.

No grupo *General* do diálogo de configuração do processamento irá encontrar duas entradas denominados de *Pré-execução do arquivo script* e *Post-execution script file* onde os nomes do arquivo dos scripts a serem executados em cada caso podem ser inseridos.

17.6 Gerenciador do histórico

17.6.1 O histórico do processamento

Cada vez que executa um algoritmo, a informação do processo é armazenado no gerenciador do histórico. Juntamente com os parâmetros usados, a data e o tempo de execução são também salvos.

Dessa forma, é fácil rastrear e controlar todo o trabalho que tem sido desenvolvido utilizando a estrutura de processamento e facilmente reproduzi-lo.

O gerenciador do histórico é um conjunto de entradas de registro agrupados de acordo com sua data de execução, tornando-o mais fácil de encontrar informações sobre um algoritmo executado a qualquer momento particular.

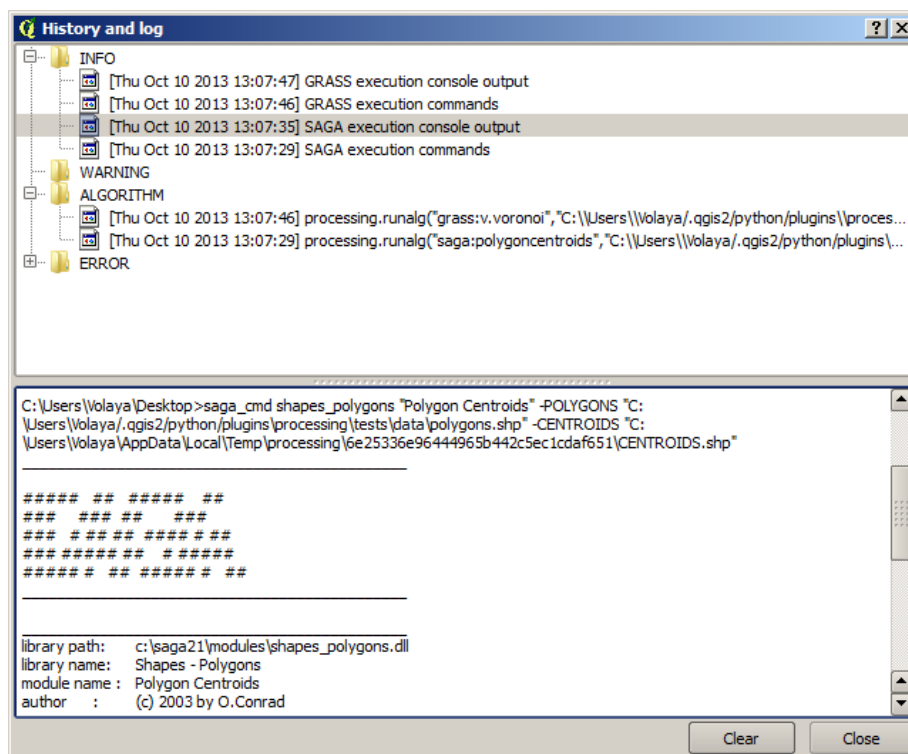


Figura 17.28: Histórico

A informação do processo é mantida como uma expressão de linha de comandos, mesmo se o algoritmo seja lançado a partir da barra de ferramentas. Isto permite ser útil para aqueles que querem aprender como se usa a interface da linha de comandos, uma vez que podem chamar o algoritmo usando a barra de ferramentas e verificar o gerenciador do histórico para ver como o mesmo algoritmo pode ser chamado a partir da linha de comandos.

Além de navegar nas entradas no registro, você também pode re-executar os processos, basta um duplo clique sobre a entrada correspondente.

Junto com a execuções do algoritmo de gravação, a estrutura de processamento se comunica com o usuário por meio de outros grupos de registro, a saber *Erros*, *Advertências* e *Informações*. No caso de algo não está funcionando corretamente, olhando para o *Erros* poderá ajudá-lo a ver o que está acontecendo. Se você entrar em contato com um desenvolvedor para relatar um bug ou erro, a informação nesse grupo vai ser muito útil para ele ou para ele descobrir o que está acontecendo de errado.

Os algoritmos de terceiros normalmente são executados chamando suas interfaces de linha de comando, que se comunicam com o usuário através do terminal. Embora o terminal não seja mostrado, um despejo completo do que é armazenado no grupo *Informações* cada vez que você executar um desses algoritmos. Se, por exemplo, você está tendo problemas executando um algoritmo SAGA, procure uma entrada chamada 'execução do SAGA no terminal de saída' para verificar todas as mensagens geradas pelo SAGA e tentar descobrir onde está o problema.

Alguns algoritmos, mesmo se eles produzirem um resultado com os dados de entrada fornecidos, pode adicionar comentários ou informações adicionais ao bloco *Atenção* se detectar possíveis problemas com os dados, a fim de avisá-lo. Certifique-se de verificar essas mensagens se você está tendo resultados inesperados.

17.7 Escrevendo novos algoritmos de processamento com scripts python

Você pode criar seus próprios algoritmos escrevendo com o código Python correspondente e adicionando algumas linhas extras para fornecer informações adicionais necessárias para definir a semântica do algoritmo. Você pode encontrar um menu *Criar um novo script* no grupo *Ferramentas* no bloco algoritmo *Script* da caixa de ferramentas. Dê um duplo clique sobre ele para abrir o diálogo edição script. É onde você deve digitar seu código. Salvando o script na pasta: 'scripts' (o padrão quando você abrir o diálogo de salvar o arquivo), com: file: 'extensão .py', criará automaticamente o algoritmo correspondente.

O nome do algoritmo (o que você vai ver na caixa de ferramentas) é criado a partir do nome do arquivo, removendo a sua extensão e substituindo baixos hífen com espaços em branco.

Vamos ter o seguinte código, que calcula o índice de vertente topográfica (TWI) diretamente de um MDE

```
##dem=raster
##twi=output raster
ret_slope = processing.runalg("saga:slopeaspectcurvature", dem, 0, None,
                             None, None, None, None)
ret_area = processing.runalg("saga:catchmentarea", dem,
                             0, False, False, False, False, None, None, None, None, None)
processing.runalg("saga:topographicwetnessindextwi", ret_slope['SLOPE'],
                 ret_area['AREA'], None, 1, 0, twi)
```

Como você pode ver, envolve 3 algoritmos, todos eles provenientes da SAGA. O último deles calcula o TWI, mas precisa de uma camada de inclinação e uma camada de acumulação de fluxo. Não temos estas, mas uma vez que temos o MDE, podemos calculá-los chamando os algoritmos SAGA correspondentes.

A parte do código onde esta transformação substitui não é difícil de entender se você já leu o capítulo anterior. As primeiras linhas, no entanto, precisam de alguma explicação adicional. Eles fornecem a informação que é necessário para transformar o seu código em um algoritmo que pode ser executado a partir de qualquer um dos componentes da GUI, como a caixa de ferramentas ou o modelador gráfico.

Estas linhas de começo com um comentário símbolo duplo Python (##) e tem a seguinte estrutura

```
[parameter_name]=[parameter_type] [optional_values]
```

Aqui está uma lista de todos os tipos de parâmetros que são suportadas no processamento de scripts, sua sintaxe e alguns exemplos.

- `raster`. Uma camada raster
- `vetor`. Uma camada vetor
- `tabela`. Uma tabela

- `número`. Um valor numérico. Um valor padrão deve ser fornecido. Por exemplo, `profundidade=número 2.4`
- `string`. Uma cadeia de texto. Como no caso de valores numéricos, um valor padrão deve ser adicionado. Por exemplo, `nome=texto Vitor`
- `longstring`. Igual um texto, mas uma caixa de texto maior será mostrado, por isso, é mais adequado para textos longos, como para um script esperam um trecho de código pequeno.
- `“booleano“`. Um valor booleano. Adicionar `Verdadeiro` ou `“Falso“` depois dele, para definir o valor padrão. Por exemplo, `verbose=booleano Verdadeiro`.
- `raster multiplo`. Um conjunto de camadas raster de entrada.
- `vetor multiplo`. Um conjunto de camadas vetor de entrada.
- `Campo`. Um campo na tabela de atributos de uma camada de vetor. O nome da camada tem que ser adicionado após a tag `Campo`. Por exemplo, se você declarou um vetor de entrada com `minhacamada=vetor`, você poderia usar o `minhacamada=campo minhacamada` para adicionar um campo a partir dessa camada como parâmetro.
- `pasta`. Um pasta
- `arquivo`. Um arquivo
- `src`. Um Sistema de Referência de Coordenada

O nome do parâmetro é o nome que será mostrado para o usuário durante a execução do algoritmo, e também o nome da variável a ser usado no código de script. O valor digitado pelo usuário para esse parâmetro será atribuído a uma variável com esse nome.

Ao mostrar o nome do parâmetro para o nome, o usuário irá editado-lo para melhorar a sua aparência, substituindo hífen baixos com espaços. Assim, por exemplo, se você quiser que o usuário veja um parâmetro denominado `Um valor numérico`, você pode usar o nome da variável `A_numerical_value`.

Valores de tabelas e camadas são sequências contendo o caminho de arquivo do objeto correspondente. Para transformá-los em um objeto do QGIS, você pode usar a função `processing.getObjectFromUri()`. Várias entradas também têm um valor de texto, que contém os caminhos de arquivos para todos os objetos selecionados, separados por ponto e vírgula (;).

Saídas são definidas de forma semelhante, usando as seguintes etiquetas:

- `raster de saída`
- `vetor de saída`
- `tabela de saída`
- `html de saída`
- `arquivo de saída`
- `número de saída`
- `texto de saída`
- `extensão de saída`

O valor atribuído às variáveis de saída é sempre um texto com um caminho de arquivo. Ele vai corresponder a um caminho de arquivo temporário, caso o usuário não informe qualquer arquivo de saída.

Além das marcas para os parâmetros e saídas, você também pode definir o grupo em que o algoritmo será mostrado, usando a etiqueta `grupo`.

A última marca que você pode usar em seu cabeçalho roteiro é `##nomodeler`. Use isso quando você não quer que seu algoritmo seja mostrado na janela de modelador. Isto deve ser utilizado para algoritmos que não têm uma sintaxe clara (por exemplo, se o número de camadas a serem criadas não é conhecido antecipadamente, a tempo de desenho), o que os torna inadequados para a modelagem gráfica

17.8 Entregando os dados produzidos pelo algoritmo

Quando você declara uma saída que representa uma camada (raster, vetor ou tabela), o algoritmo irá tentar adicioná-la ao QGIS uma vez que ela for concluída. Essa é a razão pela qual, apesar de o método `runalg()` não carregar as camadas que produz, a camada final *TWI* será carregada, uma vez que será salva no arquivo digitado pelo usuário, que é o valor da saída correspondente.

Não use o método `load()` em seus algoritmos de script, mas apenas quando se trabalha com a linha de console. Se uma camada é criada como saída de um algoritmo, que deve ser declarada como tal. Caso contrário, você não será capaz de usar corretamente o algoritmo no modelador, desde a sua sintaxe (como definido pelas etiquetas explicadas acima) não coincidir com o que o algoritmo realmente cria.

Saídas ocultas (números e textos) não tem um valor. Em vez disso, é você quem tem que atribuir um valor a elas. Para isso, basta definir o valor de uma variável com o nome usado para declarar que a saída. Por exemplo, se você usou esta declaração,

```
##average=output number
```

a seguinte linha foi definido o valor de saída para 5:

```
average = 5
```

17.9 Comunicação com o usuário

Se o algoritmo leva um longo tempo para processar, é uma boa idéia informar ao usuário. Você tem um chamado progresso global disponível, com dois métodos disponíveis: `setText(texto)` e `setPercentage(porcento)` para modificar o progresso do texto e a barra de progresso.

Se você tiver que fornecer algumas informações para o usuário, não relacionada com a evolução do algoritmo, você pode usar o método `SetInfo(texto)`, também o progresso do objeto.

Se o seu script tiver algum problema, a maneira correta de propagação é para levantar uma exceção do tipo `GeoAlgorithmExecutionException()`. Você pode passar uma mensagem como argumento para o construtor da exceção. Processing vai cuidar de manuseá-lo e se comunicar com o usuário, dependendo de onde o algoritmo estiver sendo executado a partir da (caixa de ferramentas, modelador, console do Python ...)

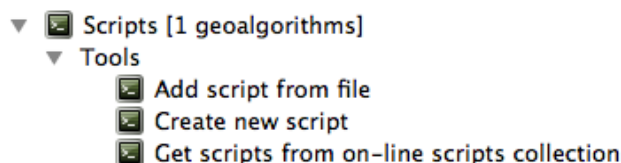
17.10 Documentando seus scripts

Como no caso dos modelos, você pode criar a documentação adicional para o seu roteiro, para explicar o que eles fazem e como usá-los. Na caixa de diálogo de edição de script que você vai encontrar o botão **[Editar roteiro ajuda]**. Clique nele e ele irá levá-lo para o diálogo ajuda a edição. Confira o capítulo sobre o modelador gráfico para saber mais sobre este diálogo e como usá-lo.

Arquivos de ajuda são salvos na mesma pasta que o próprio script, acrescentando a extensão `.help` ao nome do arquivo. Observe que você pode editar a ajuda de seu script antes de salvá-la pela primeira vez. Se mais tarde você fechar o script de diálogo edição sem salvar o script (ou seja, você descartá-lo), o conteúdo da ajuda que você escreveu será perdidos. Se o seu roteiro já foi salvo e está associado a um nome de arquivo, a gravação será feita automaticamente.

17.11 Exemplos de scripts

Vários exemplos estão disponíveis na coleção on-line de scripts, que podem ser acessados selecionando o script *Adquirir script da coleção on-line* a ferramenta sob o *Scripts/ferramentas* entrada na caixa de ferramentas.



Por favor, verifique-os para ver exemplos reais de como criar algoritmos usando as classes de estrutura de processamento. Você pode clicar o botão direito do mouse em qualquer algoritmo de roteiro e escolha: *guilabel: Editar script* para editar o seu código ou apenas para vê-lo.

17.12 Melhores práticas para algoritmos de script escrito

Aqui está um rápido resumo das idéias a serem consideradas ao criar seus algoritmos de script e, especialmente, se você quer compartilhar com outros usuários do QGIS. Seguindo estas regras simples irá assegurar a coerência entre os diferentes elementos de processamento, tais como a caixa de ferramentas, o modelador ou a interface de processamento em lote.

- Não coloque resultados das camadas. Vamos trabalhar o Processamento com seus resultados e carregar suas camadas se necessárias.
- Sempre declare as saídas de seu algoritmo criadas. Evite coisas como `decalring` uma saída e, em seguida, usando o nome do arquivo de destino definido para que a saída para criar uma coleção deles. Isso vai quebrar a semântica correta do algoritmo e tornar impossível usá-lo com segurança no modelador. Se você tem que escrever um algoritmo assim, certifique-se de adicionar a etiqueta `##nomodeler`.
- Não mostrar caixas de mensagens ou usar qualquer elemento GUI do script. Se você quer se comunicar com o usuário, use o método `SetInfo()` ou lançar uma `GeoAlgorithmExecutionException`.
- Como regra geral, não se esqueça que o seu algoritmo pode ser executado em um contexto que não seja a caixa de ferramentas de processamento.

17.13 Pré e pós-execução de ganchos de script

Os scripts também pode serem usados para definir ganchos pré e pós-execução que são executados antes e depois que algoritmo seja executado. Esta pode ser usada para automatizar as tarefas que devem ser executadas sempre que um algoritmo é executado.

A sintaxe é idêntica à sintaxe explicada acima, mas uma variável global adicional chamado `alg` está disponível, que representa o algoritmo que acaba de ser (ou está prestes a ser) executado.

No grupo *Geral* da janela de configuração de processamento, você vai encontrar duas entradas com o nome *Arquivo de script de pré-execução* e *Arquivo de script de pós-execução* onde o nome do arquivo dos scripts a serem executado em cada caso, podem ser inseridos.

17.14 Configurando as aplicações externas

The processing framework can be extended using additional applications. Currently, SAGA, GRASS, OTB (Orfeo Toolbox) and R are supported, along with some other command-line applications that provide spatial data analysis functionalities. Algorithms relying on an external application are managed by their own algorithm provider.

Este capítulo irá mostrar como configurar a infraestrutura do processamento para incluir estas aplicações adicionais, e irá explicar algumas características particulares baseado nos algoritmos. Uma vez feita a correta con-

figuração do sistema, terá possibilidade de executar algoritmos externos a partir de qualquer componente como a caixa de ferramentas ou o modelador gráfico, assim como faz para outro geolgoritmo.

Por padrão, todos os algoritmos que dependem de um aplicativo externo, ou seja, que não são disponibilizado com o QGIS não estão habilitados no programa. Você pode habilitá-los na janela de configurações do SEXTANTE. Antes de ativá-los, certifique-se de que o aplicativo correspondente já está instalado em seu sistema. A habilitação de um provedor de algoritmos sem ter o programa correspondente instalado no sistema permitirá que os algoritmos apareçam na caixa de ferramentas, mas um erro será exibido quando você tentar executá-los.

Isto porque os procedimentos característicos dos algoritmos (necessário para criar a janela de diálogo de parâmetros e dar as informações necessárias sobre o algoritmo) estão incluídos em cada aplicativo. Isto é, eles fazem parte do QGIS, então você os têm em sua instalação, mesmo que você não tenha instalado outro software qualquer. Entretanto, a execução do algoritmo, precisa dos códigos binários do software externo para ser instalado em seu sistema.

17.14.1 Uma nota para usuários Windows

Se você não for um usuário avançado e você estiver executando QGIS no Windows, você pode não estar interessado em ler o resto deste capítulo. Certifique-se de instalar QGIS em seu sistema usando o instalador independente. Isso irá instalar automaticamente SAGA, GRASS e OTB em seu sistema e configurá-los para que eles possam ser executados a partir QGIS. Todos os algoritmos da visão simplificada da caixa de ferramentas estarão prontos para serem executados sem a necessidade de qualquer configuração adicional. Se estiver instalando através da aplicação OSGeo4W, certifique-se que você selecionou SAGA para instalação e também o OTB.

Se desejar saber mais sobre como os fornecedores trabalham, ou deseja usar alguns algoritmos que não estão incluídos na caixa de ferramentas simplificada (como por exemplo os scripts R), continue a leitura.

17.14.2 Uma nota para os formatos dos arquivos

Ao usar um software externo, abrindo um arquivo no QGIS não significa que ele se comportará assim como no outro software. Na maioria dos casos, ele pode ler o que você abriu no QGIS, mas em alguns casos, isso pode dar incompatibilidade de tipo de arquivo. Ao usar bancos de dados ou formatos de arquivos incomuns, seja para camadas raster ou vetoriais, alguns problemas podem surgir. Se isso acontecer, tente usar formatos de arquivos conhecidos, que você tem certeza de que são aceitos por ambos os programas, e verifique a saída do console (na janela de histórico de registros) para saber mais sobre o que está acontecendo de errado.

Usando as camadas raster GRASS, por exemplo, um dos casos em que pode ter problema e não seja possível completar o seu trabalho é se chamar um algoritmo externo usando uma camada como arquivo de entrada. Por esta razão, essas camadas não irão aparecer como disponíveis para os algoritmos.

Você deve, no entanto, não encontrar problemas sobretudo com camadas vetoriais, pois o qgl converte automaticamente do formato de arquivo original para ser aceito pelo aplicativo externo antes de passar a camada para ele. Isso adiciona um tempo de processamento extra, que pode ser significativo se a camada tem um tamanho grande, por isso não se surpreenda se for preciso mais tempo para processar uma camada em uma conexão com banco de dados do que uma camada de tamanho similar armazenada num shapefile.

Provedores que não usa aplicações externas podem processar qualquer camada aberta no QGIS.

Quanto aos formatos de saída, todos os formatos suportados pelo QGIS pode ser usado, tanto para camadas raster e vetores. Alguns provedores não suportam determinados formatos, mas todos podem exportar para formatos camada raster em comuns que podem mais tarde ser transformados pelo qgl automaticamente. Como no caso de camadas de entrada, se for necessária a conversão, pode aumentar o tempo de processamento

Se a extensão do nome de arquivo especificado ao chamar um algoritmo não coincide com a extensão de qualquer um dos formatos suportados pelo qgl, então o sufixo será adicionado para definir um formato padrão. No caso de camadas raster, a: file:. Extensão *.tif* 'é usado, enquanto que: file:'.sh' é usado para camadas vetoriais.

17.14.3 Uma nota para as seleções da camada vetorial

Aplicações externas podem também estar cientes das seleções que existem em camadas vetoriais dentro do QGIS. No entanto, isso requer reescrever todas as camadas de entrada de vetores, como se fossem originalmente em um formato não suportado pelo aplicativo externo. Somente quando não houver seleção, ou o *Use somente recursos selecionados* a opção não está habilitado na configuração geral processamento, pode ser uma camada diretamente passada para um aplicativo externo.

Noutros casos, exportando apenas os elementos seleccionados é necessário, o que em alguns casos os tempos de execução serão mais longos.

SAGA

Algoritmos SAGA pode ser executado a partir de QGIS se você tiver SAGA instalado em seu sistema e configurar a estrutura de processamento adequada para que ele possa encontrar seus executáveis. Em particular, o executável de linha de comando SAGA é necessário para executar seus algoritmos.

Caso esteja utilizando o sistema Windows, os instaladores stand-alone ou o instalador OSGeo4W, ambos instalam o SAGA juntamente com o QGIS, e o caminho é automaticamente configurado, portanto não é necessário fazer mais nada.

Se você instalou o SAGA (lembre-se, você precisará da versão 2.1), o caminho para o executável SAGA deve ser configurado. Para fazer isso, abra a janela de configuração. No: guilabel: “SAGA” block, você vai encontrar um ambiente com o nome: guilabel: *SAGA Folder*.. Digite o caminho para a pasta onde está instalado SAGA. Feche a janela de configuração, e agora você está pronto para executar algoritmos de SAGA a partir do QGIS.

Se estiver utilizado o linux, os binários SAGA não estão incluídos no SEXTANTE, portanto precisa baixar e instalar o software. Por favor verifique o sítio na internet do SAGA para mais informações. É necessário o SAGA 2.1.

Neste caso não necessita de configurar o caminho para o executável SAGA, e não irá ver aquelas pastas. Em vez disso, deve garantir que o SAGA está instalado corretamente e a sua pasta adicionada à variável de ambiente PATH. Apenas abra o console e digite `saga_cmd` para verificar se o sistema encontra onde os binários SAGA estão localizados.

17.14.4 Sobre as limitações do sistema de grid do SAGA

A maior parte dos algoritmos SAGA que requerem várias camadas raster de entrada, exigem que elas tenham o mesmo sistema de projeção. Ou seja, para cobrir a mesma área geográfica e tenham uma mesma resolução espacial. Ao chamar algoritmos SAGA do QGIS, você pode usar qualquer camada, independentemente da sua resolução espacial e extensão, de forma individual. Quando múltiplas camadas raster são utilizados como entrada para um algoritmo SAGA, o QGIS irá adaptá-los a um sistema de coordenadas comum e, em seguida, transportará ao SAGA (ao menos que o algoritmo SAGA não possa operar com camadas de diferentes sistemas de projeção).

A definição do sistema de projeção comum é controlado pelo usuário, você vai encontrar vários parâmetros no grupo SAGA da janela de configuração para defini-lo. Existem duas maneiras de definir o sistema de projeção:

- Configure-o manualmente. Você define a extensão configurando os valores dos seguintes parâmetros:
 - *Reamostragem do X min*
 - *Reamostragem do X máx*
 - *Reamostragem do Y min*
 - *Reamostragem do Y máx*
 - *Reamostragem do tamanho da célula*

Tenha em atenção que o QGIS irá reamostrar as camadas de entrada para essa extensão, mesmo que não se sobreponham.

- Configurando automaticamente a partir das camadas de entrada. Para selecionar esta opção, verifique a opção: `guilabel:Use min covering grid system for resampling`. Todas as outras configurações irão ser ignoradas e a extensão mínima que cobre todas as camadas de entrada serão usadas. O tamanho de célula da camada de destino é o máximo de tamanho de célula de todas as camadas de entrada.

Para algoritmos que não usam camadas raster múltiplas, ou para aquelas que não necessitam de um único sistema de grid de entrada, não será feito uma reamostragem antes de chamar o SAGA, e esses parâmetros não serão usados.

17.14.5 Limitações para camadas multi-banda

Ao contrário do `lqg l`, o SAGA não tem suporte para multi-camadas da banda. Se você quiser usar uma camada multibanda (como um RGB ou imagem multiespectral), você primeiro tem que dividi-la em imagens individuais. Para fazer isso, você pode usar o ‘SAGA/Grid - Tools/Split RGB image’ algoritmo (que cria três imagens de uma imagem RGB) ou o ‘SAGA/Grid - Tools/Extract band’ algoritmo (para extrair uma única faixa).

17.14.6 Limitações na resolução espacial

O SAGA pressupõe que as camadas raster têm o mesmo tamanho de célula no eixo X e Y. Se estiver trabalhando com uma camada com diferentes valores para o tamanho de célula horizontal e vertical, deverá obter resultados inesperados. Nesse caso, um aviso será adicionado ao registro do processamento, indicando que a camada de entrada não se adapta de forma a ser processado pelo SAGA.

17.14.7 Registrando

Quando o QGIS chama o SAGA, ele faz isso usando a interface de linha de comando, passando assim um conjunto de comandos para executar toda a operação requerida. O SAGA mostra seu progresso, escrevendo informações para o console, que inclui a porcentagem de processamento já realizado, juntamente com o conteúdo adicional. Esta saída é filtrada e usada para atualizar a barra de progresso durante a execução do algoritmo.

Ambos os comandos enviados pelo QGIS e as informações adicionais impressas pelo SAGA pode ser registrada juntamente com outras mensagens de registro de processamento, e você pode encontrá-los úteis para acompanhar em detalhes o que está acontecendo quando o `lqg l` executa um algoritmo SAGA. Você vai encontrar dois cenários, a saber: *Log console output* e *Log execution commands*, para ativar esse mecanismo de registro.

A maioria dos outros fornecedores que usam uma aplicação externa e chamam a partir da linha de comandos têm opções semelhantes, portanto irá encontrar noutros sítios da lista de configurações do processamento.

R. Creating R scripts

A integração da linguagem R com o QGIS é diferente do SAGA em que não há um conjunto de algoritmos predefinido que pode rodar (com exceção de alguns exemplos). Em vez disso, você deve escrever seus scripts e chamar os comandos R, e de uma forma muito semelhante ao que vimos no capítulo dedicado ao processamento de scripts. Este capítulo mostra a sintaxe para usar e para chamar os comandos de R no SEXTANTE além de como usar objetos SEXTANTE (camadas, tabelas) nestes scripts.

A primeira coisa que você tem que fazer, como vimos no caso da SAGA, é dizer ao QGIS onde seus binários R estão localizados. Você pode fazer isso usando o: `guilabel:R folder` entrando na janela de configuração de processamento. Depois de definir esse parâmetro, você pode começar a criar e executar seus próprios scripts R.

Uma vez mais, isto é diferente no Linux, e só tem de ter certeza que a pasta do R está incluída na variável de ambiente PATH. Se conseguir iniciar o R, apenas introduza R na consola, e estará pronto a começar.

Para adicionar um novo algoritmo que usa uma função R (ou um script R mais complexo que você desenvolveu e que você gostaria de ter disponível a partir do QGIS), você tem que criar um arquivo de script que informa a estrutura de processamento e como proceder com essa operação e inserir os comandos correspondentes em R.

Os arquivos de script R têm a extensão `.rsx` e é fácil criá-los se tiver o conhecimento básico da sintaxe do R e do código R. Eles devem ficar armazenados na pasta `scripts R`. Pode definir esta pasta no grupo de configurações `R` (disponível a partir do diálogo de configurações do processamento), como faz com a pasta para scripts normais do processamento.

Vamos dar uma olhada em um arquivo muito simples de script, que chama o método R `spsample` para criar uma rede aleatória dentro do limite dos polígonos em uma dada camada de polígono. Este método pertencem ao pacote `maptools`. Uma vez que quase todos os algoritmos que você gostaria de incorporar ao `qgis` vai usar ou gerar dados espaciais, conhecimento de pacotes espaciais como `maptools` e, especialmente, `sp`, é obrigatório.

```
##polyg=vector
##numpoints=number 10
##output=output vector
##sp=group
pts=spsample(polyg,numpoints,type="random")
output=SpatialPointsDataFrame(pts, as.data.frame(pts))
```

As primeiras linhas, que começam com um comentário sinal duplo Python (`##`), dizem entradas do algoritmo QGIS descrito no arquivo e as saídas que irá gerar. Eles trabalham com exatamente a mesma sintaxe que os scripts de Sextante que já vimos, para que eles não sejam descritos aqui novamente.

Quando você declara um parâmetro de entrada, o QGIS usa essa informação para duas coisas: criar a interface do usuário para pedi-lo o valor desse parâmetro e criando uma variável de R correspondente, que pode mais tarde ser usado como entrada para os comandos R.

No exemplo acima, estamos declarando uma entrada do tipo `vetorial` chamado `polyg`. Ao executar o algoritmo, o QGIS vai abrir em R a camada selecionada pelo usuário e armazená-la em uma variável também chamado `polyg`. Assim, o nome de um parâmetro também é o nome da variável que pode usar em R para concordar com o valor daquele parâmetro (portanto, você deve evitar o uso de palavras reservadas ao R como nomes de parâmetro).

Elementos espaciais como camadas vetoriais e raster são lidos usando os comandos `readOGR()` e `brick()` (você não tem que se preocupar com a adição desses comandos para o seu arquivo de descrição pois o QGIS fará), e eles são armazenados como objetos `Spatial*DataFrame`. Campos da tabela são armazenados como strings contendo o nome do campo selecionado.

Tabelas são abertas usando o comando `read.csv()`. Se uma tabela inserida pelo usuário não está no formato CSV, será convertida antes de importá-la para R.

Adicionalmente, os arquivos raster pode ser lidos usando o comando `readGDAL()` em vez de `brick()`, usando o `##usereadgdal`.

Se você é um usuário avançado e não quer o `qgis` para criar o objeto que representa a camada, você pode usar o `## passfilename` tag para indicar que você prefere um texto com o nome do arquivo em seu lugar. Neste caso, cabe a você abrir o arquivo antes de realizar qualquer operação com os dados que ele contém.

Com a informação em cima, podemos agora perceber a primeira linha do nosso primeiro script exemplo (a primeira linha que não começa com o comentário Python).

```
pts=spsample(polyg,numpoints,type="random")
```

A variável `polyg` já contém o objecto `SpatialPolygonsDataFrame`, portanto pode ser usado para chamar o método `spsample`, tal como o `numpoints`, que indica o número de pontos a ser adicionados ao grid de amostra criada.

Desde que declarado uma saída do tipo vetor chamado `out`, temos que criar uma variável chamada `out` e armazenar um `Spatial*DataFrame` objeto nele (neste caso, um `SpatialPointsDataFrame`). Você pode usar qualquer nome para as variáveis intermediárias. Apenas certifique-se de que a variável que armazena o resultado final tem o mesmo nome que você usou para declará-la, e que contém um valor adequado.

Neste caso, o resultado obtido a partir do método `spsample` tem que ser convertido diretamente no `SpatialPointsDataFrame` objeto, uma vez que é por si só um objeto de classe `ppp`, o que não é uma classe adequada para ser devolvida ao `qgis`.

Se o seu algoritmo gera camadas raster, a maneira como eles são salvos vai depender se você tem usado ou não usou a opção `##dontuserasterpackag`. O que você tem usado, as camadas são salvas usando o método `writeGDAL()`. Se não, o `writeRaster()` método do pacote de `raster` será usado.

Se usou a opção `#passfilename`, os arquivos de saída são gerados usando o pacote `raster` (com `writeRaster()`), mesmo que não seja usado nos arquivos de entrada.

Se o seu algoritmo não gera qualquer camada, mas sim um resultado de texto no console em vez disso, você tem que indicar que deseja que o console seja exibido uma vez que a execução seja concluída. Para isso, basta começar as linhas de comando que produzem os resultados que você deseja imprimir com a `>` ('greater') sign. A saída de todas as outras linhas não serão mostradas. Por exemplo, aqui está o arquivo de descrição de um algoritmo que realiza um teste de normalidade em um determinado campo (coluna) dos atributos de uma camada vetorial:

```
##layer=vector
##field=field layer
##nortest=group
library(nortest)
>lillie.test(layer[[field]])
```

A saída da última linha é impressa, mas a saída da primeira não é (e nem são as saídas de outras linhas de comando adicionadas automaticamente pelo QGIS).

Se o seu algoritmo criar qualquer tipo de gráficos (usando o método `plot()`), adicione a seguinte linha:

```
##showplots
```

Isto fará com que o QGIS redirecione todas as saídas gráficas R para um ficheiro temporário, que poderá ser mais tarde aberta quando a execução do R é finalizada.

Tanto os gráficos como os resultados do console serão exibidos no gestor de resultados do processamento.

Para mais informações, consulte os arquivos de script fornecidos com o SEXTANTE. A maioria deles são bastante simples e vai lhe ajudar muito a entender como criar as seus próprios scripts.

Nota: Bibliotecas `rgdal` and `maptools` são carregados por padrão, então você não tem que adicionar os comandos correspondentes `library()` (você só tem que ter certeza de que esses dois pacotes são instalados no seu R distribuição). No entanto, outras bibliotecas adicionais que você pode precisar ter de ser explicitamente carregado. Basta adicionar os comandos necessários no início do seu script. Você também tem que ter certeza de que os pacotes correspondentes são instalados na distribuição usada por R QGIS. A estrutura de processamento não vai cuidar de qualquer instalação do pacote. Se você executar um script que requer um pacote que não está instalado a execução irá falhar e o SEXTANTE irá tentar detectar quais pacotes estão faltando. Você deve instalar as bibliotecas que faltam manualmente antes de executar o algoritmo.

GRASS

Configurar o GRASS não é muito diferente de configurar o SAGA. Primeiro, o caminho para a pasta GRASS tem de ser definida, mas apenas se estiver a correr o Windows. Adicionalmente, um interpretador shell, (normalmente o `msys.exe`, que pode ser encontrado na maioria das distribuições do GRASS para o Windows) tem de ser definido e o seu caminho configurado.

Por padrão, a estrutura de processamento tenta configurar seu conector GRASS para usar a distribuição que conecta junto com o `lqg`. Isso deve funcionar sem problemas na maioria dos sistemas, mas se você tiver problemas, você pode ter que configurar o conector GRASS manualmente. Além disso, se você quiser usar uma instalação GRASS diferente, você pode alterar essa configuração e aponte para a pasta onde a outra versão está instalada. GRASS 6.4 é necessário para os algoritmos funcionarem corretamente.

Se está a trabalhar em Linux, só necessita de ter a certeza que o GRASS está corretamente instalado, e pode ser executado sem problema a partir do console.

Algoritmos GRASS usar uma região para os cálculos. Esta região pode ser definida manualmente usando valores semelhantes aos encontrados na configuração `saga`, ou automaticamente, tendo a extensão mínima que abrange

todas as camadas de entrada usadas para executar o algoritmo de cada vez. Se esta abordagem é o comportamento que você preferir, basta verificar o *Use min covering region* nos parâmetros de configuração GRASS.

O último parâmetro que tem que ser configurado está relacionada com o mapset. O mapset é necessário para executar o GRASS, e a estrutura de processamento cria um mapset temporário para cada execução. Você tem que informar ao sistema se os dados que você está trabalhando são geográficos (latitude/longitude) ou coordenadas projetadas.

GDAL

Não é necessário configurações adicionais para correr os algoritmos GDAL, uma vez que já está incorporado no QGIS e os algoritmos podem inferir na sua configuração a partir dele.



Orfeo Toolbox

Algoritmos do Orfeo Toolbox (OTB) pode ser executado a partir de QGIS se este sistema estiver instalado em seu sistema e configurou QGIS corretamente, para que ele possa encontrar todos os arquivos necessários (ferramentas de linha de comando e bibliotecas).

Se estiver utilizado o Linux, os binários SAGA não estarão incluídos no QGIS, portanto precisa de transferir e instalar o software. Por favor verifique o sítio na internet do OTB para mais informação sendo necessário o SAGA 2.1.

Uma vez instalado o OTB, inicie o QGIS, abra a janela de configuração do processamento e configure o provedor de algoritmo OTB. No :guilabel: 'Orfeo Toolbox (image analysis)'bloco, você vai encontrar todas as definições relacionadas com a OTB. Em primeiro lugar, garanta que os algoritmos estão habilitados.

Em seguida, configure o caminho para a pasta onde estão as ferramentas da linha de comandos OTB e as bibliotecas estão instaladas:

-  normalmente *OTB applications folder* encaminha para `/usr/lib/otb/applications` e *OTB command line tools folder* é `/usr/bin`
-  | Se você usar o instalador OSGeo4W, instale o pacote `otb-bin` e entre `C:\OSGeo4W\apps\orfeotoolbox\applications` como *OTB applications folder* e `C:\OSGeo4W\bin` as *OTB command line tools folder*. Estes valores devem ser configurados por padrão, mas se você tiver uma instalação OTB diferente, configure-os para os valores correspondentes no seu sistema.

TauDEM

Para usar este provedor necessita de instalar as ferramentas da linha de comandos do TauDEM.

17.14.8 Windows

Por favor visite o [sítio na internet do TauDEM](#) para as instruções de instalação e os binários dos sistemas para 32bit e 64bit precompilados. **IMPORTANTE:** necessita os executáveis do TauDEM 5.0.6. A versão 5.2 não é suportada atualmente.

17.14.9 Linux

Não existem pacotes para a distribuição Linux, então você deve compilar o TauDEM manualmente. Como o TauDEM usa MPICH2, tem que instalá-lo primeiro usando algum gerenciador de pacotes de aplicativos. Como alternativa, o TauDEM trabalha bem com o Open MPI, assim é possível usá-lo no lugar do MPICH2.

Baixe o **'código-fonte do TauDEM 5.0.6'** <http://hydrology.usu.edu/taudem/taudem5.0/TauDEM5PCsrc_506.zip> e extraia os arquivos numa pasta.

Abra o arquivo `linearpart.h` e depois a linha

```
#include "mpi.h"
```

adicione uma nova linha com

```
#include <stdint.h>
```

e irá obter

```
#include "mpi.h"  
#include <stdint.h>
```

Salve as alterações e feche o arquivo. Agora abra o arquivo `tiffIO.h`, encontre a linha `#include "stdint.h"` e substitua as entre aspas ("") com `<>`, para que possa obter

```
#include <stdint.h>
```

Salve as alterações e feche o arquivo. Crie um diretório de compilação e `cd` para ele

```
mkdir build  
cd build
```

Configure a sua compilação com o comando

```
CXX=mpicxx cmake -DCMAKE_INSTALL_PREFIX=/usr/local ..
```

e de seguida compile

```
make
```

Finalmente, instale o TauDEM em `/usr/local/bin` e execute

```
sudo make install
```

17.15 O QGIS Comando

A transformação inclui uma ferramenta prática que permite a execução de algoritmos sem ter que usar a caixa de ferramentas, mas apenas digitando o nome do algoritmo que você deseja executar.

Esta ferramenta é conhecida como o *comando QGIS*, e é apenas uma caixa de texto simples para autocompletar onde você digita o comando que você deseja executar.

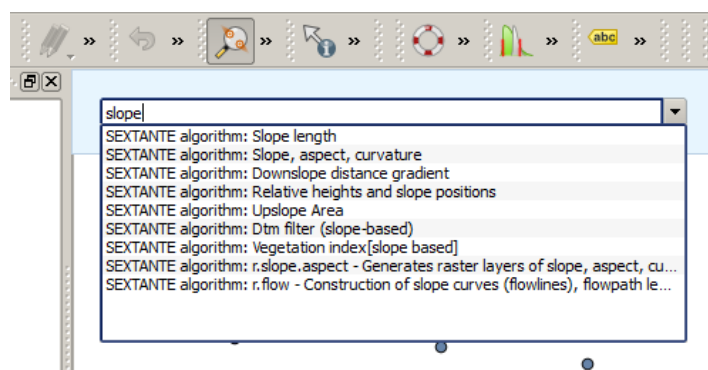


Figura 17.29: O QGIS Comando

O Comando é iniciado a partir do menu *Análise* ou, da forma mais prática, pressionando `Shift + Ctrl + M` (você pode mudar esse atalho de teclado padrão nas configurações QGIS, se você preferir um diferente). Para além da execução de algoritmos de processamento, o Comando lhe dá acesso a maioria das funcionalidades no

QGIS, o que significa que ele dá-lhe uma maneira prática e eficiente de executar tarefas QGIS e permite que você controle QGIS com uso reduzido de botões e menus.

Além disso, o comando é configurável, assim você pode adicionar seus comandos personalizados e tê-los apenas a algumas teclas de distância, tornando-se uma ferramenta poderosa para ajudá-lo a se tornar mais produtivo no seu trabalho diário com QGIS.

17.15.1 Comandos disponíveis

Os comandos disponíveis no Comando entram nas seguintes categorias:

- Algoritmos de processamento. Estes são mostrados como “ algoritmo de processamento: <name of the algorithm> “. .
- Ítens do Menu. Estes são mostrados como item de menu:<menu texto entrada>. Todos os itens de menus disponíveis a partir da interface do QGIS estão disponíveis, mesmo que eles estejam incluídos em um submenu.
- Funções Python. Você pode criar funções Python curtas que serão depois incluídas na lista de comandos disponíveis. Elas são mostradas como Função: <nome da função>.

Para executar qualquer um dos acima, basta começar a digitar e, em seguida, selecione o elemento correspondente da lista de comandos disponíveis que aparece depois de filtrar toda a lista de comandos com o texto que você digitou.

No caso de chamar uma função Python, pode selecionar uma entrada da lista, que tem o prefixo de Função: (para a instância, Função: removeall), ou apenas escreva diretamente o nome da função (``removeall no exemplo anterior). Não existe necessidade de adicionar parêntesis após o nome da função.

17.15.2 Criando funções personalizadas

As funções personalizadas são adicionadas ao introduzir o código Python correspondente no arquivo `commands.py` que pode ser encontrado `.qgis/sextante/commander directory` na pasta do usuário. É apenas um arquivo Python simples onde pode adicionar as funções que necessita.

O arquivo é criado com alguns exemplo funciona a primeira vez que você abrir o Comandante. Se você ainda não lançou o comandante, no entanto, você pode criar o arquivo você mesmo. Para editar o arquivo de comandos, utilize o seu editor de texto favorito. Você também pode usar um built-in editor chamando o comando edição do Comando. Ele vai abrir o editor com o arquivo de comandos, e você pode editá-lo diretamente e, em seguida, salvar as alterações.

Por exemplo, pode adicionar a seguinte função, que remove todas as camadas:

```
from qgis.gui import *

def removeall():
    mapreg = QgsMapLayerRegistry.instance()
    mapreg.removeAllMapLayers()
```

Depois de adicionar a função, ele estará disponível no Comando, e você pode chamá-lo digitando `removeall`. Não há necessidade de fazer nada além de escrever a própria função.

As funções podem receber parâmetros. Adicionar `*args` para a sua definição de função para receber argumentos. Ao chamar a função do Comando, os parâmetros têm de ser passados separados por espaços.

Aqui está um exemplo de uma função que carrega uma camada e que tome como parâmetro o nome do arquivo da camada para carregar.

```
import processing

def load(*args):
    processing.load(args[0])
```

Se você quiser carregar a camada em `/home/myuser/points.shp`, tipo carregue `/home/myuser/points.shp` na caixa de texto comando.

.

Fornece algoritmos e processamento

18.1 Provedor de algoritmo GDAL

GDAL (Geospatial Data Abstraction Library) é uma biblioteca de tradução para formatos de dados geoespaciais raster e vetor.

18.1.1 Análises GDAL

Aspecto

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de Entrada [raster] <coloque aqui a descrição do parâmetro>

Número da Banda [number] <coloque aqui a descrição do parâmetro>

Padrão: *1*

Computar bordas [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Usar fórmula de Zevenbergen&Thorne (e não a de Horn) [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Retornar ângulo trigonométrico (ao invés de azimute) [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Retornar 0 para 'flat' (ao invés de -9999) [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Saídas

Arquivo de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:aspect', input, band, compute_edges, zevenbergen, trig_angle, zero_flat)
```

Veja também

Cor do relevo

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de Entrada [raster] <coloque aqui a descrição do parâmetro>

Número da Banda [number] <coloque aqui a descrição do parâmetro>

Padrão: *1*

Computar bordas [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Arquivo de Configuração de Cores [file] <coloque aqui a descrição do parâmetro>

Modo de Correspondência [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — “0,0,0,0” RGBA
- 1 — Cor exata
- 2 — Cor próxima

Padrão: *0*

Saídas

Arquivo de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:colorrelief', input, band, compute_edges, color_table, match_mode, out)
```

Veja também

Preencher sem dados

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de Entrada [raster] <coloque aqui a descrição do parâmetro>

Distância de busca [number] <coloque aqui a descrição do parâmetro>

Padrão: *100*

Suavizar interações [number] <coloque aqui a descrição do parâmetro>

Padrão: *0*

Operar na Banda [number] <coloque aqui a descrição do parâmetro>

Padrão: *1*

Máscara de validação [raster] Opcional.

<coloque aqui a descrição do parâmetro>

Não usar máscara de validação padrão [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Saídas

Camada de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:fillnodata', input, distance, iterations, band, mask, no_default_mask,
```

Veja também

Grid (Média móvel)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [vector: point] <coloque aqui a descrição do parâmetro>

Campo Z [tablefield: numeric] Opcional.

<coloque aqui a descrição do parâmetro>

Ráio 1 [number] <coloque aqui a descrição do parâmetro>

Padrão: *0.0*

Ráio 2 [number] <coloque aqui a descrição do parâmetro>

Padrão: *0.0*

Mínimo de Pontos [number] <coloque aqui a descrição do parâmetro>

Padrão: *0.0*

Ângulo [number] <coloque aqui a descrição do parâmetro>

Padrão: *0.0*

Sem dados [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Tipo da saída raster [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Padrão: 5

Saídas

Arquivo de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:gridaverage', input, z_field, radius_1, radius_2, min_points, angle, n
```

Veja também

Grid (Dados métricos)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [vector: point] <coloque aqui a descrição do parâmetro>

Campo Z [tablefield: numeric] Opcional.

<coloque aqui a descrição do parâmetro>

Métricas [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — Mínimo
- 1 — Máximo
- 2 — Intervalo

- 3 — Contagem
- 4 — Média da distância
- 5 — Média da distância entre pontos

Padrão: 0

Ráio 1 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ráio 2 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Mínimo de Pontos [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ângulo [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Sem dados [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Tipo do raster de saída [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Padrão: 5

Saídas

Arquivo de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:griddatametrics', input, z_field, metric, radius_1, radius_2, min_point
```

Veja também

Grid (Inverso da Distância à Potência)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [vector: point] <coloque aqui a descrição do parâmetro>

Campo Z [tablefield: numeric] Opcional.

<coloque aqui a descrição do parâmetro>

Potência [number] <coloque aqui a descrição do parâmetro>

Padrão: 2.0

Suavização [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ráio 1 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ráio 2 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Máximos de Pontos [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Mínimo de Pontos [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ângulo [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Sem dados [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Tipo da saída raster [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32

- 10 — CFloat64

Padrão: 5

Saídas

Arquivo de saída [raster] <coloque aqui a descrição da saída>

Usando o console

```
processing.runalg('gdalogr:gridinvdist', input, z_field, power, smothing, radius_1, radius_2, max.
```

Veja tambem

Grid (Vizinho mais próximo)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [vector: point] <coloque aqui a descrição do parâmetro>

Campo Z [tablefield: numeric] Opcional.

<coloque aqui a descrição do parâmetro>

Ráio 1 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ráio 2 [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Ângulo [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Sem dados [number] <coloque aqui a descrição do parâmetro>

Padrão: 0.0

Tipo da saída raster [selection] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32

- 9 — CFloat32
- 10 — CFloat64

Padrão: 5

Saídas

Arquivo de saída [raster] <coloque aqui a descrição da saída>

Usando o console

```
processing.runalg('gdalogr:gridnearestneighbor', input, z_field, radius_1, radius_2, angle, nodat
```

Veja tambem

Sombreamento

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de Entrada [raster] <coloque aqui a descrição do parâmetro>

Número da Banda [number] <coloque aqui a descrição do parâmetro>

Padrão: 1

Computar bordas [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Usar fórmula de Zevenbergen&Thorne (e não a de Horn) [boolean] <coloque aqui a descrição do parâmetro>

Padrão: *False*

Fator Z (exagero vertical) [number] <coloque aqui a descrição do parâmetro>

Padrão: 1.0

Escala (razão entre un. vert. e horiz.) [number] <coloque aqui a descrição do parâmetro>

Padrão: 1.0

Azimute de insolação [number] <coloque aqui a descrição do parâmetro>

Padrão: 315.0

Altitude de insolação [number] <coloque aqui a descrição do parâmetro>

Padrão: 45.0

Saídas

Arquivo de saída [raster] <coloque aqui a descrição da saída>

Usando o console

```
processing.runalg('gdalogr:hillshade', input, band, compute_edges, zevenbergen, z_factor, scale, a
```

Veja também

Preto próximo

Descrição

<coloque a descrição do algoritmo aqui>

Parâmetros

Camada de entrada [raster] <coloque a descrição do parâmetro aqui>

Como longe de preto (branco) [número] <coloque a descrição do parâmetro aqui>

Padrão: 15

Procure pixels quase branco em vez de quase preto [booleano] <coloque a descrição do parâmetro aqui>

Padrão: *Falso*

Saídas

Camada de saída [raster] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('gdalogr:nearblack', input, near, white, output)
```

ver igual

Proximidade (distância raster)

Descrição

<coloque a descrição do algoritmo aqui>

Parâmetros

Camada de entrada [raster] <coloque a aqui a descrição do parâmetro>

Valores [texto] <coloque a aqui a descrição do parâmetro>

Padrão: *(não definido)*

Unidades de distância [seleção] <coloque a aqui a descrição do parâmetro>

Opções:

- 0 — GEO
- 1 — PIXEL

Padrão: 0

Dist Máx (ignorar valores negativos) [número] <coloque a aqui a descrição do parâmetro>

Padrão: -1

sem dado (ignorar valores negativos) [número] <coloque a aqui a descrição do parâmetro>

Padrão: -1

Val buf fixado (ignora valor negativo) [número] <coloque a aqui a descrição do parâmetro>

Padrão: -1

tipo raster de saída [seleção] <coloque a aqui a descrição do parâmetro>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Padrão: 5

Saídas

Camada de saída [raster] <coloque aqui a descrição da saída>

Uso do console

```
processing.runalg('gdalogr:proximity', input, values, units, max_dist, nodata, buf_val, rtype, out)
```

Veja mais

Rugosidade

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

número de banda [número] <coloque aqui a descrição do parâmetros>

Padrão: 1

Computar bordas [booleano] <coloque aqui a descrição do parâmetro>

Padrão: *Falso*

Saídas

arquivo de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:roughness', input, band, compute_edges, output)
```

Veja também

Crivo

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetro>

Limiar [número] <coloque aqui a descrição do parâmetro>

Padrão: *2*

Conexão com pixel [seleção] <coloque aqui a descrição do parâmetro>

Opções:

- 0 — 4
- 1 — 8

Padrão: *0*

Saídas

Camadas de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:sieve', input, threshold, connections, output)
```

Veja também

Declividade

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetro>

Número de banda [número] <coloque aqui a descrição do parâmetro>

Padrão: *1*

Computar bordas [booleano] <coloque aqui a descrição do parâmetro>

Padrão: *Falso*

Use fórmula Zevenbergen&Thorne (em vez de aspas) [booleano] <coloque aqui a descrição do parâmetro>

Padrão: *Falso*

Declividade expressa em percentual (em vez de graus) [booleano] <coloque aqui a descrição do parâmetro>

Padrão: *Falso*

Escala (raio de unidades vert. para horiz.) [número] <coloque aqui a descrição do parâmetro>

Padrão: *1.0*

Saídas

arquivo de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:slope', input, band, compute_edges, zevenbergen, as_percent, scale, out)
```

Veja também

IPT (Índice de Posição Topográfica)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

número de bandas [número] <coloque aqui a descrição do parâmetros>

Padrão: *1*

Computar bordas [booleano] <coloque aqui a descrição do parâmetros>

Padrão: *Falso*

Saídas

arquivo de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:tpitopographicpositionindex', input, band, compute_edges, output)
```

Veja também

IRT (Índice de Rugosidade do Terreno)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

número de bandas [número] <coloque aqui a descrição do parâmetros>

Padrão: *1*

Computar bordas [booleano] <coloque aqui a descrição do parâmetros>

Padrão: *Falso*

Saídas

arquivo de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:triterrainruggednessindex', input, band, compute_edges, output)
```

Veja também

.

18.1.2 Conversão GDAL

gdal2xyz

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

número de banda [número] <coloque aqui a descrição do parâmetros>

Padrão: *1*

Saídas

arquivo de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:gdal2xyz', input, band, output)
```

Veja também

PCT para RGB

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

número de banda [número] <coloque aqui a descrição do parâmetros>

Opções:

- 0 — 1
- 1 — 2
- 2 — 3
- 3 — 4
- 4 — 5
- 5 — 6
- 6 — 7
- 7 — 8
- 8 — 9
- 9 — 10
- 10 — 11
- 11 — 12
- 12 — 13
- 13 — 14
- 14 — 15
- 15 — 16
- 16 — 17
- 17 — 18
- 18 — 19
- 19 — 20
- 20 — 21

- 21 — 22
- 22 — 23
- 23 — 24
- 24 — 25

Padrão: 0

Saídas

Camada de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:pcttorgb', input, nband, output)
```

Veja também

Poligonizar (raster para vetor)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

Nome do campo de saída [texto] <coloque aqui a descrição do parâmetros>

Padrão: *DN*

Saídas

Camada de saída [vetor] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:polygonize', input, field, output)
```

Veja também

Rasterizar (vetor para raster)

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [vetor: qualquer] <coloque aqui a descrição do parâmetros>

Tabela de atributo [campo da tabela: qualquer] <coloque aqui a descrição do parâmetros>

Adicione valores dentro de uma camada raster existente (*) [booleano] <coloque aqui a descrição do parâmetros>

Padrão: *Falso*

Defina o tamanho do raster de saída (ignorada se a opção acima for verificada) [seleção] <coloque aqui a descrição do parâmetros>

Opções:

- 0 — tamanho de saída em pixels
- 1 — Resolução de saída em unidades do mapa por pixel

Padrão: *1*

Horizontal [número] <coloque aqui a descrição do parâmetros>

Padrão: *100.0*

Vertical [número] <coloque aqui a descrição do parâmetros>

Padrão: *100.0*

Tipo do Raster [seleção] <coloque aqui a descrição do parâmetros>

Opções:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Padrão: *0*

Saídas

Camada de saída: obrigatório escolher uma camada existente, se a opção (*) for selecionada <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:rasterize', input, field, writeover, dimensions, width, height, rtype,
```


Veja também

RGB para PCT

Descrição

<coloque aqui a descrição do algoritmo>

Parâmetros

Camada de entrada [raster] <coloque aqui a descrição do parâmetros>

Número de cores [número] <coloque aqui a descrição do parâmetros>

Padrão: 2

Saídas

Camada de saída [raster] <coloque aqui a descrição da saída>

Usar console

```
processing.runalg('gdalogr:rgbttopct', input, ncolors, output)
```

Veja também

Translate (convert format)

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Set the size of the output file (In pixels or %) [number] <put parameter description here>

Default: *100*

Output size is a percentage of input size [boolean] <put parameter description here>

Default: *True*

Nodata value, leave as none to take the nodata value from input [string] <put parameter description here>

Default: *none*

Expand [selection] <put parameter description here>

Options:

- 0 — none
- 1 — gray
- 2 — rgb

- 3 — rgba

Default: 0

Output projection for output file [leave blank to use input projection] [crs]
<put parameter description here>

Default: *None*

Subset based on georeferenced coordinates [extent] <put parameter description here>

Default: *0,1,0,1*

Copy all subdatasets of this file to individual output files [boolean] <put parameter description here>

Default: *False*

Additional creation parameters [string] Optional.

<put parameter description here>

Default: *(not set)*

Output raster type [selection] <put parameter description here>

Options:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Default: 5

Outputs

Output layer [raster] <put output description here>

Console usage

```
processing.runalg('gdalogr:translate', input, outsize, outsize_perc, no_data, expand, srs, projwin)
```

See also

.

18.1.3 Extração GDAL

Clip raster by extent

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Nodata value, leave as none to take the nodata value from input [string] <put parameter description here>

Default: *none*

Clipping extent [extent] <put parameter description here>

Default: *0,1,0,1*

Additional creation parameters [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output layer [raster] <put output description here>

Console usage

```
processing.runalg('gdalogr:cliprasterbyextent', input, no_data, projwin, extra, output)
```

See also

Clip raster by mask layer

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Mask layer [vector: polygon] <put parameter description here>

Nodata value, leave as none to take the nodata value from input [string] <put parameter description here>

Default: *none*

Create and output alpha band [boolean] <put parameter description here>

Default: *False*

Keep resolution of output raster [boolean] <put parameter description here>

Default: *False*

Additional creation parameters [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output layer [raster] <put output description here>

Console usage

```
processing.runalg('gdalogr:cliprasterbymasklayer', input, mask, no_data, alpha_band, keep_resolut
```

See also

Contour

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Interval between contour lines [number] <put parameter description here>

Default: *10.0*

Attribute name (if not set, no elevation attribute is attached) [string]

Optional.

<put parameter description here>

Default: *ELEV*

Additional creation parameters [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output file for contour lines (vector) [vector] <put output description here>

Console usage

```
processing.runalg('gdalogr:contour', input_raster, interval, field_name, extra, output_vector)
```

See also

.

18.1.4 Miscelaneos GDAL

Build Virtual Raster

Description

<put algorithm description here>

Parameters

Input layers [**multipleinput: rasters**] <put parameter description here>

Resolution [**selection**] <put parameter description here>

Options:

- 0 — average
- 1 — highest
- 2 — lowest

Default: *0*

Layer stack [**boolean**] <put parameter description here>

Default: *True*

Allow projection difference [**boolean**] <put parameter description here>

Default: *False*

Outputs

Output layer [**raster**] <put output description here>

Console usage

```
processing.runalg('gdalogr:buildvirtualraster', input, resolution, separate, proj_difference, out)
```

See also

Merge

Description

<put algorithm description here>

Parameters

Input layers [**multipleinput: rasters**] <put parameter description here>

Grab pseudocolor table from first layer [**boolean**] <put parameter description here>

Default: *False*

Layer stack [**boolean**] <put parameter description here>

Default: *False*

Output raster type [**selection**] <put parameter description here>

Options:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Default: 5

Outputs

Output layer [**raster**] <put output description here>

Console usage

```
processing.runalg('gdalogr:merge', input, pct, separate, rtype, output)
```

See also

Build overviews (pyramids)

Description

<put algorithm description here>

Parameters

Input layer [**raster**] <put parameter description here>

Overview levels [**string**] <put parameter description here>

Default: 2 4 8 16

Remove all existing overviews [boolean] <put parameter description here>

Default: *False*

Resampling method [selection] <put parameter description here>

Options:

- 0 — nearest
- 1 — average
- 2 — gauss
- 3 — cubic
- 4 — average_mp
- 5 — average_magphase
- 6 — mode

Default: *0*

Overview format [selection] <put parameter description here>

Options:

- 0 — Internal (if possible)
- 1 — External (GTiff .ovr)
- 2 — External (ERDAS Imagine .aux)

Default: *0*

Outputs

Output layer [raster] <put output description here>

Console usage

```
processing.runalg('gdalogr:overviews', input, levels, clean, resampling_method, format)
```

See also

Information

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Suppress GCP info [boolean] <put parameter description here>

Default: *False*

Suppress metadata info [boolean] <put parameter description here>

Default: *False*

Outputs

Layer information [html] <put output description here>

Console usage

```
processing.runalg('gdalorg:rasterinfo', input, nogcp, nometadata, output)
```

See also

.

18.1.5 Projeções GDAL

Extract projection

Description

<put algorithm description here>

Parameters

Input file [raster] <put parameter description here>

Create also .prj file [boolean] <put parameter description here>

Default: *False*

Outputs

Console usage

```
processing.runalg('gdalorg:extractprojection', input, prj_file)
```

See also

Warp (reproject)

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Source SRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Destination SRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Output file resolution in target georeferenced units (leave 0 for no change) [number]
 <put parameter description here>

Default: *0.0*

Resampling method [selection] <put parameter description here>

Options:

- 0 — near
- 1 — bilinear
- 2 — cubic
- 3 — cubicspline
- 4 — lanczos

Default: *0*

Additional creation parameters [string] Optional.

<put parameter description here>

Default: *(not set)*

Output raster type [selection] <put parameter description here>

Options:

- 0 — Byte
- 1 — Int16
- 2 — UInt16
- 3 — UInt32
- 4 — Int32
- 5 — Float32
- 6 — Float64
- 7 — CInt16
- 8 — CInt32
- 9 — CFloat32
- 10 — CFloat64

Default: *5*

Outputs

Output layer [raster] <put output description here>

Console usage

```
processing.runalg('gdal:warpreproject', input, source_srs, dest_srs, tr, method, extra, rtype,
```

See also

.

18.1.6 Conversão OGR

Convert format

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Destination Format [selection] <put parameter description here>

Options:

- 0 — ESRI Shapefile
- 1 — GeoJSON
- 2 — GeoRSS
- 3 — SQLite
- 4 — GMT
- 5 — MapInfo File
- 6 — INTERLIS 1
- 7 — INTERLIS 2
- 8 — GML
- 9 — Geoconcept
- 10 — DXF
- 11 — DGN
- 12 — CSV
- 13 — BNA
- 14 — S57
- 15 — KML
- 16 — GPX
- 17 — PGDump
- 18 — GPSTrackMaker
- 19 — ODS
- 20 — XLSX
- 21 — PDF

Default: 0

Creation Options [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('gdalogr:convertformat', input_layer, format, options, output_layer)
```

See also

.

18.1.7 Geoprocessamento OGR

Clip vectors by extent

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Clip extent [extent] <put parameter description here>

Default: *0,1,0,1*

Additional creation Options [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('gdalogr:clipvectorsbyextent', input_layer, clip_extent, options, output_layer)
```

See also

Clip vectors by polygon

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Clip layer [vector: polygon] <put parameter description here>

Additional creation Options [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('gdalogr:clipvectorsbypolygon', input_layer, clip_layer, options, output_layer)
```

See also

.

18.1.8 Miscelaneos OGR

Execute SQL

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

SQL [string] <put parameter description here>

Default: *(not set)*

Outputs

SQL result [vector] <put output description here>

Console usage

```
processing.runalg('gdalogr:executesql', input, sql, output)
```

See also**Import Vector into PostGIS database (available connections)****Description**

<put algorithm description here>

Parameters

Database (connection name) [selection] <put parameter description here>

Options:

- 0 — local

Default: 0

Input layer [vector: any] <put parameter description here>

Output geometry type [selection] <put parameter description here>

Options:

- 0 —
- 1 — NONE
- 2 — GEOMETRY
- 3 — POINT
- 4 — LINESTRING
- 5 — POLYGON
- 6 — GEOMETRYCOLLECTION
- 7 — MULTIPOINT
- 8 — MULTIPOLYGON
- 9 — MULTILINESTRING

Default: 5

Input CRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Output CRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Schema name [string] Optional.

<put parameter description here>

Default: *public*

Table name, leave blank to use input name [string] Optional.

<put parameter description here>

Default: *(not set)*

Primary Key [string] Optional.

<put parameter description here>

Default: *id*

Geometry column name [string] Optional.

<put parameter description here>

Default: *geom*

Vector dimensions [selection] <put parameter description here>

Options:

- 0 — 2
- 1 — 3

Default: *0*

Distance tolerance for simplification [string] Optional.

<put parameter description here>

Default: *(not set)*

Maximum distance between 2 nodes (densification) [string] Optional.

<put parameter description here>

Default: *(not set)*

Select features by extent (defined in input layer CRS) [extent] <put parameter description here>

Default: *0,1,0,1*

Clip the input layer using the above (rectangle) extent [boolean] <put parameter description here>

Default: *False*

Select features using a SQL "WHERE" statement (Ex: column="value") [string] Optional.

<put parameter description here>

Default: *(not set)*

Group "n" features per transaction (Default: 20000) [string] Optional.

<put parameter description here>

Default: *(not set)*

Overwrite existing table? [boolean] <put parameter description here>

Default: *True*

Append to existing table? [boolean] <put parameter description here>

Default: *False*

Append and add new fields to existing table? [boolean] <put parameter description here>

Default: *False*

Do not launder columns/table name/s? [boolean] <put parameter description here>

Default: *False*

Do not create Spatial Index? [boolean] <put parameter description here>

Default: *False*

Continue after a failure, skipping the failed feature [boolean] <put parameter description here>

Default: *False*

Additional creation options [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Console usage

```
processing.runalg('gdalogr:importvectorintopostgisdatabaseavailableconnections', database, input_
```

See also

Import Vector into PostGIS database (new connection)

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Output geometry type [selection] <put parameter description here>

Options:

- 0 —
- 1 — NONE
- 2 — GEOMETRY
- 3 — POINT
- 4 — LINESTRING
- 5 — POLYGON
- 6 — GEOMETRYCOLLECTION
- 7 — MULTIPOINT
- 8 — MULTIPOLYGON
- 9 — MULTILINESTRING

Default: 5

Input CRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Output CRS (EPSG Code) [crs] <put parameter description here>

Default: *EPSG:4326*

Host [string] <put parameter description here>

Default: *localhost*

Port [string] <put parameter description here>

Default: *5432*

Username [string] <put parameter description here>

Default: *(not set)*

Database Name [string] <put parameter description here>

Default: *(not set)*

Password [string] <put parameter description here>

Default: *(not set)*

Schema name [string] Optional.

<put parameter description here>

Default: *public*

Table name, leave blank to use input name [string] Optional.

<put parameter description here>

Default: *(not set)*

Primary Key [string] Optional.

<put parameter description here>

Default: *id*

Geometry column name [string] Optional.

<put parameter description here>

Default: *geom*

Vector dimensions [selection] <put parameter description here>

Options:

- 0 — 2
- 1 — 3

Default: *0*

Distance tolerance for simplification [string] Optional.

<put parameter description here>

Default: *(not set)*

Maximum distance between 2 nodes (densification) [string] Optional.

<put parameter description here>

Default: *(not set)*

Select features by extent (defined in input layer CRS) [extent] <put parameter description here>

Default: *0,1,0,1*

Clip the input layer using the above (rectangle) extent [boolean] <put parameter description here>

Default: *False*

Select features using a SQL "WHERE" statement (Ex: column="value") [string] Optional.

<put parameter description here>

Default: *(not set)*

Group "n"features per transaction (Default: 20000) [string] Optional.

<put parameter description here>

Default: *(not set)*

Overwrite existing table? [boolean] <put parameter description here>

Default: *True*

Append to existing table? [boolean] <put parameter description here>

Default: *False*

Append and add new fields to existing table? [boolean] <put parameter description here>

Default: *False*

Do not launder columns/table name/s? [boolean] <put parameter description here>

Default: *False*

Do not create Spatial Index? [boolean] <put parameter description here>

Default: *False*

Continue after a failure, skipping the failed feature [boolean] <put parameter description here>

Default: *False*

Additional creation options [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Console usage

```
processing.runalg('gdalogr:importvectorintopostgisdatabasewconnection', input_layer, gtype, s_s
```

See also

Information

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Outputs

Layer information [html] <put output description here>

Console usage

```
processing.runalg('gdalogr:information', input, output)
```

See also

.

18.2 LAStools

LAStools is a collection of highly efficient, multicore command line tools for LiDAR data processing.

18.2.1 las2las_filter

Description

<put algorithm description here>

Parameters

verbose [**boolean**] <put parameter description here>

Default: *False*

input LAS/LAZ file [**file**] Optional.

<put parameter description here>

filter (by return, classification, flags) [**selection**] <put parameter description here>

Options:

- 0 — —
- 1 — keep_last
- 2 — keep_first
- 3 — keep_middle
- 4 — keep_single
- 5 — drop_single
- 6 — keep_double
- 7 — keep_class 2
- 8 — keep_class 2 8
- 9 — keep_class 8
- 10 — keep_class 6
- 11 — keep_class 9
- 12 — keep_class 3 4 5
- 13 — keep_class 2 6
- 14 — drop_class 7
- 15 — drop_withheld

Default: 0

second filter (by return, classification, flags) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — keep_last
- 2 — keep_first
- 3 — keep_middle
- 4 — keep_single
- 5 — drop_single
- 6 — keep_double
- 7 — keep_class 2
- 8 — keep_class 2 8
- 9 — keep_class 8
- 10 — keep_class 6
- 11 — keep_class 9
- 12 — keep_class 3 4 5
- 13 — keep_class 2 6
- 14 — drop_class 7
- 15 — drop_withheld

Default: 0

filter (by coordinate, intensity, GPS time, ...) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — clip_x_above
- 2 — clip_x_below
- 3 — clip_y_above
- 4 — clip_y_below
- 5 — clip_z_above
- 6 — clip_z_below
- 7 — drop_intensity_above
- 8 — drop_intensity_below
- 9 — drop_gps_time_above
- 10 — drop_gps_time_below
- 11 — drop_scan_angle_above
- 12 — drop_scan_angle_below
- 13 — keep_point_source
- 14 — drop_point_source
- 15 — drop_point_source_above

- 16 — drop_point_source_below
- 17 — keep_user_data
- 18 — drop_user_data
- 19 — drop_user_data_above
- 20 — drop_user_data_below
- 21 — keep_every_nth
- 22 — keep_random_fraction
- 23 — thin_with_grid

Default: 0

value for filter (by coordinate, intensity, GPS time, ...) [string] <put parameter description here>

Default: *(not set)*

second filter (by coordinate, intensity, GPS time, ...) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — clip_x_above
- 2 — clip_x_below
- 3 — clip_y_above
- 4 — clip_y_below
- 5 — clip_z_above
- 6 — clip_z_below
- 7 — drop_intensity_above
- 8 — drop_intensity_below
- 9 — drop_gps_time_above
- 10 — drop_gps_time_below
- 11 — drop_scan_angle_above
- 12 — drop_scan_angle_below
- 13 — keep_point_source
- 14 — drop_point_source
- 15 — drop_point_source_above
- 16 — drop_point_source_below
- 17 — keep_user_data
- 18 — drop_user_data
- 19 — drop_user_data_above
- 20 — drop_user_data_below
- 21 — keep_every_nth
- 22 — keep_random_fraction
- 23 — thin_with_grid

Default: 0

value for second filter (by coordinate, intensity, GPS time, ...) [string] <put parameter description here>

Default: *(not set)*

Outputs

output LAS/LAZ file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:las2lasfilter', verbose, input_laslaz, filter_return_class_flags1, ...)
```

See also

18.2.2 las2las_project

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

source projection [selection] <put parameter description here>

Options:

- 0 — —
- 1 — utm
- 2 — sp83
- 3 — sp27
- 4 — longlat
- 5 — latlong

Default: *0*

source utm zone [selection] <put parameter description here>

Options:

- 0 — —
- 1 — 1 (north)
- 2 — 2 (north)
- 3 — 3 (north)
- 4 — 4 (north)
- 5 — 5 (north)
- 6 — 6 (north)

- 7 — 7 (north)
- 8 — 8 (north)
- 9 — 9 (north)
- 10 — 10 (north)
- 11 — 11 (north)
- 12 — 12 (north)
- 13 — 13 (north)
- 14 — 14 (north)
- 15 — 15 (north)
- 16 — 16 (north)
- 17 — 17 (north)
- 18 — 18 (north)
- 19 — 19 (north)
- 20 — 20 (north)
- 21 — 21 (north)
- 22 — 22 (north)
- 23 — 23 (north)
- 24 — 24 (north)
- 25 — 25 (north)
- 26 — 26 (north)
- 27 — 27 (north)
- 28 — 28 (north)
- 29 — 29 (north)
- 30 — 30 (north)
- 31 — 31 (north)
- 32 — 32 (north)
- 33 — 33 (north)
- 34 — 34 (north)
- 35 — 35 (north)
- 36 — 36 (north)
- 37 — 37 (north)
- 38 — 38 (north)
- 39 — 39 (north)
- 40 — 40 (north)
- 41 — 41 (north)
- 42 — 42 (north)
- 43 — 43 (north)
- 44 — 44 (north)
- 45 — 45 (north)

- 46 — 46 (north)
- 47 — 47 (north)
- 48 — 48 (north)
- 49 — 49 (north)
- 50 — 50 (north)
- 51 — 51 (north)
- 52 — 52 (north)
- 53 — 53 (north)
- 54 — 54 (north)
- 55 — 55 (north)
- 56 — 56 (north)
- 57 — 57 (north)
- 58 — 58 (north)
- 59 — 59 (north)
- 60 — 60 (north)
- 61 — 1 (south)
- 62 — 2 (south)
- 63 — 3 (south)
- 64 — 4 (south)
- 65 — 5 (south)
- 66 — 6 (south)
- 67 — 7 (south)
- 68 — 8 (south)
- 69 — 9 (south)
- 70 — 10 (south)
- 71 — 11 (south)
- 72 — 12 (south)
- 73 — 13 (south)
- 74 — 14 (south)
- 75 — 15 (south)
- 76 — 16 (south)
- 77 — 17 (south)
- 78 — 18 (south)
- 79 — 19 (south)
- 80 — 20 (south)
- 81 — 21 (south)
- 82 — 22 (south)
- 83 — 23 (south)
- 84 — 24 (south)

- 85 — 25 (south)
- 86 — 26 (south)
- 87 — 27 (south)
- 88 — 28 (south)
- 89 — 29 (south)
- 90 — 30 (south)
- 91 — 31 (south)
- 92 — 32 (south)
- 93 — 33 (south)
- 94 — 34 (south)
- 95 — 35 (south)
- 96 — 36 (south)
- 97 — 37 (south)
- 98 — 38 (south)
- 99 — 39 (south)
- 100 — 40 (south)
- 101 — 41 (south)
- 102 — 42 (south)
- 103 — 43 (south)
- 104 — 44 (south)
- 105 — 45 (south)
- 106 — 46 (south)
- 107 — 47 (south)
- 108 — 48 (south)
- 109 — 49 (south)
- 110 — 50 (south)
- 111 — 51 (south)
- 112 — 52 (south)
- 113 — 53 (south)
- 114 — 54 (south)
- 115 — 55 (south)
- 116 — 56 (south)
- 117 — 57 (south)
- 118 — 58 (south)
- 119 — 59 (south)
- 120 — 60 (south)

Default: 0

source state plane code [selection] <put parameter description here>

Options:

- 0 — —
- 1 — AK_10
- 2 — AK_2
- 3 — AK_3
- 4 — AK_4
- 5 — AK_5
- 6 — AK_6
- 7 — AK_7
- 8 — AK_8
- 9 — AK_9
- 10 — AL_E
- 11 — AL_W
- 12 — AR_N
- 13 — AR_S
- 14 — AZ_C
- 15 — AZ_E
- 16 — AZ_W
- 17 — CA_I
- 18 — CA_II
- 19 — CA_III
- 20 — CA_IV
- 21 — CA_V
- 22 — CA_VI
- 23 — CA_VII
- 24 — CO_C
- 25 — CO_N
- 26 — CO_S
- 27 — CT
- 28 — DE
- 29 — FL_E
- 30 — FL_N
- 31 — FL_W
- 32 — GA_E
- 33 — GA_W
- 34 — HI_1
- 35 — HI_2
- 36 — HI_3
- 37 — HI_4
- 38 — HI_5

- 39 — IA_N
- 40 — IA_S
- 41 — ID_C
- 42 — ID_E
- 43 — ID_W
- 44 — IL_E
- 45 — IL_W
- 46 — IN_E
- 47 — IN_W
- 48 — KS_N
- 49 — KS_S
- 50 — KY_N
- 51 — KY_S
- 52 — LA_N
- 53 — LA_S
- 54 — MA_I
- 55 — MA_M
- 56 — MD
- 57 — ME_E
- 58 — ME_W
- 59 — MI_C
- 60 — MI_N
- 61 — MI_S
- 62 — MN_C
- 63 — MN_N
- 64 — MN_S
- 65 — MO_C
- 66 — MO_E
- 67 — MO_W
- 68 — MS_E
- 69 — MS_W
- 70 — MT_C
- 71 — MT_N
- 72 — MT_S
- 73 — NC
- 74 — ND_N
- 75 — ND_S
- 76 — NE_N
- 77 — NE_S

- 78 — NH
- 79 — NJ
- 80 — NM_C
- 81 — NM_E
- 82 — NM_W
- 83 — NV_C
- 84 — NV_E
- 85 — NV_W
- 86 — NY_C
- 87 — NY_E
- 88 — NY_LI
- 89 — NY_W
- 90 — OH_N
- 91 — OH_S
- 92 — OK_N
- 93 — OK_S
- 94 — OR_N
- 95 — OR_S
- 96 — PA_N
- 97 — PA_S
- 98 — PR
- 99 — RI
- 100 — SC_N
- 101 — SC_S
- 102 — SD_N
- 103 — SD_S
- 104 — St.Croix
- 105 — TN
- 106 — TX_C
- 107 — TX_N
- 108 — TX_NC
- 109 — TX_S
- 110 — TX_SC
- 111 — UT_C
- 112 — UT_N
- 113 — UT_S
- 114 — VA_N
- 115 — VA_S
- 116 — VT

- 117 — WA_N
- 118 — WA_S
- 119 — WI_C
- 120 — WI_N
- 121 — WI_S
- 122 — WV_N
- 123 — WV_S
- 124 — WY_E
- 125 — WY_EC
- 126 — WY_W
- 127 — WY_WC

Default: 0

target projection [selection] <put parameter description here>

Options:

- 0 — —
- 1 — utm
- 2 — sp83
- 3 — sp27
- 4 — longlat
- 5 — latlong

Default: 0

target utm zone [selection] <put parameter description here>

Options:

- 0 — —
- 1 — 1 (north)
- 2 — 2 (north)
- 3 — 3 (north)
- 4 — 4 (north)
- 5 — 5 (north)
- 6 — 6 (north)
- 7 — 7 (north)
- 8 — 8 (north)
- 9 — 9 (north)
- 10 — 10 (north)
- 11 — 11 (north)
- 12 — 12 (north)
- 13 — 13 (north)
- 14 — 14 (north)
- 15 — 15 (north)

- 16 — 16 (north)
- 17 — 17 (north)
- 18 — 18 (north)
- 19 — 19 (north)
- 20 — 20 (north)
- 21 — 21 (north)
- 22 — 22 (north)
- 23 — 23 (north)
- 24 — 24 (north)
- 25 — 25 (north)
- 26 — 26 (north)
- 27 — 27 (north)
- 28 — 28 (north)
- 29 — 29 (north)
- 30 — 30 (north)
- 31 — 31 (north)
- 32 — 32 (north)
- 33 — 33 (north)
- 34 — 34 (north)
- 35 — 35 (north)
- 36 — 36 (north)
- 37 — 37 (north)
- 38 — 38 (north)
- 39 — 39 (north)
- 40 — 40 (north)
- 41 — 41 (north)
- 42 — 42 (north)
- 43 — 43 (north)
- 44 — 44 (north)
- 45 — 45 (north)
- 46 — 46 (north)
- 47 — 47 (north)
- 48 — 48 (north)
- 49 — 49 (north)
- 50 — 50 (north)
- 51 — 51 (north)
- 52 — 52 (north)
- 53 — 53 (north)
- 54 — 54 (north)

- 55 — 55 (north)
- 56 — 56 (north)
- 57 — 57 (north)
- 58 — 58 (north)
- 59 — 59 (north)
- 60 — 60 (north)
- 61 — 1 (south)
- 62 — 2 (south)
- 63 — 3 (south)
- 64 — 4 (south)
- 65 — 5 (south)
- 66 — 6 (south)
- 67 — 7 (south)
- 68 — 8 (south)
- 69 — 9 (south)
- 70 — 10 (south)
- 71 — 11 (south)
- 72 — 12 (south)
- 73 — 13 (south)
- 74 — 14 (south)
- 75 — 15 (south)
- 76 — 16 (south)
- 77 — 17 (south)
- 78 — 18 (south)
- 79 — 19 (south)
- 80 — 20 (south)
- 81 — 21 (south)
- 82 — 22 (south)
- 83 — 23 (south)
- 84 — 24 (south)
- 85 — 25 (south)
- 86 — 26 (south)
- 87 — 27 (south)
- 88 — 28 (south)
- 89 — 29 (south)
- 90 — 30 (south)
- 91 — 31 (south)
- 92 — 32 (south)
- 93 — 33 (south)

- 94 — 34 (south)
- 95 — 35 (south)
- 96 — 36 (south)
- 97 — 37 (south)
- 98 — 38 (south)
- 99 — 39 (south)
- 100 — 40 (south)
- 101 — 41 (south)
- 102 — 42 (south)
- 103 — 43 (south)
- 104 — 44 (south)
- 105 — 45 (south)
- 106 — 46 (south)
- 107 — 47 (south)
- 108 — 48 (south)
- 109 — 49 (south)
- 110 — 50 (south)
- 111 — 51 (south)
- 112 — 52 (south)
- 113 — 53 (south)
- 114 — 54 (south)
- 115 — 55 (south)
- 116 — 56 (south)
- 117 — 57 (south)
- 118 — 58 (south)
- 119 — 59 (south)
- 120 — 60 (south)

Default: 0

target state plane code [selection] <put parameter description here>

Options:

- 0 — —
- 1 — AK_10
- 2 — AK_2
- 3 — AK_3
- 4 — AK_4
- 5 — AK_5
- 6 — AK_6
- 7 — AK_7
- 8 — AK_8

- 9 — AK_9
- 10 — AL_E
- 11 — AL_W
- 12 — AR_N
- 13 — AR_S
- 14 — AZ_C
- 15 — AZ_E
- 16 — AZ_W
- 17 — CA_I
- 18 — CA_II
- 19 — CA_III
- 20 — CA_IV
- 21 — CA_V
- 22 — CA_VI
- 23 — CA_VII
- 24 — CO_C
- 25 — CO_N
- 26 — CO_S
- 27 — CT
- 28 — DE
- 29 — FL_E
- 30 — FL_N
- 31 — FL_W
- 32 — GA_E
- 33 — GA_W
- 34 — HI_1
- 35 — HI_2
- 36 — HI_3
- 37 — HI_4
- 38 — HI_5
- 39 — IA_N
- 40 — IA_S
- 41 — ID_C
- 42 — ID_E
- 43 — ID_W
- 44 — IL_E
- 45 — IL_W
- 46 — IN_E
- 47 — IN_W

- 48 — KS_N
- 49 — KS_S
- 50 — KY_N
- 51 — KY_S
- 52 — LA_N
- 53 — LA_S
- 54 — MA_I
- 55 — MA_M
- 56 — MD
- 57 — ME_E
- 58 — ME_W
- 59 — MI_C
- 60 — MI_N
- 61 — MI_S
- 62 — MN_C
- 63 — MN_N
- 64 — MN_S
- 65 — MO_C
- 66 — MO_E
- 67 — MO_W
- 68 — MS_E
- 69 — MS_W
- 70 — MT_C
- 71 — MT_N
- 72 — MT_S
- 73 — NC
- 74 — ND_N
- 75 — ND_S
- 76 — NE_N
- 77 — NE_S
- 78 — NH
- 79 — NJ
- 80 — NM_C
- 81 — NM_E
- 82 — NM_W
- 83 — NV_C
- 84 — NV_E
- 85 — NV_W
- 86 — NY_C

- 87 — NY_E
- 88 — NY_LI
- 89 — NY_W
- 90 — OH_N
- 91 — OH_S
- 92 — OK_N
- 93 — OK_S
- 94 — OR_N
- 95 — OR_S
- 96 — PA_N
- 97 — PA_S
- 98 — PR
- 99 — RI
- 100 — SC_N
- 101 — SC_S
- 102 — SD_N
- 103 — SD_S
- 104 — St.Croix
- 105 — TN
- 106 — TX_C
- 107 — TX_N
- 108 — TX_NC
- 109 — TX_S
- 110 — TX_SC
- 111 — UT_C
- 112 — UT_N
- 113 — UT_S
- 114 — VA_N
- 115 — VA_S
- 116 — VT
- 117 — WA_N
- 118 — WA_S
- 119 — WI_C
- 120 — WI_N
- 121 — WI_S
- 122 — WV_N
- 123 — WV_S
- 124 — WY_E
- 125 — WY_EC

- 126 — WY_W
- 127 — WY_WC

Default: 0

Outputs

output LAS/LAZ file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:las2lasproject', verbose, input_laslaz, source_projection, source_u
```

See also

18.2.3 las2las_transform

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

transform (coordinates) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — translate_x
- 2 — translate_y
- 3 — translate_z
- 4 — scale_x
- 5 — scale_y
- 6 — scale_z
- 7 — clamp_z_above
- 8 — clamp_z_below

Default: 0

value for transform (coordinates) [string] <put parameter description here>

Default: *(not set)*

second transform (coordinates) [selection] <put parameter description here>

Options:

- 0 — —

- 1 — translate_x
- 2 — translate_y
- 3 — translate_z
- 4 — scale_x
- 5 — scale_y
- 6 — scale_z
- 7 — clamp_z_above
- 8 — clamp_z_below

Default: 0

value for second transform (coordinates) [string] <put parameter description here>

Default: *(not set)*

transform (intensities, scan angles, GPS times, ...) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — scale_intensity
- 2 — translate_intensity
- 3 — clamp_intensity_above
- 4 — clamp_intensity_below
- 5 — scale_scan_angle
- 6 — translate_scan_angle
- 7 — translate_gps_time
- 8 — set_classification
- 9 — set_user_data
- 10 — set_point_source
- 11 — scale_rgb_up
- 12 — scale_rgb_down
- 13 — repair_zero_returns

Default: 0

value for transform (intensities, scan angles, GPS times, ...) [string] <put parameter description here>

Default: *(not set)*

second transform (intensities, scan angles, GPS times, ...) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — scale_intensity
- 2 — translate_intensity
- 3 — clamp_intensity_above
- 4 — clamp_intensity_below

- 5 — scale_scan_angle
- 6 — translate_scan_angle
- 7 — translate_gps_time
- 8 — set_classification
- 9 — set_user_data
- 10 — set_point_source
- 11 — scale_rgb_up
- 12 — scale_rgb_down
- 13 — repair_zero_returns

Default: 0

value for second transform (intensities, scan angles, GPS times, ...) [string]
 <put parameter description here>

Default: *(not set)*

operations (first 7 need an argument) [selection] <put parameter description here>

Options:

- 0 — —
- 1 — set_point_type
- 2 — set_point_size
- 3 — set_version_minor
- 4 — set_version_major
- 5 — start_at_point
- 6 — stop_at_point
- 7 — remove_vlr
- 8 — auto_reoffset
- 9 — week_to_adjusted
- 10 — adjusted_to_week
- 11 — scale_rgb_up
- 12 — scale_rgb_down
- 13 — remove_all_vlrs
- 14 — remove_extra
- 15 — clip_to_bounding_box

Default: 0

argument for operation [string] <put parameter description here>

Default: *(not set)*

Outputs

output LAS/LAZ file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:las2lastransform', verbose, input_laslaz, transform_coordinate1, tr
```

See also

18.2.4 las2txt

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

parse_string [string] <put parameter description here>

Default: *xyz*

Outputs

Output ASCII file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:las2txt', verbose, input_laslaz, parse_string, output)
```

See also

18.2.5 lasindex

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

is mobile or terrestrial LiDAR (not airborne) [boolean] <put parameter description here>

Default: *False*

Outputs

Console usage

```
processing.runalg('lidartools:lasindex', verbose, input_laslaz, mobile_or_terrestrial)
```

See also

18.2.6 lasinfo

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

Outputs

Output ASCII file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:lasinfo', verbose, input_laslaz, output)
```

See also

18.2.7 lasmerge

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

files are flightlines [boolean] <put parameter description here>

Default: *True*

input LAS/LAZ file [file] Optional.

<put parameter description here>

2nd file [file] Optional.

<put parameter description here>

3rd file [file] Optional.

<put parameter description here>

4th file [file] Optional.

<put parameter description here>

5th file [file] Optional.

<put parameter description here>

6th file [file] Optional.

<put parameter description here>

7th file [file] Optional.

<put parameter description here>

Outputs

output LAS/LAZ file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:lasmerge', verbose, files_are_flightlines, input_laslaz, file2, file3)
```

See also

18.2.8 lasprecision

Description

<put algorithm description here>

Parameters

verbose [boolean] <put parameter description here>

Default: *False*

input LAS/LAZ file [file] Optional.

<put parameter description here>

Outputs

Output ASCII file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:lasprecision', verbose, input_laslaz, output)
```


See also

18.2.9 lasquery

Description

<put algorithm description here>

Parameters

verbose [**boolean**] <put parameter description here>

Default: *False*

area of interest [**extent**] <put parameter description here>

Default: *0,1,0,1*

Outputs

Console usage

```
processing.runalg('lidartools:lasquery', verbose, aoi)
```

See also

18.2.10 lasvalidate

Description

<put algorithm description here>

Parameters

verbose [**boolean**] <put parameter description here>

Default: *False*

input LAS/LAZ file [**file**] Optional.

<put parameter description here>

Outputs

Output XML file [**file**] <put output description here>

Console usage

```
processing.runalg('lidartools:lasvalidate', verbose, input_laslaz, output)
```

See also

18.2.11 laszip

Description

<put algorithm description here>

Parameters

verbose [**boolean**] <put parameter description here>

Default: *False*

input LAS/LAZ file [**file**] Optional.

<put parameter description here>

only report size [**boolean**] <put parameter description here>

Default: *False*

Outputs

output LAS/LAZ file [**file**] <put output description here>

Console usage

```
processing.runalg('lidartools:laszip', verbose, input_laslaz, report_size, output_laslaz)
```

See also

18.2.12 txt2las

Description

<put algorithm description here>

Parameters

verbose [**boolean**] <put parameter description here>

Default: *False*

Input ASCII file [**file**] Optional.

<put parameter description here>

parse lines as [**string**] <put parameter description here>

Default: *xyz*

skip the first n lines [**number**] <put parameter description here>

Default: *0*

resolution of x and y coordinate [**number**] <put parameter description here>

Default: *0.01*

resolution of z coordinate [number] <put parameter description here>

Default: *0.01*

Outputs

output LAS/LAZ file [file] <put output description here>

Console usage

```
processing.runalg('lidartools:txt2las', verbose, input, parse_string, skip, scale_factor_xy, scale_factor_z)
```

See also

.

18.3 Modeler Tools

18.3.1 Calculator

Description

<put algorithm description here>

Parameters

Formula [string] <put parameter description here>

Default: *(not set)*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: *0.0*

dummy [number] <put parameter description here>

Default: 0.0

dummy [number] <put parameter description here>

Default: 0.0

Outputs

Result [number] <put output description here>

Console usage

```
processing.runalg('modelertools:calculator', formula, number0, number1, number2, number3, number4,
```

See also

18.3.2 Raster layer bounds

Description

<put algorithm description here>

Parameters

Layer [raster] <put parameter description here>

Outputs

min X [number] <put output description here>

max X [number] <put output description here>

min Y [number] <put output description here>

max Y [number] <put output description here>

Extent [extent] <put output description here>

Console usage

```
processing.runalg('modelertools:rasterlayerbounds', layer)
```

See also

18.3.3 Vector layer bounds

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Outputs

min X [number] <put output description here>

max X [number] <put output description here>

min Y [number] <put output description here>

max Y [number] <put output description here>

Extent [extent] <put output description here>

Console usage

```
processing.runalg('modelertools:vectorlayerbounds', layer)
```

See also

.

18.4 OrfeoToolbox algorithm provider

Orfeo ToolBox (OTB) is an open source library of image processing algorithms. OTB is based on the medical image processing library ITK and offers particular functionalities for remote sensing image processing in general and for high spatial resolution images in particular. Targeted algorithms for high resolution optical images (Pleiades, SPOT, QuickBird, WorldView, Landsat, Ikonos), hyperspectral sensors (Hyperion) or SAR (TerraSarX, ERS, Palsar) are available.

Nota: Please remember that Processing contains only the interface description, so you need to install OTB by yourself and configure Processing properly.

.

18.4.1 Calibração

Optical calibration

Description

<put algorithm description here>

Parameters

Input [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Calibration Level [selection] <put parameter description here>

Options:

- 0 — toa

Default: *0*

Convert to milli reflectance [boolean] <put parameter description here>

Default: *True*

Clamp of reflectivity values between [0, 100] [boolean] <put parameter description here>

Default: *True*

Relative Spectral Response File [file] Optional.

<put parameter description here>

Outputs

Output [raster] <put output description here>

Console usage

```
processing.runalg('otb:opticalcalibration', -in, -ram, -level, -milli, -clamp, -rsr, -out)
```

See also

.

18.4.2 Extração de feição

BinaryMorphologicalOperation (closing)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: *0*

The Structuring Element Radius [number] <put parameter description here>

Default: *5*

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — closing

Default: *0*

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:binarymorphologicaloperationclosing', -in, -channel, -ram, -structype, -st
```

See also

BinaryMorphologicalOperation (dilate)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: *0*

The Structuring Element Radius [number] <put parameter description here>

Default: *5*

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — dilate

Default: *0*

Foreground Value [number] <put parameter description here>

Default: *1*

Background Value [number] <put parameter description here>

Default: *0*

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:binarymorphologicaloperationdilate', -in, -channel, -ram, -structype, -stru
```

See also

BinaryMorphologicalOperation (erode)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: *0*

The Structuring Element Radius [number] <put parameter description here>

Default: *5*

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — erode

Default: *0*

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:binarymorphologicaloperationerode', -in, -channel, -ram, -structype, -stru
```

See also

BinaryMorphologicalOperation (opening)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: *0*

The Structuring Element Radius [number] <put parameter description here>

Default: *5*

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — opening

Default: *0*

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:binarymorphologicaloperationopening', -in, -channel, -ram, -structype, -st
```

See also

EdgeExtraction (gradient)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Edge feature [selection] <put parameter description here>

Options:

- 0 — gradient

Default: 0

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:edgeextractiongradient', -in, -channel, -ram, -filter, -out)
```

See also

EdgeExtraction (sobel)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Edge feature [selection] <put parameter description here>

Options:

- 0 — sobel

Default: 0

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:edgeextractionsobel', -in, -channel, -ram, -filter, -out)
```

See also

EdgeExtraction (touzi)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Edge feature [selection] <put parameter description here>

Options:

- 0 — touzi

Default: 0

The Radius [number] <put parameter description here>

Default: 1

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:edgeextractiontouzi', -in, -channel, -ram, -filter, -filter.touzi.xradius,
```

See also

GrayScaleMorphologicalOperation (closing)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: 0

The Structuring Element Radius [number] <put parameter description here>

Default: 5

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — closing

Default: 0

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:grayscalemorphologicaloperationclosing', -in, -channel, -ram, -structype, ...
```

See also

GrayScaleMorphologicalOperation (dilate)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: 0

The Structuring Element Radius [number] <put parameter description here>

Default: 5

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — dilate

Default: 0

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:grayscalemorphologicaloperationdilate', -in, -channel, -ram, -structype, -s
```

See also

GrayScaleMorphologicalOperation (erode)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: 0

The Structuring Element Radius [number] <put parameter description here>

Default: 5

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — erode

Default: 0

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:grayscalemorphologicaloperationerode', -in, -channel, -ram, -structype, -s
```

See also

GrayScaleMorphologicalOperation (opening)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Structuring Element Type [selection] <put parameter description here>

Options:

- 0 — ball

Default: *0*

The Structuring Element Radius [number] <put parameter description here>

Default: *5*

Morphological Operation [selection] <put parameter description here>

Options:

- 0 — opening

Default: *0*

Outputs

Feature Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:grayscalemorphologicaloperationopening', -in, -channel, -ram, -structype, ...
```

See also

Haralick Texture Extraction

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Selected Channel [number] <put parameter description here>

Default: *1*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

X Radius [number] <put parameter description here>

Default: *2*

Y Radius [number] <put parameter description here>

Default: 2

X Offset [number] <put parameter description here>

Default: 1

Y Offset [number] <put parameter description here>

Default: 1

Image Minimum [number] <put parameter description here>

Default: 0

Image Maximum [number] <put parameter description here>

Default: 255

Histogram number of bin [number] <put parameter description here>

Default: 8

Texture Set Selection [selection] <put parameter description here>

Options:

- 0 — simple
- 1 — advanced
- 2 — higher

Default: 0

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:haralicktextureextraction', -in, -channel, -ram, -parameters.xrad, -paramet
```

See also

Line segment detection

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

No rescaling in [0, 255] [boolean] <put parameter description here>

Default: *True*

Outputs

Output **Detected lines** [**vector**] <put output description here>

Console usage

```
processing.runalg('otb:linesegmentdetection', -in, -norescale, -out)
```

See also

Local Statistic Extraction

Description

<put algorithm description here>

Parameters

Input Image [**raster**] <put parameter description here>

Selected Channel [**number**] <put parameter description here>

Default: *1*

Available RAM (Mb) [**number**] <put parameter description here>

Default: *128*

Neighborhood radius [**number**] <put parameter description here>

Default: *3*

Outputs

Feature Output Image [**raster**] <put output description here>

Console usage

```
processing.runalg('otb:localstatisticextraction', -in, -channel, -ram, -radius, -out)
```

See also

Multivariate alteration detector

Description

<put algorithm description here>

Parameters

Input Image 1 [raster] <put parameter description here>

Input Image 2 [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Outputs

Change Map [raster] <put output description here>

Console usage

```
processing.runalg('otb:multivariatealterationdetector', -in1, -in2, -ram, -out)
```

See also

Radiometric Indices

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Blue Channel [number] <put parameter description here>

Default: *1*

Green Channel [number] <put parameter description here>

Default: *1*

Red Channel [number] <put parameter description here>

Default: *1*

NIR Channel [number] <put parameter description here>

Default: *1*

Mir Channel [number] <put parameter description here>

Default: *1*

Available Radiometric Indices [selection] <put parameter description here>

Options:

- 0 — ndvi
- 1 — tndvi
- 2 — rvi

- 3 — savi
- 4 — tsavi
- 5 — msavi
- 6 — msavi2
- 7 — gemi
- 8 — ipvi
- 9 — ndwi
- 10 — ndwi2
- 11 — mndwi
- 12 — ndpi
- 13 — ndti
- 14 — ri
- 15 — ci
- 16 — bi
- 17 — bi2

Default: 0

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:radiometricindices', -in, -ram, -channels.blue, -channels.green, -channels
```

See also

.

18.4.3 Geometria

Image Envelope

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Sampling Rate [number] <put parameter description here>

Default: 0

Projection [string] Optional.

<put parameter description here>

Default: *None*

Outputs

Output Vector Data [vector] <put output description here>

Console usage

```
processing.runalg('otb:imageenvelope', -in, -sr, -proj, -out)
```

See also

OrthoRectification (epsg)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Output Cartographic Map Projection [selection] <put parameter description here>

Options:

- 0 — epsg

Default: *0*

EPSG Code [number] <put parameter description here>

Default: *4326*

Parameters estimation modes [selection] <put parameter description here>

Options:

- 0 — autosize
- 1 — autospacing

Default: *0*

Default pixel value [number] <put parameter description here>

Default: *0*

Default elevation [number] <put parameter description here>

Default: *0*

Interpolation [selection] <put parameter description here>

Options:

- 0 — bco
- 1 — nn
- 2 — linear

Default: 0

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Resampling grid spacing [number] <put parameter description here>

Default: 4

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:orthorectificationeps', -io.in, -map, -map.epsg.code, -outputs.mode, -outp
```

See also

OrthoRectification (fit-to-ortho)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Parameters estimation modes [selection] <put parameter description here>

Options:

- 0 — orthofit

Default: 0

Model ortho-image [raster] Optional.

<put parameter description here>

Default pixel value [number] <put parameter description here>

Default: 0

Default elevation [number] <put parameter description here>

Default: 0

Interpolation [selection] <put parameter description here>

Options:

- 0 — bco
- 1 — nn
- 2 — linear

Default: 0

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Resampling grid spacing [number] <put parameter description here>

Default: 4

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:orthorectificationfittoortho', -io.in, -outputs.mode, -outputs.ortho, -outp
```

See also

OrthoRectification (lambert-WGS84)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Output Cartographic Map Projection [selection] <put parameter description here>

Options:

- 0 — lambert2
- 1 — lambert93
- 2 — wgs

Default: 0

Parameters estimation modes [selection] <put parameter description here>

Options:

- 0 — autosize
- 1 — autospacing

Default: 0

Default pixel value [number] <put parameter description here>

Default: 0

Default elevation [number] <put parameter description here>

Default: 0

Interpolation [selection] <put parameter description here>

Options:

- 0 — bco
- 1 — nn
- 2 — linear

Default: 0

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Resampling grid spacing [number] <put parameter description here>

Default: 4

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:orthorectificationlambertwgs84', -io.in, -map, -outputs.mode, -outputs.def
```

See also

OrthoRectification (utm)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Output Cartographic Map Projection [selection] <put parameter description here>

Options:

- 0 — utm

Default: 0

Zone number [number] <put parameter description here>

Default: 31

Northern Hemisphere [boolean] <put parameter description here>

Default: *True*

Parameters estimation modes [selection] <put parameter description here>

Options:

- 0 — autosize
- 1 — autospacing

Default: 0

Default pixel value [number] <put parameter description here>

Default: 0

Default elevation [number] <put parameter description here>

Default: 0

Interpolation [selection] <put parameter description here>

Options:

- 0 — bco
- 1 — nn
- 2 — linear

Default: 0

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Resampling grid spacing [number] <put parameter description here>

Default: 4

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:orthorectificationutm', -io.in, -map, -map.utm.zone, -map.utm.northhem, -o
```

See also

Pansharpening (bayes)

Description

<put algorithm description here>

Parameters

Input PAN Image [raster] <put parameter description here>

Input XS Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — bayes

Default: 0

Weight [number] <put parameter description here>

Default: 0.9999

S coefficient [number] <put parameter description here>

Default: 1

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:pansharpeningbayes', -inp, -inxs, -method, -method.bayes.lambda, -method.b
```

See also

Pansharpening (lmvm)

Description

<put algorithm description here>

Parameters

Input PAN Image [raster] <put parameter description here>

Input XS Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — lmvm

Default: 0

X radius [number] <put parameter description here>

Default: 3

Y radius [number] <put parameter description here>

Default: 3

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:pansharpeninglmvm', -inp, -inxs, -method, -method.lmvm.radiusx, -method.lm
```


See also

Pansharpening (rcs)

Description

<put algorithm description here>

Parameters

Input PAN Image [raster] <put parameter description here>

Input XS Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — rcs

Default: 0

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:pansharpeningrcs', -inp, -inxs, -method, -ram, -out)
```

See also

RigidTransformResample (id)

Description

<put algorithm description here>

Parameters

Input image [raster] <put parameter description here>

Type of transformation [selection] <put parameter description here>

Options:

- 0 — id

Default: 0

X scaling [number] <put parameter description here>

Default: 1

Y scaling [number] <put parameter description here>

Default: 1

Interpolation [selection] <put parameter description here>

Options:

- 0 — nn
- 1 — linear
- 2 — bco

Default: 2

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:rigidtransformresampleid', -in, -transform.type, -transform.type.id.scalex
```

See also

RigidTransformResample (rotation)

Description

<put algorithm description here>

Parameters

Input image [raster] <put parameter description here>

Type of transformation [selection] <put parameter description here>

Options:

- 0 — rotation

Default: 0

Rotation angle [number] <put parameter description here>

Default: 0

X scaling [number] <put parameter description here>

Default: 1

Y scaling [number] <put parameter description here>

Default: 1

Interpolation [selection] <put parameter description here>

Options:

- 0 — nn
- 1 — linear
- 2 — bco

Default: 2

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:rigidtransformresamplerotation', -in, -transform.type, -transform.type.rot
```

See also

RigidTransformResample (translation)

Description

<put algorithm description here>

Parameters

Input image [raster] <put parameter description here>

Type of transformation [selection] <put parameter description here>

Options:

- 0 — translation

Default: 0

The X translation (in physical units) [number] <put parameter description here>

Default: 0

The Y translation (in physical units) [number] <put parameter description here>

Default: 0

X scaling [number] <put parameter description here>

Default: 1

Y scaling [number] <put parameter description here>

Default: 1

Interpolation [selection] <put parameter description here>

Options:

- 0 — nn
- 1 — linear
- 2 — bco

Default: 2

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:rigidtransformresampletranslation', -in, -transform.type, -transform.type.
```

See also

Superimpose sensor

Description

<put algorithm description here>

Parameters

Reference input [raster] <put parameter description here>

The image to reproject [raster] <put parameter description here>

Default elevation [number] <put parameter description here>

Default: 0

Spacing of the deformation field [number] <put parameter description here>

Default: 4

Interpolation [selection] <put parameter description here>

Options:

- 0 — bco
- 1 — nn
- 2 — linear

Default: 0

Radius for bicubic interpolation [number] <put parameter description here>

Default: 2

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Outputs

Output image [raster] <put output description here>

Console usage

```
processing.runalg('otb:superimposesensor', -inr, -inm, -elev.default, -lms, -interpolator, -inter
```

See also

.

18.4.4 Filtragem de imagem

DimensionalityReduction (ica)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — ica

Default: *0*

number of iterations [number] <put parameter description here>

Default: *20*

Give the increment weight of W in [0, 1] [number] <put parameter description here>

Default: *1*

Number of Components [number] <put parameter description here>

Default: *0*

Normalize [boolean] <put parameter description here>

Default: *True*

Outputs

Output Image [raster] <put output description here>

“**Inverse Output Image**“ [raster] <put output description here>

Transformation matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:dimensionalityreductionica', -in, -method, -method.ica.iter, -method.ica.m
```

See also

DimensionalityReduction (maf)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — maf

Default: 0

Number of Components. [number] <put parameter description here>

Default: 0

Normalize. [boolean] <put parameter description here>

Default: *True*

Outputs

Output Image [raster] <put output description here>

Transformation matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:dimensionalityreductionmaf', -in, -method, -nbcomp, -normalize, -out, -out
```

See also

DimensionalityReduction (napca)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — napca

Default: 0

Set the x radius of the sliding window. [number] <put parameter description here>

Default: 1

Set the y radius of the sliding window. [number] <put parameter description here>

Default: 1

Number of Components. [number] <put parameter description here>

Default: 0

Normalize. [boolean] <put parameter description here>

Default: *True*

Outputs

Output Image [raster] <put output description here>

“**Inverse Output Image**“ [raster] <put output description here>

Transformation matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:dimensionalityreductionnapca', -in, -method, -method.napca.radiusx, -method
```

See also

DimensionalityReduction (pca)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Algorithm [selection] <put parameter description here>

Options:

- 0 — pca

Default: 0

Number of Components. [number] <put parameter description here>

Default: 0

Normalize. [boolean] <put parameter description here>

Default: *True*

Outputs

Output Image [raster] <put output description here>

“ **Inverse Output Image**“ [raster] <put output description here>

Transformation matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:dimensionalityreductionpca', -in, -method, -nbcomp, -normalize, -out, -out.
```

See also

Mean Shift filtering (can be used as Exact Large-Scale Mean-Shift segmentation, step 1)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Spatial radius [number] <put parameter description here>

Default: *5*

Range radius [number] <put parameter description here>

Default: *15*

Mode convergence threshold [number] <put parameter description here>

Default: *0.1*

Maximum number of iterations [number] <put parameter description here>

Default: *100*

Range radius coefficient [number] <put parameter description here>

Default: *0*

Mode search. [boolean] <put parameter description here>

Default: *True*

Outputs

Filtered output [raster] <put output description here>

Spatial image [raster] <put output description here>

Console usage

```
processing.runalg('otb:meanshiftfilteringcanbeusedasexactlargescalemeanshiftsegmentationstep1', -
```

See also

Smoothing (anidif)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Smoothing Type [selection] <put parameter description here>

Options:

- 0 — anidif

Default: *2*

Time Step [number] <put parameter description here>

Default: *0.125*

Nb Iterations [number] <put parameter description here>

Default: *10*

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:smoothinganidif', -in, -ram, -type, -type.anidif.timestep, -type.anidif.nb
```

See also

Smoothing (gaussian)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Smoothing Type [selection] <put parameter description here>

Options:

- 0 — gaussian

Default: 2

Radius [number] <put parameter description here>

Default: 2

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:smoothinggaussian', -in, -ram, -type, -type.gaussian.radius, -out)
```

See also

Smoothing (mean)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Smoothing Type [selection] <put parameter description here>

Options:

- 0 — mean

Default: 2

Radius [number] <put parameter description here>

Default: 2

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:smoothingmean', -in, -ram, -type, -type.mean.radius, -out)
```

See also

.

18.4.5 Manipulação de imagem

ColorMapping (continuous)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Operation [selection] <put parameter description here>

Options:

- 0 — labeltocolour

Default: *0*

Color mapping method [selection] <put parameter description here>

Options:

- 0 — continuous

Default: *0*

Look-up tables [selection] <put parameter description here>

Options:

- 0 — red
- 1 — green
- 2 — blue
- 3 — grey
- 4 — hot
- 5 — cool
- 6 — spring
- 7 — summer
- 8 — autumn
- 9 — winter
- 10 — copper

- 11 — jet
- 12 — hsv
- 13 — overunder
- 14 — relief

Default: 0

Mapping range lower value [number] <put parameter description here>

Default: 0

Mapping range higher value [number] <put parameter description here>

Default: 255

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:colormappingcontinuous', -in, -ram, -op, -method, -method.continuous.lut, ...
```

See also

ColorMapping (custom)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Operation [selection] <put parameter description here>

Options:

- 0 — labeltcolor

Default: 0

Color mapping method [selection] <put parameter description here>

Options:

- 0 — custom

Default: 0

Look-up table file [file] <put parameter description here>

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:colormappingcustom', -in, -ram, -op, -method, -method.custom.lut, -out)
```

See also

ColorMapping (image)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Operation [selection] <put parameter description here>

Options:

- 0 — labeltocolor

Default: *0*

Color mapping method [selection] <put parameter description here>

Options:

- 0 — image

Default: *0*

Support Image [raster] <put parameter description here>

NoData value [number] <put parameter description here>

Default: *0*

lower quantile [number] <put parameter description here>

Default: *2*

upper quantile [number] <put parameter description here>

Default: *2*

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:colormappingimage', -in, -ram, -op, -method, -method.image.in, -method.ima
```

See also

ColorMapping (optimal)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Operation [selection] <put parameter description here>

Options:

- 0 — labeltocolor

Default: *0*

Color mapping method [selection] <put parameter description here>

Options:

- 0 — optimal

Default: *0*

Background label [number] <put parameter description here>

Default: *0*

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:colormappingoptimal', -in, -ram, -op, -method, -method.optimal.background,
```

See also

ExtractROI (fit)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Extraction mode [selection] <put parameter description here>

Options:

- 0 — fit

Default: 0

Reference image [raster] <put parameter description here>

Default elevation [number] <put parameter description here>

Default: 0

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:extractroi', -in, -ram, -mode, -mode.fit.ref, -mode.fit.elev.default, -
```

See also

ExtractROI (standard)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Extraction mode [selection] <put parameter description here>

Options:

- 0 — standard

Default: 0

Start X [number] <put parameter description here>

Default: 0

Start Y [number] <put parameter description here>

Default: 0

Size X [number] <put parameter description here>

Default: 0

Size Y [number] <put parameter description here>

Default: 0

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:extractroistandard', -in, -ram, -mode, -startx, -starty, -sizex, -sizey, -o
```

See also

Images Concatenation

Description

<put algorithm description here>

Parameters

Input images list [multipleinput: rasters] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:imagesconcatenation', -il, -ram, -out)
```

See also

Image Tile Fusion

Description

<put algorithm description here>

Parameters

Input Tile Images [**multipleinput: rasters**] <put parameter description here>

Number of tile columns [**number**] <put parameter description here>

Default: *0*

Number of tile rows [**number**] <put parameter description here>

Default: *0*

Outputs

Output Image [**raster**] <put output description here>

Console usage

```
processing.runalg('otb:imagetilefusion', -il, -cols, -rows, -out)
```

See also

Read image information

Description

<put algorithm description here>

Parameters

Input Image [**raster**] <put parameter description here>

Display the OSSIM keywordlist [**boolean**] <put parameter description here>

Default: *True*

GCPs Id [**string**] <put parameter description here>

Default: *None*

GCPs Info [**string**] <put parameter description here>

Default: *None*

GCPs Image Coordinates [**string**] <put parameter description here>

Default: *None*

GCPs Geographic Coordinates [**string**] <put parameter description here>

Default: *None*

Outputs

Console usage

```
processing.runalg('otb:readimageinformation', -in, -keywordlist, -gcp.ids, -gcp.info, -gcp.imcoord
```

See also

Rescale Image

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Output min value [number] <put parameter description here>

Default: 0

Output max value [number] <put parameter description here>

Default: 255

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:rescaleimage', -in, -ram, -outmin, -outmax, -out)
```

See also

Split Image

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output Image [file] <put output description here>

Console usage

```
processing.runalg('otb:splitimage', -in, -ram, -out)
```

See also

18.4.6 Aprendizado

Classification Map Regularization

Description

<put algorithm description here>

Parameters

Input classification image [raster] <put parameter description here>

Structuring element radius (in pixels) [number] <put parameter description here>

Default: *1*

Multiple majority: Undecided(X)/Original [boolean] <put parameter description here>

Default: *True*

Label for the NoData class [number] <put parameter description here>

Default: *0*

Label for the Undecided class [number] <put parameter description here>

Default: *0*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Outputs

Output regularized image [raster] <put output description here>

Console usage

```
processing.runalg('otb:classificationmapregularization', -io.in, -ip.radius, -ip.suvbool, -ip.nod
```

See also

ComputeConfusionMatrix (raster)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Ground truth [selection] <put parameter description here>

Options:

- 0 — raster

Default: 0

Input reference image [raster] <put parameter description here>

Value for nodata pixels [number] <put parameter description here>

Default: 0

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:computeconfusionmatrixraster', -in, -ref, -ref.raster.in, -nodatalabel, -r
```

See also

ComputeConfusionMatrix (vector)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Ground truth [selection] <put parameter description here>

Options:

- 0 — vector

Default: 0

Input reference vector data [file] <put parameter description here>

Field name [string] Optional.

<put parameter description here>

Default: *Class*

Value for nodata pixels [number] <put parameter description here>

Default: 0

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Matrix output [file] <put output description here>

Console usage

```
processing.runalg('otb:computeconfusionmatrixvector', -in, -ref, -ref.vector.in, -ref.vector.field)
```

See also

Compute Images second order statistics

Description

<put algorithm description here>

Parameters

Input images [multipleinput: rasters] <put parameter description here>

Background Value [number] <put parameter description here>

Default: 0.0

Outputs

Output XML file [file] <put output description here>

Console usage

```
processing.runalg('otb:computeimagessecondorderstatistics', -il, -bv, -out)
```

See also

FusionOfClassifications (dempstershafer)

Description

<put algorithm description here>

Parameters

Input classifications [multipleinput: rasters] <put parameter description here>

Fusion method [selection] <put parameter description here>

Options:

- 0 — dempstershafer

Default: 0

Confusion Matrices [**multipleinput: files**] <put parameter description here>

Mass of belief measurement [**selection**] <put parameter description here>

Options:

- 0 — precision
- 1 — recall
- 2 — accuracy
- 3 — kappa

Default: 0

Label for the NoData class [**number**] <put parameter description here>

Default: 0

Label for the Undecided class [**number**] <put parameter description here>

Default: 0

Outputs

The output classification image [**raster**] <put output description here>

Console usage

```
processing.runalg('otb:fusionofclassificationsdempstershafer', -il, -method, -method.dempstershafer)
```

See also

FusionOfClassifications (majorityvoting)

Description

<put algorithm description here>

Parameters

Input classifications [**multipleinput: rasters**] <put parameter description here>

Fusion method [**selection**] <put parameter description here>

Options:

- 0 — majorityvoting

Default: 0

Label for the NoData class [**number**] <put parameter description here>

Default: 0

Label for the Undecided class [**number**] <put parameter description here>

Default: 0

Outputs

The output classification image [raster] <put output description here>

Console usage

```
processing.runalg('otb:fusionofclassificationsmajorityvoting', -il, -method, -nodatalabel, -undec
```

See also

Image Classification

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Input Mask [raster] Optional.

<put parameter description here>

Model file [file] <put parameter description here>

Statistics file [file] Optional.

<put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:imageclassification', -in, -mask, -model, -imstat, -ram, -out)
```

See also

SOM Classification

Description

<put algorithm description here>

Parameters

InputImage [raster] <put parameter description here>

ValidityMask [raster] Optional.

<put parameter description here>

TrainingProbability [number] <put parameter description here>

Default: 1

TrainingSetSize [number] <put parameter description here>

Default: 0

StreamingLines [number] <put parameter description here>

Default: 0

SizeX [number] <put parameter description here>

Default: 32

SizeY [number] <put parameter description here>

Default: 32

NeighborhoodX [number] <put parameter description here>

Default: 10

NeighborhoodY [number] <put parameter description here>

Default: 10

NumberIteration [number] <put parameter description here>

Default: 5

BetaInit [number] <put parameter description here>

Default: 1

BetaFinal [number] <put parameter description here>

Default: 0.1

InitialValue [number] <put parameter description here>

Default: 0

Available RAM (Mb) [number] <put parameter description here>

Default: 128

set user defined seed [number] <put parameter description here>

Default: 0

Outputs

OutputImage [raster] <put output description here>

SOM Map [raster] <put output description here>

Console usage

```
processing.runalg('otb:somclassification', -in, -vm, -tp, -ts, -sl, -sx, -sy, -nx, -ny, -ni, -bi,
```


See also

TrainImagesClassifier (ann)

Description

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: *0*

Maximum training sample size per class [**number**] <put parameter description here>

Default: *1000*

Maximum validation sample size per class [**number**] <put parameter description here>

Default: *1000*

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: *True*

Training and validation sample ratio [**number**] <put parameter description here>

Default: *0.5*

Name of the discrimination field [**string**] <put parameter description here>

Default: *Class*

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — ann

Default: *0*

Train Method Type [**selection**] <put parameter description here>

Options:

- 0 — reg
- 1 — back

Default: *0*

Number of neurons in each intermediate layer [**string**] <put parameter description here>

Default: *None*

Neuron activation function type [**selection**] <put parameter description here>

Options:

- 0 — ident
- 1 — sig

- 2 — gau

Default: 1

Alpha parameter of the activation function [number] <put parameter description here>

Default: 1

Beta parameter of the activation function [number] <put parameter description here>

Default: 1

Strength of the weight gradient term in the BACKPROP method [number] <put parameter description here>

Default: 0.1

Strength of the momentum term (the difference between weights on the 2 previous iterations) [number] <put parameter description here>

Default: 0.1

Initial value Delta_0 of update-values Delta_{ij} in RPROP method [number] <put parameter description here>

Default: 0.1

Update-values lower limit Delta_{min} in RPROP method [number] <put parameter description here>

Default: 1e-07

Termination criteria [selection] <put parameter description here>

Options:

- 0 — iter
- 1 — eps
- 2 — all

Default: 2

Epsilon value used in the Termination criteria [number] <put parameter description here>

Default: 0.01

Maximum number of iterations used in the Termination criteria [number] <put parameter description here>

Default: 1000

set user defined seed [number] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierann', -io.il, -io.vd, -io.imstat, -elev.default, -sam
```

See also**TrainImagesClassifier (bayes)****Description**

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: 0

Maximum training sample size per class [**number**] <put parameter description here>

Default: 1000

Maximum validation sample size per class [**number**] <put parameter description here>

Default: 1000

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: *True*

Training and validation sample ratio [**number**] <put parameter description here>

Default: 0.5

Name of the discrimination field [**string**] <put parameter description here>

Default: *Class*

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — bayes

Default: 0

set user defined seed [**number**] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [**file**] <put output description here>

Output model [**file**] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierbayes', -io.il, -io.vd, -io.imstat, -elev.default, -s
```

See also

TrainImagesClassifier (boost)

Description

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: 0

Maximum training sample size per class [**number**] <put parameter description here>

Default: 1000

Maximum validation sample size per class [**number**] <put parameter description here>

Default: 1000

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: True

Training and validation sample ratio [**number**] <put parameter description here>

Default: 0.5

Name of the discrimination field [**string**] <put parameter description here>

Default: Class

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — boost

Default: 0

Boost Type [**selection**] <put parameter description here>

Options:

- 0 — discrete
- 1 — real
- 2 — logit
- 3 — gentle

Default: 1

Weak count [**number**] <put parameter description here>

Default: 100

Weight Trim Rate [**number**] <put parameter description here>

Default: 0.95

Maximum depth of the tree [number] <put parameter description here>

Default: *1*

set user defined seed [number] <put parameter description here>

Default: *0*

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierboost', -io.il, -io.vd, -io.imstat, -elev.default, -s
```

See also

TrainImagesClassifier (dt)

Description

<put algorithm description here>

Parameters

Input Image List [multipleinput: rasters] <put parameter description here>

Input Vector Data List [multipleinput: any vectors] <put parameter description here>

Input XML image statistics file [file] Optional.

<put parameter description here>

Default elevation [number] <put parameter description here>

Default: *0*

Maximum training sample size per class [number] <put parameter description here>

Default: *1000*

Maximum validation sample size per class [number] <put parameter description here>

Default: *1000*

On edge pixel inclusion [boolean] <put parameter description here>

Default: *True*

Training and validation sample ratio [number] <put parameter description here>

Default: *0.5*

Name of the discrimination field [string] <put parameter description here>

Default: *Class*

Classifier to use for the training [selection] <put parameter description here>

Options:

- 0 — dt

Default: 0

Maximum depth of the tree [number] <put parameter description here>

Default: 65535

Minimum number of samples in each node [number] <put parameter description here>

Default: 10

Termination criteria for regression tree [number] <put parameter description here>

Default: 0.01

Cluster possible values of a categorical variable into K <= cat clusters to find a suboptimal
<put parameter description here>

Default: 10

K-fold cross-validations [number] <put parameter description here>

Default: 10

Set UseSelfRule flag to false [boolean] <put parameter description here>

Default: True

Set TruncatePrunedTree flag to false [boolean] <put parameter description here>

Default: True

set user defined seed [number] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierdt', -io.il, -io.vd, -io.imstat, -elev.default, -samp
```

See also

TrainImagesClassifier (gbt)

Description

<put algorithm description here>

Parameters

Input Image List [multipleinput: rasters] <put parameter description here>

Input Vector Data List [multipleinput: any vectors] <put parameter description here>

Input XML image statistics file [file] Optional.

<put parameter description here>

Default elevation [number] <put parameter description here>

Default: *0*

Maximum training sample size per class [number] <put parameter description here>

Default: *1000*

Maximum validation sample size per class [number] <put parameter description here>

Default: *1000*

On edge pixel inclusion [boolean] <put parameter description here>

Default: *True*

Training and validation sample ratio [number] <put parameter description here>

Default: *0.5*

Name of the discrimination field [string] <put parameter description here>

Default: *Class*

Classifier to use for the training [selection] <put parameter description here>

Options:

- 0 — gbt

Default: *0*

Number of boosting algorithm iterations [number] <put parameter description here>

Default: *200*

Regularization parameter [number] <put parameter description here>

Default: *0.01*

Portion of the whole training set used for each algorithm iteration [number]

<put parameter description here>

Default: *0.8*

Maximum depth of the tree [number] <put parameter description here>

Default: *3*

set user defined seed [number] <put parameter description here>

Default: *0*

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifiergbt', -io.il, -io.vd, -io.imstat, -elev.default, -sam
```

See also

TrainImagesClassifier (knn)

Description

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: 0

Maximum training sample size per class [**number**] <put parameter description here>

Default: 1000

Maximum validation sample size per class [**number**] <put parameter description here>

Default: 1000

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: *True*

Training and validation sample ratio [**number**] <put parameter description here>

Default: 0.5

Name of the discrimination field [**string**] <put parameter description here>

Default: *Class*

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — knn

Default: 0

Number of Neighbors [**number**] <put parameter description here>

Default: 32

set user defined seed [**number**] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [**file**] <put output description here>

Output model [**file**] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierknn', -io.il, -io.vd, -io.imstat, -elev.default, -sam
```

See also

TrainImagesClassifier (libsvm)

Description

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: *0*

Maximum training sample size per class [**number**] <put parameter description here>

Default: *1000*

Maximum validation sample size per class [**number**] <put parameter description here>

Default: *1000*

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: *True*

Training and validation sample ratio [**number**] <put parameter description here>

Default: *0.5*

Name of the discrimination field [**string**] <put parameter description here>

Default: *Class*

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — libsvm

Default: *0*

SVM Kernel Type [**selection**] <put parameter description here>

Options:

- 0 — linear
- 1 — rbf
- 2 — poly
- 3 — sigmoid

Default: *0*

Cost parameter C [number] <put parameter description here>

Default: *1*

Parameters optimization [boolean] <put parameter description here>

Default: *True*

set user defined seed [number] <put parameter description here>

Default: *0*

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierlibsvm', -io.il, -io.vd, -io.imstat, -elev.default, -
```

See also

TrainImagesClassifier (rf)

Description

<put algorithm description here>

Parameters

Input Image List [multipleinput: rasters] <put parameter description here>

Input Vector Data List [multipleinput: any vectors] <put parameter description here>

Input XML image statistics file [file] Optional.

<put parameter description here>

Default elevation [number] <put parameter description here>

Default: *0*

Maximum training sample size per class [number] <put parameter description here>

Default: *1000*

Maximum validation sample size per class [number] <put parameter description here>

Default: *1000*

On edge pixel inclusion [boolean] <put parameter description here>

Default: *True*

Training and validation sample ratio [number] <put parameter description here>

Default: *0.5*

Name of the discrimination field [string] <put parameter description here>

Default: *Class*

Classifier to use for the training [selection] <put parameter description here>

Options:

- 0 — rf

Default: 0

Maximum depth of the tree [number] <put parameter description here>

Default: 5

Minimum number of samples in each node [number] <put parameter description here>

Default: 10

Termination Criteria for regression tree [number] <put parameter description here>

Default: 0

Cluster possible values of a categorical variable into K <= cat clusters to find a suboptimal <put parameter description here>

Default: 10

Size of the randomly selected subset of features at each tree node [number] <put parameter description here>

Default: 0

Maximum number of trees in the forest [number] <put parameter description here>

Default: 100

Sufficient accuracy (OOB error) [number] <put parameter description here>

Default: 0.01

set user defined seed [number] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifierrf', -io.il, -io.vd, -io.imstat, -elev.default, -samp
```

See also

TrainImagesClassifier (svm)

Description

<put algorithm description here>

Parameters

Input Image List [**multipleinput: rasters**] <put parameter description here>

Input Vector Data List [**multipleinput: any vectors**] <put parameter description here>

Input XML image statistics file [**file**] Optional.

<put parameter description here>

Default elevation [**number**] <put parameter description here>

Default: 0

Maximum training sample size per class [**number**] <put parameter description here>

Default: 1000

Maximum validation sample size per class [**number**] <put parameter description here>

Default: 1000

On edge pixel inclusion [**boolean**] <put parameter description here>

Default: *True*

Training and validation sample ratio [**number**] <put parameter description here>

Default: 0.5

Name of the discrimination field [**string**] <put parameter description here>

Default: *Class*

Classifier to use for the training [**selection**] <put parameter description here>

Options:

- 0 — svm

Default: 0

SVM Model Type [**selection**] <put parameter description here>

Options:

- 0 — csvc
- 1 — nusvc
- 2 — oneclass

Default: 0

SVM Kernel Type [**selection**] <put parameter description here>

Options:

- 0 — linear
- 1 — rbf
- 2 — poly
- 3 — sigmoid

Default: 0

Cost parameter C [**number**] <put parameter description here>

Default: 1

Parameter nu of a SVM optimization problem (NU_SVC / ONE_CLASS) [**number**] <put parameter description here>

Default: 0

Parameter coef0 of a kernel function (POLY / SIGMOID) [number] <put parameter description here>

Default: 0

Parameter gamma of a kernel function (POLY / RBF / SIGMOID) [number] <put parameter description here>

Default: 1

Parameter degree of a kernel function (POLY) [number] <put parameter description here>

Default: 1

Parameters optimization [boolean] <put parameter description here>

Default: *True*

set user defined seed [number] <put parameter description here>

Default: 0

Outputs

Output confusion matrix [file] <put output description here>

Output model [file] <put output description here>

Console usage

```
processing.runalg('otb:trainimagesclassifiersvm', -io.il, -io.vd, -io.imstat, -elev.default, -sam
```

See also

Unsupervised KMeans image classification

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Validity Mask [raster] Optional.

<put parameter description here>

Training set size [number] <put parameter description here>

Default: 100

Number of classes [number] <put parameter description here>

Default: 5

Maximum number of iterations [number] <put parameter description here>

Default: 1000

Convergence threshold [number] <put parameter description here>

Default: *0.0001*

Outputs

Output Image [raster] <put output description here>

Centroid filename [file] <put output description here>

Console usage

```
processing.runalg('otb:unsupervisedkmeansimageclassification', -in, -ram, -vm, -ts, -nc, -maxit,
```

See also

.

18.4.7 Miscelaneos

Band Math

Description

<put algorithm description here>

Parameters

Input image list [multipleinput: rasters] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Expression [string] <put parameter description here>

Default: *None*

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:bandmath', -il, -ram, -exp, -out)
```

See also

ComputeModulusAndPhase-one (OneEntry)

Description

<put algorithm description here>

Parameters

Number Of inputs [selection] <put parameter description here>

Options:

- 0 — one

Default: 0

Input image [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Modulus [raster] <put output description here>

Phase [raster] <put output description here>

Console usage

```
processing.runalg('otb:computemodulusandphaseoneentry', -nbininput, -nbininput.one.in, -ram, -mod,
```

See also

ComputeModulusAndPhase-two (TwoEntries)

Description

<put algorithm description here>

Parameters

Number Of inputs [selection] <put parameter description here>

Options:

- 0 — two

Default: 0

Real part input [raster] <put parameter description here>

Imaginary part input [raster] <put parameter description here>

Available RAM (Mb) [number] <put parameter description here>

Default: 128

Outputs

Modulus [raster] <put output description here>

Phase [raster] <put output description here>

Console usage

```
processing.runalg('otb:computemodulusandphasetwoentries', -nbinput, -nbinput.two.re, -nbinput.t
```

See also

Images comparaison

Description

<put algortithm description here>

Parameters

Reference image [raster] <put parameter description here>

Reference image channel [number] <put parameter description here>

Default: *1*

Measured image [raster] <put parameter description here>

Measured image channel [number] <put parameter description here>

Default: *1*

Start X [number] <put parameter description here>

Default: *0*

Start Y [number] <put parameter description here>

Default: *0*

Size X [number] <put parameter description here>

Default: *0*

Size Y [number] <put parameter description here>

Default: *0*

Outputs

Console usage

```
processing.runalg('otb:imagescomparaison', -ref.in, -ref.channel, -meas.in, -meas.channel, -roi.st
```

See also

Image to KMZ Export

Description

<put algortithm description here>

Parameters

Input image [raster] <put parameter description here>

Tile Size [number] <put parameter description here>

Default: *512*

Image logo [raster] Optional.

<put parameter description here>

Image legend [raster] Optional.

<put parameter description here>

Default elevation [number] <put parameter description here>

Default: *0*

Outputs

Output .kmz product [file] <put output description here>

Console usage

```
processing.runalg('otb:imagetokmzexport', -in, -tilesize, -logo, -legend, -elev.default, -out)
```

See also

.

18.4.8 Segmentação

Connected Component Segmentation

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Mask expression [string] Optional.

<put parameter description here>

Default: *None*

Connected Component Expression [string] <put parameter description here>

Default: *None*

Minimum Object Size [number] <put parameter description here>

Default: *2*

OBIA Expression [string] Optional.

<put parameter description here>

Default: *None*

Default elevation [number] <put parameter description here>

Default: *0*

Outputs

Output Shape [vector] <put output description here>

Console usage

```
processing.runalg('otb:connectedcomponentsegmentation', -in, -mask, -expr, -minsize, -obia, -elev
```

See also

Exact Large-Scale Mean-Shift segmentation, step 2

Description

<put algorithm description here>

Parameters

Filtered image [raster] <put parameter description here>

Spatial image [raster] Optional.

<put parameter description here>

Range radius [number] <put parameter description here>

Default: *15*

Spatial radius [number] <put parameter description here>

Default: *5*

Minimum Region Size [number] <put parameter description here>

Default: *0*

Size of tiles in pixel (X-axis) [number] <put parameter description here>

Default: *500*

Size of tiles in pixel (Y-axis) [number] <put parameter description here>

Default: *500*

Directory where to write temporary files [file] Optional.

<put parameter description here>

Temporary files cleaning [boolean] <put parameter description here>

Default: *True*

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:exactlargescalemeanshiftsegmentationstep2', -in, -inpos, -ranger, -spatial,
```

See also

Exact Large-Scale Mean-Shift segmentation, step 3 (optional)

Description

<put algorithm description here>

Parameters

Input image [raster] <put parameter description here>

Segmented image [raster] <put parameter description here>

Minimum Region Size [number] <put parameter description here>

Default: 50

Size of tiles in pixel (X-axis) [number] <put parameter description here>

Default: 500

Size of tiles in pixel (Y-axis) [number] <put parameter description here>

Default: 500

Outputs

Output Image [raster] <put output description here>

Console usage

```
processing.runalg('otb:exactlargescalemeanshiftsegmentationstep3optional', -in, -inseg, -minsize,
```

See also

Exact Large-Scale Mean-Shift segmentation, step 4

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Segmented image [raster] <put parameter description here>

Size of tiles in pixel (X-axis) [number] <put parameter description here>

Default: *500*

Size of tiles in pixel (Y-axis) [number] <put parameter description here>

Default: *500*

Outputs

Output GIS vector file [vector] <put output description here>

Console usage

```
processing.runalg('otb:exactlargescalemeanshiftsegmentationstep4', -in, -inseg, -tilesizex, -tile
```

See also

Hoover compare segmentation

Description

<put algorithm description here>

Parameters

Input ground truth [raster] <put parameter description here>

Input machine segmentation [raster] <put parameter description here>

Background label [number] <put parameter description here>

Default: *0*

Overlapping threshold [number] <put parameter description here>

Default: *0.75*

Correct detection score [number] <put parameter description here>

Default: *0.0*

Over-segmentation score [number] <put parameter description here>

Default: *0.0*

Under-segmentation score [number] <put parameter description here>

Default: *0.0*

Missed detection score [number] <put parameter description here>

Default: *0.0*

Outputs

Colored ground truth output [raster] <put output description here>

Colored machine segmentation output [raster] <put output description here>

Console usage

```
processing.runalg('otb:hoovercomparesegmentation', -ingt, -inms, -bg, -th, -rc, -rf, -ra, -rm, -o
```

See also

Segmentation (cc)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Segmentation algorithm [selection] <put parameter description here>

Options:

- 0 — cc

Default: 0

Condition [string] <put parameter description here>

Default: *None*

Processing mode [selection] <put parameter description here>

Options:

- 0 — vector

Default: 0

Writing mode for the output vector file [selection] <put parameter description here>

Options:

- 0 — ulco
- 1 — ovw
- 2 — ulovw
- 3 — ulu

Default: 0

Mask Image [raster] Optional.

<put parameter description here>

8-neighbor connectivity [boolean] <put parameter description here>

Default: *True*

Stitch polygons [**boolean**] <put parameter description here>

Default: *True*

Minimum object size [**number**] <put parameter description here>

Default: *1*

Simplify polygons [**number**] <put parameter description here>

Default: *0.1*

Layer name [**string**] <put parameter description here>

Default: *layer*

Geometry index field name [**string**] <put parameter description here>

Default: *DN*

Tiles size [**number**] <put parameter description here>

Default: *1024*

Starting geometry index [**number**] <put parameter description here>

Default: *1*

OGR options for layer creation [**string**] Optional.

<put parameter description here>

Default: *None*

Outputs

Output vector file [**vector**] <put output description here>

Console usage

```
processing.runalg('otb:segmentationcc', -in, -filter, -filter.cc.expr, -mode, -mode.vector.outmode)
```

See also

Segmentation (edison)

Description

<put algorithm description here>

Parameters

Input Image [**raster**] <put parameter description here>

Segmentation algorithm [**selection**] <put parameter description here>

Options:

- 0 — edison

Default: *0*

Spatial radius [**number**] <put parameter description here>

Default: *5*

Range radius [number] <put parameter description here>

Default: *15*

Minimum region size [number] <put parameter description here>

Default: *100*

Scale factor [number] <put parameter description here>

Default: *1*

Processing mode [selection] <put parameter description here>

Options:

- 0 — vector

Default: *0*

Writing mode for the output vector file [selection] <put parameter description here>

Options:

- 0 — ulco
- 1 — ovw
- 2 — ulovw
- 3 — ulu

Default: *0*

Mask Image [raster] Optional.

<put parameter description here>

8-neighbor connectivity [boolean] <put parameter description here>

Default: *True*

Stitch polygons [boolean] <put parameter description here>

Default: *True*

Minimum object size [number] <put parameter description here>

Default: *1*

Simplify polygons [number] <put parameter description here>

Default: *0.1*

Layer name [string] <put parameter description here>

Default: *layer*

Geometry index field name [string] <put parameter description here>

Default: *DN*

Tiles size [number] <put parameter description here>

Default: *1024*

Starting geometry index [number] <put parameter description here>

Default: *1*

OGR options for layer creation [string] Optional.

<put parameter description here>

Default: *None*

Outputs

Output vector file [**vector**] <put output description here>

Console usage

```
processing.runalg('otb:segmentationedison', -in, -filter, -filter.edison.spatialr, -filter.edison
```

See also

Segmentation (meanshift)

Description

<put algorithm description here>

Parameters

Input Image [**raster**] <put parameter description here>

Segmentation algorithm [**selection**] <put parameter description here>

Options:

- 0 — meanshift

Default: 0

Spatial radius [**number**] <put parameter description here>

Default: 5

Range radius [**number**] <put parameter description here>

Default: 15

Mode convergence threshold [**number**] <put parameter description here>

Default: 0.1

Maximum number of iterations [**number**] <put parameter description here>

Default: 100

Minimum region size [**number**] <put parameter description here>

Default: 100

Processing mode [**selection**] <put parameter description here>

Options:

- 0 — vector

Default: 0

Writing mode for the output vector file [**selection**] <put parameter description here>

Options:

- 0 — ulco
- 1 — ovw
- 2 — ulovw

- 3 — ulu

Default: *0*

Mask Image [raster] Optional.

<put parameter description here>

8-neighbor connectivity [boolean] <put parameter description here>

Default: *True*

Stitch polygons [boolean] <put parameter description here>

Default: *True*

Minimum object size [number] <put parameter description here>

Default: *1*

Simplify polygons [number] <put parameter description here>

Default: *0.1*

Layer name [string] <put parameter description here>

Default: *layer*

Geometry index field name [string] <put parameter description here>

Default: *DN*

Tiles size [number] <put parameter description here>

Default: *1024*

Starting geometry index [number] <put parameter description here>

Default: *1*

OGR options for layer creation [string] Optional.

<put parameter description here>

Default: *None*

Outputs

Output vector file [vector] <put output description here>

Console usage

```
processing.runalg('otb:segmentationmeanshift', -in, -filter, -filter.meanshift.spatialr, -filter.
```

See also

Segmentation (mprofiles)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Segmentation algorithm [selection] <put parameter description here>

Options:

- 0 — mprofiles

Default: 0

Profile Size [number] <put parameter description here>

Default: 5

Initial radius [number] <put parameter description here>

Default: 1

Radius step. [number] <put parameter description here>

Default: 1

Threshold of the final decision rule [number] <put parameter description here>

Default: 1

Processing mode [selection] <put parameter description here>

Options:

- 0 — vector

Default: 0

Writing mode for the output vector file [selection] <put parameter description here>

Options:

- 0 — ulco
- 1 — ovw
- 2 — ulovw
- 3 — ulu

Default: 0

Mask Image [raster] Optional.

<put parameter description here>

8-neighbor connectivity [boolean] <put parameter description here>

Default: *True*

Stitch polygons [boolean] <put parameter description here>

Default: *True*

Minimum object size [number] <put parameter description here>

Default: 1

Simplify polygons [number] <put parameter description here>

Default: 0.1

Layer name [string] <put parameter description here>

Default: *layer*

Geometry index field name [string] <put parameter description here>

Default: *DN*

Tiles size [number] <put parameter description here>

Default: *1024*

Starting geometry index [number] <put parameter description here>

Default: *1*

OGR options for layer creation [string] Optional.

<put parameter description here>

Default: *None*

Outputs

Output vector file [vector] <put output description here>

Console usage

```
processing.runalg('otb:segmentationmprofiles', -in, -filter, -filter.mprofiles.size, -filter.mpro
```

See also

Segmentation (watershed)

Description

<put algorithm description here>

Parameters

Input Image [raster] <put parameter description here>

Segmentation algorithm [selection] <put parameter description here>

Options:

- 0 — watershed

Default: *0*

Depth Threshold [number] <put parameter description here>

Default: *0.01*

Flood Level [number] <put parameter description here>

Default: *0.1*

Processing mode [selection] <put parameter description here>

Options:

- 0 — vector

Default: *0*

Writing mode for the output vector file [selection] <put parameter description here>

Options:

- 0 — ulco
- 1 — ovw
- 2 — ulovw
- 3 — ulu

Default: 0

Mask Image [raster] Optional.

<put parameter description here>

8-neighbor connectivity [boolean] <put parameter description here>

Default: *True*

Stitch polygons [boolean] <put parameter description here>

Default: *True*

Minimum object size [number] <put parameter description here>

Default: 1

Simplify polygons [number] <put parameter description here>

Default: 0.1

Layer name [string] <put parameter description here>

Default: *layer*

Geometry index field name [string] <put parameter description here>

Default: *DN*

Tiles size [number] <put parameter description here>

Default: 1024

Starting geometry index [number] <put parameter description here>

Default: 1

OGR options for layer creation [string] Optional.

<put parameter description here>

Default: *None*

Outputs

Output vector file [vector] <put output description here>

Console usage

```
processing.runalg('otb:segmentationwatershed', -in, -filter, -filter.watershed.threshold, -filter
```

See also

.

18.4.9 Estéreo

Stereo Framework

Description

<put algorithm description here>

Parameters

Input images list [**multipleinput: rasters**] <put parameter description here>

Couples list [**string**] Optional.

<put parameter description here>

Default: *None*

Image channel used for the block matching [**number**] <put parameter description here>

Default: *1*

Default elevation [**number**] <put parameter description here>

Default: *0*

Output resolution [**number**] <put parameter description here>

Default: *1*

NoData value [**number**] <put parameter description here>

Default: *-32768*

Method to fuse measures in each DSM cell [**selection**] <put parameter description here>

Options:

- 0 — max
- 1 — min
- 2 — mean
- 3 — acc

Default: *0*

Parameters estimation modes [**selection**] <put parameter description here>

Options:

- 0 — fit
- 1 — user

Default: *0*

Upper Left X [**number**] <put parameter description here>

Default: *0.0*

Upper Left Y [**number**] <put parameter description here>

Default: *0.0*

Size X [**number**] <put parameter description here>

Default: *0*

Size Y [number] <put parameter description here>

Default: 0

Pixel Size X [number] <put parameter description here>

Default: 0.0

Pixel Size Y [number] <put parameter description here>

Default: 0.0

Output Cartographic Map Projection [selection] <put parameter description here>

Options:

- 0 — utm
- 1 — lambert2
- 2 — lambert93
- 3 — wgs
- 4 — epsg

Default: 3

Zone number [number] <put parameter description here>

Default: 31

Northern Hemisphere [boolean] <put parameter description here>

Default: *True*

EPSG Code [number] <put parameter description here>

Default: 4326

Step of the deformation grid (in pixels) [number] <put parameter description here>

Default: 16

Sub-sampling rate for epipolar grid inversion [number] <put parameter description here>

Default: 10

Block-matching metric [selection] <put parameter description here>

Options:

- 0 — ssdmean
- 1 — ssd
- 2 — ncc
- 3 — lp

Default: 0

p value [number] <put parameter description here>

Default: 1

Radius of blocks for matching filter (in pixels) [number] <put parameter description here>

Default: 2

Minimum altitude offset (in meters) [number] <put parameter description here>

Default: -20

Maximum altitude offset (in meters) [number] <put parameter description here>

Default: *20*

Use bijection consistency in block matching strategy [boolean] <put parameter description here>

Default: *True*

Use median disparities filtering [boolean] <put parameter description here>

Default: *True*

Correlation metric threshold [number] <put parameter description here>

Default: *0.6*

Input left mask [raster] Optional.

<put parameter description here>

Input right mask [raster] Optional.

<put parameter description here>

Discard pixels with low local variance [number] <put parameter description here>

Default: *50*

Available RAM (Mb) [number] <put parameter description here>

Default: *128*

Outputs

Output DSM [raster] <put output description here>

Console usage

```
processing.runalg('otb:stereoframework', -input.il, -input.co, -input.channel, -elev.default, -out
```

See also

.

18.4.10 Vector

Concatenate

Description

<put algorithm description here>

Parameters

Input VectorDatas to concatenate [multipleinput: any vectors] <put parameter description here>

Outputs

Concatenated VectorData [**vector**] <put output description here>

Console usage

```
processing.runalg('otb:concatenate', -vd, -out)
```

See also

.

18.5 QGIS algorithm provider

QGIS algorithm provider implements various analysis and geoprocessing operations using mostly only QGIS API. So almost all algorithms from this provider will work “out of the box” without any additional configuration.

This provider incorporates fTools functionality, some algorithms from mmQGIS plugin and also adds its own algorithms.

.

18.5.1 Database

Import into PostGIS

Description

<put algorithm description here>

Parameters

Layer to import [**vector: any**] <put parameter description here>

Database (connection name) [**selection**] <put parameter description here>

Options:

- 0 — local

Default: 0

Schema (schema name) [**string**] <put parameter description here>

Default: *public*

Table to import to (leave blank to use layer name) [**string**] <put parameter description here>

Default: *(not set)*

Primary key field [**tablefield: any**] Optional.

<put parameter description here>

Geometry column [**string**] <put parameter description here>

Default: *geom*

Overwrite [boolean] <put parameter description here>

Default: *True*

Create spatial index [boolean] <put parameter description here>

Default: *True*

Convert field names to lowercase [boolean] <put parameter description here>

Default: *True*

Drop length constraints on character fields [boolean] <put parameter description here>

Default: *False*

Outputs

Console usage

```
processing.runalg('qgis:importintopostgis', input, database, schema, tablename, primary_key, geom
```

See also

PostGIS execute SQL

Description

<put algorithm description here>

Parameters

Database [string] <put parameter description here>

Default: *(not set)*

SQL query [string] <put parameter description here>

Default: *(not set)*

Outputs

Console usage

```
processing.runalg('qgis:postgisexecutesql', database, sql)
```

See also

.

18.5.2 Raster general

Set style for raster layer

Description

<put algorithm description here>

Parameters

Raster layer [**raster**] <put parameter description here>

Style file [**file**] <put parameter description here>

Outputs

Styled layer [**raster**] <put output description here>

Console usage

```
processing.runalg('qgis:setstyleforrasterlayer', input, style)
```

See also

.

18.5.3 Raster

Hypsometric curves

Description

Calculate hypsometric curves for features of polygon layer and save them as CSV file for further processing.

Parameters

DEM to analyze [**raster**] DEM to use for calculating altitudes.

Boundary layer [**vector: polygon**] Polygonal vector layer with boundaries of areas used to calculate hypsometric curves.

Step [**number**] Distanse between curves.

Default: *100.0*

Use % of area instead of absolute value [**boolean**] Write area percentage to “Area” field of the CSV file instead of absolute area value.

Default: *False*

Outputs

Output directory [directory] Directory where output will be saved. For each feature from input vector layer CSV file with area and altitude values will be created.

File name consists of prefix `hystogram_` followed by layer name and feature ID.

Console usage

```
processing.runalg('qgis:hypsometriccurves', input_dem, boundary_layer, step, use_percentage, output_directory)
```

See also

Raster layer statistics

Description

Calculates basic statistics of the raster layer.

Parameters

Input layer [raster] Raster to analyze.

Outputs

Statistics [html] Analysis results in HTML format.

Minimum value [number] Minimum cell value.

Maximum value [number] Maximum cell value.

Sum [number] Sum of all cells values.

Mean value [number] Mean cell value.

valid cells count [number] Number of cell with data.

No-data cells count [number] Number of NODATA cells.

Standard deviation [number] Standard deviation of cells values.

Console usage

```
processing.runalg('qgis:rasterlayerstatistics', input_raster, output_html_file)
```

See also

Zonal Statistics

Description

Calculates some statistics values for pixels of input raster inside certain zones, defined as polygon layer.

Following values calculated for each zone:

- minimum

- maximum
- sum
- count
- mean
- standard deviation
- number of unique values
- range
- variance

Parameters

Raster layer [**raster**] Raster to analyze.

Raster band [**number**] Number of raster band to analyze.

Default: *1*

Vector layer containing zones [**vector: polygon**] Layer with zones boundaries.

Output column prefix [**string**] Prefix for output fields.

Default: *_*

Load whole raster in memory [**boolean**] Determines if raster band will be loaded in memory (**True**) or readed by chunks (**False**). Useful only when disk IO or raster scanning inefficiencies are your limiting factor.

Default: *True*

Outputs

Output layer [**vector**] The resulting layer. Basically this is same layer as zones layer with new columns containing statistics added.

Console usage

```
processing.runalg('qgis:zonalstatistics', input_raster, raster_band, input_vector, column_prefix,
```

See also

.

18.5.4 Table

Frequency analysis

Description

<put algorithm description here>

Parameters

input [vector: any] <put parameter description here>

fields [string] <put parameter description here>

Default: *(not set)*

Outputs

output [table] <put output description here>

Console usage

```
processing.runalg('qgis:frequencyanalysis', input, fields, output)
```

See also

.

18.5.5 Vector analysis

Count points in polygon

Description

Counts the number of points present in each feature of a polygon layer.

Parameters

Polygons [vector: polygon] Polygons layer.

Points [vector: point] Points layer.

Count field name [string] The name of the attribute table column containing the points number.

Default: *NUMPOINTS*

Outputs

Result [vector] Resulting layer with the attribute table containing the new column of the points count.

Console usage

```
processing.runalg('qgis:countpointsinpolygon', polygons, points, field, output)
```

See also

Count points in polygon (weighted)

Description

Counts the number of points in each feature of a polygon layer and calculates the mean of the selected field for each feature of the polygon layer. These values will be added to the attribute table of the resulting polygon layer.

Parameters

Polygons [vector: polygon] Polygons layer.

Points [vector: point] Points layer.

Weight field [tablefield: any] Weight field of the points attribute table.

Count field name [string] Name of the column for the new weighted field.

Default: *NUMPOINTS*

Outputs

Result [vector] The resulting polygons layer.

Console usage

```
processing.runalg('qgis:countpointsinpolygonweighted', polygons, points, weight, field, output)
```

See also

Count unique points in polygon

Description

Counts the number of unique values of a points in a polygons layer. Creates a new polygons layer with an extra column in the attribute table containing the count of unique values for each feature.

Parameters

Polygons [vector: polygon] Polygons layer.

Points [vector: point] Points layer.

Class field [tablefield: any] Points layer column name of the unique value chosen.

Count field name [string] Column name containing the count of unique values in the resulting polygons layer.

Default: *NUMPOINTS*

Outputs

Result [vector] The resulting polygons layer.

Console usage

```
processing.runalg('qgis:countuniquepointsinpolygon', polygons, points, classfield, field, output)
```

See also

Distance matrix

Description

<put algorithm description here>

Parameters

Input point layer [vector: point] <put parameter description here>

Input unique ID field [tablefield: any] <put parameter description here>

Target point layer [vector: point] <put parameter description here>

Target unique ID field [tablefield: any] <put parameter description here>

Output matrix type [selection] <put parameter description here>

Options:

- 0 — Linear (N*k x 3) distance matrix
- 1 — Standard (N x T) distance matrix
- 2 — Summary distance matrix (mean, std. dev., min, max)

Default: 0

Use only the nearest (k) target points [number] <put parameter description here>

Default: 0

Outputs

Distance matrix [table] <put output description here>

Console usage

```
processing.runalg('qgis:distancematrix', input_layer, input_field, target_layer, target_field, ma
```

See also

Distance to nearest hub

Description

<put algorithm description here>

Parameters

Source points layer [vector: any] <put parameter description here>

Destination hubs layer [vector: any] <put parameter description here>

Hub layer name attribute [tablefield: any] <put parameter description here>

Output shape type [selection] <put parameter description here>

Options:

- 0 — Point
- 1 — Line to hub

Default: 0

Measurement unit [selection] <put parameter description here>

Options:

- 0 — Meters
- 1 — Feet
- 2 — Miles
- 3 — Kilometers
- 4 — Layer units

Default: 0

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:distancetonearesthub', points, hubs, field, geometry, unit, output)
```

See also

Generate points (pixel centroids) along line

Description

<put algorithm description here>

Parameters

Raster layer [raster] <put parameter description here>

Vector layer [vector: line] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:generatepointspixelcentroidsalongline', input_raster, input_vector, output)
```

See also

Generate points (pixel centroids) inside polygons

Description

<put algorithm description here>

Parameters

Raster layer [raster] <put parameter description here>

Vector layer [vector: polygon] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:generatepointspixelcentroidsinsidepolygons', input_raster, input_vector, output)
```

See also

Hub lines

Description

Creates hub and spoke diagrams with lines drawn from points on the `Spoke Point` layer to matching points in the `Hub Point` layer. Determination of which hub goes with each point is based on a match between the `Hub ID field` on the hub points and the `Spoke ID field` on the spoke points.

Parameters

Hub point layer [vector: any] <put parameter description here>

Hub ID field [tablefield: any] <put parameter description here>

Spoke point layer [vector: any] <put parameter description here>

Spoke ID field [tablefield: any] <put parameter description here>

Outputs

Output [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:hublines', hubs, hub_field, spokes, spoke_field, output)
```

See also

Mean coordinate(s)

Description

Calculates the mean of the coordinates of a layer starting from a field of the attribute table.

Parameters

Input layer [**vector: any**] <put parameter description here>

Weight field [**tablefield: numeric**] Optional.

Field to use if you want to perform a weighted mean.

Unique ID field [**tablefield: numeric**] Optional.

Unique field on which the calculation of the mean will be made.

Outputs

Result [**vector**] The resulting points layer.

Console usage

```
processing.runalg('qgis:meancoordinates', points, weight, uid, output)
```

See also

Nearest neighbour analysis

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Outputs

Result [**html**] <put output description here>

Observed mean distance [**number**] <put output description here>

Expected mean distance [**number**] <put output description here>

Nearest neighbour index [**number**] <put output description here>

Number of points [number] <put output description here>

Z-Score [number] <put output description here>

Console usage

```
processing.runalg('qgis:nearestneighbouranalysis', points, output)
```

See also

Sum line lengths

Description

<put algorithm description here>

Parameters

Lines [vector: line] <put parameter description here>

Polygons [vector: polygon] <put parameter description here>

Lines length field name [string] <put parameter description here>

Default: *LENGTH*

Lines count field name [string] <put parameter description here>

Default: *COUNT*

Outputs

Result [vector] <put output description here>

Console usage

```
processing.runalg('qgis:sumlinelengths', lines, polygons, len_field, count_field, output)
```

See also

.

18.5.6 Vector creation

Create grid

Description

Creates a grid.

Parameters

Grid type [selection] Grid type.

Options:

- 0 — Rectangle (line)
- 1 — Rectangle (polygon)
- 2 — Diamond (polygon)
- 3 — Hexagon (polygon)

Default: *0*

Width [number] Horizontal extent of the grid.

Default: *360.0*

Height [number] Vertical extent of the grid.

Default: *180.0*

Horizontal spacing [number] X-axes spacing between the lines.

Default: *10.0*

Vertical spacing [number] Y-axes spacing between the lines.

Default: *10.0*

Center X [number] X-coordinate of the grid center.

Default: *0.0*

Center Y [number] Y-coordinate of the grid center.

Default: *0.0*

Output CRS [crs] Coordinate reference system for grid.

Default: *EPSG:4326*

Outputs

Output [vector] The resulting grid layer (lines or polygons).

Console usage

```
processing.runalg('qgis:creategrid', type, width, height, hspacing, vspacing, centerx, centery, c
```

See also

Points layer from table

Description

Creates points layer from geometryless table with columns that contain point coordinates.

Parameters

Input layer [table] Input table

X field [tablefield: any] Table column containing the X coordinate.

Y field [tablefield: any] Table column containing the Y coordinate.

Target CRS [crs] Coordinate reference system to use for layer.

Default: *EPSG:4326*

Outputs

Output layer [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:pointslayerfromtable', input, xfield, yfield, target_crs, output)
```

See also

Points to path

Description

<put algorithm description here>

Parameters

Input point layer [vector: point] <put parameter description here>

Group field [tablefield: any] <put parameter description here>

Order field [tablefield: any] <put parameter description here>

Date format (if order field is DateTime) [string] Optional.

<put parameter description here>

Default: *(not set)*

Outputs

Paths [vector] <put output description here>

Directory [directory] <put output description here>

Console usage

```
processing.runalg('qgis:pointstopath', vector, group_field, order_field, date_format, output_line)
```

See also

Random points along line

Description

<put algorithm description here>

Parameters

Input layer [vector: line] <put parameter description here>

Number of points [number] <put parameter description here>

Default: 1

Minimum distance [number] <put parameter description here>

Default: 0.0

Outputs

Random points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randompointsalongline', vector, point_number, min_distance, output)
```

See also

Random points in extent

Description

<put algorithm description here>

Parameters

Input extent [extent] <put parameter description here>

Default: 0,1,0,1

Points number [number] <put parameter description here>

Default: 1

Minimum distance [number] <put parameter description here>

Default: 0.0

Outputs

Random points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randompointsinextent', extent, point_number, min_distance, output)
```

See also

Random points in layer bounds

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon] <put parameter description here>

Points number [number] <put parameter description here>

Default: 1

Minimum distance [number] <put parameter description here>

Default: 0.0

Outputs

Random points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randompointsinlayerbounds', vector, point_number, min_distance, output)
```

See also

Random points inside polygons (fixed)

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon] <put parameter description here>

Sampling strategy [selection] <put parameter description here>

Options:

- 0 — Points count
- 1 — Points density

Default: 0

Number or density of points [number] <put parameter description here>

Default: *1.0*

Minimum distance [number] <put parameter description here>

Default: *0.0*

Outputs

Random points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randompointsinsidepolygonsfixed', vector, strategy, value, min_distance, ...)
```

See also

Random points inside polygons (variable)

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon] <put parameter description here>

Sampling strategy [selection] <put parameter description here>

Options:

- 0 — Points count
- 1 — Points density

Default: *0*

Number field [tablefield: numeric] <put parameter description here>

Minimum distance [number] <put parameter description here>

Default: *0.0*

Outputs

Random points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randompointsinsidepolygonsvariable', vector, strategy, field, min_distance, ...)
```


See also

Regular points

Description

<put algorithm description here>

Parameters

Input extent [extent] <put parameter description here>

Default: *0,1,0,1*

Point spacing/count [number] <put parameter description here>

Default: *0.0001*

Initial inset from corner (LH side) [number] <put parameter description here>

Default: *0.0*

Apply random offset to point spacing [boolean] <put parameter description here>

Default: *False*

Use point spacing [boolean] <put parameter description here>

Default: *True*

Outputs

Regular points [vector] <put output description here>

Console usage

```
processing.runalg('qgis:regularpoints', extent, spacing, inset, randomize, is_spacing, output)
```

See also

Vector grid

Description

<put algorithm description here>

Parameters

Grid extent [extent] <put parameter description here>

Default: *0,1,0,1*

X spacing [number] <put parameter description here>

Default: *0.0001*

Y spacing [number] <put parameter description here>

Default: *0.0001*

Grid type [selection] <put parameter description here>

Options:

- 0 — Output grid as polygons
- 1 — Output grid as lines

Default: 0

Outputs

Grid [vector] <put output description here>

Console usage

```
processing.runalg('qgis:vectorgrid', extent, step_x, step_y, type, output)
```

See also

.

18.5.7 Vector general

Delete duplicate geometries

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:deleteduplicategeometries', input, output)
```

See also

Join atributes by location

Description

<put algorithm description here>

Parameters

Target vector layer [vector: any] <put parameter description here>

Join vector layer [vector: any] <put parameter description here>

Attribute summary [selection] <put parameter description here>

Options:

- 0 — Take attributes of the first located feature
- 1 — Take summary of intersecting features

Default: 0

Statistics for summary (comma separated) [string] <put parameter description here>

Default: *sum,mean,min,max,median*

Output table [selection] <put parameter description here>

Options:

- 0 — Only keep matching records
- 1 — Keep all records (including non-matching target records)

Default: 0

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:joinattributesbylocation', target, join, summary, stats, keep, output)
```

See also

Join attributes table

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Input layer 2 [table] <put parameter description here>

Table field [tablefield: any] <put parameter description here>

Table field 2 [tablefield: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:joinattributetable', input_layer, input_layer_2, table_field, table_fiel
```

See also

Merge vector layers

Description

<put algorithm description here>

Parameters

Input layer 1 [vector: any] <put parameter description here>

Input layer 2 [vector: any] <put parameter description here>

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:mergevectorlayers', layer1, layer2, output)
```

See also

Polygon from layer extent

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Calculate extent for each feature separately [boolean] <put parameter description here>

Default: *False*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:polygonfromlayerextent', input_layer, by_feature, output)
```

See also

Reproject layer

Description

Reprojects a vector layer in a different CRS.

Parameters

Input layer [vector: any] Layer to reproject.

Target CRS [crs] Destination coordinate reference system.

Default: *EPSG:4326*

Outputs

Reprojected layer [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:reprojectlayer', input, target_crs, output)
```

See also

Save selected features

Description

Saves the selected features as a new layer.

Parameters

Input layer [vector: any] Layer to process.

Outputs

Output layer with selected features [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:savesselectedfeatures', input_layer, output_layer)
```

See also

Set style for vector layer

Description

<put algorithm description here>

Parameters

Vector layer [vector: any] <put parameter description here>

Style file [file] <put parameter description here>

Outputs

Styled layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:setstyleforvectorlayer', input, style)
```

See also

Snap points to grid

Description

<put algorithm description here>

Parameters

Input Layer [vector: any] <put parameter description here>

Horizontal spacing [number] <put parameter description here>

Default: 0.1

Vertical spacing [number] <put parameter description here>

Default: 0.1

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:snappointstogrid', input, hspacing, vspacing, output)
```

See also

Split vector layer

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Unique ID field [tablefield: any] <put parameter description here>

Outputs

Output directory [directory] <put output description here>

Console usage

```
processing.runalg('qgis:splitvectorlayer', input, field, output)
```

See also

.

18.5.8 Vector geometry

Concave hull

Description

<put algorithm description here>

Parameters

Input point layer [vector: point] <put parameter description here>

Threshold (0-1, where 1 is equivalent with Convex Hull) [number] <put parameter description here>

Default: *0.3*

Allow holes [boolean] <put parameter description here>

Default: *True*

Split multipart geometry into singleparts geometries [boolean] <put parameter description here>

Default: *False*

Outputs

Concave hull [vector] <put output description here>

Console usage

```
processing.runalg('qgis:concavehull', input, alpha, holes, no_multigeometry, output)
```

See also

Convert geometry type

Description

Converts a geometry type to another one.

Parameters

Input layer [vector: any] Layer in input.

New geometry type [selection] Type of conversion to perform.

Options:

- 0 — Centroids
- 1 — Nodes
- 2 — Linestrings
- 3 — Multilinestrings
- 4 — Polygons

Default: 0

Outputs

Output [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:convertgeometrytype', input, type, output)
```

See also

Convex hull

Description

<put algorithm description here>

Parameters

Input layer [**vector: any**] <put parameter description here>

Field (optional, only used if creating convex hulls by classes) [**tablefield: any**]

Optional.

<put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — Create single minimum convex hull
- 1 — Create convex hulls based on field

Default: 0

Outputs

Convex hull [**vector**] <put output description here>

Console usage

```
processing.runalg('qgis:convexhull', input, field, method, output)
```

See also

Create points along lines

Description

<put algorithm description here>

Parameters

lines [**vector: any**] <put parameter description here>

distance [**number**] <put parameter description here>

Default: 1

startpoint [**number**] <put parameter description here>

Default: 0

endpoint [**number**] <put parameter description here>

Default: 0

Outputs

output [**vector**] <put output description here>

Console usage

```
processing.runalg('qgis:createpointsalonglines', lines, distance, startpoint, endpoint, output)
```

See also

Delaunay triangulation

Description

<put algorithm description here>

Parameters

Input layer [vector: point] <put parameter description here>

Outputs

Delaunay triangulation [vector] <put output description here>

Console usage

```
processing.runalg('qgis:delaunaytriangulation', input, output)
```

See also

Densify geometries given an interval

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon, line] <put parameter description here>

Interval between Vertices to add [number] <put parameter description here>

Default: 1.0

Outputs

Densified layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:densifygeometriesgivenaninterval', input, interval, output)
```

See also

Densify geometries

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon, line] <put parameter description here>

Vertices to add [number] <put parameter description here>

Default: 1

Outputs

Densified layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:densifygeometries', input, vertices, output)
```

See also

Dissolve

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon, line] <put parameter description here>

Dissolve all (do not use field) [boolean] <put parameter description here>

Default: *True*

Unique ID field [tablefield: any] Optional.

<put parameter description here>

Outputs

Dissolved [vector] <put output description here>

Console usage

```
processing.runalg('qgis:dissolve', input, dissolve_all, field, output)
```

See also

Eliminate sliver polygons

Description

<put algorithm description here>

Parameters

Input layer [**vector: polygon**] <put parameter description here>

Use current selection in input layer (works only if called from toolbox) [**boolean**]
<put parameter description here>

Default: *False*

Selection attribute [**tablefield: any**] <put parameter description here>

Comparison [**selection**] <put parameter description here>

Options:

- 0 — ==
- 1 — !=
- 2 — >
- 3 — >=
- 4 — <
- 5 — <=
- 6 — begins with
- 7 — contains

Default: *0*

Value [**string**] <put parameter description here>

Default: *0*

Merge selection with the neighbouring polygon with the [**selection**] <put parameter description here>

Options:

- 0 — Largest area
- 1 — Smallest Area
- 2 — Largest common boundary

Default: *0*

Outputs

Cleaned layer [**vector**] <put output description here>

Console usage

```
processing.runalg('qgis:eliminatesliverpolygons', input, keepselection, attribute, comparison, co
```

See also

Explode lines

Description

<put algorithm description here>

Parameters

Input layer [vector: line] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:explodelines', input, output)
```

See also

Extract nodes

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon, line] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:extractnodes', input, output)
```

See also

Fill holes

Description

<put algorithm description here>

Parameters

Polygons [vector: any] <put parameter description here>

Max area [number] <put parameter description here>

Default: *100000*

Outputs

Results [vector] <put output description here>

Console usage

```
processing.runalg('qgis:fillholes', polygons, max_area, results)
```

See also

Fixed distance buffer

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Distance [number] <put parameter description here>

Default: *10.0*

Segments [number] <put parameter description here>

Default: *5*

Dissolve result [boolean] <put parameter description here>

Default: *False*

Outputs

Buffer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:fixeddistancebuffer', input, distance, segments, dissolve, output)
```

See also

Keep n biggest parts

Description

<put algorithm description here>

Parameters

Polygons [vector: polygon] <put parameter description here>

To keep [number] <put parameter description here>

Default: 1

Outputs

Results [vector] <put output description here>

Console usage

```
processing.runalg('qgis:keepnbiggestparts', polygons, to_keep, results)
```

See also

Lines to polygons

Description

<put algorithm description here>

Parameters

Input layer [vector: line] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:linestopolygons', input, output)
```

See also

Multipart to singleparts

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:multiparttosingleparts', input, output)
```

See also

Points displacement

Description

Moves overlapped points at small distance, that they all become visible. The result is very similar to the output of the “Point displacement” renderer but it is permanent.

Parameters

Input layer [vector: point] Layer with overlapped points.

Displacement distance [number] Desired displacement distance **NOTE:** displacement distance should be in same units as layer.

Default: *0.00015*

Horizontal distribution for two point case [boolean] Controls distribution direction in case of two overlapped points. If *True* points will be distributed horizontally, otherwise they will be distributed vertically.

Default: *True*

Outputs

Output layer [vector] The resulting layer with shifted overlapped points.

Console usage

```
processing.runalg('qgis:pointsdisplacement', input_layer, distance, horizontal, output_layer)
```


See also

Polygon centroids

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:polygoncentroids', input_layer, output_layer)
```

See also

Polygonize

Description

<put algorithm description here>

Parameters

Input layer [vector: line] <put parameter description here>

Keep table structure of line layer [boolean] <put parameter description here>

Default: *False*

Create geometry columns [boolean] <put parameter description here>

Default: *True*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:polygonize', input, fields, geometry, output)
```

See also

Polygons to lines

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:polygonstolines', input, output)
```

See also

Simplify geometries

Description

<put algorithm description here>

Parameters

Input layer [vector: polygon, line] <put parameter description here>

Tolerance [number] <put parameter description here>

Default: *1.0*

Outputs

Simplified layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:simplifygeometries', input, tolerance, output)
```

See also

Singleparts to multipart

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Unique ID field [tablefield: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:singlepartstomultipart', input, field, output)
```

See also

Variable distance buffer

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Distance field [tablefield: any] <put parameter description here>

Segments [number] <put parameter description here>

Default: 5

Dissolve result [boolean] <put parameter description here>

Default: *False*

Outputs

Buffer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:variabledistancebuffer', input, field, segments, dissolve, output)
```

See also

Voronoi polygons

Description

<put algorithm description here>

Parameters

Input layer [vector: point] <put parameter description here>

Buffer region [number] <put parameter description here>

Default: *0.0*

Outputs

Voronoi polygons [vector] <put output description here>

Console usage

```
processing.runalg('qgis:voronoipolygons', input, buffer, output)
```

See also

.

18.5.9 Vector overlay

Clip

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Clip layer [vector: any] <put parameter description here>

Outputs

Clipped [vector] <put output description here>

Console usage

```
processing.runalg('qgis:clip', input, overlay, output)
```

See also

Difference

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Difference layer [vector: any] <put parameter description here>

Outputs

Difference [vector] <put output description here>

Console usage

```
processing.runalg('qgis:difference', input, overlay, output)
```

See also

Intersection

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Intersect layer [vector: any] <put parameter description here>

Outputs

Intersection [vector] <put output description here>

Console usage

```
processing.runalg('qgis:intersection', input, input2, output)
```

See also

Line intersections

Description

<put algorithm description here>

Parameters

Input layer [vector: line] <put parameter description here>

Intersect layer [vector: line] <put parameter description here>

Input unique ID field [tablefield: any] <put parameter description here>

Intersect unique ID field [tablefield: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:lineintersections', input_a, input_b, field_a, field_b, output)
```

See also

Symmetrical difference

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Difference layer [vector: any] <put parameter description here>

Outputs

Symmetrical difference [vector] <put output description here>

Console usage

```
processing.runalg('qgis:symmetricaldifference', input, overlay, output)
```

See also

Union

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Input layer 2 [vector: any] <put parameter description here>

Outputs

Union [vector] <put output description here>

Console usage

```
processing.runalg('qgis:union', input, input2, output)
```

See also

lignoraratualizaçãol

18.5.10 Selecionar vetor

Extract by attribute

Description

<put algorithm description here>

Parameters

Input Layer [vector: any] <put parameter description here>

Selection attribute [tablefield: any] <put parameter description here>

Operator [selection] <put parameter description here>

Options:

- 0 — =
- 1 — !=
- 2 — >
- 3 — >=
- 4 — <
- 5 — <=
- 6 — begins with

- 7 — contains

Default: 0

Value [string] <put parameter description here>

Default: *(not set)*

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:extractbyattribute', input, field, operator, value, output)
```

See also

Extract by location

Description

<put algorithm description here>

Parameters

Layer to select from [vector: any] <put parameter description here>

Additional layer (intersection layer) [vector: any] <put parameter description here>

Include input features that touch the selection features [boolean] <put parameter description here>

Default: *False*

Include input features that overlap/cross the selection features [boolean] <put parameter description here>

Default: *False*

Include input features completely within the selection features [boolean] <put parameter description here>

Default: *False*

Outputs

Selection [vector] <put output description here>

Console usage

```
processing.runalg('qgis:extractbylocation', input, intersect, touches, overlaps, within, output)
```


See also

Random extract

Description

<put algorithm description here>

Parameters

Input layer [**vector: any**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — Number of selected features
- 1 — Percentage of selected features

Default: 0

Number/percentage of selected features [**number**] <put parameter description here>

Default: 10

Outputs

Selection [**vector**] <put output description here>

Console usage

```
processing.runalg('qgis:randomextract', input, method, number, output)
```

See also

Random extract within subsets

Description

<put algorithm description here>

Parameters

Input layer [**vector: any**] <put parameter description here>

ID Field [**tablefield: any**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — Number of selected features
- 1 — Percentage of selected features

Default: 0

Number/percentage of selected features [number] <put parameter description here>

Default: *10*

Outputs

Selection [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randomextractwithinsubsets', input, field, method, number, output)
```

See also

Random selection

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — Number of selected features
- 1 — Percentage of selected features

Default: *0*

Number/percentage of selected features [number] <put parameter description here>

Default: *10*

Outputs

Selection [vector] <put output description here>

Console usage

```
processing.runalg('qgis:randomselection', input, method, number)
```

See also

Random selection within subsets

Description

<put algorithm description here>

Parameters

Input layer [**vector: any**] <put parameter description here>

ID Field [**tablefield: any**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — Number of selected features
- 1 — Percentage of selected features

Default: 0

Number/percentage of selected features [**number**] <put parameter description here>

Default: 10

Outputs

Selection [**vector**] <put output description here>

Console usage

```
processing.runalg('qgis:randomselectionwithinsubsets', input, field, method, number)
```

See also

Select by attribute

Description

Selects and saves as new layer all features from input layer that satisfy condition.

NOTE: algorithm is case-sensitive (“qgis” is different from “Qgis” and “QGIS”)

Parameters

Input Layer [**vector: any**] Layer to process.

Selection attribute [**tablefield: any**] Field on which perform the selection.

Operator [**selection**] Comparison operator.

Options:

- 0 — =
- 1 — !=
- 2 — >
- 3 — >=
- 4 — <
- 5 — <=
- 6 — begins with
- 7 — contains

Default: 0

Value [string] Value to compare.

Default: *(not set)*

Outputs

Output [vector] The resulting layer.

Console usage

```
processing.runalg('qgis:selectbyattribute', input, field, operator, value, output)
```

See also

Select by expression

Description

<put algorithm description here>

Parameters

Input Layer [vector: any] <put parameter description here>

Expression [string] <put parameter description here>

Default: *(not set)*

Modify current selection by [selection] <put parameter description here>

Options:

- 0 — creating new selection
- 1 — adding to current selection
- 2 — removing from current selection

Default: 0

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:selectbyexpression', layername, expression, method)
```

See also

Select by location

Description

<put algorithm description here>

Parameters

Layer to select from [vector: any] <put parameter description here>

Additional layer (intersection layer) [vector: any] <put parameter description here>

Include input features that touch the selection features [boolean] <put parameter description here>

Default: *False*

Include input features that overlap/cross the selection features [boolean] <put parameter description here>

Default: *False*

Include input features completely within the selection features [boolean] <put parameter description here>

Default: *False*

Modify current selection by [selection] <put parameter description here>

Options:

- 0 — creating new selection
- 1 — adding to current selection
- 2 — removing from current selection

Default: *0*

Outputs

Selection [vector] <put output description here>

Console usage

```
processing.runalg('qgis:selectbylocation', input, intersect, touches, overlaps, within, method)
```

See also

[lignoraratualização](#)

18.5.11 Vetor da tabela

Add autoincremental field

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:addautoincrementalfield', input, output)
```

See also

Add field to attributes table

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Field name [string] <put parameter description here>

Default: *(not set)*

Field type [selection] <put parameter description here>

Options:

- 0 — Integer
- 1 — Float
- 2 — String

Default: *0*

Field length [number] <put parameter description here>

Default: *10*

Field precision [number] <put parameter description here>

Default: *0*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:addfieldtoattributetable', input_layer, field_name, field_type, field_length)
```

See also

Advanced Python field calculator

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Result field name [string] <put parameter description here>

Default: *NewField*

Field type [selection] <put parameter description here>

Options:

- 0 — Integer
- 1 — Float
- 2 — String

Default: *0*

Field length [number] <put parameter description here>

Default: *10*

Field precision [number] <put parameter description here>

Default: *0*

Global expression [string] Optional.

<put parameter description here>

Default: *(not set)*

Formula [string] <put parameter description here>

Default: *value =*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:advancedpythonfieldcalculator', input_layer, field_name, field_type, field
```

See also

Basic statistics for numeric fields

Description

<put algorithm description here>

Parameters

Input vector layer [**vector: any**] <put parameter description here>

Field to calculate statistics on [**tablefield: numeric**] <put parameter description here>

Outputs

Statistics for numeric field [**html**] <put output description here>

Coefficient of Variation [**number**] <put output description here>

Minimum value [**number**] <put output description here>

Maximum value [**number**] <put output description here>

Sum [**number**] <put output description here>

Mean value [**number**] <put output description here>

Count [**number**] <put output description here>

Range [**number**] <put output description here>

Median [**number**] <put output description here>

Number of unique values [**number**] <put output description here>

Standard deviation [**number**] <put output description here>

Console usage

```
processing.runalg('qgis:basicstatisticsfornumericfields', input_layer, field_name, output_html_fi
```

See also

Basic statistics for text fields

Description

<put algorithm description here>

Parameters

Input vector layer [vector: any] <put parameter description here>

Field to calculate statistics on [tablefield: string] <put parameter description here>

Outputs

Statistics for text field [html] <put output description here>

Minimum length [number] <put output description here>

Maximum length [number] <put output description here>

Mean length [number] <put output description here>

Count [number] <put output description here>

Number of empty values [number] <put output description here>

Number of non-empty values [number] <put output description here>

Number of unique values [number] <put output description here>

Console usage

```
processing.runalg('qgis:basicstatisticsfortextfields', input_layer, field_name, output_html_file)
```

See also

Create equivalent numerical field

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Class field [tablefield: any] <put parameter description here>

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:createequivalentnumericalfield', input, field, output)
```

See also

Delete column

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Field to delete [tablefield: any] <put parameter description here>

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:deletecolumn', input, column, output)
```

See also

Export/Add geometry columns

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Calculate using [selection] <put parameter description here>

Options:

- 0 — Layer CRS
- 1 — Project CRS
- 2 — Ellipsoidal

Default: 0

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:exportaddgeometrycolumns', input, calc_method, output)
```

See also

Field calculator

Description

<put algorithm description here>

Parameters

Input layer [vector: any] <put parameter description here>

Result field name [string] <put parameter description here>

Default: *(not set)*

Field type [selection] <put parameter description here>

Options:

- 0 — Float
- 1 — Integer
- 2 — String
- 3 — Date

Default: *0*

Field length [number] <put parameter description here>

Default: *10*

Field precision [number] <put parameter description here>

Default: *3*

Create new field [boolean] <put parameter description here>

Default: *True*

Formula [string] <put parameter description here>

Default: *(not set)*

Outputs

Output layer [vector] <put output description here>

Console usage

```
processing.runalg('qgis:fieldcalculator', input_layer, field_name, field_type, field_length, field...
```

See also

List unique values

Description

Lists unique values of an attribute table field and counts their number.

Parameters

Input layer [vector: any] Layer to analyze.

Target field [tablefield: any] Field to analyze.

Outputs

Unique values [html] Analysis results in HTML format.

Total unique values [number] Total number of unique values in given field.

Unique values [string] List of all unique values in given field.

Console usage

```
processing.runalg('qgis:listuniquevalues', input_layer, field_name, output)
```

See also

Number of unique values in classes

Description

<put algorithm description here>

Parameters

input [vector: any] <put parameter description here>

class field [tablefield: any] <put parameter description here>

value field [tablefield: any] <put parameter description here>

Outputs

output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:numberofuniquevaluesinclasses', input, class_field, value_field, output)
```

See also

Statistics by categories

Description

<put algorithm description here>

Parameters

Input vector layer [vector: any] <put parameter description here>

Field to calculate statistics on [tablefield: numeric] <put parameter description here>

Field with categories [tablefield: any] <put parameter description here>

Outputs

Statistics [table] <put output description here>

Console usage

```
processing.runalg('qgis:statisticsbycategories', input_layer, values_field_name, categories_field)
```

See also

Text to float

Description

<put algorithm description here>

Parameters

Input Layer [vector: any] <put parameter description here>

Text attribute to convert to float [tablefield: string] <put parameter description here>

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('qgis:texttofloat', input, field, output)
```

See also

.

18.6 Provedor de algoritmo R

R também chamado de GNU S, é uma linguagem fortemente funcional e ambiente para explorar estatisticamente conjuntos de dados, fazem muitas representações gráficas de dados a partir de conjuntos de dados personalizados

Nota: Por favor, lembre-se que o processamento contém apenas scripts R, então você precisa instalar R sozinho e configurar corretamente o Processamento.

18.6.1 Estatísticas básicas

Tabela de frequencia

Descricao

<colocar algoritmo descrição aqui>

Parametros

Camada [vetor: any] <colocar parametro de descrição aqui>

Campo [tablefield: any] <colocar parametro de descrição aqui>

Saidas

Saida do R Console [html] <colocar a descrição da saída aqui>

Utilizacao do console

```
processing.runalg('r:frequencytable', layer, field, r_console_output)
```

Veja tambem

Teste Kolmogrov-Smirnov

Descrição

<ponha a descrição do algoritmo aqui>

Parâmetros

Camada [vetor: qualquer] <ponha o parâmetro do algoritmo aqui>

Campo [tablefield: qualquer] <ponha o parâmetro do algoritmo aqui>

Saídas

Arquivo de Saída da Linha de comandos R [html] <ponha a descrição de saída aqui>

Uso da Linha de Comandos

```
processing.runalg('r:kolmogrovsmirnovtest', layer, field, r_console_output)
```

Veja também

Estatísticas resumidas

Descricao

<coloque a descrição do algoritmo aqui>

Parametros

Camada [vector: any] <coloque a descrição do parametro aqui>

Campo [tablefield: any] <coloque a descrição do parametro aqui>

Saida

Saida do R Console [html] <coloque a descrição da saida aqui>

Uso do console

```
processing.runalg('r:summarystatistics', layer, field, r_console_output)
```

Veja também

.

18.6.2 Home range

Characteristic hull method

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Field [tablefield: any] <put parameter description here>

Outputs

Home_ranges [vector] <put output description here>

Console usage

```
processing.runalg('r:characteristichullmethod', layer, field, home_ranges)
```

See also

Kernel h ref

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Field [tablefield: any] <put parameter description here>

Grid [number] <put parameter description here>

Default: *10.0*

Percentage [number] <put parameter description here>

Default: *10.0*

Folder [directory] Optional.

<put parameter description here>

Outputs

Home_ranges [vector] <put output description here>

Console usage

```
processing.runalg('r:kernelhref', layer, field, grid, percentage, folder, home_ranges)
```

See also

Minimum convex polygon

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Percentage [number] <put parameter description here>

Default: *10.0*

Field [tablefield: any] <put parameter description here>

Outputs

Home_ranges [vector] <put output description here>

Console usage

```
processing.runalg('r:minimumconvexpolygon', layer, percentage, field, home_ranges)
```

See also

Single-linkage cluster analysis

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Field [tablefield: any] <put parameter description here>

Percentage [number] <put parameter description here>

Default: *10.0*

Outputs

R Plots [html] <put output description here>

Home_ranges [vector] <put output description here>

Console usage

```
processing.runalg('r:singlelinkageclusteranalysis', layer, field, percentage, rplots, home_ranges)
```

See also

.

18.6.3 Point pattern

F function

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Nsim [number] <put parameter description here>

Default: *10.0*

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:ffunction', layer, nsim, rplots)
```

See also

G function

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Nsim [number] <put parameter description here>

Default: *10.0*

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:gfunction', layer, nsim, rplots)
```

See also

Monte-Carlo spatial randomness

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Simulations [number] <put parameter description here>

Default: *100.0*

Optional plot name [string] <put parameter description here>

Default: *(not set)*

Outputs

R Plots [html] <put output description here>

R Console Output [html] <put output description here>

Console usage

```
processing.runalg('r:montecarlospatialrandomness', layer, simulations, optional_plot_name, rplots)
```

See also

Quadrat analysis

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Outputs

R Plots [html] <put output description here>

R Console Output [html] <put output description here>

Console usage

```
processing.runalg('r:quadratanalysis', layer, rplots, r_console_output)
```

See also

Gríde de amostragem aleatória

Descrição

<coloque o algoritmo de descrição aqui>

Parâmetros

Camada [vetor: any] <coloque o parâmetro de descrição aqui>

Tamanho [numero] <coloque o parâmetro de descrição aqui>

Padrao: *10.0*

Saídas

Saída [vetor] <coloque a descrição de saída aqui>

Usando o console

```
processing.runalg('r:randomsamplinggrid', layer, size, output)
```

Veja também

Regular sampling grid

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Size [number] <put parameter description here>

Default: *10.0*

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('r:regularsamplinggrid', layer, size, output)
```

See also

Relative distribution (distance covariate)

Description

<put algorithm description here>

Parameters

Layer [**vector: any**] <put parameter description here>

Covariate [**vector: any**] <put parameter description here>

Covariate name [**string**] <put parameter description here>

Default: *mandatory_covariate_name_(no_spaces)*

x label [**string**] <put parameter description here>

Default: *(not set)*

Plot name [**string**] <put parameter description here>

Default: *(not set)*

Legend position [**string**] <put parameter description here>

Default: *float*

Outputs

R Plots [**html**] <put output description here>

Console usage

```
processing.runalg('r:relativedistributiondistancecovariate', layer, covariate, covariate_name, x_
```

See also

Relative distribution (raster covariate)

Description

<put algorithm description here>

Parameters

points [**vector: any**] <put parameter description here>

covariate [**raster**] <put parameter description here>

covariate name [**string**] <put parameter description here>

Default: *mandatory_covariate_name_(no_spaces)*

x label [**string**] <put parameter description here>

Default: *(not set)*

plot name [**string**] <put parameter description here>

Default: *(not set)*

legend position [**string**] <put parameter description here>

Default: *float*

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:relativedistributionrastercovariate', points, covariate, covariate_name, x_l
```

See also

Ripley - Rasson spatial domain

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('r:ripleyrassonspatialdomain', layer, output)
```

See also

.

18.6.4 Raster processing

Advanced raster histogram

Description

<put algorithm description here>

Parameters

Layer [raster] <put parameter description here>

Dens or Hist [string] <put parameter description here>

Default: *Hist*

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:advancedrasterhistogram', layer, dens_or_hist, rplots)
```

See also

Raster histogram

Description

<put algorithm description here>

Parameters

Layer [raster] <put parameter description here>

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:rasterhistogram', layer, rplots)
```

See also

.

18.6.5 Vector processing

Histogram

Description

<put algorithm description here>

Parameters

Layer [vector: any] <put parameter description here>

Field [tablefield: any] <put parameter description here>

Outputs

R Plots [html] <put output description here>

Console usage

```
processing.runalg('r:histogram', layer, field, rplots)
```

See also

.

18.7 SAGA algorithm provider

SAGA (System for Automated Geoscientific Analyses) is a free, hybrid, cross-platform GIS software. SAGA provides many geoscientific methods which are bundled in so-called module libraries.

Nota: Please remember that Processing contains only the interface description, so you need to install SAGA by yourself and configure Processing properly.

18.7.1 Geostatistics

Directional statistics for single grid

Description

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Points [**vector: any**] Optional.

<put parameter description here>

Direction [**Degree**] [**number**] <put parameter description here>

Default: *0.0*

Tolerance [**Degree**] [**number**] <put parameter description here>

Default: *0.0*

Maximum Distance [**Cells**] [**number**] <put parameter description here>

Default: *0*

Distance Weighting [**selection**] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *1.0*

Outputs

Arithmetic Mean [raster] <put output description here>

Difference from Arithmetic Mean [raster] <put output description here>

Minimum [raster] <put output description here>

Maximum [raster] <put output description here>

Range [raster] <put output description here>

Variance [raster] <put output description here>

Standard Deviation [raster] <put output description here>

Mean less Standard Deviation [raster] <put output description here>

Mean plus Standard Deviation [raster] <put output description here>

Deviation from Arithmetic Mean [raster] <put output description here>

Percentile [raster] <put output description here>

Directional Statistics for Points [vector] <put output description here>

Console usage

```
processing.runalg('saga:directionalstatisticsforsinglegrid', grid, points, direction, tolerance, n)
```

See also

Fast representativeness

Description

<put algorithm description here>

Parameters

Input [raster] <put parameter description here>

Level of Generalisation [number] <put parameter description here>

Default: *16*

Outputs

Output [raster] <put output description here>

Output Lod [raster] <put output description here>

Output Seeds [raster] <put output description here>

Console usage

```
processing.runalg('saga:fastrepresentativeness', input, lod, result, result_lod, seeds)
```

See also

Geographically weighted multiple regression (points/grids)

Description

<put algorithm description here>

Parameters

Predictors [multipleinput: rasters] <put parameter description here>

Output of Regression Parameters [boolean] <put parameter description here>

Default: *True*

Points [vector: point] <put parameter description here>

Dependent Variable [tablefield: any] <put parameter description here>

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *1.0*

Search Range [selection] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [number] <put parameter description here>

Default: 100

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of observations
- 1 — [1] all points

Default: 0

Maximum Number of Observations [number] <put parameter description here>

Default: 10

Minimum Number of Observations [number] <put parameter description here>

Default: 4

Outputs

Regression [raster] <put output description here>

Coefficient of Determination [raster] <put output description here>

Regression Parameters [raster] <put output description here>

Residuals [vector] <put output description here>

Console usage

```
processing.runalg('saga:geographicallyweightedmultipleregressionpointsgrids', predictors, parameter
```

See also

Geographically weighted multiple regression (points)

Description

<put algorithm description here>

Parameters

Points [vector: any] <put parameter description here>

Dependent Variable [tablefield: any] <put parameter description here>

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [number] <put parameter description here>

Default: 1

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: 1.0

Search Range [selection] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [number] <put parameter description here>

Default: 100

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of observations
- 1 — [1] all points

Default: 0

Maximum Number of Observations [number] <put parameter description here>

Default: 10

Minimum Number of Observations [number] <put parameter description here>

Default: 4

Outputs

Regression [vector] <put output description here>

Console usage

```
processing.runalg('saga:geographicallyweightedmultipleregressionpoints', points, dependent, distanc
```

See also

Geographically weighted multiple regression

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Dependent Variable [tablefield: any] <put parameter description here>

Target Grids [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [number] <put parameter description here>

Default: 1

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: 1

Search Range [selection] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [number] <put parameter description here>

Default: 100

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of observations
- 1 — [1] all points

Default: 0

Maximum Number of Observations [number] <put parameter description here>

Default: 10

Minimum Number of Observations [number] <put parameter description here>

Default: 4

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Quality [raster] <put output description here>

Intercept [raster] <put output description here>

Quality [raster] <put output description here>

Intercept [raster] <put output description here>

Console usage

```
processing.runalg('saga:geographicallyweightedmultipleregression', points, dependent, target, dist
```

See also

Geographically weighted regression (points/grid)

Description

<put algorithm description here>

Parameters

Predictor [raster] <put parameter description here>

Points [vector: point] <put parameter description here>

Dependent Variable [tablefield: any] <put parameter description here>

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [number] <put parameter description here>

Default: 1

Inverse Distance Offset [boolean] <put parameter description here>

Default: True

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: 1.0

Search Range [selection] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [number] <put parameter description here>

Default: 0

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of observations
- 1 — [1] all points

Default: 0

Maximum Number of Observations [number] <put parameter description here>

Default: 10

Minimum Number of Observations [number] <put parameter description here>

Default: 4

Outputs

Regression [raster] <put output description here>

Coefficient of Determination [raster] <put output description here>

Intercept [raster] <put output description here>

Slope [raster] <put output description here>

Residuals [vector] <put output description here>

Console usage

```
processing.runalg('saga:geographicallyweightedregressionpointsgrid', predictor, points, dependent,
```

See also

Geographically weighted regression

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Dependent Variable [**tablefield: any**] <put parameter description here>

Predictor [**tablefield: any**] <put parameter description here>

Target Grids [**selection**] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Distance Weighting [**selection**] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [**number**] <put parameter description here>

Default: 0

Inverse Distance Offset [**boolean**] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [**number**] <put parameter description here>

Default: 0.0

Search Range [**selection**] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [**number**] <put parameter description here>

Default: 100

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of observations
- 1 — [1] all points

Default: 0

Maximum Number of Observations [number] <put parameter description here>

Default: 10

Minimum Number of Observations [number] <put parameter description here>

Default: 4

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Quality [raster] <put output description here>

Intercept [raster] <put output description here>

Slope [raster] <put output description here>

Console usage

```
processing.runalg('saga:geographicallyweightedregression', points, dependent, predictor, target, 0, 1, 0, 1, 100.0)
```

See also

Global moran's i for grids

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Case of contiguity [selection] <put parameter description here>

Options:

- 0 — [0] Rook
 - 1 — [1] Queen
- Default: 0

Outputs

Result [table] <put output description here>

Console usage

```
processing.runalg('saga:globalmoransiforgrids', grid, contiguity, result)
```

See also

Minimum distance analysis

Description

Performs a complete distance analysis of a point layer:

- minimum distance of points
- maximum distance of points
- average distance of all the points
- standard deviation of the distance
- duplicated points

Parameters

Points [vector: point] Layer to analyze.

Outputs

Minimum Distance Analysis [table] The resulting table.

Console usage

```
processing.runalg('saga:minimumdistanceanalysis', points, table)
```

See also

Multi-band variation

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Radius [**Cells**] [**number**] <put parameter description here>

Default: *1*

Distance Weighting [**selection**] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [**number**] <put parameter description here>

Default: *1*

Inverse Distance Offset [**boolean**] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [**number**] <put parameter description here>

Default: *1.0*

Outputs

Mean Distance [**raster**] <put output description here>

Standard Deviation [**raster**] <put output description here>

Distance [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:multibandvariation', bands, radius, distance_weighting_weighting, distance_offset)
```

See also

Multiple regression analysis (grid/grids)

Description

<put algorithm description here>

Parameters

Dependent [**raster**] <put parameter description here>

Grids [**multipleinput: rasters**] <put parameter description here>

Grid Interpolation [**selection**] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Include X Coordinate [boolean] <put parameter description here>

Default: *True*

Include Y Coordinate [boolean] <put parameter description here>

Default: *True*

Method [selection] <put parameter description here>

Options:

- 0 — [0] include all
- 1 — [1] forward
- 2 — [2] backward
- 3 — [3] stepwise

Default: 0

P in [number] <put parameter description here>

Default: 5

P out [number] <put parameter description here>

Default: 5

Outputs

Regression [raster] <put output description here>

Residuals [raster] <put output description here>

Details: Coefficients [table] <put output description here>

Details: Model [table] <put output description here>

Details: Steps [table] <put output description here>

Console usage

```
processing.runalg('saga:multipleregressionanalysisgridgrids', dependent, grids, interpol, coord_x,
```

See also

Multiple regression analysis (points/grids)

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Shapes [**vector: any**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Grid Interpolation [**selection**] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Include X Coordinate [**boolean**] <put parameter description here>

Default: *True*

Include Y Coordinate [**boolean**] <put parameter description here>

Default: *True*

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] include all
- 1 — [1] forward
- 2 — [2] backward
- 3 — [3] stepwise

Default: 0

P in [**number**] <put parameter description here>

Default: 5

P out [**number**] <put parameter description here>

Default: 5

Outputs

Details: **Coefficients** [**table**] <put output description here>

Details: **Model** [**table**] <put output description here>

Details: **Steps** [**table**] <put output description here>

Residuals [**vector**] <put output description here>

Regression [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:multipleregressionanalysispointsgrids', grids, shapes, attribute, interpo
```

See also

Polynomial regression

Description

<put algorithm description here>

Parameters

Points [vector: any] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Polynom [selection] <put parameter description here>

Options:

- 0 — [0] simple planar surface
- 1 — [1] bi-linear saddle
- 2 — [2] quadratic surface
- 3 — [3] cubic surface
- 4 — [4] user defined

Default: 0

Maximum X Order [number] <put parameter description here>

Default: 4

Maximum Y Order [number] <put parameter description here>

Default: 4

Maximum Total Order [number] <put parameter description here>

Default: 4

Trend Surface [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Residuals [vector] <put output description here>

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:polynomialregression', points, attribute, polynom, xorder, yorder, torder)
```

See also

Radius of variance (grid)

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Standard Deviation [number] <put parameter description here>

Default: 1.0

Maximum Search Radius (cells) [number] <put parameter description here>

Default: 20

Type of Output [selection] <put parameter description here>

Options:

- 0 — [0] Cells
- 1 — [1] Map Units

Default: 0

Outputs

Variance Radius [raster] <put output description here>

Console usage

```
processing.runalg('saga:radiusofvariancegrid', input, variance, radius, output, result)
```

See also

Regression analysis

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Shapes [vector: any] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Grid Interpolation [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Regression Function [selection] <put parameter description here>

Options:

- 0 — [0] $Y = a + b * X$ (linear)
- 1 — [1] $Y = a + b / X$
- 2 — [2] $Y = a / (b - X)$
- 3 — [3] $Y = a * X^b$ (power)
- 4 — [4] $Y = a e^{(b * X)}$ (exponential)
- 5 — [5] $Y = a + b * \ln(X)$ (logarithmic)

Default: 0

Outputs

Regression [raster] <put output description here>

Residuals [vector] <put output description here>

Console usage

```
processing.runalg('saga:regressionanalysis', grid, shapes, attribute, interpol, method, regression)
```

See also

Representativeness

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Radius (Cells) [number] <put parameter description here>

Default: *10*

Exponent [number] <put parameter description here>

Default: *1*

Outputs

Representativeness [raster] <put output description here>

Console usage

```
processing.runalg('saga:representativeness', input, radius, exponent, result)
```

See also

Residual analysis

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Radius (Cells) [number] <put parameter description here>

Default: *7*

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *1.0*

Outputs

Mean Value [raster] <put output description here>
Difference from Mean Value [raster] <put output description here>
Standard Deviation [raster] <put output description here>
Value Range [raster] <put output description here>
Minimum Value [raster] <put output description here>
Maximum Value [raster] <put output description here>
Deviation from Mean Value [raster] <put output description here>
Percentile [raster] <put output description here>

Console usage

```
processing.runalg('saga:residualanalysis', grid, radius, distance_weighting_weighting, distance_w
```

See also

Spatial point pattern analysis

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>
Vertex Distance [Degree] [number] <put parameter description here>
Default: 5

Outputs

Mean Centre [vector] <put output description here>
Standard Distance [vector] <put output description here>
Bounding Box [vector] <put output description here>

Console usage

```
processing.runalg('saga:spatialpointpatternanalysis', points, step, centre, stddist, bbox)
```

See also

Statistics for grids

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Outputs

Arithmetic Mean [**raster**] <put output description here>

Minimum [**raster**] <put output description here>

Maximum [**raster**] <put output description here>

Variance [**raster**] <put output description here>

Standard Deviation [**raster**] <put output description here>

Mean less Standard Deviation [**raster**] <put output description here>

Mean plus Standard Deviation [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:statisticsforgrids', grids, mean, min, max, var, stddev, stddevlo, stddevhi)
```

See also**Variogram cloud****Description**

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Maximum Distance [**number**] <put parameter description here>

Default: *0.0*

Skip Number [**number**] <put parameter description here>

Default: *1*

Outputs

Variogram Cloud [**table**] <put output description here>

Console usage

```
processing.runalg('saga:variogramcloud', points, field, distmax, nskip, result)
```

See also

Variogram surface

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Number of Distance Classes [number] <put parameter description here>

Default: 10

Skip Number [number] <put parameter description here>

Default: 1

Outputs

Number of Pairs [raster] <put output description here>

Variogram Surface [raster] <put output description here>

Covariance Surface [raster] <put output description here>

Console usage

```
processing.runalg('saga:variogramsurface', points, field, distcount, nskip, count, variance, covar
```

See also

Zonal grid statistics

Description

<put algorithm description here>

Parameters

Zone Grid [raster] <put parameter description here>

Categorical Grids [multipleinput: rasters] Optional.

<put parameter description here>

Grids to analyse [multipleinput: rasters] Optional.

<put parameter description here>

Aspect [raster] Optional.

<put parameter description here>

Short Field Names [boolean] <put parameter description here>

Default: *True*

Outputs

Zonal Statistics [table] <put output description here>

Console usage

```
processing.runalg('saga:zonalgridstatistics', zones, catlist, statlist, aspect, shortnames, outta
```

See also

.

18.7.2 Grid analysis

Accumulated cost (anisotropic)

Description

<put algorithm description here>

Parameters

Cost Grid [raster] <put parameter description here>

Direction of max cost [raster] <put parameter description here>

Destination Points [raster] <put parameter description here>

k factor [number] <put parameter description here>

Default: *1*

Threshold for different route [number] <put parameter description here>

Default: *0*

Outputs

Accumulated Cost [raster] <put output description here>

Console usage

```
processing.runalg('saga:accumulatedcostanisotropic', cost, direction, points, k, threshold, accco
```

See also

Accumulated cost (isotropic)

Description

<put algorithm description here>

Parameters

Cost Grid [raster] <put parameter description here>

Destination Points [raster] <put parameter description here>

Threshold for different route [number] <put parameter description here>

Default: 0.0

Outputs

Accumulated Cost [raster] <put output description here>

Closest Point [raster] <put output description here>

Console usage

```
processing.runalg('saga:accumulatedcostisotropic', cost, points, threshold, acccost, closestpt)
```

See also

Aggregation index

Description

<put algorithm description here>

Parameters

Input Grid [raster] <put parameter description here>

Max. Number of Classes [number] <put parameter description here>

Default: 5

Outputs

Result [table] <put output description here>

Console usage

```
processing.runalg('saga:aggregationindex', input, maxnumclass, result)
```

See also**Analytical hierarchy process****Description**

<put algorithm description here>

Parameters

Input Grids [**multipleinput: rasters**] <put parameter description here>

Pairwise Comparisons Table [**table**] <put parameter description here>

Outputs

Output Grid [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:analyticalhierarchyprocess', grids, table, output)
```

See also**Cross-classification and tabulation****Description**

<put algorithm description here>

Parameters

Input Grid 1 [**raster**] <put parameter description here>

Input Grid 2 [**raster**] <put parameter description here>

Max. Number of Classes [**number**] <put parameter description here>

Default: 5

Outputs

Cross-Classification Grid [**raster**] <put output description here>

Cross-Tabulation Table [**table**] <put output description here>

Console usage

```
processing.runalg('saga:crossclassificationandtabulation', input, input2, maxnumclass, resultgrid)
```

See also

Fragmentation (alternative)

Description

<put algorithm description here>

Parameters

Classification [raster] <put parameter description here>

Class Identifier [number] <put parameter description here>

Default: 1

Neighborhood Min [number] <put parameter description here>

Default: 1

Neighborhood Max [number] <put parameter description here>

Default: 1

Level Aggregation [selection] <put parameter description here>

Options:

- 0 — [0] average
- 1 — [1] multiplicative

Default: 0

Add Border [boolean] <put parameter description here>

Default: *True*

Connectivity Weighting [number] <put parameter description here>

Default: 1.1

Minimum Density [Percent] [number] <put parameter description here>

Default: 10

Minimum Density for Interior Forest [Percent] [number] <put parameter description here>

Default: 99

Search Distance Increment [number] <put parameter description here>

Default: 0.0

Density from Neighbourhood [boolean] <put parameter description here>

Default: *True*

Outputs

Density [Percent] [raster] <put output description here>

Connectivity [Percent] [raster] <put output description here>

Fragmentation [raster] <put output description here>

Summary [table] <put output description here>

Console usage

```
processing.runalg('saga:fragmentationalternative', classes, class, neighborhood_min, neighborhood_max)
```

See also

Fragmentation classes from density and connectivity

Description

<put algorithm description here>

Parameters

Density [Percent] [raster] <put parameter description here>

Connectivity [Percent] [raster] <put parameter description here>

Add Border [boolean] <put parameter description here>

Default: *True*

Connectivity Weighting [number] <put parameter description here>

Default: *0*

Minimum Density [Percent] [number] <put parameter description here>

Default: *10*

Minimum Density for Interior Forest [Percent] [number] <put parameter description here>

Default: *99*

Outputs

Fragmentation [raster] <put output description here>

Console usage

```
processing.runalg('saga:fragmentationclassesfromdensityandconnectivity', density, connectivity, border)
```

See also

Fragmentation (standard)

Description

<put algorithm description here>

Parameters

Classification [raster] <put parameter description here>

Class Identifier [number] <put parameter description here>

Default: 1

Neighborhood Min [number] <put parameter description here>

Default: 1

Neighborhood Max [number] <put parameter description here>

Default: 3

Level Aggregation [selection] <put parameter description here>

Options:

- 0 — [0] average
- 1 — [1] multiplicative

Default: 0

Add Border [boolean] <put parameter description here>

Default: *True*

Connectivity Weighting [number] <put parameter description here>

Default: 1.1

Minimum Density [Percent] [number] <put parameter description here>

Default: 10

Minimum Density for Interior Forest [Percent] [number] <put parameter description here>

Default: 99

Neighborhood Type [selection] <put parameter description here>

Options:

- 0 — [0] square
- 1 — [1] circle

Default: 0

Include diagonal neighbour relations [boolean] <put parameter description here>

Default: *True*

Outputs

Density [Percent] [raster] <put output description here>

Connectivity [Percent] [raster] <put output description here>

Fragmentation [raster] <put output description here>

Summary [table] <put output description here>

Console usage

```
processing.runalg('saga:fragmentationstandard', classes, class, neighborhood_min, neighborhood_max)
```

See also

Layer of extreme value

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] Maximum
- 1 — [1] Minimum

Default: 0

Outputs

Result [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:layerofextremevalue', grids, criteria, result)
```

See also

Least cost paths

Description

<put algorithm description here>

Parameters

Source Point (s) [**vector: point**] <put parameter description here>

Accumulated cost [**raster**] <put parameter description here>

Values [**multipleinput: rasters**] Optional.

<put parameter description here>

Outputs

Profile (points) [**vector**] <put output description here>

Profile (lines) [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:leastcostpaths', source, dem, values, points, line)
```

See also

Ordered Weighted Averaging

Description

<put algorithm description here>

Parameters

Input Grids [**multipleinput: rasters**] <put parameter description here>

Weights [**fixedtable**] <put parameter description here>

Outputs

Output Grid [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:orderedweightedaveraging', grids, weights, output)
```

See also

Pattern analysis

Description

<put algorithm description here>

Parameters

Input Grid [**raster**] <put parameter description here>

Size of Analysis Window [**selection**] <put parameter description here>

Options:

- 0 — [0] 3 X 3
- 1 — [1] 5 X 5
- 2 — [2] 7 X 7

Default: 0

Max. Number of Classes [**number**] <put parameter description here>

Default: 0

Outputs

Relative Richness [raster] <put output description here>

Diversity [raster] <put output description here>

Dominance [raster] <put output description here>

Fragmentation [raster] <put output description here>

Number of Different Classes [raster] <put output description here>

Center Versus Neighbours [raster] <put output description here>

Console usage

```
processing.runalg('saga:patternanalysis', input, winsize, maxnumclass, relative, diversity, domin
```

See also

Soil texture classification

Description

<put algorithm description here>

Parameters

Sand [raster] Optional.

<put parameter description here>

Silt [raster] Optional.

<put parameter description here>

Clay [raster] Optional.

<put parameter description here>

Outputs

Soil Texture [raster] <put output description here>

Sum [raster] <put output description here>

Console usage

```
processing.runalg('saga:soiltextureclassification', sand, silt, clay, texture, sum)
```

See also

.

18.7.3 Grid calculus

Function

Description

<put algorithm description here>

Parameters

xmin [number] <put parameter description here>

Default: *0.0*

xmax [number] <put parameter description here>

Default: *0.0*

ymin [number] <put parameter description here>

Default: *0.0*

ymax [number] <put parameter description here>

Default: *0.0*

Formula [string] <put parameter description here>

Default: *(not set)*

Outputs

Function [raster] <put output description here>

Console usage

```
processing.runalg('saga:function', xmin, xmax, ymin, ymax, formul, result)
```

See also

Fuzzify

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

A [number] <put parameter description here>

Default: *0.0*

B [number] <put parameter description here>

Default: *0.0*

C [number] <put parameter description here>

Default: *0.0*

D [number] <put parameter description here>

Default: *0.0*

Membership Function Type [selection] <put parameter description here>

Options:

- 0 — [0] linear
- 1 — [1] sigmoidal
- 2 — [2] j-shaped

Default: *0*

Adjust to Grid [boolean] <put parameter description here>

Default: *True*

Outputs

Fuzzified Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:fuzzify', input, a, b, c, d, type, autofit, output)
```

See also

Fuzzy intersection (and)

Description

<put algorithm description here>

Parameters

Grids [multipleinput: rasters] <put parameter description here>

Operator Type [selection] <put parameter description here>

Options:

- 0 — [0] min(a, b) (non-interactive)
- 1 — [1] a * b
- 2 — [2] max(0, a + b - 1)

Default: *0*

Outputs

Intersection [raster] <put output description here>

Console usage

```
processing.runalg('saga:fuzzyintersectionand', grids, type, and)
```

See also

Fuzzy union (or)

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Operator Type [**selection**] <put parameter description here>

Options:

- 0 — [0] $\max(a, b)$ (non-interactive)
- 1 — [1] $a + b - a * b$
- 2 — [2] $\min(1, a + b)$

Default: 0

Outputs

Union [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:fuzzyunionor', grids, type, or)
```

See also

Geometric figures

Description

Draws simple geometric figures.

Parameters

Cell Count [**number**] Number of cells to use.

Default: 0

Cell Size [**number**] Size of the single cell.

Default: 0

Figure [selection] Type of the figure.

Options:

- 0 — [0] Cone (up)
- 1 — [1] Cone (down)
- 2 — [2] Plane

Default: 0

Direction of Plane [Degree] [number] Rotation factor in degrees.

Default: 0

Outputs

Result [raster] The resulting layer.

Console usage

```
processing.runalg('saga:geometricfigures', cell_count, cell_size, figure, plane, result)
```

See also

Gradient vector from cartesian to polar coordinates

Description

<put algorithm description here>

Parameters

X Component [raster] <put parameter description here>

Y Component [raster] <put parameter description here>

Polar Angle Units [selection] <put parameter description here>

Options:

- 0 — [0] radians
- 1 — [1] degree

Default: 0

Polar Coordinate System [selection] <put parameter description here>

Options:

- 0 — [0] mathematical
- 1 — [1] geographical
- 2 — [2] user defined

Default: 0

User defined Zero Direction [number] <put parameter description here>

Default: 0.0

User defined Orientation [selection] <put parameter description here>

Options:

- 0 — [0] clockwise
- 1 — [1] counterclockwise

Default: 0

Outputs

Direction [raster] <put output description here>

Length [raster] <put output description here>

Console usage

```
processing.runalg('saga:gradientvectorfromcartesiantopolarcoordinates', dx, dy, units, system, sy
```

See also

Gradient vector from polar to cartesian coordinates

Description

<put algorithm description here>

Parameters

Direction [raster] <put parameter description here>

Length [raster] <put parameter description here>

Polar Angle Units [selection] <put parameter description here>

Options:

- 0 — [0] radians
- 1 — [1] degree

Default: 0

Polar Coordinate System [selection] <put parameter description here>

Options:

- 0 — [0] mathematical
- 1 — [1] geographical
- 2 — [2] user defined

Default: 0

User defined Zero Direction [number] <put parameter description here>

Default: 0.0

User defined Orientation [selection] <put parameter description here>

Options:

- 0 — [0] clockwise

- 1 — [1] counterclockwise

Default: 0

Outputs

X Component [raster] <put output description here>

Y Component [raster] <put output description here>

Console usage

```
processing.runalg('saga:gradientvectorfrompolarcartesiancoordinates', dir, len, units, system, ...)
```

See also

Grid difference

Description

Creates a new grid layer as the result of the difference between two other grid layers.

Parameters

A [raster] First layer.

B [raster] Second layer.

Outputs

Difference (A - B) [raster] The resulting layer.

Console usage

```
processing.runalg('saga:griddifference', a, b, c)
```

See also

Grid division

Description

Creates a new grid layer as the result of the division between two other grid layers.

Parameters

Dividend [raster] First layer.

Divisor [raster] Second layer.

Outputs

Quotient [raster] The resulting layer.

Console usage

```
processing.runalg('saga:griddivision', a, b, c)
```

See also

Grid normalisation

Description

Normalises the grid values according to minimum and maximum values chosen.

Parameters

Grid [raster] Grid to normalize.

Target Range (min) [number] Minimum value.

Default: *0*

Target Range (max) [number] Maximum value.

Default: *1*

Outputs

Normalised Grid [raster] The resulting layer.

Console usage

```
processing.runalg('saga:gridnormalisation', input, range_min, range_max, output)
```

See also

Grids product

Description

<put algorithm description here>

Parameters

Grids [multipleinput: rasters] <put parameter description here>

Outputs

Product [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridsproduct', grids, result)
```

See also

Grids sum

Description

Creates a new grid layer as the result of the sum of two or more grid layers.

Parameters

Grids [multipleinput: rasters] Grid layers to sum

Outputs

Sum [raster] The resulting layer.

Console usage

```
processing.runalg('saga:gridssum', grids, result)
```

See also

Grid standardisation

Description

Standardises the grid layer values.

Parameters

Grid [raster] Grid to process.

Stretch Factor [number] stretching factor.

Default: *1.0*

Outputs

Standardised Grid [raster] The resulting layer.

Console usage

```
processing.runalg('saga:gridstandardisation', input, stretch, output)
```

See also

Grid volume

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Count Only Above Base Level
- 1 — [1] Count Only Below Base Level
- 2 — [2] Subtract Volumes Below Base Level
- 3 — [3] Add Volumes Below Base Level

Default: 0

Base Level [number] <put parameter description here>

Default: 0.0

Outputs

Console usage

```
processing.runalg('saga:gridvolume', grid, method, level)
```

See also

Metric conversions

Description

Performs numerical conversions of the grid values.

Parameters

Grid [raster] Grid to process.

Conversion [selection] Conversion type.

Options:

- 0 — [0] radians to degree
- 1 — [1] degree to radians
- 2 — [2] Celsius to Fahrenheit
- 3 — [3] Fahrenheit to Celsius

Default: 0

Outputs

Converted Grid [raster] The resulting layer.

Console usage

```
processing.runalg('saga:metricconversions', grid, conversion, conv)
```

See also

Polynomial trend from grids

Description

<put algorithm description here>

Parameters

Dependent Variables [multipleinput: rasters] <put parameter description here>

Independent Variable (per Grid and Cell) [multipleinput: rasters] Optional.

<put parameter description here>

Independent Variable (per Grid) [fixedtable] <put parameter description here>

Type of Approximated Function [selection] <put parameter description here>

Options:

- 0 — [0] first order polynom (linear regression)
- 1 — [1] second order polynom
- 2 — [2] third order polynom
- 3 — [3] fourth order polynom
- 4 — [4] fifth order polynom

Default: 0

Outputs

Polynomial Coefficients [raster] <put output description here>

Coefficient of Determination [raster] <put output description here>

Console usage

```
processing.runalg('saga:polynomialtrendfromgrids', grids, y_grids, y_table, polynom, parms, quality)
```

See also

Random field

Description

Generates a random grid layer.

Parameters

Width (Cells) [number] Width of the layer in cells.

Default: *100*

Height (Cells) [number] Height of the layer in cells.

Default: *100*

Cellsize [number] Cell size to use.

Default: *100.0*

West [number] West coordinate of the bottom-left corner of the grid.

Default: *0.0*

South [number] South coordinate of the bottom-left corner of the grid.

Default: *0.0*

Method [selection] Statistical method used for the calculation.

Options:

- 0 — [0] Uniform
- 1 — [1] Gaussian

Default: *0*

Range Min [number] Minimum cell value to use.

Default: *0.0*

Range Max [number] Maximum cell value to use.

Default: *1.0*

Arithmetic Mean [number] Mean of all the cell values to use.

Default: *0.0*

Standard Deviation [number] Standard deviation of all the cell values to use.

Default: *1.0*

Outputs

Random Field [raster] The resulting layer.

Console usage

```
processing.runalg('saga:randomfield', nx, ny, cellsize, xmin, ymin, method, range_min, range_max,
```


See also

Random terrain generation

Description

<put algorithm description here>

Parameters

Radius (cells) [number] <put parameter description here>

Default: *10*

Iterations [number] <put parameter description here>

Default: *10*

Target Dimensions [selection] <put parameter description here>

Options:

- 0 — [0] User defined

Default: *0*

Grid Size [number] <put parameter description here>

Default: *1.0*

Cols [number] <put parameter description here>

Default: *100*

Rows [number] <put parameter description here>

Default: *100*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:randomterraingeneration', radius, iterations, target_type, user_cell_size)
```

See also

Raster calculator

Description

<put algorithm description here>

Parameters

Main input layer [raster] <put parameter description here>

Additional layers [multipleinput: rasters] Optional.

<put parameter description here>

Formula [string] <put parameter description here>

Default: *(not set)*

Outputs

Result [raster] <put output description here>

Console usage

```
processing.runalg('saga:rastercalculator', grids, xgrids, formula, result)
```

See also

.

18.7.4 Grid filter

Dtm filter (slope-based)

Description

<put algorithm description here>

Parameters

Grid to filter [raster] <put parameter description here>

Search Radius [number] <put parameter description here>

Default: 2

Approx. Terrain Slope [number] <put parameter description here>

Default: 30.0

Use Confidence Interval [boolean] <put parameter description here>

Default: *True*

Outputs

Bare Earth [raster] <put output description here>

Removed Objects [raster] <put output description here>

Console usage

```
processing.runalg('saga:dtmfilterslopebased', input, radius, terrainslope, stddev, ground, nongro
```

See also

Filter clumps

Description

<put algorithm description here>

Parameters

Input Grid [raster] <put parameter description here>

Min. Size [number] <put parameter description here>

Default: 10

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:filterclumps', grid, threshold, output)
```

See also

Gaussian filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Standard Deviation [number] <put parameter description here>

Default: 1

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Search Radius [number] <put parameter description here>

Default: 3

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gaussianfilter', input, sigma, mode, radius, result)
```

See also

Laplacian filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] standard kernel 1
- 1 — [1] standard kernel 2
- 2 — [2] Standard kernel 3
- 3 — [3] user defined kernel

Default: 0

Standard Deviation (Percent of Radius) [number] <put parameter description here>

Default: 0

Radius [number] <put parameter description here>

Default: 1

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] square
- 1 — [1] circle

Default: 0

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:laplacianfilter', input, method, sigma, radius, mode, result)
```

See also**Majority filter****Description**

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Search Mode [**selection**] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Radius [**number**] <put parameter description here>

Default: 1

Threshold [**Percent**] [**number**] <put parameter description here>

Default: 0

Outputs

Filtered Grid [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:majorityfilter', input, mode, radius, threshold, result)
```

See also**Morphological filter****Description**

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Search Mode [**selection**] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Radius [number] <put parameter description here>

Default: *1*

Method [selection] <put parameter description here>

Options:

- 0 — [0] Dilation
- 1 — [1] Erosion
- 2 — [2] Opening
- 3 — [3] Closing

Default: *0*

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:morphologicalfilter', input, mode, radius, method, result)
```

See also

Multi direction lee filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Estimated Noise (absolute) [number] <put parameter description here>

Default: *1.0*

Estimated Noise (relative) [number] <put parameter description here>

Default: *1.0*

Weighted [boolean] <put parameter description here>

Default: *True*

Method [selection] <put parameter description here>

Options:

- 0 — [0] noise variance given as absolute value
- 1 — [1] noise variance given relative to mean standard deviation
- 2 — [2] original calculation (Ringeler)

Default: *0*

Outputs

Filtered Grid [raster] <put output description here>

Minimum Standard Deviation [raster] <put output description here>

Direction of Minimum Standard Deviation [raster] <put output description here>

Console usage

```
processing.runalg('saga:multidirectionleefilter', input, noise_abs, noise_rel, weighted, method, ...)
```

See also

Rank filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Radius [number] <put parameter description here>

Default: 1

Rank [Percent] [number] <put parameter description here>

Default: 50

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:rankfilter', input, mode, radius, rank, result)
```

See also

Simple filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Filter [selection] <put parameter description here>

Options:

- 0 — [0] Smooth
- 1 — [1] Sharpen
- 2 — [2] Edge

Default: 0

Radius [number] <put parameter description here>

Default: 2

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:simplefilter', input, mode, method, radius, result)
```

See also

User defined filter

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Filter Matrix [table] Optional.

<put parameter description here>

Default Filter Matrix (3x3) [fixedtable] <put parameter description here>

Outputs

Filtered Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:userdefinedfilter', input, filter, filter_3x3, result)
```

See also

.

18.7.5 Grid gridding**Inverse distance weighted****Description**

Inverse distance grid interpolation from irregular distributed points.

Parameters

Points [**vector:** **point**] <put parameter description here>

Attribute [**tablefield:** **any**] <put parameter description here>

Target Grid [**selection**] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Distance Weighting [**selection**] <put parameter description here>

Options:

- 0 — [0] inverse distance to a power
- 1 — [1] linearly decreasing within search radius
- 2 — [2] exponential weighting scheme
- 3 — [3] gaussian weighting scheme

Default: 0

Inverse Distance Power [**number**] <put parameter description here>

Default: 2

Exponential and Gaussian Weighting Bandwidth [**number**] <put parameter description here>

Default: 1

Search Range [**selection**] <put parameter description here>

Options:

- 0 — [0] search radius (local)
- 1 — [1] no search radius (global)

Default: 0

Search Radius [**number**] <put parameter description here>

Default: 100.0

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Number of Points [selection] <put parameter description here>

Options:

- 0 — [0] maximum number of points
- 1 — [1] all points

Default: 0

Maximum Number of Points [number] <put parameter description here>

Default: 10

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:inversedistanceweighted', shapes, field, target, weighting, power, bandwi
```

See also

Kernel density estimation

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Weight [tablefield: any] <put parameter description here>

Radius [number] <put parameter description here>

Default: 10

Kernel [selection] <put parameter description here>

Options:

- 0 — [0] quartic kernel
- 1 — [1] gaussian kernel

Default: 0

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:kerneldensityestimation', points, population, radius, kernel, target, out)
```

See also

Modifed quadratic shepard

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Quadratic Neighbors [number] <put parameter description here>

Default: 13

Weighting Neighbors [number] <put parameter description here>

Default: 19

Left [number] <put parameter description here>

Default: 0.0

Right [number] <put parameter description here>

Default: 0.0

Bottom [number] <put parameter description here>

Default: *0.0*

Top [number] <put parameter description here>

Default: *0.0*

Cellsize [number] <put parameter description here>

Default: *100.0*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:modifiedquadraticshepard', shapes, field, target, quadratic_neighbors, wei
```

See also

Natural neighbour

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Sibson [boolean] <put parameter description here>

Default: *True*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Cellsize [number] <put parameter description here>

Default: *100.0*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:naturalneighbour', shapes, field, target, sibson, output_extent, user_size)
```

See also

Nearest neighbour

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:nearestneighbour', shapes, field, target, output_extent, user_size, user_size)
```

See also

Shapes to grid

Description

<put algorithm description here>

Parameters

Shapes [vector: any] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Method for Multiple Values [selection] <put parameter description here>

Options:

- 0 — [0] first
- 1 — [1] last
- 2 — [2] minimum
- 3 — [3] maximum
- 4 — [4] mean

Default: 0

Method for Lines [selection] <put parameter description here>

Options:

- 0 — [0] thin
- 1 — [1] thick

Default: 0

Preferred Target Grid Type [selection] <put parameter description here>

Options:

- 0 — [0] Integer (1 byte)
- 1 — [1] Integer (2 byte)
- 2 — [2] Integer (4 byte)
- 3 — [3] Floating Point (4 byte)
- 4 — [4] Floating Point (8 byte)

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:shapestogrid', input, field, multiple, line_type, grid_type, output_extent)
```

See also**Triangulation****Description**

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:triangulation', shapes, field, target, output_extent, user_size, user_grid)
```

See also

.

18.7.6 Grid spline**B-spline approximation****Description**

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Resolution [number] <put parameter description here>

Default: 1.0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:bsplineapproximation', shapes, field, target, level, output_extent, user_
```

See also

Cubic spline approximation

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Minimal Number of Points [number] <put parameter description here>

Default: 3

Maximal Number of Points [number] <put parameter description here>

Default: 20

Points per Square [number] <put parameter description here>

Default: 5

Tolerance [number] <put parameter description here>

Default: 140.0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:cubicsplineapproximation', shapes, field, target, npmin, npmax, nppc, k, c
```

See also

Multilevel b-spline interpolation (from grid)

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Method [selection] <put parameter description here>

Options:

- 0 — [0] without B-spline refinement
- 1 — [1] with B-spline refinement

Default: 0

Threshold Error [number] <put parameter description here>

Default: 0.0001

Maximum Level [number] <put parameter description here>

Default: 11.0

Data Type [selection] <put parameter description here>

Options:

- 0 — [0] same as input grid
- 1 — [1] floating point

Default: 0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:multilevelbsplineinterpolationfromgrid', gridpoints, target, method, epsi
```

See also

Multilevel b-spline interpolation

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Method [selection] <put parameter description here>

Options:

- 0 — [0] without B-spline refinement
- 1 — [1] with B-spline refinement

Default: 0

Threshold Error [number] <put parameter description here>

Default: 0.0001

Maximum Level [number] <put parameter description here>

Default: 11.0

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: *100.0*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:multilevelbsplineinterpolation', shapes, field, target, method, epsilon, .
```

See also

Thin plate spline (global)

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Regularisation [number] <put parameter description here>

Default: *0.0*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Cellsize [number] <put parameter description here>

Default: *100.0*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:thinplatesplineglobal', shapes, field, target, regul, output_extent, user.
```

See also

Thin plate spline (local)

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Regularisation [number] <put parameter description here>

Default: 0.0001

Search Radius [number] <put parameter description here>

Default: 100.0

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] all directions
- 1 — [1] quadrants

Default: 0

Points Selection [selection] <put parameter description here>

Options:

- 0 — [0] all points in search radius
- 1 — [1] maximum number of points

Default: 0

Maximum Number of Points [number] <put parameter description here>

Default: 10

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:thinplatesplinelocal', shapes, field, target, regul, radius, mode, select,
```

See also

Thin plate spline (tin)

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Regularisation [number] <put parameter description here>

Default: 0.0

Neighbourhood [selection] <put parameter description here>

Options:

- 0 — [0] immediate
- 1 — [1] level 1
- 2 — [2] level 2

Default: 0

Add Frame [boolean] <put parameter description here>

Default: True

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Cellsize [number] <put parameter description here>

Default: 100.0

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:thinplatesplinetin', shapes, field, target, regul, level, frame, output_e
```

See also

.

18.7.7 Grid tools

Aggregate

Description

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Aggregation Size [**number**] <put parameter description here>

Default: 3

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] Sum
- 1 — [1] Min
- 2 — [2] Max

Default: 0

Outputs

Console usage

```
processing.runalg('saga:aggregate', input, size, method)
```

See also

Change grid values

Description

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Replace Condition [**selection**] <put parameter description here>

Options:

- 0 — [0] Grid value equals low value
- 1 — [1] Low value < grid value < high value

- 2 — [2] Low value \leq grid value $<$ high value

Default: 0

Lookup Table [fixedtable] <put parameter description here>

Outputs

Changed Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:changegridvalues', grid_in, method, lookup, grid_out)
```

See also

Close gaps

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Mask [raster] Optional.

<put parameter description here>

Tension Threshold [number] <put parameter description here>

Default: 0.1

Outputs

Changed Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:closegaps', input, mask, threshold, result)
```

See also

Close gaps with spline

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Mask [raster] Optional.

<put parameter description here>

Only Process Gaps with Less Cells [number] <put parameter description here>

Default: *0*

Maximum Points [number] <put parameter description here>

Default: *1000*

Number of Points for Local Interpolation [number] <put parameter description here>

Default: *10*

Extended Neighbourhood [boolean] <put parameter description here>

Default: *True*

Neighbourhood [selection] <put parameter description here>

Options:

- 0 — [0] Neumann
- 1 — [1] Moore

Default: *0*

Radius (Cells) [number] <put parameter description here>

Default: *0*

Relaxation [number] <put parameter description here>

Default: *0.0*

Outputs

Closed Gaps Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:closegapswithspline', grid, mask, maxgapcells, maxpoints, localpoints, ex
```

See also

Close one cell gaps

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Outputs

Changed Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:closeonecellgaps', input, result)
```

See also

Convert data storage type

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Data storage type [selection] <put parameter description here>

Options:

- 0 — [0] bit
- 1 — [1] unsigned 1 byte integer
- 2 — [2] signed 1 byte integer
- 3 — [3] unsigned 2 byte integer
- 4 — [4] signed 2 byte integer
- 5 — [5] unsigned 4 byte integer
- 6 — [6] signed 4 byte integer
- 7 — [7] 4 byte floating point number
- 8 — [8] 8 byte floating point number

Default: 0

Outputs

Converted Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:convertdatastoragetype', input, type, output)
```

See also

Crop to data

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Outputs

Cropped layer [raster] <put output description here>

Console usage

```
processing.runalg('saga:croptodata', input, output)
```

See also

Grid buffer

Description

<put algorithm description here>

Parameters

Features Grid [raster] <put parameter description here>

Distance [number] <put parameter description here>

Default: *1000*

Buffer Distance [selection] <put parameter description here>

Options:

- 0 — [0] Fixed
- 1 — [1] Cell value

Default: *0*

Outputs

Buffer Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridbuffer', features, dist, buffertype, buffer)
```

See also

Grid masking

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Mask [raster] <put parameter description here>

Outputs

Masked Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridmasking', grid, mask, masked)
```

See also

Grid orientation

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Copy
- 1 — [1] Flip
- 2 — [2] Mirror
- 3 — [3] Invert

Default: 0

Outputs

Changed Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridorientation', input, method, result)
```

See also

Grid proximity buffer

Description

<put algorithm description here>

Parameters

Source Grid [raster] <put parameter description here>

Buffer distance [number] <put parameter description here>

Default: *500.0*

Equidistance [number] <put parameter description here>

Default: *100.0*

Outputs

Distance Grid [raster] <put output description here>

Allocation Grid [raster] <put output description here>

Buffer Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridproximitybuffer', source, dist, ival, distance, alloc, buffer)
```

See also

Grid shrink/expand

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Operation [selection] <put parameter description here>

Options:

- 0 — [0] Shrink
- 1 — [1] Expand

Default: 0

Search Mode [selection] <put parameter description here>

Options:

- 0 — [0] Square
- 1 — [1] Circle

Default: 0

Radius [number] <put parameter description here>

Default: 1

Method [selection] <put parameter description here>

Options:

- 0 — [0] min
- 1 — [1] max
- 2 — [2] mean
- 3 — [3] majority

Default: 0

Outputs

Result Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:gridshrinkexpand', input, operation, mode, radius, method_expand, result)
```

See also

Invert data/no-data

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Outputs

Result [raster] <put output description here>

Console usage

```
processing.runalg('saga:invertdatanodata', input, output)
```

See also

Merge raster layers

Description

<put algorithm description here>

Parameters

Grids to Merge [**multipleinput: rasters**] <put parameter description here>

Preferred data storage type [**selection**] <put parameter description here>

Options:

- 0 — [0] 1 bit
- 1 — [1] 1 byte unsigned integer
- 2 — [2] 1 byte signed integer
- 3 — [3] 2 byte unsigned integer
- 4 — [4] 2 byte signed integer
- 5 — [5] 4 byte unsigned integer
- 6 — [6] 4 byte signed integer
- 7 — [7] 4 byte floating point
- 8 — [8] 8 byte floating point

Default: 0

Interpolation [**selection**] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Overlapping Cells [**selection**] <put parameter description here>

Options:

- 0 — [0] mean value
- 1 — [1] first value in order of grid list

Default: 0

Outputs

Merged Grid [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:mergerasterlayers', grids, type, interpol, overlap, merged)
```

See also

Patching

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Patch Grid [raster] <put parameter description here>

Interpolation Method [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Outputs

Completed Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:patching', original, additional, interpolation, completed)
```

See also

Proximity grid

Description

<put algorithm description here>

Parameters

Features [raster] <put parameter description here>

Outputs

Distance [raster] <put output description here>

Direction [raster] <put output description here>

Allocation [raster] <put output description here>

Console usage

```
processing.runalg('saga:proximitygrid', features, distance, direction, allocation)
```

See also

Reclassify grid values

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] single
- 1 — [1] range
- 2 — [2] simple table

Default: 0

old value (for single value change) [number] <put parameter description here>

Default: 0.0

new value (for single value change) [number] <put parameter description here>

Default: 1.0

operator (for single value change) [selection] <put parameter description here>

Options:

- 0 — [0] =
- 1 — [1] <
- 2 — [2] <=
- 3 — [3] >=
- 4 — [4] >

Default: 0

minimum value (for range) [number] <put parameter description here>

Default: 0.0

maximum value (for range) [number] <put parameter description here>

Default: *1.0*

new value (for range) [number] <put parameter description here>

Default: *2.0*

operator (for range) [selection] <put parameter description here>

Options:

- 0 — [0] <=
- 1 — [1] <

Default: *0*

Lookup Table [fixedtable] <put parameter description here>

operator (for table) [selection] <put parameter description here>

Options:

- 0 — [0] min <= value < max
- 1 — [1] min <= value <= max
- 2 — [2] min < value <= max
- 3 — [3] min < value < max

Default: *0*

replace no data values [boolean] <put parameter description here>

Default: *True*

new value for no data values [number] <put parameter description here>

Default: *0.0*

replace other values [boolean] <put parameter description here>

Default: *True*

new value for other values [number] <put parameter description here>

Default: *0.0*

Outputs

Reclassified Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:reclassifygridvalues', input, method, old, new, soperator, min, max, rnew,
```

See also

Resampling

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Preserve Data Type [boolean] <put parameter description here>

Default: *True*

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Interpolation Method (Scale Up) [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation
- 5 — [5] Mean Value
- 6 — [6] Mean Value (cell area weighted)
- 7 — [7] Minimum Value
- 8 — [8] Maximum Value
- 9 — [9] Majority

Default: *0*

Interpolation Method (Scale Down) [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: *0*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Cellsize [number] <put parameter description here>

Default: *100.0*

Outputs

Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:resampling', input, keep_type, target, scale_up_method, scale_down_method)
```

See also

Sort grid

Description

<put algorithm description here>

Parameters

Input Grid [raster] <put parameter description here>

Down sort [boolean] <put parameter description here>

Default: *True*

Outputs

Sorted Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:sortgrid', grid, down, output)
```

See also

Split RGB bands

Description

<put algorithm description here>

Parameters

Input layer [raster] <put parameter description here>

Outputs

Output R band layer [raster] <put output description here>

Output G band layer [raster] <put output description here>

Output B band layer [raster] <put output description here>

Console usage

```
processing.runalg('saga:splitrgrbbands', input, r, g, b)
```

See also

Threshold buffer

Description

<put algorithm description here>

Parameters

Features Grid [raster] <put parameter description here>

Value Grid [raster] <put parameter description here>

Threshold Grid [raster] Optional.

<put parameter description here>

Threshold [number] <put parameter description here>

Default: *0.0*

Threshold Type [selection] <put parameter description here>

Options:

- 0 — [0] Absolute
- 1 — [1] Relative from cell value

Default: *0*

Outputs

Buffer Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:thresholdbuffer', features, value, thresholdgrid, threshold, thresholdtype)
```

See also

.

18.7.8 Grid visualization

Histogram surface

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] rows
- 1 — [1] columns
- 2 — [2] circle

Default: 0

Outputs

Histogram [raster] <put output description here>

Console usage

```
processing.runalg('saga:histogramsurface', grid, method, hist)
```

See also

Rgb composite

Description

<put algorithm description here>

Parameters

R [raster] <put parameter description here>

G [raster] <put parameter description here>

B [raster] <put parameter description here>

Method for R value [selection] <put parameter description here>

Options:

- 0 — 0 - 255
- 1 — Rescale to 0 - 255
- 2 — User defined rescale
- 3 — Percentiles
- 4 — Percentage of standard deviation

Default: 0

Method for G value [selection] <put parameter description here>

Options:

- 0 — 0 - 255
- 1 — Rescale to 0 - 255

- 2 — User defined rescale
- 3 — Percentiles
- 4 — Percentage of standard deviation

Default: 0

Method for B value [selection] <put parameter description here>

Options:

- 0 — 0 - 255
- 1 — Rescale to 0 - 255
- 2 — User defined rescale
- 3 — Percentiles
- 4 — Percentage of standard deviation

Default: 0

Rescale Range for RED min [number] <put parameter description here>

Default: 0

Rescale Range for RED max [number] <put parameter description here>

Default: 255

Percentiles Range for RED max [number] <put parameter description here>

Default: 1

Percentiles Range for RED max [number] <put parameter description here>

Default: 99

Percentage of standard deviation for RED [number] <put parameter description here>

Default: 150.0

Rescale Range for GREEN min [number] <put parameter description here>

Default: 0

Rescale Range for GREEN max [number] <put parameter description here>

Default: 255

Percentiles Range for GREEN max [number] <put parameter description here>

Default: 1

Percentiles Range for GREEN max [number] <put parameter description here>

Default: 99

Percentage of standard deviation for GREEN [number] <put parameter description here>

Default: 150.0

Rescale Range for BLUE min [number] <put parameter description here>

Default: 0

Rescale Range for BLUE max [number] <put parameter description here>

Default: 255

Percentiles Range for BLUE max [number] <put parameter description here>

Default: 1

Percentiles Range for BLUE max [number] <put parameter description here>

Default: *99*

Percentage of standard deviation for BLUE [number] <put parameter description here>

Default: *150.0*

Outputs

Output RGB [raster] <put output description here>

Console usage

```
processing.runalg('saga:rgbcomposite', grid_r, grid_g, grid_b, r_method, g_method, b_method, r_ra
```

See also

.

18.7.9 Imagery classification

Change detection

Description

<put algorithm description here>

Parameters

Initial State [raster] <put parameter description here>

Look-up Table [table] Optional.

<put parameter description here>

Value [tablefield: any] <put parameter description here>

Value (Maximum) [tablefield: any] <put parameter description here>

Name [tablefield: any] <put parameter description here>

Final State [raster] <put parameter description here>

Look-up Table [table] Optional.

<put parameter description here>

Value [tablefield: any] <put parameter description here>

Value (Maximum) [tablefield: any] <put parameter description here>

Name [tablefield: any] <put parameter description here>

Report Unchanged Classes [boolean] <put parameter description here>

Default: *True*

Output as... [selection] <put parameter description here>

Options:

- 0 — [0] cells
- 1 — [1] percent
- 2 — [2] area

Default: 0

Outputs

Changes [raster] <put output description here>

Changes [table] <put output description here>

Console usage

```
processing.runalg('saga:changedetection', initial, ini_lut, ini_lut_min, ini_lut_max, ini_lut_name)
```

See also

Cluster analysis for grids

Description

<put algorithm description here>

Parameters

Grids [multipleinput: rasters] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Iterative Minimum Distance (Forgy 1965)
- 1 — [1] Hill-Climbing (Rubin 1967)
- 2 — [2] Combined Minimum Distance / Hillclimbing

Default: 0

Clusters [number] <put parameter description here>

Default: 5

Normalise [boolean] <put parameter description here>

Default: *True*

Old Version [boolean] <put parameter description here>

Default: *True*

Outputs

Clusters [raster] <put output description here>

Statistics [table] <put output description here>

Console usage

```
processing.runalg('saga:clusteranalysisforgrids', grids, method, ncluster, normalise, oldversion,
```

See also

Supervised classification

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Training Areas [**vector: polygon**] <put parameter description here>

Class Identifier [**tablefield: any**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] Binary Encoding
- 1 — [1] Parallelepiped
- 2 — [2] Minimum Distance
- 3 — [3] Mahalanobis Distance
- 4 — [4] Maximum Likelihood
- 5 — [5] Spectral Angle Mapping
- 6 — [6] Winner Takes All

Default: *0*

Normalise [**boolean**] <put parameter description here>

Default: *True*

Distance Threshold [**number**] <put parameter description here>

Default: *0.0*

Probability Threshold (Percent) [**number**] <put parameter description here>

Default: *0.0*

Probability Reference [**selection**] <put parameter description here>

Options:

- 0 — [0] absolute
- 1 — [1] relative

Default: *0*

Spectral Angle Threshold (Degree) [**number**] <put parameter description here>

Default: *0.0*

Outputs

Class Information [table] <put output description here>

Classification [raster] <put output description here>

Quality [raster] <put output description here>

Console usage

```
processing.runalg('saga:supervisedclassification', grids, roi, roi_id, method, normalise, thresho
```

See also

.

18.7.10 Imagery RGA

Fast region growing algorithm

Description

<put algorithm description here>

Parameters

Input Grids [multipleinput: rasters] <put parameter description here>

Seeds Grid [raster] <put parameter description here>

Smooth Rep [raster] Optional.

<put parameter description here>

Outputs

Segmente [raster] <put output description here>

Mean [raster] <put output description here>

Console usage

```
processing.runalg('saga:fastregiongrowingalgorithm', input, start, rep, result, mean)
```

See also

.

18.7.11 Imagery segmentation

Grid skeletonization

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Standard
- 1 — [1] Hilditch's Algorithm
- 2 — [2] Channel Skeleton

Default: 0

Initialisation [selection] <put parameter description here>

Options:

- 0 — [0] Less than
- 1 — [1] Greater than

Default: 0

Threshold (Init.) [number] <put parameter description here>

Default: 0.0

Convergence [number] <put parameter description here>

Default: 3.0

Outputs

Skeleton [raster] <put output description here>

Skeleton [vector] <put output description here>

Console usage

```
processing.runalg('saga:gridskeletonization', input, method, init_method, init_threshold, converg
```

See also

Seed generation

Description

<put algorithm description here>

Parameters

Features [**multipleinput: rasters**] <put parameter description here>

Bandwidth (Cells) [**number**] <put parameter description here>

Default: 2

Type of Surface [**selection**] <put parameter description here>

Options:

- 0 — [0] smoothed surface
- 1 — [1] variance (a)
- 2 — [2] variance (b)

Default: 0

Extraction of... [**selection**] <put parameter description here>

Options:

- 0 — [0] minima
- 1 — [1] maxima
- 2 — [2] minima and maxima

Default: 0

Feature Aggregation [**selection**] <put parameter description here>

Options:

- 0 — [0] additive
- 1 — [1] multiplicative

Default: 0

Normalized [**boolean**] <put parameter description here>

Default: *True*

Outputs

Surface [**raster**] <put output description here>

Seeds Grid [**raster**] <put output description here>

Seeds [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:seedgeneration', grids, factor, type_surface, type_seeds, type_merge, norm
```

See also

Simple region growing

Description

<put algorithm description here>

Parameters

Seeds [**raster**] <put parameter description here>

Features [**multipleinput: rasters**] <put parameter description here>

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] feature space and position
- 1 — [1] feature space

Default: 0

Neighbourhood [**selection**] <put parameter description here>

Options:

- 0 — [0] 4 (von Neumann)
- 1 — [1] 8 (Moore)

Default: 0

Variance in Feature Space [**number**] <put parameter description here>

Default: 1.0

Variance in Position Space [**number**] <put parameter description here>

Default: 1.0

Threshold - Similarity [**number**] <put parameter description here>

Default: 0.0

Refresh [**boolean**] <put parameter description here>

Default: *True*

Leaf Size (for Speed Optimisation) [**number**] <put parameter description here>

Default: 256

Outputs

Segments [**raster**] <put output description here>

Similarity [**raster**] <put output description here>

Seeds [**table**] <put output description here>

Console usage

```
processing.runalg('saga:simpleregiongrowing', seeds, features, method, neighbour, sig_1, sig_2, t
```

See also

Watershed segmentation

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Output [selection] <put parameter description here>

Options:

- 0 — [0] Seed Value
- 1 — [1] Segment ID

Default: 0

Method [selection] <put parameter description here>

Options:

- 0 — [0] Minima
- 1 — [1] Maxima

Default: 0

Join Segments based on Threshold Value [selection] <put parameter description here>

Options:

- 0 — [0] do not join
- 1 — [1] seed to saddle difference
- 2 — [2] seeds difference

Default: 0

Threshold [number] <put parameter description here>

Default: 0

Allow Edge Pixels to be Seeds [boolean] <put parameter description here>

Default: *True*

Borders [boolean] <put parameter description here>

Default: *True*

Outputs

Segments [raster] <put output description here>

Seed Points [vector] <put output description here>

Borders [raster] <put output description here>

Console usage

```
processing.runalg('saga:watershedsegmentation', grid, output, down, join, threshold, edge, bborder)
```

See also

.

18.7.12 Imagery tools

Vegetation index[distance based]

Description

<put algorithm description here>

Parameters

Near Infrared Band [raster] <put parameter description here>

Red Band [raster] <put parameter description here>

Slope of the soil line [number] <put parameter description here>

Default: 0.0

Intercept of the soil line [number] <put parameter description here>

Default: 0.0

Outputs

PVI (Richardson and Wiegand) [raster] <put output description here>

PVI (Perry & Lautenschlager) [raster] <put output description here>

PVI (Walther & Shabaani) [raster] <put output description here>

PVI (Qi, et al) [raster] <put output description here>

Console usage

```
processing.runalg('saga:vegetationindexdistancebased', nir, red, slope, intercept, pvi, pvi1, pvi2)
```

See also

Vegetation index[slope based]

Description

<put algorithm description here>

Parameters

Near Infrared Band [raster] <put parameter description here>

Red Band [raster] <put parameter description here>

Outputs

Normalized Difference Vegetation Index [raster] <put output description here>

Ratio Vegetation Index [raster] <put output description here>

Transformed Vegetation Index [raster] <put output description here>

Corrected Transformed Vegetation Index [raster] <put output description here>

Thiam's Transformed Vegetation Index [raster] <put output description here>

Normalized Ratio Vegetation Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:vegetationindexslopebased', nir, red, ndvi, ratio, tvi, ctvi, ttvi, nratio)
```

See also

.

18.7.13 Kriging

Ordinary kriging (global)

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Create Variance Grid [boolean] <put parameter description here>

Default: *True*

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Variogram Model [selection] <put parameter description here>

Options:

- 0 — [0] Spherical Model
- 1 — [1] Exponential Model
- 2 — [2] Gaussian Model
- 3 — [3] Linear Regression
- 4 — [4] Exponential Regression
- 5 — [5] Power Function Regression

Default: 0

Block Kriging [boolean] <put parameter description here>

Default: *True*

Block Size [number] <put parameter description here>

Default: 100

Logarithmic Transformation [boolean] <put parameter description here>

Default: *True*

Nugget [number] <put parameter description here>

Default: 0.0

Sill [number] <put parameter description here>

Default: 0.0

Range [number] <put parameter description here>

Default: 0.0

Linear Regression [number] <put parameter description here>

Default: 1.0

Exponential Regression [number] <put parameter description here>

Default: 0.1

Power Function - A [number] <put parameter description here>

Default: 1.0

Power Function - B [number] <put parameter description here>

Default: 0.5

Grid Size [number] <put parameter description here>

Default: 1.0

Fit Extent [boolean] <put parameter description here>

Default: *True*

Output extent [extent] <put parameter description here>

Default: 0,1,0,1

Outputs

Grid [raster] <put output description here>

Variance [raster] <put output description here>

Console usage

```
processing.runalg('saga:ordinarykrigingglobal', shapes, field, bvariance, target, model, block, d
```

See also

Ordinary kriging

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Create Variance Grid [**boolean**] <put parameter description here>

Default: *True*

Target Grid [**selection**] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Variogram Model [**selection**] <put parameter description here>

Options:

- 0 — [0] Spherical Model
- 1 — [1] Exponential Model
- 2 — [2] Gaussian Model
- 3 — [3] Linear Regression
- 4 — [4] Exponential Regression
- 5 — [5] Power Function Regression

Default: *0*

Block Kriging [**boolean**] <put parameter description here>

Default: *True*

Block Size [**number**] <put parameter description here>

Default: *100*

Logarithmic Transformation [**boolean**] <put parameter description here>

Default: *True*

Nugget [**number**] <put parameter description here>

Default: *0.0*

Sill [**number**] <put parameter description here>

Default: *10.0*

Range [**number**] <put parameter description here>

Default: *100.0*

Linear Regression [**number**] <put parameter description here>

Default: *1.0*

Exponential Regression [number] <put parameter description here>

Default: *0.1*

Power Function - A [number] <put parameter description here>

Default: *1*

Power Function - B [number] <put parameter description here>

Default: *0.5*

Maximum Search Radius (map units) [number] <put parameter description here>

Default: *1000.0*

Min.Number of m_Points [number] <put parameter description here>

Default: *4*

Max. Number of m_Points [number] <put parameter description here>

Default: *20*

Grid Size [number] <put parameter description here>

Default: *1.0*

Fit Extent [boolean] <put parameter description here>

Default: *True*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Outputs

Grid [raster] <put output description here>

Variance [raster] <put output description here>

Console usage

```
processing.runalg('saga:ordinarykriging', shapes, field, bvariance, target, model, block, dblock,
```

See also

Universal kriging (global)

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Create Variance Grid [boolean] <put parameter description here>

Default: *True*

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: 0

Variogram Model [selection] <put parameter description here>

Options:

- 0 — [0] Spherical Model
- 1 — [1] Exponential Model
- 2 — [2] Gaussian Model
- 3 — [3] Linear Regression
- 4 — [4] Exponential Regression
- 5 — [5] Power Function Regression

Default: 0

Block Kriging [boolean] <put parameter description here>

Default: *True*

Block Size [number] <put parameter description here>

Default: 100

Logarithmic Transformation [boolean] <put parameter description here>

Default: *True*

Nugget [number] <put parameter description here>

Default: 0.0

Sill [number] <put parameter description here>

Default: 0.0

Range [number] <put parameter description here>

Default: 0.0

Linear Regression [number] <put parameter description here>

Default: 1

Exponential Regression [number] <put parameter description here>

Default: 0.5

Power Function - A [number] <put parameter description here>

Default: 1.0

Power Function - B [number] <put parameter description here>

Default: 0.1

Grids [multipleinput: rasters] <put parameter description here>

Grid Interpolation [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation

- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: *0*

Grid Size [number] <put parameter description here>

Default: *1.0*

Fit Extent [boolean] <put parameter description here>

Default: *True*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Outputs

Grid [raster] <put output description here>

Variance [raster] <put output description here>

Console usage

```
processing.runalg('saga:universalkrigingglobal', shapes, field, bvariance, target, model, block, ...)
```

See also

Universal kriging

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Create Variance Grid [boolean] <put parameter description here>

Default: *True*

Target Grid [selection] <put parameter description here>

Options:

- 0 — [0] user defined

Default: *0*

Variogram Model [selection] <put parameter description here>

Options:

- 0 — [0] Spherical Model
- 1 — [1] Exponential Model
- 2 — [2] Gaussian Model
- 3 — [3] Linear Regression

- 4 — [4] Exponential Regression
- 5 — [5] Power Function Regression

Default: 0

Block Kriging [boolean] <put parameter description here>

Default: *True*

Block Size [number] <put parameter description here>

Default: 100

Logarithmic Transformation [boolean] <put parameter description here>

Default: *True*

Nugget [number] <put parameter description here>

Default: 0.0

Sill [number] <put parameter description here>

Default: 0.0

Range [number] <put parameter description here>

Default: 0.0

Linear Regression [number] <put parameter description here>

Default: 1.0

Exponential Regression [number] <put parameter description here>

Default: 0.1

Power Function - A [number] <put parameter description here>

Default: 1

Power Function - B [number] <put parameter description here>

Default: 0.5

Grids [multipleinput: rasters] <put parameter description here>

Grid Interpolation [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Min.Number of m_Points [number] <put parameter description here>

Default: 4

Max. Number of m_Points [number] <put parameter description here>

Default: 20

Maximum Search Radius (map units) [number] <put parameter description here>

Default: 1000.0

Grid Size [number] <put parameter description here>

Default: *1.0*

Fit Extent [boolean] <put parameter description here>

Default: *True*

Output extent [extent] <put parameter description here>

Default: *0,1,0,1*

Outputs

Grid [raster] <put output description here>

Variance [raster] <put output description here>

Console usage

```
processing.runalg('saga:universalkriging', shapes, field, bvariance, target, model, block, dblock,
```

See also

.

18.7.14 Shapes grid

Add grid values to points

Description

Creates a new vector layer as a result of the union of a points layer with the interpolated value of one or more base background grid layer(s). This way, the new layer created will have a new column in the attribute table that reflects the interpolated value of the background grid.

Parameters

Points [vector: point] Input layer.

Grids [multipleinput: rasters] Background grid layer(s)

Interpolation [selection] interpolation method to use.

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: *0*

Outputs

Result [vector] The resulting layer.

Console usage

```
processing.runalg('saga:addgridvaluestopoints', shapes, grids, interpol, result)
```

See also

Add grid values to shapes

Description

<put algorithm description here>

Parameters

Shapes [vector: any] <put parameter description here>

Grids [multipleinput: rasters] <put parameter description here>

Interpolation [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbor
- 1 — [1] Bilinear Interpolation
- 2 — [2] Inverse Distance Interpolation
- 3 — [3] Bicubic Spline Interpolation
- 4 — [4] B-Spline Interpolation

Default: 0

Outputs

Result [vector] <put output description here>

Console usage

```
processing.runalg('saga:addgridvaluestoshapes', shapes, grids, interpol, result)
```

See also

Clip grid with polygon

Description

<put algorithm description here>

Parameters

Input [raster] <put parameter description here>

Polygons [vector: polygon] <put parameter description here>

Outputs

Output [raster] <put output description here>

Console usage

```
processing.runalg('saga:clipgridwithpolygon', input, polygons, output)
```

See also

Contour lines from grid

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Minimum Contour Value [number] <put parameter description here>

Default: *0.0*

Maximum Contour Value [number] <put parameter description here>

Default: *10000.0*

Equidistance [number] <put parameter description here>

Default: *100.0*

Outputs

Contour Lines [vector] <put output description here>

Console usage

```
processing.runalg('saga:contourlinesfromgrid', input, zmin, zmax, zstep, contour)
```

See also

Gradient vectors from directional components

Description

<put algorithm description here>

Parameters

X Component [raster] <put parameter description here>

Y Component [raster] <put parameter description here>

Step [number] <put parameter description here>

Default: 1

Size Range Min [number] <put parameter description here>

Default: 25.0

Size Range Max [number] <put parameter description here>

Default: 100.0

Aggregation [selection] <put parameter description here>

Options:

- 0 — [0] nearest neighbour
- 1 — [1] mean value

Default: 0

Style [selection] <put parameter description here>

Options:

- 0 — [0] simple line
- 1 — [1] arrow
- 2 — [2] arrow (centered to cell)

Default: 0

Outputs

Gradient Vectors [vector] <put output description here>

Console usage

```
processing.runalg('saga:gradientvectorsfromdirectionalcomponents', x, y, step, size_min, size_max)
```

See also

Gradient vectors from direction and length

Description

<put algorithm description here>

Parameters

Direction [raster] <put parameter description here>

Length [raster] <put parameter description here>

Step [number] <put parameter description here>

Default: 1

Size Range Min [number] <put parameter description here>

Default: 25.0

Size Range Max [number] <put parameter description here>

Default: 100.0

Aggregation [selection] <put parameter description here>

Options:

- 0 — [0] nearest neighbour
- 1 — [1] mean value

Default: 0

Style [selection] <put parameter description here>

Options:

- 0 — [0] simple line
- 1 — [1] arrow
- 2 — [2] arrow (centered to cell)

Default: 0

Outputs

Gradient Vectors [vector] <put output description here>

Console usage

```
processing.runalg('saga:gradientvectorsfromdirectionandlength', dir, len, step, size_min, size_max)
```

See also

Gradient vectors from surface

Description

<put algorithm description here>

Parameters

Surface [raster] <put parameter description here>

Step [number] <put parameter description here>

Default: 1

Size Range Min [number] <put parameter description here>

Default: 25.0

Size Range Max [number] <put parameter description here>

Default: 100.0

Aggregation [selection] <put parameter description here>

Options:

- 0 — [0] nearest neighbour
- 1 — [1] mean value

Default: 0

Style [selection] <put parameter description here>

Options:

- 0 — [0] simple line
- 1 — [1] arrow
- 2 — [2] arrow (centered to cell)

Default: 0

Outputs

Gradient Vectors [vector] <put output description here>

Console usage

```
processing.runalg('saga:gradientvectorsfromsurface', surface, step, size_min, size_max, aggr, sty
```

See also

Grid statistics for polygons

Description

<put algorithm description here>

Parameters

Grids [multipleinput: rasters] <put parameter description here>

Polygons [vector: polygon] <put parameter description here>

Number of Cells [boolean] <put parameter description here>

Default: *True*

Minimum [boolean] <put parameter description here>

Default: *True*

Maximum [boolean] <put parameter description here>

Default: *True*

Range [boolean] <put parameter description here>

Default: *True*

Sum [boolean] <put parameter description here>

Default: *True*

Mean [boolean] <put parameter description here>

Default: *True*

Variance [boolean] <put parameter description here>

Default: *True*

Standard Deviation [boolean] <put parameter description here>

Default: *True*

Quantiles [number] <put parameter description here>

Default: *0*

Outputs

Statistics [vector] <put output description here>

Console usage

```
processing.runalg('saga:gridstatisticsforpolygons', grids, polygons, count, min, max, range, sum,
```

See also

Grid values to points (randomly)

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Frequency [number] <put parameter description here>

Default: *100*

Outputs

Points [vector] <put output description here>

Console usage

```
processing.runalg('saga:gridvaluestopointsrandomly', grid, freq, points)
```

See also

Grid values to points

Description

<put algorithm description here>

Parameters

Grids [**multipleinput: rasters**] <put parameter description here>

Polygons [**vector: any**] Optional.

<put parameter description here>

Exclude NoData Cells [**boolean**] <put parameter description here>

Default: *True*

Type [**selection**] <put parameter description here>

Options:

- 0 — [0] nodes
- 1 — [1] cells

Default: *0*

Outputs

Shapes [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:gridvaluestopoints', grids, polygons, nodata, type, shapes)
```

See also

Local minima and maxima

Description

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Outputs

Minima [**vector**] <put output description here>

Maxima [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:localminimaandmaxima', grid, minima, maxima)
```

See also**Vectorising grid classes****Description**

<put algorithm description here>

Parameters

Grid [**raster**] <put parameter description here>

Class Selection [**selection**] <put parameter description here>

Options:

- 0 — [0] one single class specified by class identifier
- 1 — [1] all classes

Default: 0

Class Identifier [**number**] <put parameter description here>

Default: 0

Vectorised class as... [**selection**] <put parameter description here>

Options:

- 0 — [0] one single (multi-)polygon object
- 1 — [1] each island as separated polygon

Default: 0

Outputs

Polygons [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:vectorisinggridclasses', grid, class_all, class_id, split, polygons)
```

See also

.

18.7.15 Shapes lines**Convert points to line(s)****Description**

Converts points to lines.

Parameters

Points [vector: point] Points to convert.

Order by... [tablefield: any] Lines will be ordered following this field.

Separate by... [tablefield: any] Lines will be grouped according to this field.

Outputs

Lines [vector] The resulting layer.

Console usage

```
processing.runalg('saga:convertpointstolines', points, order, separate, lines)
```

See also

Convert polygons to lines

Description

Creates lines from polygons.

Parameters

Polygons [vector: polygon] Layer to process.

Outputs

Lines [vector] The resulting layer.

Console usage

```
processing.runalg('saga:convertpolygonstolines', polygons, lines)
```

See also

Line dissolve

Description

<put algorithm description here>

Parameters

Lines [vector: any] <put parameter description here>

1. **Attribute** [tablefield: any] <put parameter description here>
2. **Attribute** [tablefield: any] <put parameter description here>

3. Attribute [tablefield: any] <put parameter description here>

Dissolve... [selection] <put parameter description here>

Options:

- 0 — [0] lines with same attribute value(s)
- 1 — [1] all lines

Default: 0

Outputs

Dissolved Lines [vector] <put output description here>

Console usage

```
processing.runalg('saga:linedissolve', lines, field_1, field_2, field_3, all, dissolved)
```

See also

Line-polygon intersection

Description

<put algorithm description here>

Parameters

Lines [vector: line] <put parameter description here>

Polygons [vector: polygon] <put parameter description here>

Output [selection] <put parameter description here>

Options:

- 0 — [0] one multi-line per polygon
- 1 — [1] keep original line attributes

Default: 0

Outputs

Intersection [vector] <put output description here>

Console usage

```
processing.runalg('saga:linepolygonintersection', lines, polygons, method, intersect)
```

See also

Line properties

Description

Calculates some information on each line of the layer.

Parameters

Lines [**vector: line**] Layer to analyze.

Number of Parts [**boolean**] Determines whether to calculate number of segments in line.

Default: *True*

Number of Vertices [**boolean**] Determines whether to calculate number of vertices in line.

Default: *True*

Length [**boolean**] Determines whether to calculate total line length.

Default: *True*

Outputs

Lines with Property Attributes [**vector**] The resulting layer.

Console usage

```
processing.runalg('saga:lineproperties', lines, bparts, bpoints, blength, output)
```

See also

Line simplification

Description

Simplifies the geometry of a lines layer.

Parameters

Lines [**vector: line**] Layer to process.

Tolerance [**number**] Simplification tolerance.

Default: *1.0*

Outputs

Simplified Lines [**vector**] The resulting layer.

Console usage

```
processing.runalg('saga:linesimplification', lines, tolerance, output)
```

See also

.

18.7.16 Shapes points

Add coordinates to points

Description

Adds the X and Y coordinates of feature in the attribute table of input layer.

Parameters

Points [vector: point] Input layer.

Outputs

Output [vector] Resulting layer with the updated attribute table.

Console usage

```
processing.runalg('saga:addcoordinatestopoints', input, output)
```

See also

Add polygon attributes to points

Description

Adds the specified field of the polygons layer to the attribute table of the points layer. The new attributes added for each point depend on the value of the background polygon layer.

Parameters

Points [vector: point] Points layer.

Polygons [vector: polygon] Background polygons layer.

Attribute [tablefield: any] Attribute of the polygons layer that will be added to the points layer.

Outputs

Result [vector] The resulting layer.

Console usage

```
processing.runalg('saga:addpolygonattributestopoints', input, polygons, field, output)
```

See also

Aggregate point observations

Description

<put algorithm description here>

Parameters

Reference Points [**vector: any**] <put parameter description here>

ID [**tablefield: any**] <put parameter description here>

Observations [**table**] <put parameter description here>

X [**tablefield: any**] <put parameter description here>

Y [**tablefield: any**] <put parameter description here>

Track [**tablefield: any**] <put parameter description here>

Date [**tablefield: any**] <put parameter description here>

Time [**tablefield: any**] <put parameter description here>

Parameter [**tablefield: any**] <put parameter description here>

Maximum Time Span (Seconds) [**number**] <put parameter description here>

Default: *60.0*

Maximum Distance [**number**] <put parameter description here>

Default: *0.002*

Outputs

Aggregated [**table**] <put output description here>

Console usage

```
processing.runalg('saga:aggregatepointobservations', reference, reference_id, observations, x, y,
```

See also

Clip points with polygons

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Polygons [vector: polygon] <put parameter description here>

Add Attribute to Clipped Points [tablefield: any] <put parameter description here>

Clipping Options [selection] <put parameter description here>

Options:

- 0 — [0] one layer for all points
- 1 — [1] separate layer for each polygon

Default: 0

Outputs

Clipped Points [vector] <put output description here>

Console usage

```
processing.runalg('saga:clippointswithpolygons', points, polygons, field, method, clips)
```

See also

Convert lines to points

Description

Converts lines layer into a points.

Parameters

Lines [vector: line] Lines layer to convert.

Insert Additional Points [boolean] Determines whether to add additional nodes or not.

Default: *True*

Insert Distance [number] Distance between the additional points.

Default: *1.0*

Outputs

Points [vector] The resulting layer.

Console usage

```
processing.runalg('saga:convertlinestopoints', lines, add, dist, points)
```

See also

Convert multipoints to points

Description

<put algorithm description here>

Parameters

Multipoints [**vector: point**] <put parameter description here>

Outputs

Points [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:convertmultipointstopoints', multipoints, points)
```

See also

Convex hull

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Hull Construction [**selection**] <put parameter description here>

Options:

- 0 — [0] one hull for all shapes
- 1 — [1] one hull per shape
- 2 — [2] one hull per shape part

Default: 0

Outputs

Convex Hull [**vector**] <put output description here>

Minimum Bounding Box [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:convexhull', shapes, polypoints, hulls, boxes)
```

See also

Distance matrix

Description

Generates a distance matrix between each point of the input layer. A unique ID will be created in the first row of the resulting matrix (symmetric matrix), while every other cell reflects the distance between the points.

Parameters

Points [vector: point] Input layer.

Outputs

Distance Matrix Table [table] The resulting table.

Console usage

```
processing.runalg('saga:distancematrix', points, table)
```

See also

Fit n points to shape

Description

<put algorithm description here>

Parameters

Shapes [vector: polygon] <put parameter description here>

Number of points [number] <put parameter description here>

Default: 10

Outputs

Points [vector] <put output description here>

Console usage

```
processing.runalg('saga:fitnpointstoshape', shapes, numpoints, points)
```

See also

Points filter

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Radius [**number**] <put parameter description here>

Default: *1*

Minimum Number of Points [**number**] <put parameter description here>

Default: *0*

Maximum Number of Points [**number**] <put parameter description here>

Default: *0*

Quadrants [**boolean**] <put parameter description here>

Default: *True*

Filter Criterion [**selection**] <put parameter description here>

Options:

- 0 — [0] keep maxima (with tolerance)
- 1 — [1] keep minima (with tolerance)
- 2 — [2] remove maxima (with tolerance)
- 3 — [3] remove minima (with tolerance)
- 4 — [4] remove below percentile
- 5 — [5] remove above percentile

Default: *0*

Tolerance [**number**] <put parameter description here>

Default: *0.0*

Percentile [**number**] <put parameter description here>

Default: *50*

Outputs

Filtered Points [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:pointsfilter', points, field, radius, minnum, maxnum, quadrants, method, t
```


See also

Points thinning

Description

<put algorithm description here>

Parameters

Points [**vector: point**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Resolution [**number**] <put parameter description here>

Default: *1.0*

Outputs

Thinned Points [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:pointsthinning', points, field, resolution, thinned)
```

See also

Remove duplicate points

Description

<put algorithm description here>

Parameters

Points [**vector: any**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Point to Keep [**selection**] <put parameter description here>

Options:

- 0 — [0] first point
- 1 — [1] last point
- 2 — [2] point with minimum attribute value
- 3 — [3] point with maximum attribute value

Default: *0*

Numeric Attribute Values [**selection**] <put parameter description here>

Options:

- 0 — [0] take value from the point to be kept

- 1 — [1] minimum value of all duplicates
- 2 — [2] maximum value of all duplicates
- 3 — [3] mean value of all duplicates

Default: 0

Outputs

Result [vector] <put output description here>

Console usage

```
processing.runalg('saga:removeduplicatepoints', points, field, method, numeric, result)
```

See also

Separate points by direction

Description

<put algorithm description here>

Parameters

Points [vector: point] <put parameter description here>

Number of Directions [number] <put parameter description here>

Default: 4

Tolerance (Degree) [number] <put parameter description here>

Default: 5

Outputs

Output [vector] <put output description here>

Console usage

```
processing.runalg('saga:separatepointsbydirection', points, directions, tolerance, output)
```

See also

.

18.7.17 Shapes polygons

Convert lines to polygons

Description

Converts lines to polygons.

Parameters

Lines [vector: line] Lines to convert.

Outputs

Polygons [vector] The resulting layer.

Console usage

```
processing.runalg('saga:convertlinestopolygons', lines, polygons)
```

See also

Convert polygon/line vertices to points

Description

Converts the line or polygon vertices into points.

Parameters

Shapes [vector: any] Layer to process.

Outputs

Points [vector] The resulting layer.

Console usage

```
processing.runalg('saga:convertpolygonlineverticestopoints', shapes, points)
```

See also

Polygon centroids

Description

Calculates the centroids of polygons.

Parameters

Polygons [vector: polygon] Input layer.

Centroids for each part [boolean] Determines whether centroids should be calculated for each part of multipart polygon or not.

Default: *True*

Outputs

Centroids [vector] The resulting layer.

Console usage

```
processing.runalg('saga:polygoncentroids', polygons, method, centroids)
```

See also

Polygon dissolve

Description

<put algorithm description here>

Parameters

Polygons [vector: polygon] <put parameter description here>

1. **Attribute** [tablefield: any] Optional.

<put parameter description here>

2. **Attribute** [tablefield: any] Optional.

<put parameter description here>

3. **Attribute** [tablefield: any] Optional.

<put parameter description here>

Dissolve... [selection] <put parameter description here>

Options:

- 0 — [0] polygons with same attribute value
- 1 — [1] all polygons
- 2 — [2] polygons with same attribute value (keep inner boundaries)
- 3 — [3] all polygons (keep inner boundaries)

Default: *0*

Outputs

Dissolved Polygons [vector] <put output description here>

Console usage

```
processing.runalg('saga:polygondissolve', polygons, field_1, field_2, field_3, dissolve, dissolve)
```

See also

Polygon-line intersection

Description

<put algorithm description here>

Parameters

Polygons [vector: polygon] <put parameter description here>

Lines [vector: line] <put parameter description here>

Outputs

Intersection [vector] <put output description here>

Console usage

```
processing.runalg('saga:polygonlineintersection', polygons, lines, intersect)
```

See also

Polygon parts to separate polygons

Description

<put algorithm description here>

Parameters

Polygons [vector: polygon] <put parameter description here>

Ignore Lakes [boolean] <put parameter description here>

Default: *True*

Outputs

Polygon Parts [vector] <put output description here>

Console usage

```
processing.runalg('saga:polygonpartstoseparatepolygons', polygons, lakes, parts)
```

See also

Polygon properties

Description

<put algorithm description here>

Parameters

Polygons [**vector: polygon**] <put parameter description here>

Number of Parts [**boolean**] <put parameter description here>

Default: *True*

Number of Vertices [**boolean**] <put parameter description here>

Default: *True*

Perimeter [**boolean**] <put parameter description here>

Default: *True*

Area [**boolean**] <put parameter description here>

Default: *True*

Outputs

Polygons with Property Attributes [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:polygonproperties', polygons, bparts, bpoints, blength, barea, output)
```

See also

Polygon shape indices

Description

Calculates spatial statistics for polygons. This includes:

- area
- perimeter
- perimeter / area
- perimeter / square root of the area
- maximum distance
- maximum distance / area
- maximum distance / square root of the area
- shape index

Parameters

Shapes [vector: polygon] Layer to analyze.

Outputs

Shape Index [vector] The resulting layer.

Console usage

```
processing.runalg('saga:polygonsshapeindices', shapes, index)
```

See also

Polygons to edges and nodes

Description

Extracts boundaries and nodes of polygons in separate files.

Parameters

Polygons [vector: polygon] Input layer.

Outputs

Edges [vector] Resulting line layer with polygons boundaries.

Nodes [vector] Resulting line layer with polygons nodes.

Console usage

```
processing.runalg('saga:polygontoedgesandnodes', polygons, edges, nodes)
```

See also

.

18.7.18 Shapes tools

Create graticule

Description

Creates a grid.

Parameters

Extent [vector: any] Optional.

Grid will be created according to the selected layer.

Output extent [extent] Extent of the grid.

Default: *0,1,0,1*

Division Width [number] X-axes spacing between the lines.

Default: *1.0*

Division Height [number] Y-axes spacing between the lines.

Default: *1.0*

Type [selection] Geometry type of the resulting grid.

Options:

- 0 — [0] Lines
- 1 — [1] Rectangles

Default: *0*

Outputs

Gaticule [vector] The resulting layer.

Console usage

```
processing.runalg('saga:creategraticule', extent, output_extent, distx, disty, type, graticule)
```

See also

Cut shapes layer

Description

<put algorithm description here>

Parameters

Vector layer to cut [vector: any] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] completely contained
- 1 — [1] intersects
- 2 — [2] center

Default: *0*

Cutting polygons [vector: any] <put parameter description here>

Outputs

Result [vector] <put output description here>

Extent [vector] <put output description here>

Console usage

```
processing.runalg('saga:cutshapeslayer', shapes, method, polygons_polygons, cut, extent)
```

See also

Get shapes extents

Description

Creates polygons according to the extent of the input layer features.

Parameters

Shapes [vector: any] Input layer.

Parts [boolean] Determines whether create polygon for each feature (`True`) or just create single polygon for whole layer (`False`).

Default: *True*

Outputs

Extents [vector] The resulting layer.

Console usage

```
processing.runalg('saga:getshapesextents', shapes, parts, extents)
```

See also

Merge shapes layers

Description

Merges two or more input layer into a unique resulting layer. You can merge together only layer of the same type (polygons with polygons, lines with lines, points with points).

The attribute table of the resulting layer will include only the attributes of the first input layer. Two additional columns will be added: one corresponding to the ID of every merged layer and the other one corresponding to the original name of the merged layer.

Parameters

Main Layer [vector: any] Initial layer.

Additional Layers [multipleinput: any vectors] Optional.

Layer(s) to merge with.

Outputs

Merged Layer [vector] The resulting layer.

Console usage

```
processing.runalg('saga:mergeshapelayers', main, layers, out)
```

See also

Polar to cartesian coordinates

Description

<put algorithm description here>

Parameters

Polar Coordinates [vector: any] <put parameter description here>

Exaggeration [tablefield: any] <put parameter description here>

Exaggeration Factor [number] <put parameter description here>

Default: *1*

Radius [number] <put parameter description here>

Default: *6371000.0*

Degree [boolean] <put parameter description here>

Default: *True*

Outputs

Cartesian Coordinates [vector] <put output description here>

Console usage

```
processing.runalg('saga:polartocartesiancoordinates', polar, f_exagg, d_exagg, radius, degree, ca
```

See also**Quadtree structure to shapes****Description**

<put algorithm description here>

Parameters

Shapes [**vector: any**] <put parameter description here>

Attribute [**tablefield: any**] <put parameter description here>

Outputs

Polygons [**vector**] <put output description here>

Lines [**vector**] <put output description here>

Duplicated Points [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:quadtreestructuretoshapes', shapes, attribute, polygons, lines, points)
```

See also**Shapes buffer****Description**

Creates buffer around features based on fixed distance or distance field.

Parameters

Shapes [**vector: any**] Input layer.

Buffer Distance [**selection**] Buffering method.

Options:

- 0 — [0] fixed value
- 1 — [1] attribute field

Default: 0

Buffer Distance (Fixed) [**number**] Buffer distance for “fixed value” method.

Default: 100.0

Buffer Distance (Attribute) [**tablefield: any**] Name of the distance field for “attribute field” method.

Scaling Factor for Attribute Value [**number**] <put parameter description here>

Default: 1.0

Number of Buffer Zones [number] Number of buffer(s) to generate.

Default: *1.0*

Circle Point Distance [Degree] [number] Smoothness of the buffer borders: great numbers means rough borders.

Default: *5.0*

Dissolve Buffers [boolean] Determines whether to dissolve results or not.

Default: *True*

Outputs

Buffer [vector] The resulting layer.

Console usage

```
processing.runalg('saga:shapesbuffer', shapes, buf_type, buf_dist, buf_field, buf_scale, buf_zones)
```

See also

Split shapes layer randomly

Description

Splits the input layer randomly in two parts.

Parameters

Shapes [vector: any] Layer to split.

Split ratio (%) [number] Split ratio between the resulting layers.

Default: *50*

Outputs

Group A [vector] First resulting layer.

Group B [vector] Second resulting layer.

Console usage

```
processing.runalg('saga:splitshapeslayerrandomly', shapes, percent, a, b)
```

See also

Transform shapes

Description

<put algorithm description here>

Parameters

Shapes [**vector: any**] <put parameter description here>

dX [**number**] <put parameter description here>

Default: *0.0*

dY [**number**] <put parameter description here>

Default: *0.0*

Angle [**number**] <put parameter description here>

Default: *0.0*

Scale Factor X [**number**] <put parameter description here>

Default: *1.0*

Scale Factor Y [**number**] <put parameter description here>

Default: *1.0*

X [**number**] <put parameter description here>

Default: *0.0*

Y [**number**] <put parameter description here>

Default: *0.0*

Outputs

Output [**vector**] <put output description here>

Console usage

```
processing.runalg('saga:transformshapes', in, dx, dy, angle, scalex, scaley, anchorx, anchory, out)
```

See also

.

18.7.19 Shapes transect

Transect through polygon shapefile

Description

<put algorithm description here>

Parameters

Line Transect (s) [**vector: line**] <put parameter description here>

Theme [**vector: any**] <put parameter description here>

Theme Field [**tablefield: any**] <put parameter description here>

Outputs

Result table [table] <put output description here>

Console usage

```
processing.runalg('saga:transectthroughpolygonshapefile', transect, theme, theme_field, transect_
```

See also

.

18.7.20 Simulation fire

Fire risk analysis

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Fuel Model [raster] <put parameter description here>

Wind Speed [raster] <put parameter description here>

Wind Direction [raster] <put parameter description here>

Dead Fuel Moisture 1H [raster] <put parameter description here>

Dead Fuel Moisture 10H [raster] <put parameter description here>

Dead Fuel Moisture 100H [raster] <put parameter description here>

Herbaceous Fuel Moisture [raster] <put parameter description here>

Wood Fuel Moisture [raster] <put parameter description here>

Value [raster] Optional.

<put parameter description here>

Base Probability [raster] Optional.

<put parameter description here>

Number of Events [number] <put parameter description here>

Default: *1000*

Fire Length [number] <put parameter description here>

Default: *100*

Outputs

Danger [raster] <put output description here>

Compound Probability [raster] <put output description here>

Priority Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:fireriskanalysis', dem, fuel, windspd, winddir, m1h, m10h, m100h, mherb, m
```

See also

Simulation

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Fuel Model [raster] <put parameter description here>

Wind Speed [raster] <put parameter description here>

Wind Direction [raster] <put parameter description here>

Dead Fuel Moisture 1H [raster] <put parameter description here>

Dead Fuel Moisture 10H [raster] <put parameter description here>

Dead Fuel Moisture 100H [raster] <put parameter description here>

Herbaceous Fuel Moisture [raster] <put parameter description here>

Wood Fuel Moisture [raster] <put parameter description here>

Ignition Points [raster] <put parameter description here>

Update View [boolean] <put parameter description here>

Default: *True*

Outputs

Time [raster] <put output description here>

Flame Length [raster] <put output description here>

Intensity [raster] <put output description here>

Console usage

```
processing.runalg('saga:simulation', dem, fuel, windspd, winddir, m1h, m10h, m100h, mherb, mwood,
```

See also

.

18.7.21 Simulation hydrology

Overland flow - kinematic wave d8

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Gauges [vector: any] Optional.

<put parameter description here>

Simulation Time [h] [number] <put parameter description here>

Default: 24

Simulation Time Step [h] [number] <put parameter description here>

Default: 0.1

Manning's Roughness [number] <put parameter description here>

Default: 0.03

Max. Iterations [number] <put parameter description here>

Default: 100

Epsilon [number] <put parameter description here>

Default: 0.0001

Precipitation [selection] <put parameter description here>

Options:

- 0 — [0] Homogenous
- 1 — [1] Above Elevation
- 2 — [2] Left Half

Default: 0

Threshold Elevation [number] <put parameter description here>

Default: 0.0

Outputs

Runoff [raster] <put output description here>

Flow at Gauges [table] <put output description here>

Console usage

```
processing.runalg('saga:overlandflowkinematicwaved8', dem, gauges, time_span, time_step, roughnes
```

See also

Water retention capacity

Description

<put algorithm description here>

Parameters

Plot Holes [vector: any] <put parameter description here>

DEM [raster] <put parameter description here>

Outputs

Final Parameters [vector] <put output description here>

Water Retention Capacity [raster] <put output description here>

Console usage

```
processing.runalg('saga:waterretentioncapacity', shapes, dem, output, retention)
```

See also

.

18.7.22 Table calculus

Fill gaps in records

Description

<put algorithm description here>

Parameters

Table [table] <put parameter description here>

Order [tablefield: any] <put parameter description here>

Interpolation [selection] <put parameter description here>

Options:

- 0 — [0] Nearest Neighbour
- 1 — [1] Linear

- 2 — [2] Spline

Default: 0

Outputs

Table without Gaps [table] <put output description here>

Console usage

```
processing.runalg('saga:fillgapsinrecords', table, order, method, nogaps)
```

See also

Principle components analysis

Description

<put algorithm description here>

Parameters

Table [table] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] correlation matrix
- 1 — [1] variance-covariance matrix
- 2 — [2] sums-of-squares-and-cross-products matrix

Default: 0

Number of Components [number] <put parameter description here>

Default: 3

Outputs

Principle Components [table] <put output description here>

Console usage

```
processing.runalg('saga:principlecomponentsanalysis', table, method, nfirst, pca)
```

See also

Running average

Description

<put algorithm description here>

Parameters

Input [table] <put parameter description here>

Attribute [tablefield: any] <put parameter description here>

Number of Records [number] <put parameter description here>

Default: 10

Outputs

Output [table] <put output description here>

Console usage

```
processing.runalg('saga:runningaverage', input, field, count, output)
```

See also

.

18.7.23 Table tools

Change date format

Description

Converts the date format of the input layer.

Parameters

Table [table] Input table.

Date Field [tablefield: any] Attribute the date.

Input Format [selection] Input date format.

Options:

- 0 — [0] dd.mm.yy
- 1 — [1] yy.mm.dd
- 2 — [2] dd:mm:yy
- 3 — [3] yy:mm:dd
- 4 — [4] ddmmyyyy, fix size
- 5 — [5] yyyymmdd, fix size
- 6 — [6] ddmmyy, fix size
- 7 — [7] yymmdd, fix size
- 8 — [8] Julian Day

Default: 0

Output Format [selection] Output date format.

Options:

- 0 — [0] dd.mm.yy
- 1 — [1] yy.mm.dd
- 2 — [2] dd:mm:yy
- 3 — [3] yy:mm:dd
- 4 — [4] ddmmyyyy, fix size
- 5 — [5] yyyymmdd, fix size
- 6 — [6] ddmmyy, fix size
- 7 — [7] yymmdd, fix size
- 8 — [8] Julian Day

Default: 0

Outputs

Output [table] The resulting table.

Console usage

```
processing.runalg('saga:changedateformat', table, field, fmt_in, fmt_out, output)
```

See also

Change time format

Description

Converts the time format of the input layer.

Parameters

Table [table] Input table.

Time Field [tablefield: any] Attribute with time.

Input Format [selection] Input time format.

Options:

- 0 — [0] hh.mm.ss
- 1 — [1] hh:mm:ss
- 2 — [2] hhmmss, fix size
- 3 — [3] hours
- 4 — [4] minutes
- 5 — [5] seconds

Default: 0

Output Format [selection] Output time format.

Options:

- 0 — [0] hh.mm.ss
- 1 — [1] hh:mm:ss
- 2 — [2] hhmmss, fix size
- 3 — [3] hours
- 4 — [4] minutes
- 5 — [5] seconds

Default: 0

Outputs

Output [table] The resulting table.

Console usage

```
processing.runalg('saga:changetimeformat', table, field, fmt_in, fmt_out, output)
```

See also

.

18.7.24 Terrain channels

Channel network and drainage basins

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Threshold [number] <put parameter description here>

Default: 5.0

Outputs

Flow Direction [raster] <put output description here>

Flow Connectivity [raster] <put output description here>

Strahler Order [raster] <put output description here>

Drainage Basins [raster] <put output description here>

Channels [vector] <put output description here>

Drainage Basins [vector] <put output description here>

Junctions [vector] <put output description here>

Console usage

```
processing.runalg('saga:channelnetworkanddrainagebasins', dem, threshold, direction, connection, c
```

See also

Channel network

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Flow Direction [raster] Optional.

<put parameter description here>

Initiation Grid [raster] <put parameter description here>

Initiation Type [selection] <put parameter description here>

Options:

- 0 — [0] Less than
- 1 — [1] Equals
- 2 — [2] Greater than

Default: 0

Initiation Threshold [number] <put parameter description here>

Default: 0.0

Divergence [raster] Optional.

<put parameter description here>

Tracing: Max. Divergence [number] <put parameter description here>

Default: 10

Tracing: Weight [raster] Optional.

<put parameter description here>

Min. Segment Length [number] <put parameter description here>

Default: 10

Outputs

Channel Network [raster] <put output description here>

Channel Direction [raster] <put output description here>

Channel Network [vector] <put output description here>

Console usage

```
processing.runalg('saga:channelnetwork', elevation, sinkroute, init_grid, init_method, init_value)
```

See also

Overland flow distance to channel network

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Channel Network [raster] <put parameter description here>

Flow Algorithm [selection] <put parameter description here>

Options:

- 0 — [0] D8
- 1 — [1] MFD

Default: 0

Outputs

Overland Flow Distance [raster] <put output description here>

Vertical Overland Flow Distance [raster] <put output description here>

Horizontal Overland Flow Distance [raster] <put output description here>

Console usage

```
processing.runalg('saga:overlandflowdistancetochannelnetwork', elevation, channels, method, distanceto)
```

See also

Strahler order

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Outputs

Strahler Order [raster] <put output description here>

Console usage

```
processing.runalg('saga:strahlerorder', dem, strahler)
```

See also

Vertical distance to channel network

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Channel Network [raster] <put parameter description here>

Tension Threshold [Percentage of Cell Size] [number] <put parameter description here>

Default: *1*

Keep Base Level below Surface [boolean] <put parameter description here>

Default: *True*

Outputs

Vertical Distance to Channel Network [raster] <put output description here>

Channel Network Base Level [raster] <put output description here>

Console usage

```
processing.runalg('saga:verticaldistancetochannelnetwork', elevation, channels, threshold, nounde
```

See also

Watershed basins

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Channel Network [raster] <put parameter description here>

Sink Route [raster] Optional.

<put parameter description here>

Min. Size [number] <put parameter description here>

Default: 0

Outputs

Watershed Basins [raster] <put output description here>

Console usage

```
processing.runalg('saga:watershedbasins', elevation, channels, sinkroute, minsize, basins)
```

See also

.

18.7.25 Terrain hydrology

Burn stream network into dem

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Streams [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] simply decrease cell's value by epsilon
- 1 — [1] lower cell's value to neighbours minimum value minus epsilon

Default: 0

Epsilon [number] <put parameter description here>

Default: 1.0

Outputs

Processed DEM [raster] <put output description here>

Console usage

```
processing.runalg('saga:burnstreamnetworkintodem', dem, stream, method, epsilon, burn)
```

See also

Catchment area (flow tracing)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Sink Routes [raster] Optional.

<put parameter description here>

Weight [raster] Optional.

<put parameter description here>

Material [raster] Optional.

<put parameter description here>

Target [raster] Optional.

<put parameter description here>

Step [number] <put parameter description here>

Default: *1*

Method [selection] <put parameter description here>

Options:

- 0 — [0] Rho 8
- 1 — [1] Kinematic Routing Algorithm
- 2 — [2] DEMON

Default: *0*

DEMON - Min. DQV [number] <put parameter description here>

Default: *0.0*

Flow Correction [boolean] <put parameter description here>

Default: *True*

Outputs

Catchment Area [raster] <put output description here>

Catchment Height [raster] <put output description here>

Catchment Slope [raster] <put output description here>

Total accumulated Material [raster] <put output description here>

Accumulated Material from _left_ side [raster] <put output description here>

Accumulated Material from _right_ side [raster] <put output description here>

Console usage

```
processing.runalg('saga:catchmentareaflowtracing', elevation, sinkroute, weight, material, target,
```

See also

Catchment area (recursive)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Sink Routes [raster] Optional.

<put parameter description here>

Weight [raster] Optional.

<put parameter description here>

Material [raster] Optional.

<put parameter description here>

Target [raster] Optional.

<put parameter description here>

Step [number] <put parameter description here>

Default: 1

Target Areas [raster] Optional.

<put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8
- 1 — [1] Rho 8
- 2 — [2] Deterministic Infinity
- 3 — [3] Multiple Flow Direction

Default: 0

Convergence [number] <put parameter description here>

Default: 1.1

Outputs

Catchment Area [raster] <put output description here>

Catchment Height [raster] <put output description here>

Catchment Slope [raster] <put output description here>

Total accumulated Material [raster] <put output description here>

Accumulated Material from _left_ side [raster] <put output description here>

Accumulated Material from _right_ side [raster] <put output description here>

Flow Path Length [raster] <put output description here>

Console usage

```
processing.runalg('saga:catchmentarearecursive', elevation, sinkroute, weight, material, target, ...)
```

See also

Catchment Area

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8
- 1 — [1] Rho 8
- 2 — [2] Braunschweiger Reliefmodell
- 3 — [3] Deterministic Infinity
- 4 — [4] Multiple Flow Direction
- 5 — [5] Multiple Triangular Flow Directon

Default: 0

Outputs

Catchment Area [raster] <put output description here>

Console usage

```
processing.runalg('saga:catchmentarea', elevation, method, carea)
```

See also

Cell balance

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Parameter [raster] Optional.

<put parameter description here>

Default Weight [number] <put parameter description here>

Default: *1.0*

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8
- 1 — [1] Multiple Flow Direction

Default: *0*

Outputs

Cell Balance [raster] <put output description here>

Console usage

```
processing.runalg('saga:cellbalance', dem, weights, weight, method, balance)
```

See also

Edge contamination

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Outputs

Edge Contamination [raster] <put output description here>

Console usage

```
processing.runalg('saga:edgecontamination', dem, contamination)
```

See also

Fill Sinks

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Minimum Slope [Degree] [number] <put parameter description here>

Default: *0.01*

Outputs

Filled DEM [raster] <put output description here>

Console usage

```
processing.runalg('saga:fillsinks', dem, minslope, result)
```

See also

Fill sinks (wang & liu)

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Minimum Slope [Degree] [number] <put parameter description here>

Default: *0.01*

Outputs

Filled DEM [raster] <put output description here>

Flow Directions [raster] <put output description here>

Watershed Basins [raster] <put output description here>

Console usage

```
processing.runalg('saga:fillsinkswangliu', elev, minslope, filled, fdir, wshed)
```

See also

Fill sinks xxi (wang & liu)

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Minimum Slope [Degree] [number] <put parameter description here>

Default: *0.01*

Outputs

Filled DEM [raster] <put output description here>

Console usage

```
processing.runalg('saga:fillsinksxxlwangliu', elev, minslope, filled)
```

See also

Flat detection

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Flat Area Values [selection] <put parameter description here>

Options:

- 0 — [0] elevation
- 1 — [1] enumeration

Default: *0*

Outputs

No Flats [raster] <put output description here>

Flat Areas [raster] <put output description here>

Console usage

```
processing.runalg('saga:flatdetection', dem, flat_output, noflats, flats)
```

See also

Flow path length

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Seeds [raster] Optional.

<put parameter description here>

Seeds Only [boolean] <put parameter description here>

Default: *True*

Flow Routing Algorithm [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8 (D8)
- 1 — [1] Multiple Flow Direction (FD8)

Default: *0*

Convergence (FD8) [number] <put parameter description here>

Default: *1.1*

Outputs

Flow Path Length [raster] <put output description here>

Console usage

```
processing.runalg('saga:flowpathlength', elevation, seed, seeds_only, method, convergence, length)
```


See also

Flow width and specific catchment area

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Total Catchment Area (TCA) [raster] Optional.

<put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8
- 1 — [1] Multiple Flow Direction (Quinn et al. 1991)
- 2 — [2] Aspect

Default: 0

Outputs

Flow Width [raster] <put output description here>

Specific Catchment Area (SCA) [raster] <put output description here>

Console usage

```
processing.runalg('saga:flowwidthandspecificcatchmentarea', dem, tca, method, width, sca)
```

See also

Lake flood

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Seeds [raster] <put parameter description here>

Absolute Water Levels [boolean] <put parameter description here>

Default: *True*

Outputs

Lake [raster] <put output description here>

Surface [raster] <put output description here>

Console usage

```
processing.runalg('saga:lakeflood', elev, seeds, level, outdepth, outlevel)
```

See also

Ls factor

Description

<put algorithm description here>

Parameters

Slope [raster] <put parameter description here>

Catchment Area [raster] <put parameter description here>

Area to Length Conversion [selection] <put parameter description here>

Options:

- 0 — [0] no conversion (areas already given as specific catchment area)
- 1 — [1] 1 / cell size (specific catchment area)
- 2 — [2] square root (catchment length)

Default: 0

Method (LS) [selection] <put parameter description here>

Options:

- 0 — [0] Moore et al. 1991
- 1 — [1] Desmet & Govers 1996
- 2 — [2] Bochner & Selige 2006

Default: 0

Rill/Interrill Erosivity [number] <put parameter description here>

Default: 0.0

Stability [selection] <put parameter description here>

Options:

- 0 — [0] stable
- 1 — [1] instable (thawing)

Default: 0

Outputs

LS Factor [raster] <put output description here>

Console usage

```
processing.runalg('saga:lsfactor', slope, area, conv, method, erosivity, stability, ls)
```

See also

Saga wetness index

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

t [number] <put parameter description here>

Default: *10*

Outputs

Catchment area [raster] <put output description here>

Catchment slope [raster] <put output description here>

Modified catchment area [raster] <put output description here>

Wetness index [raster] <put output description here>

Console usage

```
processing.runalg('saga:sagawetnessindex', dem, t, c, gn, cs, sb)
```

See also

Sink drainage route detection

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Threshold [boolean] <put parameter description here>

Default: *True*

Threshold Height [number] <put parameter description here>

Default: *100.0*

Outputs

Sink Route [raster] <put output description here>

Console usage

```
processing.runalg('saga:sinkdrainageroutedetection', elevation, threshold, thrsheight, sinkroute)
```

See also

Sink removal

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Sink Route [raster] Optional.

<put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deepen Drainage Routes
- 1 — [1] Fill Sinks

Default: *0*

Threshold [boolean] <put parameter description here>

Default: *True*

Threshold Height [number] <put parameter description here>

Default: *100.0*

Outputs

Preprocessed DEM [raster] <put output description here>

Console usage

```
processing.runalg('saga:sinkremoval', dem, sinkroute, method, threshold, thrsheight, dem_preproc)
```

See also

Slope length

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Outputs

Slope Length [raster] <put output description here>

Console usage

```
processing.runalg('saga:slopelength', dem, length)
```

See also

Stream power index

Description

<put algorithm description here>

Parameters

Slope [raster] <put parameter description here>

Catchment Area [raster] <put parameter description here>

Area Conversion [selection] <put parameter description here>

Options:

- 0 — [0] no conversion (areas already given as specific catchment area)
- 1 — [1] 1 / cell size (pseudo specific catchment area)

Default: 0

Outputs

Stream Power Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:streampowerindex', slope, area, conv, spi)
```

See also

Topographic wetness index (twi)

Description

<put algorithm description here>

Parameters

Slope [raster] <put parameter description here>

Catchment Area [raster] <put parameter description here>

Transmissivity [raster] Optional.

<put parameter description here>

Area Conversion [selection] <put parameter description here>

Options:

- 0 — [0] no conversion (areas already given as specific catchment area)
- 1 — [1] 1 / cell size (pseudo specific catchment area)

Default: 0

Method (TWI) [selection] <put parameter description here>

Options:

- 0 — [0] Standard
- 1 — [1] TOPMODEL

Default: 0

Outputs

Topographic Wetness Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:topographicwetnessindextwi', slope, area, trans, conv, method, twi)
```

See also

Upslope Area

Description

<put algorithm description here>

Parameters

Target Area [raster] Optional.

<put parameter description here>

Target X coordinate [number] <put parameter description here>

Default: *0.0*

Target Y coordinate [number] <put parameter description here>

Default: *0.0*

Elevation [raster] <put parameter description here>

Sink Routes [raster] Optional.

<put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Deterministic 8
- 1 — [1] Deterministic Infinity
- 2 — [2] Multiple Flow Direction

Default: *0*

Convergence [number] <put parameter description here>

Default: *1.1*

Outputs

Upslope Area [raster] <put output description here>

Console usage

```
processing.runalg('saga:upslopearea', target, target_pt_x, target_pt_y, elevation, sinkroute, met
```

See also

.

18.7.26 Terrain lighting

Analytical hillshading

Description

<put algorithm description here>

Parameters

Elevation [**raster**] <put parameter description here>

Shading Method [**selection**] <put parameter description here>

Options:

- 0 — [0] Standard
- 1 — [1] Standard (max. 90Degree)
- 2 — [2] Combined Shading
- 3 — [3] Ray Tracing

Default: 0

Azimuth [**Degree**] [**number**] <put parameter description here>

Default: 315.0

Declination [**Degree**] [**number**] <put parameter description here>

Default: 45.0

Exaggeration [**number**] <put parameter description here>

Default: 4.0

Outputs

Analytical Hillshading [**raster**] <put output description here>

Console usage

```
processing.runalg('saga:analyticalhillshading', elevation, method, azimuth, declination, exaggeration)
```

See also

Sky view factor

Description

<put algorithm description here>

Parameters

Elevation [**raster**] <put parameter description here>

Maximum Search Radius [**number**] <put parameter description here>

Default: 10000

Method [**selection**] <put parameter description here>

Options:

- 0 — [0] multi scale
- 1 — [1] sectors

Default: 0

Multi Scale Factor [number] <put parameter description here>

Default: 3

Number of Sectors [number] <put parameter description here>

Default: 8

Outputs

Visible Sky [raster] <put output description here>

Sky View Factor [raster] <put output description here>

Sky View Factor (Simplified) [raster] <put output description here>

Terrain View Factor [raster] <put output description here>

Console usage

```
processing.runalg('saga:skyviewfactor', dem, maxradius, method, level_inc, ndirs, visible, svf, s
```

See also

Topographic correction

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Original Image [raster] <put parameter description here>

Azimuth [number] <put parameter description here>

Default: 180.0

Height [number] <put parameter description here>

Default: 45.0

Method [selection] <put parameter description here>

Options:

- 0 — [0] Cosine Correction (Teillet et al. 1982)
- 1 — [1] Cosine Correction (Civco 1989)
- 2 — [2] Minnaert Correction
- 3 — [3] Minnaert Correction with Slope (Riano et al. 2003)
- 4 — [4] Minnaert Correction with Slope (Law & Nichol 2004)
- 5 — [5] C Correction
- 6 — [6] Normalization (after Civco, modified by Law & Nichol)

Default: 0

Minnaert Correction [number] <put parameter description here>

Default: 0.5

Maximum Cells (C Correction Analysis) [number] <put parameter description here>

Default: 1000

Value Range [selection] <put parameter description here>

Options:

- 0 — [0] 1 byte (0-255)
- 1 — [1] 2 byte (0-65535)

Default: 0

Outputs

Corrected Image [raster] <put output description here>

Console usage

```
processing.runalg('saga:topographiccorrection', dem, original, azi, hgt, method, minnaert, maxcel.
```

See also

.

18.7.27 Terrain morphometry

Convergence index

Description

Calculates an index of convergence/divergence regarding to overland flow. By its meaning it is similar to plan or horizontal curvature, but gives much smoother results. The calculation uses the aspects of surrounding cells, i.e. it looks to which degree surrounding cells point to the center cell. The result is given as percentages, negative values correspond to convergent, positive to divergent flow conditions. Minus 100 would be like a peak of a cone, plus 100 a pit, and 0 an even slope.

Parameters

Elevation [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Aspect
- 1 — [1] Gradient

Default: 0

Gradient Calculation [selection] <put parameter description here>

Options:

- 0 — [0] 2 x 2

- 1 — [1] 3 x 3

Default: 0

Outputs

Convergence Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:convergenceindex', elevation, method, neighbours, result)
```

See also

- Koethe, R. / Lehmeier, F. (1996): 'SARA, System zur Automatischen Relief-Analyse', Benutzerhandbuch, 2. Auflage [Geogr. Inst. Univ. Goettingen, unpublished]

Convergence index (search radius)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Radius [Cells] [number] <put parameter description here>

Default: 10

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [number] <put parameter description here>

Default: 1

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: 1

Gradient [boolean] <put parameter description here>

Default: *True*

Difference [selection] <put parameter description here>

Options:

- 0 — [0] direction to the center cell
- 1 — [1] center cell's aspect direction

Default: 0

Outputs

Convergence Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:convergenceindexsearchradius', elevation, radius, distance_weighting_weight)
```

See also

Curvature classification

Description

<put algorithm description here>

Parameters

Plan Curvature [raster] <put parameter description here>

Profile Curvature [raster] <put parameter description here>

Threshold for plane [number] <put parameter description here>

Default: 0.001

Outputs

Curvature Classification [raster] <put output description here>

Console usage

```
processing.runalg('saga:curvatureclassification', cplan, cprof, threshold, class)
```

See also

Diurnal anisotropic heating

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Alpha Max (Degree) [number] <put parameter description here>

Default: *202.5*

Outputs

Diurnal Anisotropic Heating [raster] <put output description here>

Console usage

```
processing.runalg('saga:diurnalanisotropicheating', dem, alpha_max, dah)
```

See also

Downslope distance gradient

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Vertical Distance [number] <put parameter description here>

Default: *10*

Output [selection] <put parameter description here>

Options:

- 0 — [0] distance
- 1 — [1] gradient (tangens)
- 2 — [2] gradient (degree)

Default: *0*

Outputs

Gradient [raster] <put output description here>

Gradient Difference [raster] <put output description here>

Console usage

```
processing.runalg('saga:downslopedistancegradient', dem, distance, output, gradient, difference)
```

See also

Effective air flow heights

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Wind Direction [raster] Optional.

<put parameter description here>

Wind Speed [raster] Optional.

<put parameter description here>

Constant Wind Direction [Degree] [number] <put parameter description here>

Default: *135*

Old Version [boolean] <put parameter description here>

Default: *True*

Search Distance [km] [number] <put parameter description here>

Default: *300*

Acceleration [number] <put parameter description here>

Default: *1.5*

Use Pyramids with New Version [boolean] <put parameter description here>

Default: *True*

Lee Factor [number] <put parameter description here>

Default: *0.5*

Luv Factor [number] <put parameter description here>

Default: *1.0*

Wind Direction Units [selection] <put parameter description here>

Options:

- 0 — [0] radians
- 1 — [1] degree

Default: *0*

Wind Speed Scale Factor [number] <put parameter description here>

Default: *1.0*

Outputs

Effective Air Flow Heights [raster] <put output description here>

Console usage

```
processing.runalg('saga:effectiveairflowheights', dem, dir, len, dir_const, oldver, maxdist, acce
```

See also

Hypsometry

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Number of Classes [number] <put parameter description here>

Default: *100.0*

Sort [selection] <put parameter description here>

Options:

- 0 — [0] up
- 1 — [1] down

Default: *0*

Classification Constant [selection] <put parameter description here>

Options:

- 0 — [0] height
- 1 — [1] area

Default: *0*

Use Z-Range [boolean] <put parameter description here>

Default: *True*

Z-Range Min [number] <put parameter description here>

Default: *0.0*

Z-Range Max [number] <put parameter description here>

Default: *1000.0*

Outputs

Hypsometry [table] <put output description here>

Console usage

```
processing.runalg('saga:hypsometry', elevation, count, sorting, method, bzrange, zrange_min, zran
```

See also

Land surface temperature

Description

<put algorithm description here>

Parameters

Elevation [m] [raster] <put parameter description here>

Short Wave Radiation [kW/m2] [raster] <put parameter description here>

Leaf Area Index [raster] <put parameter description here>

Elevation at Reference Station [m] [number] <put parameter description here>

Default: 0.0

Temperature at Reference Station [Deg.Celsius] [number] <put parameter description here>

Default: 0.0

Temperature Gradient [Deg.Celsius/km] [number] <put parameter description here>

Default: 6.5

C Factor [number] <put parameter description here>

Default: 1.0

Outputs

Land Surface Temperature [Deg.Celsius] [raster] <put output description here>

Console usage

```
processing.runalg('saga:landsurfacetemperature', dem, swr, lai, z_reference, t_reference, t_gradient)
```

See also

Mass balance index

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Vertical Distance to Channel Network [raster] Optional.

<put parameter description here>

T Slope [number] <put parameter description here>

Default: *15.0*

T Curvature [number] <put parameter description here>

Default: *0.01*

T Vertical Distance to Channel Network [number] <put parameter description here>

Default: *15.0*

Outputs

Mass Balance Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:massbalanceindex', dem, hrel, tslope, tcurve, threl, mbi)
```

See also

Morphometric protection index

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Radius [number] <put parameter description here>

Default: *2000.0*

Outputs

Protection Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:morphometricprotectionindex', dem, radius, protection)
```

See also

Multiresolution index of valley bottom flatness (mrvbf)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Initial Threshold for Slope [number] <put parameter description here>

Default: *16*

Threshold for Elevation Percentile (Lowness) [number] <put parameter description here>

Default: *0.4*

Threshold for Elevation Percentile (Upness) [number] <put parameter description here>

Default: *0.35*

Shape Parameter for Slope [number] <put parameter description here>

Default: *4.0*

Shape Parameter for Elevation Percentile [number] <put parameter description here>

Default: *3.0*

Update Views [boolean] <put parameter description here>

Default: *True*

Classify [boolean] <put parameter description here>

Default: *True*

Maximum Resolution (Percentage) [number] <put parameter description here>

Default: *100*

Outputs

MRVBF [raster] <put output description here>

MRRTF [raster] <put output description here>

Console usage

```
processing.runalg('saga:multiresolutionindexofvalleybottomflatnessmrvbf', dem, t_slope, t_pctl_v,
```

See also

Real area calculation

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Outputs

Real Area Grid [raster] <put output description here>

Console usage

```
processing.runalg('saga:realareacalculation', dem, area)
```

See also

Relative heights and slope positions

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

w [number] <put parameter description here>

Default: *0.5*

t [number] <put parameter description here>

Default: *10.0*

e [number] <put parameter description here>

Default: *2.0*

Outputs

Slope Height [raster] <put output description here>

Valley Depth [raster] <put output description here>

Normalized Height [raster] <put output description here>

Standardized Height [raster] <put output description here>

Mid-Slope Positon [raster] <put output description here>

Console usage

```
processing.runalg('saga:relativeheightsandslopepositions', dem, w, t, e, ho, hu, nh, sh, ms)
```

See also

Slope, aspect, curvature

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Maximum Slope (Travis et al. 1975)
- 1 — [1] Maximum Triangle Slope (Tarboton 1997)
- 2 — [2] Least Squares Fitted Plane (Horn 1981, Costa-Cabral & Burgess 1996)
- 3 — [3] Fit 2.Degree Polynom (Bauer, Rohdenburg, Bork 1985)
- 4 — [4] Fit 2.Degree Polynom (Heerdegen & Beran 1982)
- 5 — [5] Fit 2.Degree Polynom (Zevenbergen & Thorne 1987)
- 6 — [6] Fit 3.Degree Polynom (Haralick 1983)

Default: 5

Outputs

Slope [raster] <put output description here>

Aspect [raster] <put output description here>

Curvature [raster] <put output description here>

Plan Curvature [raster] <put output description here>

Profile Curvature [raster] <put output description here>

Console usage

```
processing.runalg('saga:slopeaspectcurvature', elevation, method, slope, aspect, curv, hcurv, vcurv)
```

See also

Surface specific points

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Method [selection] <put parameter description here>

Options:

- 0 — [0] Mark Highest Neighbour
- 1 — [1] Opposite Neighbours
- 2 — [2] Flow Direction
- 3 — [3] Flow Direction (up and down)

- 4 — [4] Peucker & Douglas

Default: 0

Threshold [number] <put parameter description here>

Default: 2.0

Outputs

Result [raster] <put output description here>

Console usage

```
processing.runalg('saga:surfacespecificpoints', elevation, method, threshold, result)
```

See also

Terrain ruggedness index (tri)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Radius (Cells) [number] <put parameter description here>

Default: 1

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: 0

Inverse Distance Weighting Power [number] <put parameter description here>

Default: 1

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: 1.0

Outputs

Terrain Ruggedness Index (TRI) [raster] <put output description here>

Console usage

```
processing.runalg('saga:terrainruggednessindextri', dem, radius, distance_weighting_weighting, di
```

See also

Topographic position index (tpi)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Standardize [boolean] <put parameter description here>

Default: *True*

Min Radius [number] <put parameter description here>

Default: *0.0*

Max Radius [number] <put parameter description here>

Default: *100.0*

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *75.0*

Outputs

Topographic Position Index [raster] <put output description here>

Console usage

```
processing.runalg('saga:topographicpositionindextpi', dem, standard, radius_min, radius_max, dist
```

See also

Tpi based landform classification

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Min Radius A [number] <put parameter description here>

Default: *0*

Max Radius A [number] <put parameter description here>

Default: *100*

Min Radius B [number] <put parameter description here>

Default: *0*

Max Radius B [number] <put parameter description here>

Default: *1000*

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *75.0*

Outputs

Landforms [raster] <put output description here>

Console usage

```
processing.runalg('saga:tpibasedlandformclassification', dem, radius_a_min, radius_a_max, radius_b_min, radius_b_max, distance_weighting, inverse_distance_weighting_power, inverse_distance_offset, gaussian_and_exponential_weighting_bandwidth)
```

See also

Vector ruggedness measure (vrm)

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Radius (Cells) [number] <put parameter description here>

Default: *1*

Distance Weighting [selection] <put parameter description here>

Options:

- 0 — [0] no distance weighting
- 1 — [1] inverse distance to a power
- 2 — [2] exponential
- 3 — [3] gaussian weighting

Default: *0*

Inverse Distance Weighting Power [number] <put parameter description here>

Default: *1*

Inverse Distance Offset [boolean] <put parameter description here>

Default: *True*

Gaussian and Exponential Weighting Bandwidth [number] <put parameter description here>

Default: *1*

Outputs

Vector Terrain Ruggedness (VRM) [raster] <put output description here>

Console usage

```
processing.runalg('saga:vectorruggednessmeasurevrm', dem, radius, distance_weighting_weighting, d
```

See also

Wind effect

Description

<put algorithm description here>

Parameters

Elevation [raster] <put parameter description here>

Wind Direction [raster] Optional.

<put parameter description here>

Wind Speed [raster] Optional.

<put parameter description here>

Constant Wind Direction [Degree] [number] <put parameter description here>

Default: *135*

Old Version [boolean] <put parameter description here>

Default: *True*

Search Distance [km] [number] <put parameter description here>

Default: *300.0*

Acceleration [number] <put parameter description here>

Default: *1.5*

Use Pyramids [boolean] <put parameter description here>

Default: *True*

Wind Direction Units [selection] <put parameter description here>

Options:

- 0 — [0] radians
- 1 — [1] degree

Default: *0*

Wind Speed Scale Factor [number] <put parameter description here>

Default: *1.0*

Outputs

Wind Effect [raster] <put output description here>

Windward Effect [raster] <put output description here>

Leeward Effect [raster] <put output description here>

Console usage

```
processing.runalg('saga:windeffect', dem, dir, len, dir_const, oldver, maxdist, accel, pyramids, o
```

See also

.

18.7.28 Terrain profiles

Cross profiles

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Lines [vector: line] <put parameter description here>

Profile Distance [number] <put parameter description here>

Default: *10.0*

Profile Length [number] <put parameter description here>

Default: *10.0*

Profile Samples [number] <put parameter description here>

Default: *10.0*

Outputs

Cross Profiles [vector] <put output description here>

Console usage

```
processing.runalg('saga:crossprofiles', dem, lines, dist_line, dist_profile, num_profile, profile)
```

See also

Profile from points table

Description

<put algorithm description here>

Parameters

Grid [raster] <put parameter description here>

Input [table] <put parameter description here>

X [tablefield: any] <put parameter description here>

Y [tablefield: any] <put parameter description here>

Outputs

Result [table] <put output description here>

Console usage

```
processing.runalg('saga:profilefrompointstable', grid, table, x, y, result)
```

See also

Profiles from lines

Description

<put algorithm description here>

Parameters

DEM [raster] <put parameter description here>

Values [multipleinput: rasters] Optional.

<put parameter description here>

Lines [vector: line] <put parameter description here>

Name [tablefield: any] <put parameter description here>

Each Line as new Profile [boolean] <put parameter description here>

Default: *True*

Outputs

Profiles [vector] <put output description here>

Profiles [vector] <put output description here>

Console usage

```
processing.runalg('saga:profilesfromlines', dem, values, lines, name, split, profile, profiles)
```

See also

.

18.8 TauDEM algorithm provider

TauDEM (Terrain Analysis Using Digital Elevation Models) is a set of Digital Elevation Model (DEM) tools for the extraction and analysis of hydrologic information from topography as represented by a DEM. This is software developed at Utah State University (USU) for hydrologic digital elevation model analysis and watershed delineation.

TauDEM is distributed as a set of standalone command line executable programs for a Windows and source code for compiling and use on other systems.

Nota: Please remember that Processing contains only the interface description, so you need to install TauDEM 5.0.6 by yourself and configure Processing properly.

Documentation for TauDEM algorithms derived from official [TauDEM documentation](#)

18.8.1 Basic Grid Analysis

D8 Contributing Area

Description

Calculates a grid of contributing areas using the single direction D8 flow model. The contribution of each grid cell is taken as one (or when the optional weight grid is used, the value from the weight grid). The contributing area for each grid cell is taken as its own contribution plus the contribution from upslope neighbors that drain in to it according to the D8 flow model.

If the optional outlet point shapefile is used, only the outlet cells and the cells upslope (by the D8 flow model) of them are in the domain to be evaluated.

By default, the tool checks for edge contamination. This is defined as the possibility that a contributing area value may be underestimated due to grid cells outside of the domain not being counted. This occurs when drainage is inwards from the boundaries or areas with “no data” values for elevation. The algorithm recognizes this and reports “no data” for the contributing area. It is common to see streaks of “no data” values extending inwards from boundaries along flow paths that enter the domain at a boundary. This is the desired effect and indicates that contributing area for these grid cells is unknown due to it being dependent on terrain outside of the domain of data available. Edge contamination checking may be turned off in cases where you know this is not an issue or want to ignore these problems, if for example, the DEM has been clipped along a watershed outline.

Parameters

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Outlets Shapefile [vector: point] Optional.

A point shapefile defining the outlets of interest. If this input file is used, only the cells upslope of these outlet cells are considered to be within the domain being evaluated.

Weight Grid [raster] Optional.

A grid giving contribution to flow for each cell. These contributions (also sometimes referred to as weights or loadings) are used in the contributing area accumulation. If this input file is not used, the contribution to flow will assumed to be one for each grid cell.

Check for edge contamination [boolean] A flag that indicates whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a contributing area value may be underestimated due to the fact that grid cells outside of the domain have not been evaluated. This occurs when drainage is inwards from the boundaries or areas with NODATA values for elevation. The algorithm recognizes this and reports NODATA for the impated cells. It is common to see streaks of NODATA values extending inwards from boundaries along flow paths that enter the domain at a boundary. This is the desired effect and indicates that contributing area for these grid cells is unknown due to it being dependent on terrain outside of the domain of available data. Edge contamination checking may be turned off in cases where you know this is not an issue, or want to ignore these problems, if for example, the DEM has been clipped along a watershed outline.

Default: *True*

Outputs

D8 Contributing Area Grid [raster] A grid of contributing area values calculated as the cells own contribution plus the contribution from upslope neighbors that drain in to it according to the D8 flow model.

Console usage

```
processing.runalg('taudem:d8contributingarea', -p, -o, -wg, -nc, -ad8)
```

See also

D8 Flow Directions

Description

Creates 2 grids. The first contains the flow direction from each grid cell to one of its adjacent or diagonal neighbors, calculated using the direction of steepest descent. The second contain the slope, as evaluated in the direction of steepest descent, and is reported as drop/distance, i.e. tan of the angle. Flow direction is reported as NODATA for any grid cell adjacent to the edge of the DEM domain, or adjacent to a NODATA value in the DEM. In flat areas, flow directions are assigned away from higher ground and towards lower ground using the method of Garbrecht and Martz (1997). The D8 flow direction algorithm may be applied to a DEM that has not had its pits filled, but it will then result in NODATA values for flow direction and slope at the lowest point of each pit.

D8 Flow Direction Coding:

- 1 — East
- 2 — Northeast
- 3 — North
- 4 — Northwest
- 5 — West
- 6 — Southwest
- 7 — South
- 8 — Southeast

The flow direction routing across flat areas is performed according to the method described by Garbrecht, J. and L. W. Martz, (1997), “The Assignment of Drainage Direction Over Flat Surfaces in Raster Digital Elevation Models”, *Journal of Hydrology*, 193: 204-213.

Parameters

Pit Filled Elevation Grid [raster] A grid of elevation values. This is usually the output of the “**Pit Remove**” tool, in which case it is elevations with pits removed. Pits are low elevation areas in digital elevation models (DEMs) that are completely surrounded by higher terrain. They are generally taken to be artifacts of the digitation process that interfere with the processing of flow across DEMs. So they are removed by raising their elevation to the point where they just drain off the domain. This step is not essential if you have reason to believe that the pits in your DEM are real. If a few pits actually exist and so should not be removed, while at the same time others are believed to be artifacts that need to be removed, the actual pits should have NODATA elevation values inserted at their lowest point. NODATA values serve to define edges of the domain in the flow field, and elevations are only raised to where flow is off an edge, so an internal NODATA value will stop a pit from being removed, if necessary.

Outputs

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope.

D8 Slope Grid [raster] A grid giving slope in the D8 flow direction. This is measured as drop/distance.

Console usage

```
processing.runalg('taudem:d8flowdirections', -fel, -p, -sd8)
```

See also

D-Infinity Contributing Area

Description

Calculates a grid of specific catchment area which is the contributing area per unit contour length using the multiple flow direction D-infinity approach. D-infinity flow direction is defined as steepest downward slope on planar triangular facets on a block centered grid. The contribution at each grid cell is taken as the grid cell length (or when the optional weight grid input is used, from the weight grid). The contributing area of each grid cell is then taken as its own contribution plus the contribution from upslope neighbors that have some fraction draining to it according to the D-infinity flow model. The flow from each cell either all drains to one neighbor, if the angle falls along a cardinal ($0, \pi/2, \pi, 3\pi/2$) or ordinal ($\pi/4, 3\pi/4, 5\pi/4, 7\pi/4$) direction, or is on an angle falling between the direct angle to two adjacent neighbors. In the latter case the flow is proportioned between these two neighbor cells according to how close the flow direction angle is to the direct angle to those cells. The contour length used here is the grid cell size. The resulting units of the specific catchment area are length units the same as those of the grid cell size.

When the optional weight grid is not used, the result is reported in terms of specific catchment area, the upslope area per unit contour length, taken here as the number of cells times grid cell length (cell area divided by cell length). This assumes that grid cell length is the effective contour length, in the definition of specific catchment area and does not distinguish any difference in contour length dependent upon the flow direction. When the optional weight grid is used, the result is reported directly as a summation of weights, without any scaling.

If the optional outlet point shapefile is used, only the outlet cells and the cells upslope (by the D-infinity flow model) of them are in the domain to be evaluated.

By default, the tool checks for edge contamination. This is defined as the possibility that a contributing area value may be underestimated due to grid cells outside of the domain not being counted. This occurs when drainage is inwards from the boundaries or areas with “no data” values for elevation. The algorithm recognizes this and reports “no data” for the contributing area. It is common to see streaks of “no data” values extending inwards from boundaries along flow paths that enter the domain at a boundary. This is the desired effect and indicates that contributing area for these grid cells is unknown due to it being dependent on terrain outside of the domain of data available. Edge contamination checking may be turned off in cases where you know it is not an issue or want to ignore these problems, if for example, the DEM has been clipped along a watershed outline.

Parameters

D-Infinity Flow Direction Grid [raster] A grid of flow directions based on the D-infinity flow method using the steepest slope of a triangular facet. Flow direction is determined as the direction of the steepest downward slope on the 8 triangular facets of a 3x3 block centered grid. Flow direction is encoded as an angle in radians, counter-clockwise from east as a continuous (floating point) quantity between 0 and 2π . The resulting flow in a grid is then usually interpreted as being proportioned between the two neighboring cells that define the triangular facet with the steepest downward slope.

Outlets Shapefile [vector: point] Optional.

A point shapefile defining the outlets of interest. If this input file is used, only the cells upslope of these outlet cells are considered to be within the domain being evaluated.

Weight Grid [raster] Optional.

A grid giving contribution to flow for each cell. These contributions (also sometimes referred to as weights or loadings) are used in the contributing area accumulation. If this input file is not used, the result is reported in terms of specific catchment area (the upslope area per unit contour length) taken as the number of cells times grid cell length (cell area divided by cell length).

Check for edge contamination [boolean] A flag that indicates whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a contributing area value may be underestimated due to the fact that grid cells outside of the domain have not been evaluated. This occurs when drainage is inwards from the boundaries or areas with NODATA values for elevation. The algorithm recognizes this and reports NODATA for the impated cells. It is common to see streaks of NODATA values extending inwards from boundaries along flow paths that enter the domain at a boundary. This is the desired effect and indicates that contributing area for these grid cells is unknown due to it being dependent on terrain outside of the domain of available data. Edge contamination checking may be turned off in cases where you know this is not an issue, or want to ignore these problems, if for example, the DEM has been clipped along a watershed outline.

Default: *True*

Outputs

D-Infinity Specific Catchment Area Grid [raster] A grid of specific catchment area which is the contributing area per unit contour length using the multiple flow direction D-infinity approach. The contributing area of each grid cell is then taken as its own contribution plus the contribution from upslope neighbors that have some fraction draining to it according to the D-infinity flow model.

Console usage

```
processing.runalg('taudem:dinfinitecontributingarea', -ang, -o, -wg, -nc, -sca)
```

See also

D-Infinity Flow Directions

Description

Assigns a flow direction based on the D-infinity flow method using the steepest slope of a triangular facet (Tarboton, 1997, "A New Method for the Determination of Flow Directions and Contributing Areas in Grid Digital Elevation Models", Water Resources Research, 33(2): 309-319). Flow direction is defined as steepest downward slope on planar triangular facets on a block centered grid. Flow direction is encoded as an angle in radians counter-clockwise from east as a continuous (floating point) quantity between 0 and 2π . The flow direction angle is determined as the direction of the steepest downward slope on the eight triangular facets formed in a 3 x 3 grid cell window centered on the grid cell of interest. The resulting flow in a grid is then usually interpreted as being proportioned between the two neighboring cells that define the triangular facet with the steepest downward slope.

A block-centered representation is used with each elevation value taken to represent the elevation of the center of the corresponding grid cell. Eight planar triangular facets are formed between each grid cell and its eight neighbors. Each of these has a downslope vector which when drawn outwards from the center may be at an angle that lies within or outside the 45 degree ($\pi/4$ radian) angle range of the facet at the center point. If the slope vector

angle is within the facet angle, it represents the steepest flow direction on that facet. If the slope vector angle is outside a facet, the steepest flow direction associated with that facet is taken along the steepest edge. The slope and flow direction associated with the grid cell is taken as the magnitude and direction of the steepest downslope vector from all eight facets. Slope is measured as drop/distance, i.e. \tan of the slope angle.

In the case where no slope vectors are positive (downslope), the flow direction is set using the method of Garbrecht and Martz (1997) for the determination of flow across flat areas. This makes flat areas drain away from high ground and towards low ground. The flow path grid to enforce drainage along existing streams is an optional input, and if used, takes precedence over elevations for the setting of flow directions.

The D-infinity flow direction algorithm may be applied to a DEM that has not had its pits filled, but it will then result in “no data” values for the D-infinity flow direction and slope associated with the lowest point of the pit.

Parameters

Pit Filled Elevation Grid [raster] A grid of elevation values. This is usually the output of the “**Pit Remove**” tool, in which case it is elevations with pits removed.

Outputs

D-Infinity Flow Directions Grid [raster] A grid of flow directions based on the D-infinity flow method using the steepest slope of a triangular facet. Flow direction is determined as the direction of the steepest downward slope on the 8 triangular facets of a 3x3 block centered grid. Flow direction is encoded as an angle in radians, counter-clockwise from east as a continuous (floating point) quantity between 0 and 2π . The resulting flow in a grid is then usually interpreted as being proportioned between the two neighboring cells that define the triangular facet with the steepest downward slope.

D-Infinity Slope Grid [raster] A grid of slope evaluated using the D-infinity method described in Tarboton, D. G., (1997), “A New Method for the Determination of Flow Directions and Contributing Areas in Grid Digital Elevation Models”, *Water Resources Research*, 33(2): 309-319. This is the steepest outwards slope on one of eight triangular facets centered at each grid cell, measured as drop/distance, i.e. \tan of the slope angle.

Console usage

```
processing.runalg('taudem:dinfiniteflowdirections', -fel, -ang, -slp)
```

See also

Grid Network

Description

Creates 3 grids that contain for each grid cell: 1) the longest path, 2) the total path, and 3) the Strahler order number. These values are derived from the network defined by the D8 flow model.

The longest upslope length is the length of the flow path from the furthest cell that drains to each cell. The total upslope path length is the length of the entire grid network upslope of each grid cell. Lengths are measured between cell centers taking into account cell size and whether the direction is adjacent or diagonal.

Strahler order is defined as follows: A network of flow paths is defined by the D8 Flow Direction grid. Source flow paths have a Strahler order number of one. When two flow paths of different order join the order of the downstream flow path is the order of the highest incoming flow path. When two flow paths of equal order join the downstream flow path order is increased by 1. When more than two flow paths join the downstream flow path order is calculated as the maximum of the highest incoming flow path order or the second highest incoming flow path order + 1. This generalizes the common definition to cases where more than two flow paths join at a point.

Where the optional mask grid and threshold value are input, the function is evaluated only considering grid cells that lie in the domain with mask grid value greater than or equal to the threshold value. Source (first order) grid cells are taken as those that do not have any other grid cells from inside the domain draining in to them, and only when two of these flow paths join is order propagated according to the ordering rules. Lengths are also only evaluated counting paths within the domain greater than or equal to the threshold.

If the optional outlet point shapefile is used, only the outlet cells and the cells upslope (by the D8 flow model) of them are in the domain to be evaluated.

Parameters

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Outlets Shapefile [vector: point] Optional.

A point shapefile defining the outlets of interest. If this input file is used, only the cells upslope of these outlet cells are considered to be within the domain being evaluated.

Mask Grid [raster] Optional.

A grid that is used to determine the domain to be analyzed. If the mask grid value \geq mask threshold (see below), then the cell will be included in the domain. While this tool does not have an edge contamination flag, if edge contamination analysis is needed, then a mask grid from a function like “**D8 Contributing Area**” that does support edge contamination can be used to achieve the same result.

Mask Threshold [number] This input parameter is used in the calculation mask grid value \geq mask threshold to determine if the grid cell is in the domain to be analyzed.

Default: *100*

Outputs

Longest Upslope Length Grid [raster] A grid that gives the length of the longest upslope D8 flow path terminating at each grid cell. Lengths are measured between cell centers taking into account cell size and whether the direction is adjacent or diagonal.

Total Upslope Length Grid [raster] The total upslope path length is the length of the entire D8 flow grid network upslope of each grid cell. Lengths are measured between cell centers taking into account cell size and whether the direction is adjacent or diagonal.

Strahler Network Order Grid [raster] A grid giving the Strahler order number for each cell. A network of flow paths is defined by the D8 Flow Direction grid. Source flow paths have a Strahler order number of one. When two flow paths of different order join the order of the downstream flow path is the order of the highest incoming flow path. When two flow paths of equal order join the downstream flow path order is increased by 1. When more than two flow paths join the downstream flow path order is calculated as the maximum of the highest incoming flow path order or the second highest incoming flow path order + 1. This generalizes the common definition to cases where more than two flow paths join at a point.

Console usage

```
processing.runalg('taudem:gridnetwork', d8_flow_dir_grid, outlets_shape, mask_grid, threshold, lo
```

See also

Pit Remove

Description

Identifies all pits in the DEM and raises their elevation to the level of the lowest pour point around their edge. Pits are low elevation areas in digital elevation models (DEMs) that are completely surrounded by higher terrain. They are generally taken to be artifacts that interfere with the routing of flow across DEMs, so are removed by raising their elevation to the point where they drain off the edge of the domain. The pour point is the lowest point on the boundary of the “watershed” draining to the pit. This step is not essential if you have reason to believe that the pits in your DEM are real. If a few pits actually exist and so should not be removed, while at the same time others are believed to be artifacts that need to be removed, the actual pits should have NODATA elevation values inserted at their lowest point. NODATA values serve to define edges in the domain, and elevations are only raised to where flow is off an edge, so an internal NODATA value will stop a pit from being removed, if necessary.

Parameters

Elevation Grid [raster] A digital elevation model (DEM) grid to serve as the base input for the terrain analysis and stream delineation.

Outputs

Pit Removed Elevation Grid [raster] A grid of elevation values with pits removed so that flow is routed off of the domain.

Console usage

```
processing.runalg('taudem:pitremove', -z, -fel)
```

See also

.

18.8.2 Specialized Grid Analysis

D8 Distance To Streams

Description

Computes the horizontal distance to stream for each grid cell, moving downslope according to the D8 flow model, until a stream grid cell is encountered.

Parameters

D8 Flow Direction Grid [raster] This input is a grid of flow directions that are encoded using the D8 method where all flow from a cells goes to a single neighboring cell in the direction of steepest descent. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Stream Raster Grid [raster] A grid indicating streams. Such a grid can be created by several of the tools in the “**Stream Network Analysis**” toolset. However, the tools in the “**Stream Network Analysis**” toolset only create grids with a value of 0 for no stream, or 1 for stream cells. This tool can also accept grids with values greater than 1, which can be used in conjunction with the `Threshold` parameter to determine the location of streams. This allows Contributing Area grids to be used to define streams as well as the normal Stream Raster grids. This grid expects integer (long integer) values and any non-integer values will be truncated to an integer before being evaluated.

Threshold [number] This value acts as threshold on the `Stream Raster Grid` to determine the location of streams. Cells with a `Stream Raster Grid` value greater than or equal to the `Threshold` value are interpreted as streams.

Default: 50

Outputs

Output Distance to Streams [raster] A grid giving the horizontal distance along the flow path as defined by the D8 Flow Directions Grid to the streams in the Stream Raster Grid.

Console usage

```
processing.runalg('taudem:d8distancetostreams', -p, -src, -thresh, -dist)
```

See also

D-Infinity Avalanche Runout

Description

Identifies an avalanche’s affected area and the flow path length to each cell in that affected area. All cells downslope from each source area cell, up to the point where the slope from the source to the affected area is less than a threshold angle called the Alpha Angle can be in the affected area. This tool uses the D-infinity multiple flow direction method for determining flow direction. This will likely cause very small amounts of flow to be dispersed to some downslope cells that might overstate the affected area, so a threshold proportion can be set to avoid this excess dispersion. The flow path length is the distance from the cell in question to the source cell that has the highest angle.

All points downslope from the source area are potentially in the affected area, but not beyond a point where the slope from the source to the affected area is less than a threshold angle called the Alpha Angle.

Slope is to be measured using the straight line distance from source point to evaluation point.

It makes more physical sense to me for the angle to be measured along the flow path. Nevertheless it is equally easy to code straight line angles as angles along the flow path, so an option that allows switching will be provided. The most practical way to evaluate avalanche runout is to keep track of the source point with the greatest angle to each point. Then the recursive upslope flow algebra approach will look at a grid cell and all its upslope neighbors that flow to it. Information from the upslope neighbors will be used to calculate the angle to the grid cell in question and retain it in the runout zone if the angle exceeds the alpha angle. This procedure makes the assumption that the maximum angle at a grid cell will be from the set of cells that have maximum angles to the inflowing neighbors. This will always be true of angle is calculated along a flow path, but I can conceive of cases where flow paths bend back on themselves where this would not be the case for straight line angles.

The D-infinity multiple flow direction field assigns flow from each grid cell to multiple downslope neighbors using proportions (P_{ik}) that vary between 0 and 1 and sum to 1 for all flows out of a grid cell. It may be desirable to specify a threshold T that this proportion has to exceed before a grid cell is counted as flowing to a downslope

grid cell, e.g. $P_{ik} > T$ ($=0.2$ say) to avoid dispersion to grid cells that get very little flow. T will be specified as a user input. If all upslope grid cells are to be used T may be input as 0.

Avalanche source sites are to be input as a short integer grid (name suffix **ass*, e.g. *demass*) comprised of positive values where avalanches may be triggered and 0 values elsewhere.

The following grids are output:

- *rz* — A runout zone indicator with value 0 to indicate that this grid cell is not in the runout zone and value > 0 to indicate that this grid cell is in the runout zone. Since there may be information in the angle to the associated source site, this variable will be assigned the angle to the source site (in degrees)
- *dm* — Along flow distance from the source site that has the highest angle to the point in question

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Pit Filled Elevation Grid [raster] This input is a grid of elevation values. As a general rule, it is recommended that you use a grid of elevation values that have had the pits removed for this input. Pits are generally taken to be artifacts that interfere with the analysis of flow across them. This grid can be obtained as the output of the “**Pit Remove**” tool, in which case it contains elevation values where the pits have been filled to the point where they just drain.

Avalanche Source Site Grid [raster] This is a grid of source areas for snow avalanches that are commonly identified manually using a mix of experience and visual interpretation of maps. Avalanche source sites are to be input as a short integer grid (name suffix **ass*, e.g. *demass*) comprised of positive values where avalanches may be triggered and 0 values elsewhere.

Proportion Threshold [number] This value is a threshold proportion that is used to limit the dispersion of flow caused by using the D-infinity multiple flow direction method for determining flow direction. The D-infinity multiple flow direction method often causes very small amounts of flow to be dispersed to some downslope cells that might overstate the affected area, so a threshold proportion can be set to avoid this excess dispersion.

Default: *0.2*

Alpha Angle Threshold [number] This value is the threshold angle, called the Alpha Angle, that is used to determine which of the cells downslope from the source cells are in the affected area. Only the cells downslope from each source area cell, up to the point where the slope from the source to the affected area is less than a threshold angle are in the affected area.

Default: *18*

Measure distance along flow path [boolean] This option selects the method used to measure the distance used to calculate the slope angle. If option is *True* then measure it along the flow path, where the *False* option causes the slope to be measure along the straight line distance from the source cell to the evaluation cell.

Default: *True*

Outputs

Runout Zone Grid [raster] This grid Identifies the avalanche’s runout zone (affected area) using a runout zone indicator with value 0 to indicate that this grid cell is not in the runout zone and value > 0 to indicate that this grid cell is in the runout zone. Since there may be information in the angle to the associated source site, this variable will be assigned the angle to the source site (in degrees).

Path Distance Grid [raster] This is a grid of the flow distance from the source site that has the highest angle to each cell.

Console usage

```
processing.runalg('taudem:dinfinityavalancherunout', -ang, -fel, -ass, -thresh, -alpha, -direct, ...)
```

See also

D-Infinity Concentration Limited Accumulation

Description

This function applies to the situation where an unlimited supply of a substance is loaded into flow at a concentration or solubility threshold C_{sol} over a region indicated by an indicator grid (dg). It a grid of the concentration of a substance at each location in the domain, where the supply of substance from a supply area is loaded into the flow at a concentration or solubility threshold. The flow is first calculated as a D-infinity weighted contributing area of an input Effective Runoff Weight Grid (notionally excess precipitation). The concentration of substance over the supply area (indicator grid) is at the concentration threshold. As the substance moves downslope with the D-infinity flow field, it is subject to first order decay in moving from cell to cell as well as dilution due to changes in flow. The decay multiplier grid gives the fractional (first order) reduction in quantity in moving from grid cell x to the next downslope cell. If the outlets shapefile is used, the tool only evaluates the part of the domain that contributes flow to the locations given by the shapefile. This is useful for a tracking a contaminant or compound from an area with unlimited supply of that compound that is loaded into a flow at a concentration or solubility threshold over a zone and flow from the zone may be subject to decay or attenuation.

The indicator grid (dg) is used to delineate the area of the substance supply using the (0, 1) indicator function $i(x)$. $A[]$ denotes the weighted accumulation operator evaluated using the D-Infinity Contributing Area function. The Effective Runoff Weight Grid gives the supply to the flow (e.g. the excess rainfall if this is overland flow) denoted as $w(x)$. The specific discharge is then given by:

$$Q(x) = A[w(x)]$$

This weighted accumulation $Q(x)$ is output as the Overland Flow Specific Discharge Grid. Over the substance supply area concentration is at the threshold (the threshold is a saturation or solubility limit). If $i(x) = 1$, then

$$C(x) = C_{sol}, \text{ and } L(x) = C_{sol} Q(x),$$

where $L(x)$ denotes the load being carried by the flow. At remaining locations, the load is determined by load accumulation and the concentration by dilution:

Here $d(x) = d(i, j)$ is a decay multiplier giving the fractional (first order) reduction in mass in moving from grid cell x to the next downslope cell. If travel (or residence) times $t(x)$ associated with flow between cells are available $d(x)$ may be evaluated as $\exp(-k t(x))$ where k is a first order decay parameter. The Concentration grid output is $C(x)$. If the outlets shapefile is used, the tool only evaluates the part of the domain that contributes flow to the locations given by the shapefile.

Useful for a tracking a contaminant released or partitioned to flow at a fixed threshold concentration.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This grid can be created by the function “D-Infinity Flow Directions”.

Disturbance Indicator Grid [raster] A grid that indicates the source zone of the area of substance supply and must be 1 inside the zone and 0 or NODATA over the rest of the domain.

Decay Multiplier Grid [raster] A grid giving the factor by which flow leaving each grid cell is multiplied before accumulation on downslope grid cells. This may be used to simulate the movement of an attenuating or decaying substance. If travel (or residence) times $t(x)$ associated with flow between cells are available $d(x)$ may be evaluated as $\exp(-k t(x))$ where k is a first order decay parameter.

Effective Runoff Weight Grid [raster] A grid giving the input quantity (notionally effective runoff or excess precipitation) to be used in the D-infinity weighted contributing area evaluation of Overland Flow Specific Discharge.

Outlets shapefile [vector: point] Optional.

This optional input is a point shapefile defining outlets of interest. If this file is used, the tool will only evaluate the area upslope of these outlets.

Concentration Threshold [number] The concentration or solubility threshold. Over the substance supply area, concentration is at this threshold.

Default: *1.0*

Check for edge contamination [boolean] This option determines whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being considered when determining contributing area.

Default: *True*

Outputs

Concentration Grid [raster] A grid giving the resulting concentration of the compound of interest in the flow.

Console usage

```
processing.runalg('taudem:dinfiniteconcentrationlimitedaccumulation', -ang, -dg, -dm, -q, -o, -cs
```

See also

D-Infinity Decaying Accumulation

Description

The D-Infinity Decaying Accumulation tool creates a grid of the accumulated quantity at each location in the domain where the quantity accumulates with the D-infinity flow field, but is subject to first order decay in moving from cell to cell. By default, the quantity contribution of each grid cell is the cell length to give a per unit width accumulation, but can optionally be expressed with a weight grid. The decay multiplier grid gives the fractional (first order) reduction in quantity in accumulating from grid cell x to the next downslope cell.

A decayed accumulation operator $DA[.]$ takes as input a mass loading field $m(x)$ expressed at each grid location as $m(i, j)$ that is assumed to move with the flow field but is subject to first order decay in moving from cell to cell. The output is the accumulated mass at each location $DA(x)$. The accumulation of m at each grid cell can be numerically evaluated.

Here $d(x) = d(i, j)$ is a decay multiplier giving the fractional (first order) reduction in mass in moving from grid cell x to the next downslope cell. If travel (or residence) times $t(x)$ associated with flow between cells are available $d(x)$ may be evaluated as $\exp(-k t(x))$ where k is a first order decay parameter. The weight grid is used to represent the mass loading $m(x)$. If not specified this is taken as 1. If the outlets shapefile is used the function is only evaluated on that part of the domain that contributes flow to the locations given by the shapefile.

Useful for a tracking contaminant or compound subject to decay or attenuation.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This grid can be created by the function “**D-Infinity Flow Directions**”.

Decay Multiplier Grid [raster] A grid giving the factor by which flow leaving each grid cell is multiplied before accumulation on downslope grid cells. This may be used to simulate the movement of an attenuating substance.

Weight Grid [raster] Optional.

A grid giving weights (loadings) to be used in the accumulation. If this optional grid is not specified, weights are taken as the linear grid cell size to give a per unit width accumulation.

Outlets Shapefile [vector: point] Optional.

This optional input is a point shapefile defining outlets of interest. If this file is used, the tool will only evaluate the area upslope of these outlets.

Check for edge contamination [boolean] This option determines whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being considered when determining contributing area.

Default: *True*

Outputs

Decayed Specific Catchment Area Grid [raster] The D-Infinity Decaying Accumulation tool creates a grid of the accumulated mass at each location in the domain where mass moves with the D-infinity flow field, but is subject to first order decay in moving from cell to cell.

Console usage

```
processing.runalg('taudem:dinfiniteaccumulation', -ang, -dm, -wg, -o, -nc, -dsca)
```

See also

D-Infinity Distance Down

Description

Calculates the distance downslope to a stream using the D-infinity flow model. The D-infinity flow model is a multiple flow direction model, because the outflow from each grid cell is proportioned between up to 2 downslope grid cells. As such, the distance from any grid cell to a stream is not uniquely defined. Flow that originates at a particular grid cell may enter the stream at a number of different cells. The statistical method may be selected as the longest, shortest or weighted average of the flow path distance to the stream. Also one of several ways of measuring distance may be selected: the total straight line path (Pythagoras), the horizontal component of the straight line path, the vertical component of the straight line path, or the total surface flow path.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Pit Filled Elevation Grid [raster] This input is a grid of elevation values. As a general rule, it is recommended that you use a grid of elevation values that have had the pits removed for this input. Pits are generally taken to be artifacts that interfere with the analysis of flow across them. This grid can be obtained as the output of the “**Pit Remove**” tool, in which case it contains elevation values where the pits have been filled to the point where they just drain.

Stream Raster Grid [raster] A grid indicating streams, by using a grid cell value of 1 on streams and 0 off streams. This is usually the output of one of the tools in the “**Stream Network Analysis**” toolset.

Weight Path Grid [raster] Optional.

A grid giving weights (loadings) to be used in the distance calculation. This might be used for example where only flow distance through a buffer is to be calculated. The weight is then 1 in the buffer and 0 outside it. Alternatively the weight may reflect some sort of cost function for travel over the surface, perhaps representing travel time or attenuation of a process. If this input file is not used, the loadings will assumed to be one for each grid cell.

Statistical Method [selection] Statistical method used to calculate the distance down to the stream. In the D-Infinity flow model, the outflow from each grid cell is proportioned between two downslope grid cells. Therefore, the distance from any grid cell to a stream is not uniquely defined. Flow that originates at a particular grid cell may enter the stream at a number of cells. The distance to the stream may be defined as the longest (maximum), shortest (minimum) or weighted average of the distance down to the stream.

Options:

- 0 — Minimum
- 1 — Maximum
- 2 — Average

Default: 2

Distance Method [selection] Distance method used to calculate the distance down to the stream. One of several ways of measuring distance may be selected: the total straight line path (Pythagoras), the horizontal component of the straight line path (horizontal), the vertical component of the straight line path (vertical), or the total surface flow path (surface).

Options:

- 0 — Pythagoras
- 1 — Horizontal
- 2 — Vertical
- 3 — Surface

Default: 1

Check for edge contamination [boolean] A flag that determines whether the tool should check for edge contamination. This is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being counted. In the context of Distance Down this occurs when part of a flow path traced downslope from a grid cell leaves the domain without reaching a stream grid cell. With edge contamination checking selected, the algorithm recognizes this and reports no data for the result. This is the desired effect and indicates that values for these grid cells is unknown due to it being dependent on terrain outside of the domain of data available. Edge contamination checking may be overridden in cases where you know this is not an issue or want to evaluate the distance using only the fraction of flow paths that terminate at a stream.

Default: *True*

Outputs

D-Infinity Drop to Stream Grid [raster] Grid containing the distance to stream calculated using the D-infinity flow model and the statistical and path methods chosen.

Console usage

```
processing.runalg('taudem:dinfinitydistancedown', dinf_flow_dir_grid, pit_filled_grid, stream_grid)
```

See also

D-Infinity Distance Up

Description

This tool calculates the distance from each grid cell up to the ridge cells along the reverse D-infinity flow directions. Ridge cells are defined to be grid cells that have no contribution from grid cells further upslope. Given the convergence of multiple flow paths at any grid cell, any given grid cell can have multiple upslope ridge cells. There are three statistical methods that this tool can use: maximum distance, minimum distance and waited flow average over these flow paths. A variant on the above is to consider only grid cells that contribute flow with a proportion greater than a user specified threshold (t) to be considered as upslope of any given grid cell. Setting $t=0.5$ would result in only one flow path from any grid cell and would give the result equivalent to a D8 flow model, rather than D-infinity flow model, where flow is proportioned between two downslope grid cells. Finally there are several different optional paths that can be measured: the total straight line path (Pythagoras), the horizontal component of the straight line path, the vertical component of the straight line path, or the total surface flow path.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Pit Filled Elevation Grid [raster] This input is a grid of elevation values. As a general rule, it is recommended that you use a grid of elevation values that have had the pits removed for this input. Pits are generally taken to be artifacts that interfere with the analysis of flow across them. This grid can be obtained as the output of the “**Pit Remove**” tool, in which case it contains elevation values where the pits have been filled to the point where they just drain.

Slope Grid [raster] This input is a grid of slope values. This is measured as drop/distance and it is most often obtained as the output of the “**D-Infinity Flow Directions**” tool.

Statistical Method [selection] Statistical method used to calculate the distance down to the stream. In the D-Infinity flow model, the outflow from each grid cell is proportioned between two downslope grid cells. Therefore, the distance from any grid cell to a stream is not uniquely defined. Flow that originates at a particular grid cell may enter the stream at a number of cells. The distance to the stream may be defined as the longest (maximum), shortest (minimum) or weighted average of the distance down to the stream.

Options:

- 0 — Minimum
- 1 — Maximum
- 2 — Average

Default: 2

Distance Method [selection] Distance method used to calculate the distance down to the stream. One of several ways of measuring distance may be selected: the total straight line path (Pythagoras), the horizontal component of the straight line path (horizontal), the vertical component of the straight line path (vertical), or the total surface flow path (surface).

Options:

- 0 — Pythagoras

- 1 — Horizontal
- 2 — Vertical
- 3 — Surface

Default: *1*

Proportion Threshold [number] The proportion threshold parameter where only grid cells that contribute flow with a proportion greater than this user specified threshold (τ) is considered to be upslope of any given grid cell. Setting $\tau=0.5$ would result in only one flow path from any grid cell and would give the result equivalent to a D8 flow model, rather than D-Infinity flow model, where flow is proportioned between two downslope grid cells.

Default: *0.5*

Check for edge contamination [boolean] A flag that determines whether the tool should check for edge contamination. This is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being counted.

Default: *True*

Outputs

D-Infinity Distance Up [raster] Grid containing the distances up to the ridge calculated using the D-Infinity flow model and the statistical and path methods chosen.

Console usage

```
processing.runalg('taudem:dinfinitydistanceup', dinf_flow_dir_grid, pit_filled_grid, slope_grid, ...)
```

See also

D-Infinity Reverse Accumulation

Description

This works in a similar way to evaluation of weighted Contributing area, except that the accumulation is by propagating the weight loadings upslope along the reverse of the flow directions to accumulate the quantity of weight loading downslope from each grid cell. The function also reports the maximum value of the weight loading downslope from each grid cell in the Maximum Downslope grid.

This function is designed to evaluate and map the hazard due to activities that may have an effect downslope. The example is land management activities that increase runoff. Runoff is sometimes a trigger for landslides or debris flows, so the weight grid here could be taken as a terrain stability map. Then the reverse accumulation provides a measure of the amount of unstable terrain downslope from each grid cell, as an indicator of the danger of activities that may increase runoff, even though there may be no potential for any local impact.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Weight Grid [raster] A grid giving weights (loadings) to be used in the accumulation.

Outputs

Reverse Accumulation Grid [raster] The grid giving the result of the “Reverse Accumulation” function. This works in a similar way to evaluation of weighted Contributing area, except that the accumulation is by propagating the weight loadings upslope along the reverse of the flow directions to accumulate the quantity of loading downslope from each grid cell.

Maximum Downslope Grid [raster] The grid giving the maximum of the weight loading grid downslope from each grid cell.

Console usage

```
processing.runalg('taudem:dinfiniteinverseaccumulation', -ang, -wg, -racc, -dmax)
```

See also

D-Infinity Transport Limited Accumulation - 2

Description

This function is designed to calculate the transport and deposition of a substance (e.g. sediment) that may be limited by both supply and the capacity of the flow field to transport it. This function accumulates substance flux (e.g. sediment transport) subject to the rule that transport out of any grid cell is the minimum between supply and transport capacity, T_{cap} . The total supply at a grid cell is calculated as the sum of the transport in from upslope grid cells, T_{in} , plus the local supply contribution, E (e.g. erosion). This function also outputs deposition, D , calculated as total supply minus actual transport.

Here E is the supply. T_{out} at each grid cell becomes T_{in} for downslope grid cells and is reported as Transport limited accumulation (t_{la}). D is deposition (t_{dep}). The function provides the option to evaluate concentration of a compound (contaminant) adhered to the transported substance. This is evaluated as follows:

Where L_{in} is the total incoming compound loading and C_{in} and T_{in} refer to the Concentration and Transport entering from each upslope grid cell.

If

else

where C_s is the concentration supplied locally and the difference in the second term on the right represents the additional supply from the local grid cell. Then,

C_{out} at each grid cell comprises is the concentration grid output from this function.

If the outlets shapefile is used the tool only evaluates that part of the domain that contributes flow to the locations given by the shapefile.

Transport limited accumulation is useful for modeling erosion and sediment delivery, including the spatial dependence of sediment delivery ratio and contaminant that adheres to sediment.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Supply Grid [raster] A grid giving the supply (loading) of material to a transport limited accumulation function. In the application to erosion, this grid would give the erosion detachment, or sediment supplied at each grid cell.

Transport Capacity Grid [raster] A grid giving the transport capacity at each grid cell for the transport limited accumulation function. In the application to erosion this grid would give the transport capacity of the carrying flow.

Input Concentration Grid [raster] A grid giving the concentration of a compound of interest in the supply to the transport limited accumulation function. In the application to erosion, this grid would give the concentration of say phosphorous adhered to the eroded sediment.

Outlets Shapefile [vector: point] Optional.

This optional input is a point shapefile defining outlets of interest. If this file is used, the tool will only evaluate the area upslope of these outlets.

Check for edge contamination [boolean] This option determines whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being considered when determining the result.

Default: *True*

Outputs

Transport Limited Accumulation Grid [raster] This grid is the weighted accumulation of supply accumulated respecting the limitations in transport capacity and reports the transport rate calculated by accumulating the substance flux subject to the rule that the transport out of any grid cell is the minimum of the total supply (local supply plus transport in) to that grid cell and the transport capacity.

Deposition Grid [raster] A grid giving the deposition resulting from the transport limited accumulation. This is the residual from the transport in to each grid cell minus the transport capacity out of the grid cell. The deposition grid is calculated as the transport in + the local supply - the transport out.

Output Concentration Grid [raster] If an input concentration in supply grid is given, then this grid is also output and gives the concentration of a compound (contaminant) adhered or bound to the transported substance (e.g. sediment) is calculated.

Console usage

```
processing.runalg('taudem:dinfiniteytransportlimitedaccumulation2', dinf_flow_dir_grid, supply_grid)
```

See also

D-Infinity Transport Limited Accumulation

Description

This function is designed to calculate the transport and deposition of a substance (e.g. sediment) that may be limited by both supply and the capacity of the flow field to transport it. This function accumulates substance flux (e.g. sediment transport) subject to the rule that transport out of any grid cell is the minimum between supply and transport capacity, T_{cap} . The total supply at a grid cell is calculated as the sum of the transport in from upslope grid cells, T_{in} , plus the local supply contribution, E (e.g. erosion). This function also outputs deposition, D , calculated as total supply minus actual transport.

Here E is the supply. T_{out} at each grid cell becomes T_{in} for downslope grid cells and is reported as Transport limited accumulation (t_{la}). D is deposition (t_{dep}). The function provides the option to evaluate concentration of a compound (contaminant) adhered to the transported substance. This is evaluated as follows:

Where L_{in} is the total incoming compound loading and C_{in} and T_{in} refer to the Concentration and Transport entering from each upslope grid cell.

If

else

where C_s is the concentration supplied locally and the difference in the second term on the right represents the additional supply from the local grid cell. Then,

C_{out} at each grid cell comprises is the concentration grid output from this function.

If the outlets shapefile is used the tool only evaluates that part of the domain that contributes flow to the locations given by the shapefile.

Transport limited accumulation is useful for modeling erosion and sediment delivery, including the spatial dependence of sediment delivery ratio and contaminant that adheres to sediment.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-infinity method. Flow direction is measured in radians, counter clockwise from east. This can be created by the tool “**D-Infinity Flow Directions**”.

Supply Grid [raster] A grid giving the supply (loading) of material to a transport limited accumulation function. In the application to erosion, this grid would give the erosion detachment, or sediment supplied at each grid cell.

Transport Capacity Grid [raster] A grid giving the transport capacity at each grid cell for the transport limited accumulation function. In the application to erosion this grid would give the transport capacity of the carrying flow.

Outlets Shapefile [vector: point] Optional.

This optional input is a point shapefile defining outlets of interest. If this file is used, the tool will only evaluate the area upslope of these outlets.

Check for edge contamination [boolean] This option determines whether the tool should check for edge contamination. Edge contamination is defined as the possibility that a value may be underestimated due to grid cells outside of the domain not being considered when determining the result.

Default: *True*

Outputs

Transport Limited Accumulation Grid [raster] This grid is the weighted accumulation of supply accumulated respecting the limitations in transport capacity and reports the transport rate calculated by accumulating the substance flux subject to the rule that the transport out of any grid cell is the minimum of the total supply (local supply plus transport in) to that grid cell and the transport capacity.

Deposition Grid [raster] A grid giving the deposition resulting from the transport limited accumulation. This is the residual from the transport in to each grid cell minus the transport capacity out of the grid cell. The deposition grid is calculated as the transport in + the local supply - the transport out.

Console usage

```
processing.runalg('taudem:dinfiniteynfinitytransportlimitedaccumulation', dinf_flow_dir_grid, supply_grid,
```

See also

D-Infinity Upslope Dependence

Description

The D-Infinity Upslope Dependence tool quantifies the amount each grid cell in the domain contributes to a destination set of grid cells. D-Infinity flow directions proportion flow from each grid cell between multiple downslope grid cells. Following this flow field downslope the amount of flow originating at each grid cell that reaches the destination zone is defined. Upslope influence is evaluated using a downslope recursion, examining grid cells downslope from each grid cell, so that the map produced identifies the area upslope where flow through the destination zone originates, or the area it depends on, for its flow.

The figures below illustrate the amount each source point in the domain x (blue) contributes to the destination point or zone y (red). If the indicator weighted contributing area function is denoted $I(y; x)$ giving the weighted contribution using a unit value (1) from specific grid cells y to grid cells x , then the upslope dependence is: $D(x; y) = I(y; x)$.

This is useful for example to track where flow or a flow related substance or contaminant that enters a destination area may come from.

Parameters

D-Infinity Flow Direction Grid [raster] A grid giving flow direction by the D-Infinity method where the flow direction angle is determined as the direction of the steepest downward slope on the eight triangular facets formed in a 3x3 grid cell window centered on the grid cell of interest. This grid can be produced using the “**D-Infinity Flow Direction**” tool.

Destination Grid [raster] A grid that encodes the destination zone that may receive flow from upslope. This grid must be 1 inside the zone y and 0 over the rest of the domain.

Outputs

Output Upslope Dependence Grid [raster] A grid quantifying the amount each source point in the domain contributes to the zone defined by the destination grid.

Console usage

```
processing.runalg('taudem:dinfinityupslopedependence', -ang, -dg, -dep)
```

See also

Slope Average Down

Description

This tool computes slope in a D8 downslope direction averaged over a user selected distance. Distance should be specified in horizontal map units.

Parameters

D8 Flow Direction Grid [raster] This input is a grid of flow directions that are encoded using the D8 method where all flow from a cells goes to a single neighboring cell in the direction of steepest descent. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Pit Filled Elevation Grid [raster] This input is a grid of elevation values. As a general rule, it is recommended that you use a grid of elevation values that have had the pits removed for this input. Pits are generally taken to be artifacts that interfere with the analysis of flow across them. This grid can be obtained as the output of the “**Pit Remove**” tool, in which case it contains elevation values where the pits have been filled to the point where they just drain.

Downslope Distance [number] Input parameter of downslope distance over which to calculate the slope (in horizontal map units).

Default: 50

Outputs

Slope Average Down Grid [raster] This output is a grid of slopes calculated in the D8 downslope direction, averaged over the selected distance.

Console usage

```
processing.runalg('taudem:slopeaveragedown', -p, -fel, -dn, -slpd)
```

See also

Slope Over Area Ratio

Description

Calculates the ratio of the slope to the specific catchment area (contributing area). This is algebraically related to the more common $\ln(a/\tan \beta)$ wetness index, but contributing area is in the denominator to avoid divide by 0 errors when slope is 0.

Parameters

Slope Grid [raster] A grid of slope. This grid can be generated using either the “**D8 Flow Directions**” tool or the “**D-Infinity Flow Directions**” tool.

Specific Catchment Area Grid [raster] A grid giving the contributing area value for each cell taken as its own contribution plus the contribution from upslope neighbors that drain in to it. Contributing area is counted in terms of the number of grid cells (or summation of weights). This grid can be generated using either the “**D8 Contributing Area**” tool or the “**D-Infinity Contributing Area**” tool.

Outputs

Slope Divided By Area Ratio Grid [raster] A grid of the ratio of slope to specific catchment area (contributing area). This is algebraically related to the more common $\ln(a/\tan \beta)$ wetness index, but contributing area is in the denominator to avoid divide by 0 errors when slope is 0.

Console usage

```
processing.runalg('taudem:slopeoverarearatio', -slp, -sca, -sar)
```

See also

18.8.3 Stream Network Analysis

D8 Extreme Upslope Value

Description

Evaluates the extreme (either maximum or minimum) upslope value from an input grid based on the D8 flow model. This is intended initially for use in stream raster generation to identify a threshold of the slope times area product that results in an optimum (according to drop analysis) stream network.

If the optional outlet point shapefile is used, only the outlet cells and the cells upslope (by the D8 flow model) of them are in the domain to be evaluated.

By default, the tool checks for edge contamination. This is defined as the possibility that a result may be underestimated due to grid cells outside of the domain not being counted. This occurs when drainage is inwards from the boundaries or areas with “no data” values for elevation. The algorithm recognizes this and reports “no data” for the result for these grid cells. It is common to see streaks of “no data” values extending inwards from boundaries along flow paths that enter the domain at a boundary. This is the desired effect and indicates that the result for these grid cells is unknown due to it being dependent on terrain outside of the domain of data available. Edge

contamination checking may be turned off in cases where you know this is not an issue or want to ignore these problems, if for example, the DEM has been clipped along a watershed outline.

Parameters

D8 Flow Directions Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Upslope Values Grid [raster] This is the grid of values of which the maximum or minimum upslope value is selected. The values most commonly used are the slope times area product needed when generating stream rasters according to drop analysis.

Outlets Shapefile [vector: point] Optional.

A point shape file defining outlets of interest. If this input file is used, only the area upslope of these outlets will be evaluated by the tool.

Check for edge contamination [boolean] A flag that indicates whether the tool should check for edge contamination.

Default: *True*

Use max upslope value [boolean] A flag to indicate whether the maximum or minimum upslope value is to be calculated.

Default: *True*

Outputs

Extreme Upslope Values Grid [raster] A grid of the maximum/minimum upslope values.

Console usage

```
processing.runalg('taudem:d8extremeupslopevalue', -p, -sa, -o, -nc, -min, -ssa)
```

See also

Length Area Stream Source

Description

Creates an indicator grid (1, 0) that evaluates $A \geq (M) (L^y)$ based on upslope path length, D8 contributing area grid inputs, and parameters M and y . This grid indicates likely stream source grid cells. This is an experimental method with theoretical basis in Hack’s law which states that for streams $L \sim A^{0.6}$. However for hillslopes with parallel flow $L \sim A$. So a transition from hillslopes to streams may be represented by $L \sim A^{0.8}$ suggesting identifying grid cells as stream cells if $A > M (L^{(1/0.8)})$.

Parameters

Length Grid [raster] A grid of the maximum upslope length for each cell. This is calculated as the length of the flow path from the furthest cell that drains to each cell. Length is measured between cell centers taking into account cell size and whether the direction is adjacent or diagonal. It is this length (L) that is used in the formula, $A > (M) (L^y)$, to determine which cells are considered stream cells. This grid can be obtained as an output from the “**Grid Network**” tool.

Contributing Area Grid [raster] A grid of contributing area values for each cell that were calculated using the D8 algorithm. The contributing area for a cell is the sum of its own contribution plus the contribution from all upslope neighbors that drain to it, measured as a number of cells. This grid is typically obtained as the output of the “**D8 Contributing Area**” tool. In this tool, it is the contributing area (A) that is compared in the formula $A > (M) (L^y)$ to determine the transition to a stream.

Threshold [number] The multiplier threshold (M) parameter which is used in the formula: $A > (M) (L^y)$, to identify the beginning of streams.

Default: *0.03*

Exponent [number] The exponent (y) parameter which is used in the formula: $A > (M) (L^y)$, to identify the beginning of streams. In branching systems, Hack’s law suggests that $L = 1/M A^{(1/y)}$ with $1/y = 0.6$ (or 0.56) (y about 1.7). In parallel flow systems L is proportional to A (y about 1). This method tries to identify the transition between these two paradigms by using an exponent y somewhere in between (y about 1.3).

Default: *1.3*

Outputs

Stream Source Grid [raster] An indicator grid (1,0) that evaluates $A \geq (M)(L^y)$, based on the maximum upslope path length, the D8 contributing area grid inputs, and parameters M and y . This grid indicates likely stream source grid cells.

Console usage

```
processing.runalg('taudem:lengthareastreamsource', length_grid, contrib_area_grid, threshold, exp
```

See also

Move Outlets To Streams

Description

Moves outlet points that are not aligned with a stream cell from a stream raster grid, downslope along the D8 flow direction until a stream raster cell is encountered, the “max_dist” number of grid cells are examined, or the flow path exits the domain (i.e. a “no data” value is encountered for the D8 flow direction). The output file is a new outlets shapefile where each point has been moved to coincide with the stream raster grid, if possible. A field “dist_moved” is added to the new outlets shapefile to indicate the changes made to each point. Points that are already on a stream cell are not moved and their “dist_moved” field is assigned a value 0. Points that are initially not on a stream cell are moved by sliding them downslope along the D8 flow direction until one of the following occurs: a) A stream raster grid cell is encountered before traversing the “max_dist” number of grid cells. In which case, the point is moved and the “dist_moved” field is assigned a value indicating how many grid cells the point was moved. b) More than the “max_number” of grid cells are traversed, or c) the traversal ends up going out of the domain (i.e., a “no data” D8 flow direction value is encountered). In which case, the point is not moved and the “dist_moved” field is assigned a value of -1.

Parameters

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Stream Raster Grid [raster] This output is an indicator grid (1, 0) that indicates the location of streams, with a value of 1 for each of the stream cells and 0 for the remainder of the cells. This file is produced by several different tools in the “**Stream Network Analysis**” toolset.

Outlets Shapefile [vector: point] A point shape file defining points of interest or outlets that should ideally be located on a stream, but may not be exactly on the stream due to the fact that the shapefile point locations may not have been accurately registered with respect to the stream raster grid.

Maximum Number of Grid Cells to traverse [number] This input parameter is the maximum number of grid cells that the points in the input outlet shapefile will be moved before they are saved to the output outlet shapefile.

Default: *50*

Outputs

Output Outlet Shapefile [vector] A point shape file defining points of interest or outlets. This file has one point in it for each point in the input outlet shapefile. If the original point was located on a stream, then the point was not moved. If the original point was not on a stream, the point was moved downslope according to the D8 flow direction until it reached a stream or the maximum distance had been reached. This file has an additional field “*dist_moved*” added to it which is the number of cells that the point was moved. This field is 0 if the cell was originally on a stream, -1 if it was not moved because there was not a stream within the maximum distance, or some positive value if it was moved.

Console usage

```
processing.runalg('taudem:moveoutletstostreams', -p, -src, -o, -md, -om)
```

See also

Peuker Douglas

Description

Creates an indicator grid (1, 0) of upward curved grid cells according to the Peuker and Douglas algorithm.

With this tool, the DEM is first smoothed by a kernel with weights at the center, sides, and diagonals. The Peuker and Douglas (1975) method (also explained in Band, 1986), is then used to identify upwardly curving grid cells. This technique flags the entire grid, then examines in a single pass each quadrant of 4 grid cells, and unflags the highest. The remaining flagged cells are deemed “upwardly curved”, and when viewed, resemble a channel network. This proto-channel network generally lacks connectivity and requires thinning, issues that were discussed in detail by Band (1986).

Parameters

Elevation Grid [raster] A grid of elevation values. This is usually the output of the “**Pit Remove**” tool, in which case it is elevations with pits removed.

Center Smoothing Weight [number] The center weight parameter used by a kernel to smooth the DEM before the tool identifies upwardly curved grid cells.

Default: *0.4*

Side Smoothing Weight [number] The side weight parameter used by a kernel to smooth the DEM before the tool identifies upwardly curved grid cells.

Default: *0.1*

Diagonal Smoothing Weight [number] The diagonal weight parameter used by a kernel to smooth the DEM before the tool identifies upwardly curved grid cells.

Default: *0.05*

Outputs

Stream Source Grid [raster] An indicator grid (1, 0) of upward curved grid cells according to the Peuker and Douglas algorithm, and if viewed, resembles a channel network. This proto-channel network generally lacks connectivity and requires thinning, issues that were discussed in detail by Band (1986).

Console usage

```
processing.runalg('taudem:peukerdouglas', elevation_grid, center_weight, side_weight, diagonal_we
```

See also

- Band, L. E., (1986), “Topographic partition of watersheds with digital elevation models”, *Water Resources Research*, 22(1): 15-24.
- Peuker, T. K. and D. H. Douglas, (1975), “Detection of surface-specific points by local parallel processing of discrete terrain elevation data”, *Comput. Graphics Image Process.*, 4: 375-387.

Slope Area Combination

Description

Creates a grid of slope-area values = $(S_m)^m (A_n)^n$ based on slope and specific catchment area grid inputs, and parameters m and n . This tool is intended for use as part of the slope-area stream raster delineation method.

Parameters

Slope Grid [raster] This input is a grid of slope values. This grid can be obtained from the “**D-Infinity Flow Directions**” tool.

Contributing Area Grid [raster] A grid giving the specific catchment area for each cell taken as its own contribution (grid cell length or summation of weights) plus the proportional contribution from upslope neighbors that drain in to it. This grid is typically obtained from the “**D-Infinity Contributing Area**” tool.

Slope Exponent [number] The slope exponent (m) parameter which will be used in the formula: $(S_m)^m (A_n)^n$, that is used to create the slope-area grid.

Default: *2*

Area Exponent [number] The area exponent (n) parameter which will be used in the formula: $(S_m)^m (A_n)^n$, that is used to create the slope-area grid.

Default: *1*

Outputs

Slope Area Grid [raster] A grid of slope-area values = $(S_m)^m (A_n)^n$ calculated from the slope grid, specific catchment area grid, m slope exponent parameter, and n area exponent parameter.

Console usage

```
processing.runalg('taudem:slopeareacombination', slope_grid, area_grid, slope_exponent, area_exponent)
```

See also

Stream Definition By Threshold

Description

Operates on any grid and outputs an indicator (1, 0) grid identifying cells with input values \geq the threshold value. The standard use is to use an accumulated source area grid to as the input grid to generate a stream raster grid as the output. If you use the optional input mask grid, it limits the domain being evaluated to cells with mask values \geq 0. When you use a D-infinity contributing area grid (**sca*) as the mask grid, it functions as an edge contamination mask. The threshold logic is:

```
src = ((ssa >= thresh) & (mask >= s0)) ? 1:0
```

Parameters

Accumulated Stream Source Grid [raster] This grid nominally accumulates some characteristic or combination of characteristics of the watershed. The exact characteristic(s) varies depending on the stream network raster algorithm being used. This grid needs to have the property that grid cell values are monotonically increasing downslope along D8 flow directions, so that the resulting stream network is continuous. While this grid is often from an accumulation, other sources such as a maximum upslope function will also produce a suitable grid.

Threshold [number] This parameter is compared to the value in the Accumulated Stream Source grid (**ssa*) to determine if the cell should be considered a stream cell. Streams are identified as grid cells for which ssa value is \geq this threshold.

Default: *100*

Mask Grid [raster] Optional.

This optional input is a grid that is used to mask the domain of interest and output is only provided where this grid is \geq 0. A common use of this input is to use a D-Infinity contributing area grid as the mask so that the delineated stream network is constrained to areas where D-infinity contributing area is available, replicating the functionality of an edge contamination mask.

Outputs

Stream Raster Grid [raster] This is an indicator grid (1, 0) that indicates the location of streams, with a value of 1 for each of the stream cells and 0 for the remainder of the cells.

Console usage

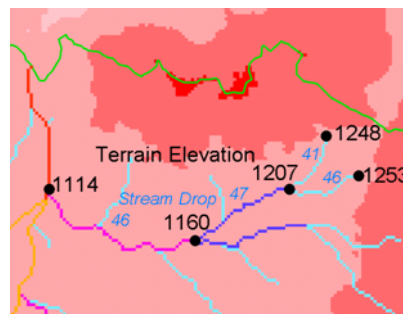
```
processing.runalg('taudem:streamdefinitionbythreshold', -ssa, -thresh, -mask, -src)
```

See also

Stream Drop Analysis

Description

Applies a series of thresholds (determined from the input parameters) to the input accumulated stream source grid (*ssa) grid and outputs the results in the *drp.txt file the stream drop statistics table. This function is designed to aid in the determination of a geomorphologically objective threshold to be used to delineate streams. Drop Analysis attempts to select the right threshold automatically by evaluating a stream network for a range of thresholds and examining the constant drop property of the resulting Strahler streams. Basically it asks the question: Is the mean stream drop for first order streams statistically different from the mean stream drop for higher order streams, using a T-test. Stream drop is the difference in elevation from the beginning to the end of a stream defined as the sequence of links of the same stream order. If the T-test shows a significant difference then the stream network does not obey this “law” so a larger threshold needs to be chosen. The smallest threshold for which the T-test does not show a significant difference gives the highest resolution stream network that obeys the constant stream drop “law” from geomorphology, and is the threshold chosen for the “objective” or automatic mapping of streams from the DEM. This function can be used in the development of stream network rasters, where the exact watershed characteristic(s) that were accumulated in the accumulated stream source grid vary based on the method being used to determine the stream network raster.



The constant stream drop “law” was identified by Broscoc (1959). For the science behind using this to determine a stream delineation threshold, see Tarboton et al. (1991, 1992), Tarboton and Ames (2001).

Parameters

D8 Contributing Area Grid [raster] A grid of contributing area values for each cell that were calculated using the D8 algorithm. The contributing area for a cell is the sum of its own contribution plus the contribution from all upslope neighbors that drain to it, measured as a number of cells or the sum of weight loadings. This grid can be obtained as the output of the “**D8 Contributing Area**” tool. This grid is used in the evaluation of drainage density reported in the stream drop table.

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the “**D8 Flow Directions**” tool.

Pit Filled Elevation Grid [raster] A grid of elevation values. This is usually the output of the “**Pit Remove**” tool, in which case it is elevations with pits removed.

Accumulated Stream Source Grid [raster] This grid must be monotonically increasing along the downslope D8 flow directions. It is compared to a series of thresholds to determine the beginning of the streams. It is often generated by accumulating some characteristic or combination of characteristics of the watershed with the “**D8 Contributing Area**” tool, or using the maximum option of the “**D8 Flow Path Extreme**” tool. The exact method varies depending on the algorithm being used.

Outlets Shapefile [vector: point] A point shapefile defining the outlets upstream of which drop analysis is performed.

Minimum Threshold [number] This parameter is the lowest end of the range searched for possible threshold values using drop analysis. This technique looks for the smallest threshold in the range where the absolute value of the t-statistic is less than 2. For the science behind the drop analysis see Tarboton et al. (1991, 1992), Tarboton and Ames (2001).

Default: 5

Maximum Threshold [number] This parameter is the highest end of the range searched for possible threshold values using drop analysis. This technique looks for the smallest threshold in the range where the absolute value of the t-statistic is less than 2. For the science behind the drop analysis see Tarboton et al. (1991, 1992), Tarboton and Ames (2001).

Default: 500

Number of Threshold Values [number] The parameter is the number of steps to divide the search range into when looking for possible threshold values using drop analysis. This technique looks for the smallest threshold in the range where the absolute value of the t-statistic is less than 2. For the science behind the drop analysis see Tarboton et al. (1991, 1992), Tarboton and Ames (2001).

Default: 10

Spacing for Threshold Values [selection] This parameter indicates whether logarithmic or linear spacing should be used when looking for possible threshold values using drop analysis.

Options:

- 0 — Logarithmic
- 1 — Linear

Default: 0

Outputs

D-Infinity Drop to Stream Grid [file] This is a comma delimited text file with the following header line:

```
:: Threshold,DrainDen,NoFirstOrd,NoHighOrd,MeanDFirstOrd,MeanDHighOrd,StdDevFirstOrd,StdDevHighOrd,T
```

The file then contains one line of data for each threshold value examined, and then a summary line that indicates the optimum threshold value. This technique looks for the smallest threshold in the range where the absolute value of the t-statistic is less than 2. For the science behind the drop analysis, see Tarboton et al. (1991, 1992), Tarboton and Ames (2001).

Console usage

```
processing.runalg('taudem:streamdropanalysis', d8_contrib_area_grid, d8_flow_dir_grid, pit_filled,
```

See also

- Broscue, A. J., (1959), “Quantitative analysis of longitudinal stream profiles of small watersheds”, Office of Naval Research, Project NR 389-042, Technical Report No. 18, Department of Geology, Columbia University, New York.
- Tarboton, D. G., R. L. Bras and I. Rodriguez-Iturbe, (1991), “On the Extraction of Channel Networks from Digital Elevation Data”, Hydrologic Processes, 5(1): 81-100.
- Tarboton, D. G., R. L. Bras and I. Rodriguez-Iturbe, (1992), “A Physical Basis for Drainage Density”, Geomorphology, 5(1/2): 59-76.
- Tarboton, D. G. and D. P. Ames, (2001), “Advances in the mapping of flow networks from digital elevation data”, World Water and Environmental Resources Congress, Orlando, Florida, May 20-24, ASCE, <http://www.engineering.usu.edu/dtarb/asce2001.pdf>.

Stream Reach and Watershed

Description

This tool produces a vector network and shapefile from the stream raster grid. The flow direction grid is used to connect flow paths along the stream raster. The Strahler order of each stream segment is computed. The subwatershed draining to each stream segment (reach) is also delineated and labeled with the value identifier that corresponds to the WSNO (watershed number) attribute in the Stream Reach Shapefile.

This tool orders the stream network according to the Strahler ordering system. Streams that don't have any other streams draining in to them are order 1. When two stream reaches of different order join the order of the downstream reach is the order of the highest incoming reach. When two reaches of equal order join the downstream reach order is increased by 1. When more than two reaches join the downstream reach order is calculated as the maximum of the highest incoming reach order or the second highest incoming reach order + 1. This generalizes the common definition to cases where more than two reaches join at a point. The network topological connectivity is stored in the Stream Network Tree file, and coordinates and attributes from each grid cell along the network are stored in the Network Coordinates file.

The stream raster grid is used as the source for the stream network, and the flow direction grid is used to trace connections within the stream network. Elevations and contributing area are used to determine the elevation and contributing area attributes in the network coordinate file. Points in the outlets shapefile are used to logically split stream reaches to facilitate representing watersheds upstream and downstream of monitoring points. The program uses the attribute field "id" in the outlets shapefile as identifiers in the Network Tree file. This tool then translates the text file vector network representation in the Network Tree and Coordinates files into a shapefile. Further attributes are also evaluated. The program has an option to delineate a single watershed by representing the entire area draining to the Stream Network as a single value in the output watershed grid.

Parameters

Pit Filled Elevation Grid [raster] A grid of elevation values. This is usually the output of the "**Pit Remove**" tool, in which case it is elevations with pits removed.

D8 Flow Direction Grid [raster] A grid of D8 flow directions which are defined, for each cell, as the direction of the one of its eight adjacent or diagonal neighbors with the steepest downward slope. This grid can be obtained as the output of the "**D8 Flow Directions**" tool.

D8 Drainage Area [raster] A grid giving the contributing area value in terms of the number of grid cells (or the summation of weights) for each cell taken as its own contribution plus the contribution from upslope neighbors that drain in to it using the D8 algorithm. This is usually the output of the "**D8 Contributing Area**" tool and is used to determine the contributing area attribute in the Network Coordinate file.

Stream Raster Grid [raster] An indicator grid indicating streams, by using a grid cell value of 1 on streams and 0 off streams. Several of the "**Stream Network Analysis**" tools produce this type of grid. The Stream Raster Grid is used as the source for the stream network.

Outlets Shapefile as Network Nodes [vector: point] Optional.

A point shape file defining points of interest. If this file is used, the tool will only delineate the stream network upstream of these outlets. Additionally, points in the Outlets Shapefile are used to logically split stream reaches to facilitate representing watersheds upstream and downstream of monitoring points. This tool **REQUIRES THAT THERE BE** an integer attribute field "id" in the Outlets Shapefile, because the "id" values are used as identifiers in the Network Tree file.

Delineate Single Watershed [boolean] This option causes the tool to delineate a single watershed by representing the entire area draining to the Stream Network as a single value in the output watershed grid. Otherwise a separate watershed is delineated for each stream reach. Default is *False* (separate watershed).

Default: *False*

Outputs

Stream Order Grid [raster] The Stream Order Grid has cells values of streams ordered according to the Strahler order system. The Strahler ordering system defines order 1 streams as stream reaches that don't have any other reaches draining in to them. When two stream reaches of different order join the order of the downstream reach is the order of the highest incoming reach. When two reaches of equal order join the downstream reach order is increased by 1. When more than two reaches join the downstream reach order is calculated as the maximum of the highest incoming reach order or the second highest incoming reach order + 1. This generalizes the common definition to cases where more than two flow paths reaches join at a point.

Watershed Grid [raster] This output grid identified each reach watershed with a unique ID number, or in the case where the delineate single watershed option was checked, the entire area draining to the stream network is identified with a single ID.

Stream Reach Shapefile [vector] This output is a polyline shapefile giving the links in a stream network. The columns in the attribute table are:

- LINKNO — Link Number. A unique number associated with each link (segment of channel between junctions). This is arbitrary and will vary depending on number of processes used
- DSLINKNO — Link Number of the downstream link. -1 indicates that this does not exist
- USLINKNO1 — Link Number of first upstream link. (-1 indicates no link upstream, i.e. for a source link)
- USLINKNO2 — Link Number of second upstream link. (-1 indicates no second link upstream, i.e. for a source link or an internal monitoring point where the reach is logically split but the network does not bifurcate)
- DSNODEID — Node identifier for node at downstream end of stream reach. This identifier corresponds to the "id" attribute from the Outlets shapefile used to designate nodes
- Order — Strahler Stream Order
- Length — Length of the link. The units are the horizontal map units of the underlying DEM grid
- Magnitude — Shreve Magnitude of the link. This is the total number of sources upstream
- DS_Cont_Ar — Drainage area at the downstream end of the link. Generally this is one grid cell upstream of the downstream end because the drainage area at the downstream end grid cell includes the area of the stream being joined
- Drop — Drop in elevation from the start to the end of the link
- Slope — Average slope of the link (computed as drop/length)
- Straight_L — Straight line distance from the start to the end of the link
- US_Cont_Ar — Drainage area at the upstream end of the link
- WSNO — Watershed number. Cross reference to the `*w.shp` and `*w` grid files giving the identification number of the watershed draining directly to the link
- DOUT_END — Distance to the eventual outlet (i.e. the most downstream point in the stream network) from the downstream end of the link
- DOUT_START — Distance to the eventual outlet from the upstream end of the link
- DOUT_MID — Distance to the eventual outlet from the midpoint of the link

Network Connectivity Tree [file] This output is a text file that details the network topological connectivity is stored in the Stream Network Tree file. Columns are as follows:

- Link Number (Arbitrary — will vary depending on number of processes used)
- Start Point Number in Network coordinates (`*coord.dat`) file (Indexed from 0)
- End Point Number in Network coordinates (`*coord.dat`) file (Indexed from 0)

- Next (Downstream) Link Number. Points to Link Number. -1 indicates no links downstream, i.e. a terminal link
- First Previous (Upstream) Link Number. Points to Link Number. -1 indicates no upstream links
- Second Previous (Upstream) Link Numbers. Points to Link Number. -1 indicates no upstream links. Where only one previous link is -1, it indicates an internal monitoring point where the reach is logically split, but the network does not bifurcate
- Strahler Order of Link
- Monitoring point identifier at downstream end of link. -1 indicates downstream end is not a monitoring point
- Network magnitude of the link, calculated as the number of upstream sources (following Shreve)

Network Coordinates [file] This output is a text file that contains the coordinates and attributes of points along the stream network. Columns are as follows:

- X coordinate
- Y Coordinate
- Distance along channels to the downstream end of a terminal link
- Elevation
- Contributing area

Console usage

```
processing.runalg('taudem:streamreachandwatershed', -fel, -p, -ad8, -src, -o, -sw, -ord, -w, -net,
```

See also

.

Compositor de Impressão

Com o compositor de impressão que você pode criar mapas agradáveis e atlas que podem ser impressos ou salvos como arquivo PDF, imagem ou arquivo SVG. Esta é uma maneira poderosa de compartilhar informação geográfica produzida com o QGIS que podem ser incluídas em relatórios ou publicadas.

O compositor de impressão fornece grandes recursos de layout e impressão. Permite que possa adicionar elementos como o enquadramento do mapa QGIS, etiquetas de texto, imagens, legendas, barras de escala, formas básicas, setas, tabelas de atributo e molduras HTML. Pode dimensionar, agrupar, alinhar, e posicionar cada elemento e ajustar as propriedades para criar o seu layout. O layout pode ser impresso ou exportado para formatos de imagens, Postscript, PDF ou para SVG (a exportação para SVG não está a trabalhar corretamente com algumas versões recentes do Qt4, deve experimentar e verificar o seu sistema individualmente). Pode guardar o layout como modelo e carregá-lo outra vez em outra sessão. Finalmente, vários mapas baseados num modelo podem ser gerados através do Gerador de Atlas. Veja a lista de ferramentas no [table_composer_1](#):












Ícone	Finalidade	Ícone	Finalidade
	Salvar Projeto		Novo Compositor
	Duplicar Compositor		Gerenciar Compositores
	Carregar a partir do modelo		Salvar como modelo
	Imprimir ou exportar como PostScript		Exportar como imagem
	Exportar como SVG		Exportar como PDF
	Reverter à última alteração		Restaurar a última alteração
	Zoom Total		Visualizar a 100%
	Aproximar		Afastar
	Atualizar vista		Zoom a uma área específica
	Movimentar		Mover conteúdo do item
	Selecionar/ Mover item		Adicionar imagem
	Adicionar novo mapa do QGIS no enquadramento do mapa		Adicionar nova legenda
	Adicionar nova etiqueta		Adicionar forma básica
	Adicionar barra de escala ao compositor de impressão		Adicionar tabela de atributos
	Adicionar seta		Desagrupar itens
	Adicionar uma HTML Frame		Desbloquear todos os itens
	Agrupar itens		Abaixar itens selecionados
	Bloquear itens selecionados		Enviar para trás
	Elevar itens selecionados		Alinhar à direita
	Trazer para a frente		Centraliza na vertical
	Alinhar à esquerda		Alinhar ao fundo
	Alinhar ao centro		Primeiro Elemento
	Alinhar ao topo		Próximo Elemento
	Pré-Visualizar Atlas		Imprimir Atlas
	Elemento Anterior		Atlas Configurações
	Último Elemento		
	Exportar Atlas como Imagem		


Tabela 1 do Compositor: Ferramentas do Compositor de Impressão

Todas as ferramentas do Compositor de Impressão estão disponíveis nos menus e como ícones na barra de ferramentas. A barra de ferramentas pode ser desligada ou ligada usando o botão direito do mouse sobre a barra de ferramentas.

19.1 Primeiros passos

19.1.1 Abrir um novo Modelo de de Compositor de Impressão

Antes de começar a trabalhar com o compositor de impressão, necessita de carregar algumas camadas vetoriais e raster no enquadramento do mapa QGIS e adaptar as suas propriedades para se ajustar à sua conveniência.

Após tudo estar renderizado ou simbolizado ao seu gosto, clique no ícone  na barra de ferramentas ou escolha *Arquivo* → *Novo Compositor de Impressão*. Será pedido para escolher um título para o novo compositor.

19.1.2 Visão geral do compositor de impressão

Abrindo o compositor de impressão ele fornece uma tela em branco que representa uma superfície de papel a utilizar para opção de impressão. Inicialmente você encontra botões do lado esquerdo do lado da tela, para adicionar itens ao compositor de mapas; o QGIS atual apresenta mapa na tela, rótulos de texto, imagens, legendas, barras de escala, formas básicas, setas, tabelas de atributos e quadros HTML. Nesta barra de ferramentas, você também encontrará os botões da barra de ferramentas para navegar, dar zoom em uma área e movimentar a visão panorâmica com os botões da barra de ferramentas e compositor para selecionar um item do compositor de mapas e para mover o conteúdo do item no mapa.

Figure_composer_overview mostra a visão inicial do compositor de impressão antes de quaisquer elementos serem adicionados.

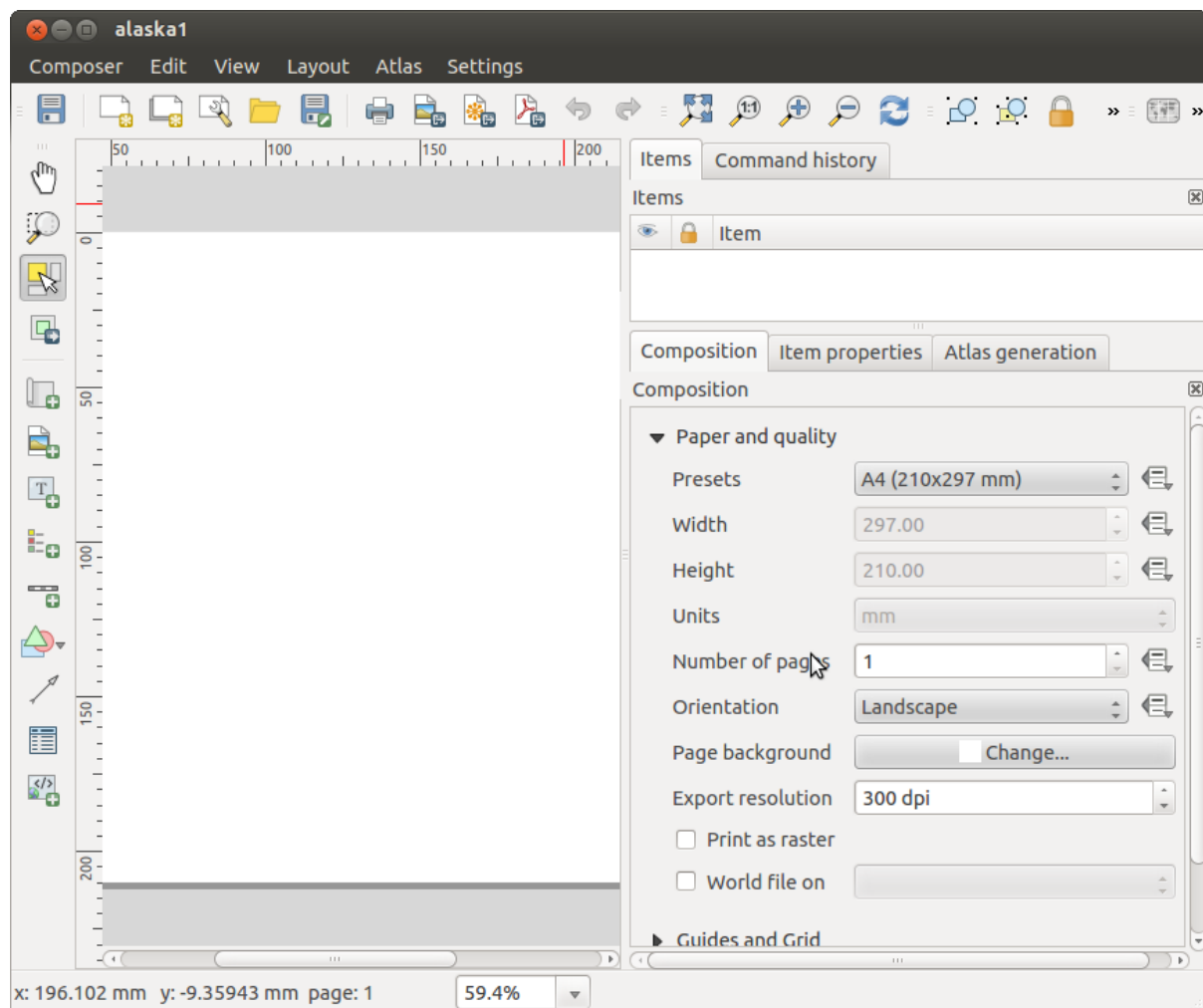





Figura 19.1: Compositor de Impressão 

À direita ao lado da tela você encontra dois painéis. O painel superior detêm as guias *Ítems* e *Histórico de Comandos* e o painel inferior contém as guias *Composição*, *propriedades do ítem* e *Geração de Atlas*.

- A guia *ítems* fornece uma lista de todos os itens do compositor de mapas adicionados à tela.
- A guia *Histórico de Comandos* exibe um histórico de todas as alterações aplicadas ao layout do compositor

de impressão. Com um clique do mouse, é possível desfazer e refazer os passos do layout para trás e para um certo status.








- A guia *Composição* permite que você defina o tamanho do papel, a orientação, o fundo da página, número de páginas e qualidade de impressão para o arquivo de saída em dpi. Além disso, você também pode ativar a caixa de seleção **caixal** *Imprimir como raster*. Isso significa que todos os itens serão convertidos para raster antes de imprimir ou salvar como PostScript ou PDF. Nesta guia, você também pode personalizar as configurações para grade e guias inteligentes.
- A guia *Propriedades di item* exibe as propriedades para o item selecionado. Clique no ícone  para selecionar um item (por exemplo, legenda, barra de escala ou rótulo) na tela. Depois clique na guia :guilabel: *Propriedades do item* e personalizar as definições para o item selecionado.
- O separador *Gerador de Atlas* permite ativar a criação de um atlas para o compositor atual e dá acesso aos seus parâmetros.
- Finalmente, você pode salvar a sua composição de impressão com o botão  Salvar projeto.

Na parte inferior da janela do compositor de impressão, você pode encontrar uma barra de status com a posição do mouse, o número da página atual e uma caixa de combinação para definir o nível de zoom.

Você pode adicionar vários elementos para o Compositor. Também é possível ver mais do que um mapa ou legenda ou barra de escala na tela do compositor de impressão, em uma ou várias páginas. Cada elemento tem as suas próprias propriedades e, no caso de o mapa, a sua própria medida. Se você quiser remover todos os elementos da tela Compositor você pode fazer isso com a tecla *Excluir* ou *Backspace*.

Ferramentas de Navegação



Para navegar no enquadramento do layout, o compositor de impressão fornece 4 ferramentas gerais:




-  Aproximar
-  Afastar
-  Ver Tudo
-  Visualizar a 100%
-  Atualiza a vista (se encontrar uma visão com um estado inconsciente)
-  *Mover Mapa*
-  modo de zoom Letreiro (zoom a uma região específica do compositor)

Você pode alterar o nível de zoom também usando a roda do mouse ou a caixa de combinação na barra de status. Se você precisa mudar para o modo panorâmico, enquanto trabalhava na área de Compositor, você pode segurar o *Espaço* ou a roda do mouse. Com *Ctrl + Barra de espaço*, você pode alternar temporariamente para o modo zoom letreiro, e com *Ctrl + Shift + Barra de espaço*, para diminuir o modo de zoom.

19.1.3 Seção de Exemplo

Para demonstrar como criar um mapa, siga as próximas instruções.

1. No site esquerdo, selecione o botão da barra de ferramentas  Adicionar novo mapa e desenhe um retângulo na tela segurando o botão esquerdo do mouse. Dentro do retângulo desenhado no mapa QGIS visto na tela.
2. Selecione o botão da barra de ferramentas  Adicionar nova barra de escala e coloque o item no mapa com o botão esquerdo do mouse sobre a tela do compositor de impressão. A barra de escala será adicionada à tela.

3. Selecione o botão da barra de ferramentas  Adicionar nova legenda e desenhe um retângulo na tela segurando o botão esquerdo do mouse. Dentro do retângulo desenhado a legenda será desenhada.
4. Selecione o ícone  Selecionar/Move item para selecionar o mapa na tela e move-lo um pouco.
5. Enquanto o item do mapa ainda está selecionado, você também pode alterar o tamanho do item do mapa. Clique enquanto mantém pressionado o botão esquerdo do mouse, em um pequeno retângulo branco em um dos cantos do item do mapa e arraste-o para um novo local para alterar o seu tamanho.
6. Clique na guia *Propriedades do item* no painel inferior esquerdo e encontre o ajuste para a Rotação. Mude o valor da configuração :guilabel: *Rotação* para '15.00**lgraus**'. Você deverá ver o item do mapa mudar de orientação.
7. Finalmente, você pode salvar a sua composição de impressão com o botão  Salvar projeto.

19.1.4 Opções do Compositor de Impressão

Em *Configurações* → *Opções do compositor* você pode definir algumas opções que serão utilizadas como padrão durante o seu trabalho.

- *Composições padrão* permitem especificar a fonte padrão a ser usada.
- Com *Aparência da grade*, você pode definir o estilo de grade e sua cor.
- *Grade Padrão* define espaçamento, deslocamento e tolerância da grade. Existem três tipos de grade: linhas de **pontos**, **Sólidas** e ****Cruzes****.
- *Guia padrão* define a tolerância para as guias.

19.1.5 Separador de Composição — Configuração geral da composição

No separador *Composição*, pode definir as configurações globais à sua composição.

- Você pode escolher um dos tamanhos *pré-definidos* para a sua folha de papel, ou digitar o de seu costume *largura e altura*.
- Composição agora podem ser divididos em várias páginas. Por exemplo, a primeira página pode mostrar um mapa da lona, e uma segunda página pode mostrar a tabela de atributos associados a uma camada, enquanto um terceiro mostra um quadro HTML com links para o seu site organização. Defina o *Número de páginas* para o valor desejado. Você pode escolher a página *Orientação* e sua *Resolução de Exportação*. Quando marcada, **lcaixal imprimir como raster** significa que todos os elementos serão rasterizados antes de imprimir ou salvar como PostScript ou PDF.
- *Grade* permite que você personalize as configurações de rede como *espaçamentos*, *deslocamentos* e *tolerância* a sua necessidade.
- Em *Atrair para alinhamentos*, você pode mudar a *Tolerância*, que é a distância máxima abaixo da qual um item é encaixado para guias inteligentes.

Alinhar à grade e / ou guias inteligentes pode ser ativado a partir do menu *Ver*. Neste menu, você também pode ocultar ou mostrar a grade e guias inteligentes.

19.1.6 Ítems do compositor opções comuns

Os itens do compositor possuem um conjunto de propriedades comuns que você vai encontrar dentro da guia *Propriedades do item*: Posição e tamanho, rotação, moldura, Fundo, ID de Item e Renderizar (Veja [figure_composer_common_1](#)).

- A janela *Posição e tamanho* permite que defina o tamanho e posição da moldura que contém o item. Pode também escolher que *Ponto de referência* será configurado nas coordenadas **X** e **Y** previamente definidas.
- A *Rotação* define a rotação do elemento (em graus).

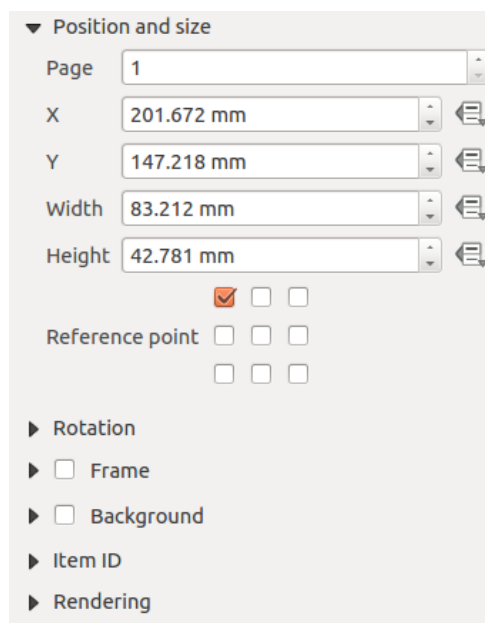





Figura 19.2: Janela comum das propriedades do item 

- A  *Moldura* mostra ou esconde a moldura à volta da etiqueta. Clique nos botões de [Cor] e [Espessura] para ajustar essas propriedades.
- A **lcaixal** *Fundo* ativa ou desativa uma cor de fundo. Clique no botão [Cor ...] para exibir uma caixa de diálogo onde você pode escolher uma cor ou escolher uma configuração personalizada. A transparência também pode ser ajustada ao longo de todo o campo **alfa**.
- Use o *ID do item* para criar uma relação com outros itens do compositor de impressão. Isto é utilizado com servidor do QGIS e qualquer cliente web potencial. Você pode definir o ID de um item (por exemplo, um mapa e uma etiqueta), e, em seguida, o cliente web pode enviar dados para definir uma propriedade (por exemplo, texto do rótulo) para esse item específico. O comando `GetProjectSettings` irá listar IDs dos itens que layout estão disponíveis.
- *Editando* moda será selecionado no campo opção. Ver [Rendering_Mode](#).

Nota:

- Se você habilitou a caixa **lcaixal** *Use live-atualizando cor seletor dialogs* nas opções gerais do QGIS, o botão de cor será atualizado assim que você escolher uma novas janelas de cores **Diálogo de Cor**. Se não, você precisa fechar o **Diálogo de Cor**.
- O ícone  *Dados definido override* ao lado de um campo significa que você pode associar o campo com dados no item do mapa ou utilizar expressões. Estes são particularmente útil na geração atlas (Ver [atlas_data_defined_overrides](#)).

19.2 Modo de Renderização

QGIS agora permite edição avançada para itens do compositor assim como camadas vetoriais e raster.

- **Transparência** **Ideslizantel**: Você pode fazer o item subjacente no Compositor visível com esta ferramenta. Use o controle deslizante para se adaptar a visibilidade do seu item para as suas necessidades. Você também pode fazer uma definição precisa do percentual de visibilidade no menu ao lado do controle deslizante.
- **lcaixal** *Excluir item da exportação*: Você pode optar por fazer um item não aparecer em todas as exportações. Depois de ativar esta opção, o item não será incluído no PDF ou, impressões etc ..

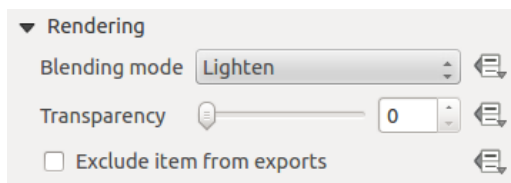



Figura 19.3: Modo de Renderização 🐧

- *modo de mistura*: Você pode conseguir efeitos de edição especiais com essas ferramentas que anteriormente só poderiam ser feitas a partir de programas gráficos. Os pixels da sua sobreposição e itens sobrepostos são misturados através das configurações descritas abaixo.
 - Normal: Este é o modo de mistura padrão, que usa o canal alfa do pixel superior para se misturar com o pixel abaixo dele; as cores não se misturam.
 - Clarear: Seleciona o máximo de cada um dos componentes do primeiro plano e dos pixels de fundo. Esteja ciente de que os resultados tendem a ser irregulares e permanentes.
 - Tela: Pixels claros da fonte são pintados sobre o destino, enquanto pixels escuros não. Este modo é muito útil para misturar a textura de uma camada com outra (por exemplo, você pode usar um monte sombreado para textura de outra camada).
 - Esquiva: Esquiva vai iluminar e saturar pixels subjacentes baseados na leveza do pixel superior. Então, quanto mais brilhantes pixels do topo causam a saturação e o brilho dos pixels subjacentes que aumentam. Isto funciona melhor se o início dos pixels não são muito brilhantes; caso contrário, o efeito é muito radical.
 - Adição: Este modo de mistura simplesmente adiciona os valores de pixel de uma camada com os valores de pixel da outra. No caso de valores acima de 1 (como no caso de RGB), o branco é exibida. Este modo é adequado para destacar feições.
 - Escurecer: Isso cria um pixel resultante que mantém os menores componentes de primeiro plano e dos pixels de fundo. Como clarear, os resultados tendem a ser irregulares e permanentes.
 - Multiplicar: Aqui, os números para cada pixel da camada superior são multiplicados com os números para o pixel correspondente da camada inferior. Os resultados são imagens mais escuras.
 - Queimar: Cores escuras na camada superior com que as camadas subjacentes a escurecer. Queimar pode ser usado para ajustar e colorir camadas subjacentes.
 - Sobreposição: Este modo combina os modos multiplicar e tela de mistura. Na imagem resultante, peças leves tornam-se mais leve e partes escuras ficam mais escuras.
 - Luz suave: Este é muito semelhante ao sobrepor, mas em vez de usar multiplicar / tela que usa a cor queimar / esquivar. Este modo deve emular brilhar uma luz suave em uma imagem.
 - Muita luz: Este modo é muito semelhante ao modo de sobreposição. É suposto simular a projeção de uma luz muito intensa numa imagem.
 - Diferença: Diferença subtrai o pixel superior a partir do pixel inferior, ou o contrário, para obter sempre um valor positivo. A mesclagem com preto não produz alterações, como a diferença com todas as cores é zero.
 - Subtrair: Este modo de mistura simplesmente subtrai os valores de pixel de uma camada com os valores de pixel da outra. No caso dos valores negativos, preto será exibido.


19.3 Itens do Compositor


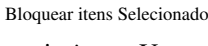
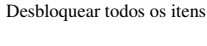
19.3.1 Ítem do mapa

Clique na barra de ferramentas botão  Adicionar novo mapa na barra de ferramentas do compositor de impressão para adicionar o mapa de tela do QGIS. Agora, arraste um retângulo para a tela Compositor com o botão esquerdo do mouse para adicionar ao mapa. Para exibir o mapa atual, você pode escolher entre três modos diferentes no guia do mapa *Propriedades do item*:

- **Retângulo** é a configuração padrão. Apenas exibe uma caixa vazia com a mensagem ‘O mapa será impresso aqui’.
- **Cache** torna o mapa na resolução de tela atual. Se você ampliar a janela Compositor dentro ou para fora, o mapa não é processado novamente, mas a imagem será redimensionada.
- **Edição** significa que, se você aumentar o zoom da janela Compositor dentro ou para fora, o mapa será processado novamente, mas por razões de espaço, apenas até uma resolução máxima.



Cache é o modo de visualização padrão para mapas do compositor de impressão recém-adicionados.

Você pode redimensionar o elemento do mapa clicando no botão  Selecionar / Mover item, selecione o elemento, e arraste uma das alças azul no canto do mapa. Com o mapa selecionado, agora você pode adaptar-se mais propriedades ao mapa guia *Propriedades do item*.

Para mover camadas dentro do elemento do mapa, selecione o elemento do mapa, clique no ícone  Mover conteúdo do item e mova as camadas dentro do quadro do item do mapa com o botão esquerdo do mouse. Depois de ter encontrado o lugar certo para o item, você pode bloquear a posição do item dentro da tela do compositor de impressão. Selecione o item do mapa e use a barra de ferramentas **bloqueado!**  ou a guia *itens* para bloquear o item. Um item bloqueado só podem ser selecionado usando a guia *itens*. Uma vez selecionado, você pode usar a guia *itens* para desbloquear itens individuais. O ícone **desbloqueado!**  vai desbloquear todos os itens bloqueados do compositor.

Propriedades principais

O diálogo *Propriedades principais* da guia do mapa *Propriedades do item* fornece as seguintes funcionalidades (ver *figure_composer_map_1*):

- A área de **Visão** permite que você defina os modos de visualização ‘retângulo’, ‘Cache’ e ‘Edição’, como descrito acima. Se você mudar a visão sobre o mapa da tela do QGIS alterando propriedades vetoriais ou raster, você pode atualizar a visualização do Compositor de impressão, selecionando o elemento do mapa no Compositor de impressão e clicando no botão **[Atualizar Visão]**.
- O campo *Escala* define a escala manual.
- O campo *Rotação* permite que você gire o conteúdo de elementos do mapa em graus. Note-se que uma estrutura de coordenada só pode ser adicionada com o valor padrão 0.
- *Desenhar itens na tela do mapa* permite mostrar as anotações que podem ser colocadas na tela do mapa na janela principal do QGIS.
- Você pode optar por bloquear as camadas mostradas em um item do mapa. Verifique **lcaixal bloquear camadas para item do mapa**. Depois que isso for verificado, qualquer camada que seria exibida ou ocultada na janela principal do QGIS não aparece ou seja é o item é escondido do Compositor de impressão. Mas o estilo e rótulos de uma camada bloqueada ainda são atualizados de acordo com a interface principal do QGIS.
- O botão  permite que você adicione rapidamente todos as visualização pre-definidas que você preparou no QGIS. Clicando no botão  você verá a lista de todos os pontos de visualização pre-definidos: basta selecionar a pre-visualização que você deseja exibir. A tela do mapa irá bloquear auto-

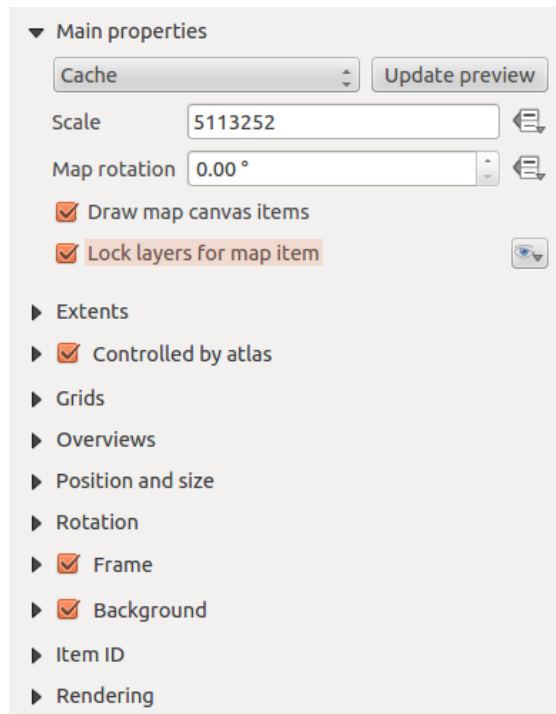



Figura 19.4: Separador das propriedades da Mapa 🐧

maticamente as camadas pré-definidas, permitindo a **caixal** bloquear camada para o item do mapa: se você quiser desmarcar a pré-visualização, basta desmarcar o botão **caixal** e pressione no botão . Veja *Legenda do Mapa* para descobrir como criar pré-visualizações.

Extensões

O diálogo *Extensão* das Propriedades do item do mapa, oferece as seguintes funcionalidades (veja *figure_composer_map_2*):

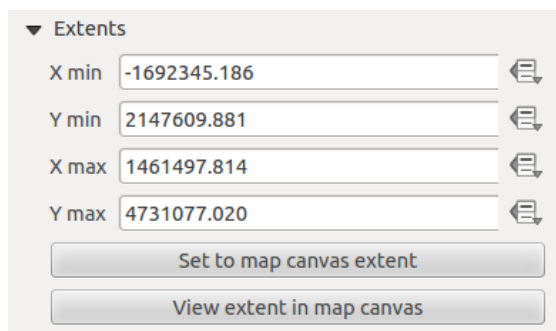


Figura 19.5: Janela de Extensões do Mapa

- A área **Extensões do Mapa** permite que você especifique a extensão do mapa usando os valores de X e Y mínimo e máximo clicando no botão **[Fixar o mapa a extensão da tela]**. Este botão define a extensão do item do mapa no compositor de impressão na medida da visualização do mapa atual da tela principal do QGIS. O botão **[Ver extensão na tela do mapa]** faz exatamente o oposto, ele atualiza a extensão da visualização do mapa na tela do aplicativo QGIS na mesma medida do item do compositor de mapa.

Se você mudar a visão sobre a tela do mapa do QGIS mudando propriedades vetoriais ou raster, você pode atualizar a visão do compositor de impressão, selecionando o elemento do compositor de impressão e clicando no botão **[Atualizar Visão]** na guia do mapa *Propriedades do item* (ver *figure_composer_map_1*).

Grades

O diálogo *Grades* do mapa guia *Propriedades do item* fornece a possibilidade de adicionar várias grades de um item do mapa.

- Com o botão de mais e menos você pode adicionar ou remover uma grade selecionada.
- Com o botão para cima e para baixo você pode mover uma grade na lista e definir a prioridade de desenho.

Quando você clicar duas vezes sobre a grade adicionado-a você pode renomear-la.

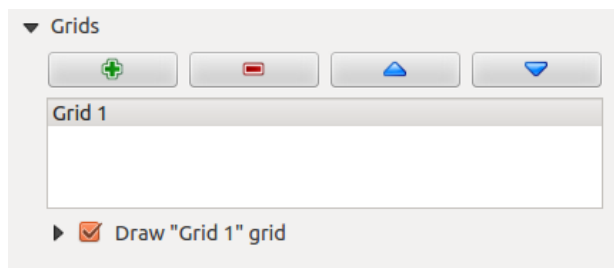


Figura 19.6: Diálogo do Grades do Mapa 🐧

Depois de ter adicionado uma grade, você pode ativar a caixa **Mostrar grade** para sobrepor uma grade sobre o elemento de mapa. Expanda esta opção que fornece um monte de opções de configurações, consulte [Figure_composer_map_4](#).

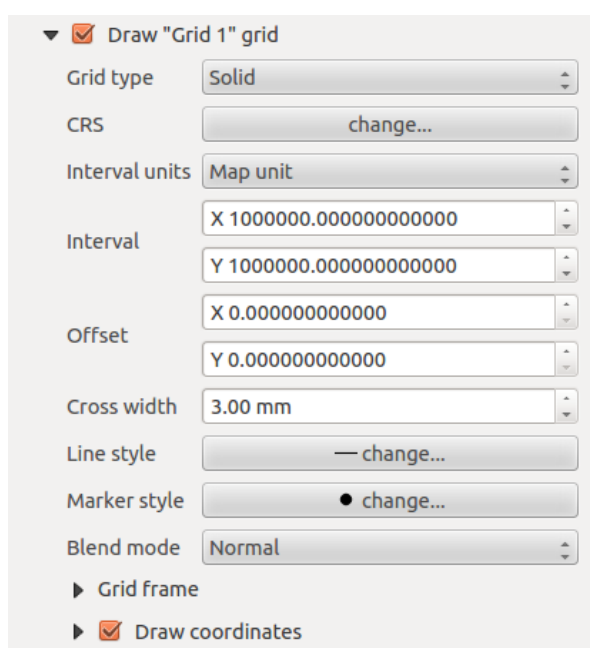


Figura 19.7: Diálogo do Grade de Desenho 🐧

Como tipo grade, você pode escolher entre usar uma linha sólida ou cruz. Simbologia da grade pode ser escolhida. Consulte a seção [Rendering_Mode](#). Além disso, você pode definir um intervalo nas direções X e Y, e compensação X e Y, e a largura usada para o tipo de grade cruzada ou linha.

- Existem diferentes opções para o estilo do quadro que contém o mapa. As seguintes opções estão disponíveis: Sem Moldura, Zebra, Linhas Interiores, Linhas Exteriores, Linhas Interiores e Exteriores e Linha de fora.
- Modo de renderização avançada também está disponível para grades (ver seção [Rendering_mode](#)).

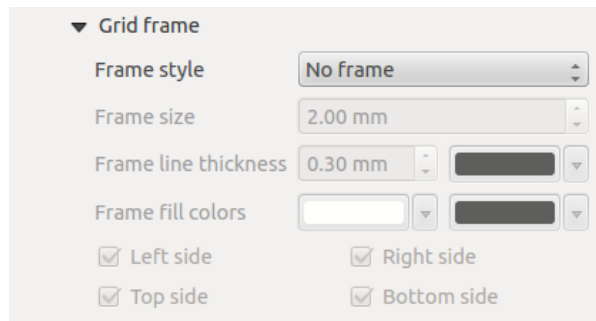


Figura 19.8: Diálogo do Quadro do Grade 🐧

- A caixa de seleção **caixal** *Desenhar coordenadas* permite que você adicione as coordenadas ao quadro do mapa. A anotação pode ser desenhada dentro ou fora do quadro do mapa. A direção da anotação pode ser definida como horizontal, vertical, horizontal e vertical, ou no sentido do limite, para cada fronteira individualmente. As unidades podem ser, em metros ou em graus. Finalmente, você pode definir a cor da grade, a fonte da anotação, a distância da anotação do quadro do mapa e a precisão das coordenadas traçadas.

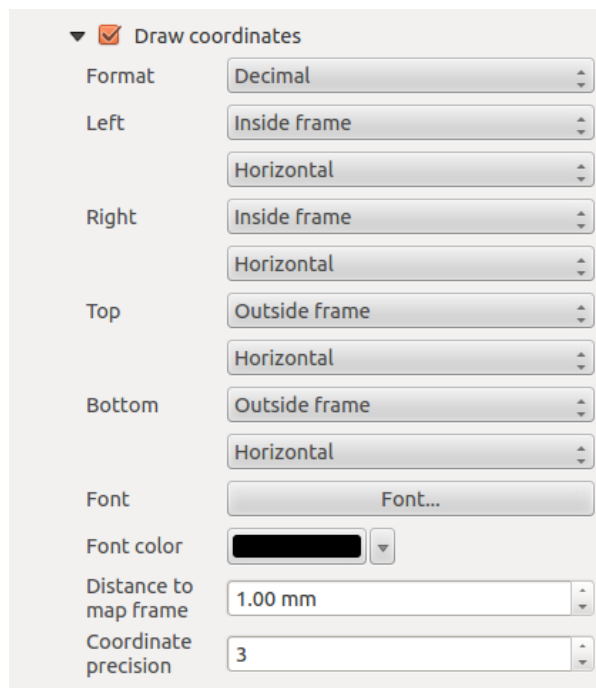


Figura 19.9: Diálogo do Desenho do Grade de Coordenadas 🐧

Enquadramentos

O diálogo *Enquadramentos* do mapa guia *Propriedades do item* apresenta as seguintes funcionalidades:

Você pode optar por criar um mapa geral, que mostra as extensões de outros mapa (s) que estão disponíveis no compositor. Primeiro você precisa criar o mapa (s) que deseja incluir no enquadramento. Em seguida, você cria o mapa que você deseja usando como enquadramento, assim como um mapa normal.

- Com o botão de mais e menos que você pode adicionar ou remover um enquadramento.
- Com o botão para cima e para baixo você pode mover um enquadramento na lista e definir a prioridade de desenho.

Abra :guilabel: *Enquadramentos* e pressione o ícone de botão mais verde para adicionar um enquadramento.

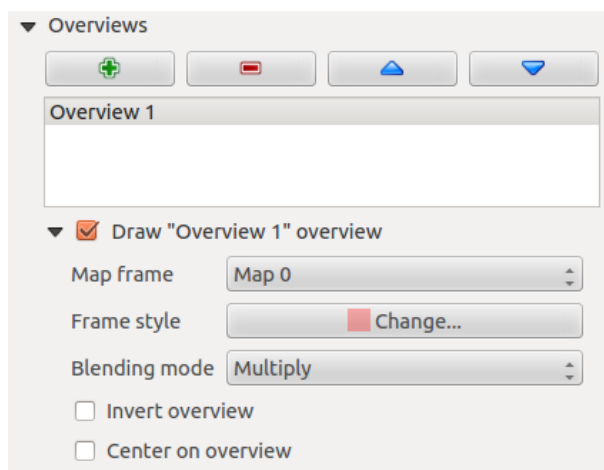



Figura 19.10: Diálogo do Mapa Enquadramentos 

Inicialmente, este enquadramento é nomeado ‘Enquadramento 1’ (ver [Figure_composer_map_7](#)). Você pode mudar o nome clicando duas vezes sobre o item introdutório na lista com o nome ‘Enquadramento 1’ e alterá-lo seu nome.

Quando você selecionar o item enquadramento na lista você pode personalizá-lo.

- A **lcaixal** *Desenho “<enquadramento I>” no enquadramento* precisa ser ativada para desenhar a extensão da moldura no mapa selecionado.
- A lista de combinação *Quadro do Mapa* pode ser usada para selecionar o item do mapa cuja extensão será desenhado no presente item do mapa.
- O *Estilo do Quadro* permite que você mude o estilo do quadro de síntese.
- A *modos de mistura* permite configurar diferentes modos de mistura de transparência. Veja [Rendering_Mode](#).
- A **lcaixal** *Inverter enquadramento* cria uma máscara em torno das extensões ativadas quando: as extensões do mapa referenciado são mostradas claramente, enquanto que todo o restante será misturado com a cor do quadro.
- A **lcaixal** *Centrar no enquadramento* coloca a extensão do quadro de visão geral no centro do mapa geral. Você só pode ativar um item da visão geral do centro, quando tiver adicionado várias visões gerais.

19.3.2 A rotulagem de item

Para adicionar um rótulo, clique no ícone  Adicionar rótulo, coloque o elemento com o botão esquerdo do mouse sobre a tela do compositor de impressão, posicione e personalize sua aparência no rótulo guia *Propriedades do item*.

A guia *Propriedades do item* do item rótulo fornece as seguintes funcionalidades para o item rótulo (ver [Figure_composer_label](#)):

Propriedades principais

- A caixa de diálogo *Propriedades do ítem* é onde o texto (HTML ou não) ou a expressão necessária para preencher o rótulo é adicionada à tela Compositor.
- As etiquetas podem ser interpretadas como códigos HTML : verifique **lcaixal** *edição como HTML*. Agora você pode inserir uma URL, uma imagem clicável com os links para uma página web ou algo mais complexo.

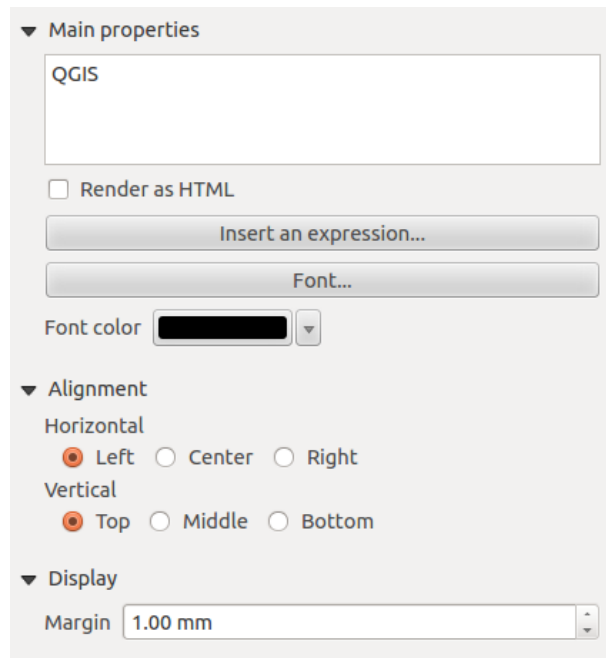




Figura 19.11: Separador das propriedades da Etiqueta 

- Você também pode inserir uma expressão. Clique em **[Inserir expressão...]** para abrir um novo diálogo. Construa uma expressão clicando nas funções disponíveis no lado esquerdo do painel. Duas categorias especiais pode ser úteis, particularmente associada com a funcionalidade atlas: funções de geometria e função registro. Na parte inferior, uma pré-visualização da Expressão é mostrada.
- Defina a *Fonte* clicando no botão **[Fonte...]** ou em *Cor da fonte* selecione uma cor usando a ferramenta de seleção de cores.

Alinhamentos e Exibição


- Você pode definir o alinhamento horizontal e vertical na zona *Alinhamento*.
- Em **Margem**, voce pode definir uma margem em mm. Esta é a margem para borda do compositor de impressão.

19.3.3 Ítem da imagem

Para adicionar uma imagem, clique no ícone  Adicionar imagem, coloque o elemento com o botão esquerdo do mouse sobre a tela do compositor de impressão, posicione e personalize sua aparência na imagem guia *item Propriedades*.

A guia figura da *Propriedades do item* apresenta as seguintes funcionalidades (ver [figure_composer_image_1](#)):

Primeiro você tem que selecionar a imagem que você deseja exibir. Existem várias formas de defini-la *fonte da imagem* na área **Propriedades principais**.

1. Use o botão de navegação  de *fonte da imagem* para selecionar um arquivo de seu computador usando a janela de navegação. A navegação iniciará na pasta da biblioteca SVG do QGIS. Além do SVG, você também pode selecionar outros formatos de imagem como .png ou .jpg.
2. Você pode inserir a fonte diretamente no campo de texto *fonte da imagem*. Você pode até mesmo fornecer um endereço URL remoto de uma imagem.
3. A partir da área **Buscar pastas**, você também pode selecionar uma imagem a partir de *Caminho de busca de imagem* para definir a origem da imagem.

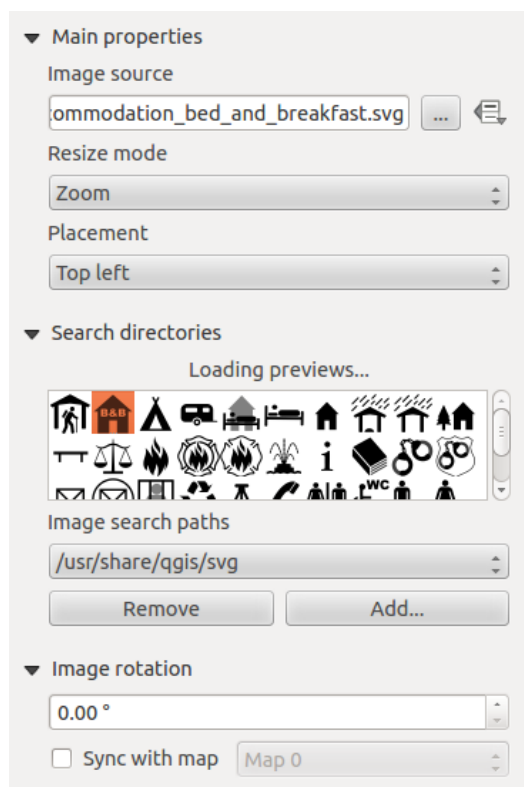




Figura 19.12: Separador Propriedades do Item da Imagem 

- Use os dados definidos no botão  para definir a fonte de imagem a partir de um registro ou usando uma expressão regular.

Com a opção *modo redimensionar*, você pode configurar como a imagem será exibida quando o quadro for alterado, ou optar por redimensionar o quadro do da imagem do item para que coincida com o tamanho original da imagem.


Voce pode seleccionar um dos seguintes modos:

- Zoom: Aumenta a imagem para o quadro, mantendo relação de aspecto da imagem.
- Ampliar: Amplia a imagem para caber dentro do quadro, ignora a relação de aspecto.
- Cortar: Utilize este modo só para imagens raster, ele define o tamanho da imagem ao tamanho da imagem original, sem escala e o quadro for usado para cortar a imagem, portanto, apenas a parte da imagem dentro do quadro será visível.
- Zoom e redimensionar quadros: Amplia a imagem para ajustar ao quadro, em seguida, redimensiona o quadro para encaixar imagem resultante.
- Redimensionar quadro para o tamanho da imagem: Configura o tamanho do quadro combinando com o tamanho original da imagem sem escala.

Modo redimensionar seleccionado pode desabilitar as opções de itens 'Posição' e 'Rotação de imagem'. A *Rotação de imagem* está ativa para o modo de redimensionamento 'Ampliar' e 'Zoom'.

Com *Posição* você pode seleccionar a posição da imagem dentro da moldura. A área **Buscar pastas** permite-lhe adicionar e remover pastas com imagens no formato SVG para o banco de dados de imagem. A pré-visualização das imagens encontradas nos diretórios seleccionados será mostrada em um painel e pode ser usada para seleccionar e definir a fonte da imagem.

As imagens podem ser giradas com a *Rotação de imagem*. Ativando a **caixa** *Sincronizar como o mapa* sincronize uma imagem da tela do QGIS (ou seja, uma seta norte rodada) com a imagem do compositor de impressão apropriada.

Também é possível selecionar uma seta norte diretamente. Se você primeiro selecionar uma imagem de seta para o norte do **Buscar pasta** e, em seguida, use o botão Procurar  do campo *fonte da imagem*, agora você pode selecionar uma seta norte da lista conforme exibido na [figure_composer_image_2](#).

Nota: Muitas das setas norte não tem um 'N' adicionado na seta de norte, isto é feito de propósito para idiomas que não usam um 'N' para Norte, para que eles possam usar outra letra.

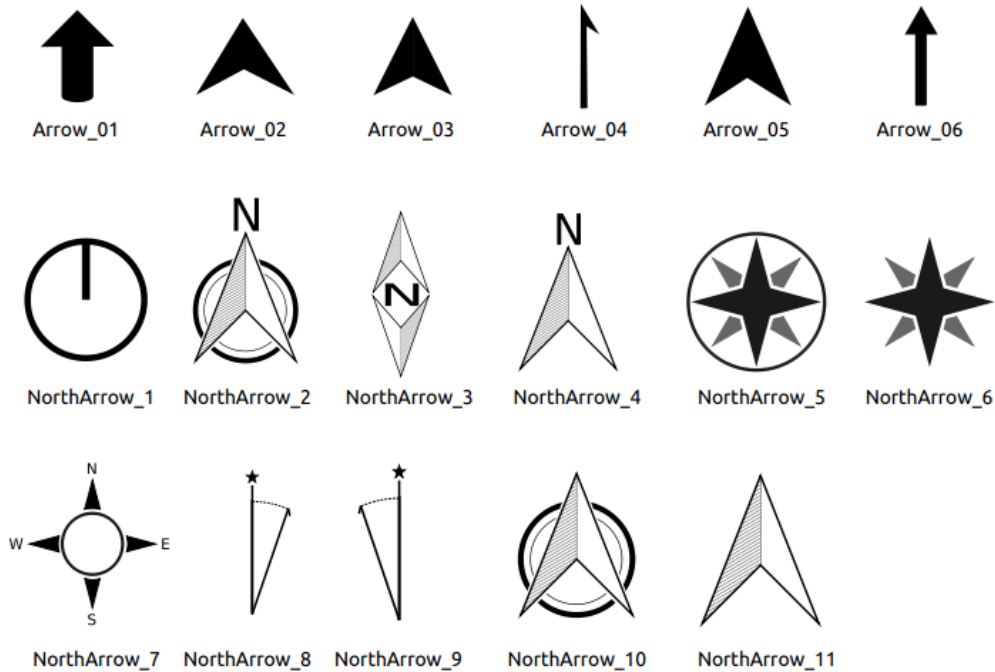



Figura 19.13: Flechas norte disponíveis para seleção na biblioteca SVG fornecida

19.3.4 Ítem da legenda

Para adicionar uma legenda do mapa, clique no ícone  Adicionar nova legenda, coloque o elemento com o botão esquerdo do mouse sobre a tela do compositor de impressão e posicione e personalize a aparência da legenda guia *Propriedades do item*.

A *Propriedades do item* da guia item de legenda oferece as seguintes funcionalidades (veja [figure_composer_legend_1](#)):

Propriedades principais

A guia *Propriedades do item* da legenda aba *Propriedades principais* apresenta as seguintes funcionalidades (ver [figure_composer_legend_2](#)):

Em *Propriedades principais* voce pode:

- Alterar o título da legenda.
- Definir o alinhamento do título para a esquerda, centro ou direita.
- Você pode escolher quais itens *Mapa* a legenda atual irá se referir na lista de seleção.
- Você pode quebrar o texto do título da legenda em um determinado caractere.

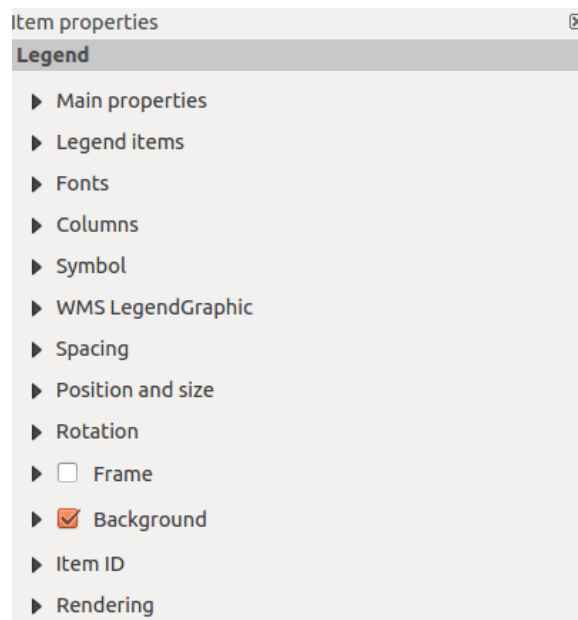



Figura 19.14: Propriedades do Separador da Legenda 

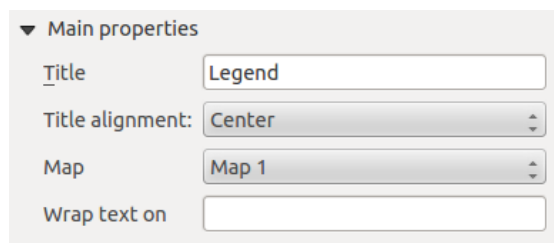



Figura 19.15: Janela das Propriedades principais da Legenda 

Itens Legenda

O diálogo *Itens da Legenda* da guia *Propriedades do item* apresenta as seguintes funcionalidades (ver [figure_composer_legend_3](#)):

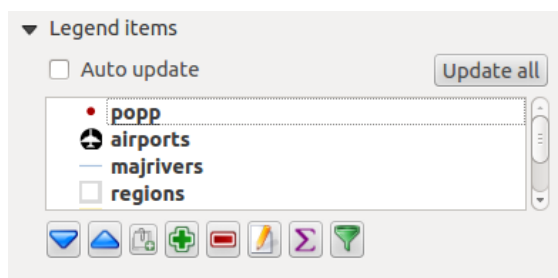


Figura 19.16: Legenda da Janela dos itens da Legenda 

- A legenda será atualizada automaticamente se a **lcaixal** *Atualização Automática* estiver marcada. Quando *Atualização Automática* estiver desmarcada isso vai lhe dar mais controle sobre os itens de legenda. Os ícones abaixo na lista de itens da legenda estão ativados.
- A janela de itens de legenda lista todos os itens de legenda e permite-lhe alterar a ordem dos itens, grupos de camadas, remover e restaurar itens na lista, editar os nomes das camadas e adicionar um filtro.
 - A ordem dos itens podem ser alterados usando os botões [**Acima**] e [**Abaixo**] ou com a funcionalidade de ‘arrastar e soltar’. A ordem não pode ser alterada para legenda gráfica WMS.
 - Use o botão [**Adicionar grupo**] para adicionar um grupo na legenda.
 - Use os botões [**Mais**] e ****[Menos]** para adicionar ou remover camadas.
 - O botão [**Editar**] é usado para editar a camada-, nome do grupo ou título, primeiro voce deve seleciona-la nos itens da legenda.
 - O botão [**Sigma**] adiciona uma contagem de feições para cada camada de vetor.
 - Use o botão [**Filtro**] para filtrar a legenda por conteúdos do mapa, apenas os itens visíveis da legenda do mapa serão listados na legenda.

Depois de alterar a simbologia na janela principal do QGIS, você pode clicar em [**Atualizar**] para carregar as mudanças dos elementos da legenda do compositor de impressão.

Fontes, Colunas e Simbolo

Os diálogos *Fontes*, *Colunas* and *Símbolo* da guia *Propriedades do item* apresenta as seguintes funcionalidades (ver [figure_composer_legend_4](#)):

- Pode mudar a fonte do título, grupo, subgrupo, e item (camada) no item legenda. Clique no botão da categoria para abrir a janela de **Selecionar fonte**.
- Você escolhe a cor dos rótulos em **Cor da fonte** usando o seletor de cores avançada, no entanto, a cor selecionada será dada a todos como fonte de todos os itens da legenda..
- Itens de legenda podem ser organizados ao longo de várias colunas. Defina o número de colunas no campo *Contagem* .
 - *Iguais larguras de colunas* define como as colunas da legenda devem ser ajustadas.
 - A opção *Dividir camadas* permite a formação de uma legenda de camada graduada ou categorizada para ser dividida entre as colunas.
- Pode alterar a largura e altura para o símbolo da legenda nesta janela.

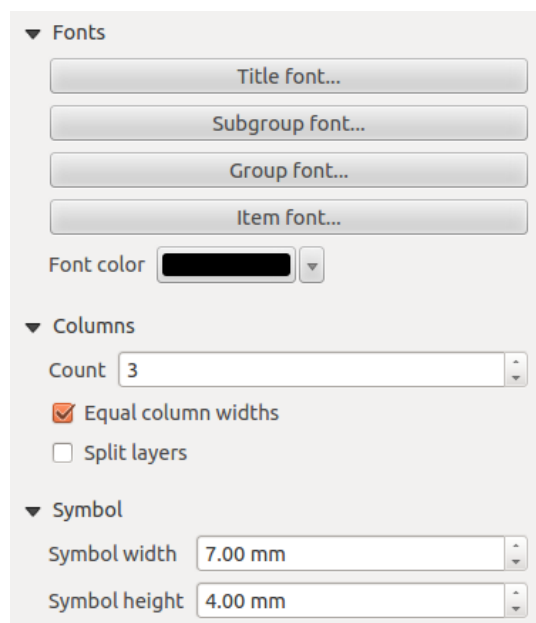


Figura 19.17: Legenda da Janela Fontes, Colunas, Símbolos e Espaçamento 

WMS LegendaGráfico e Espaçamento

Os diálogos *WMS legendGráfica* e *Espaçamento* da guia legenda *Propriedades do item* apresenta as seguintes funcionalidades (veja [figure_composer_legend_5](#)):

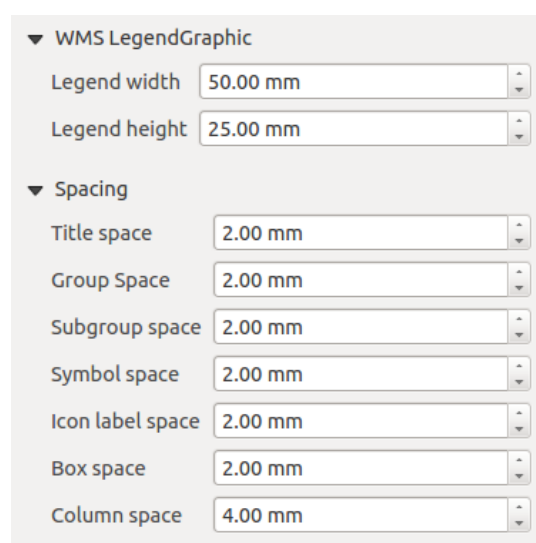



Figura 19.18: Diálogo WMS legendGraphic 

Quando tiver adicionado uma camada WMS e você inserir um item no compositor de legenda, uma solicitação será enviada para o servidor WMS para fornecer uma legenda WMS, Esta Legenda só será apresentada se o servidor WMS fornecer a capacidade GetLegendGraphic. O conteúdo legenda WMS será fornecido como uma imagem raster.

WMS legendGraphic é usado para habilitar e ajustar a *Largura da Legenda* e a *Altura da Legenda* da imagem raster da legenda WMS.

Espaçamento em torno do título, grupo, subgrupo, símbolo, ícone do rótulo, espaço da caixa ou espaço da coluna podem ser personalizados através deste diálogo.

19.3.5 Ítem da escala de barra

Para adiciona barra de espaço, clique no ícone  Adicionar nova barra de escala, colocar o elemento com o botão esquerdo do mouse sobre a tela do compositor de impressão posicionar e personalizar a aparência na barra de escala guia *Propriedades do ítem*.

A *Propriedades do item* de uma barra de escala apresentam as seguintes funcionalidades (veja [figure_composer_scalebar_1](#)):

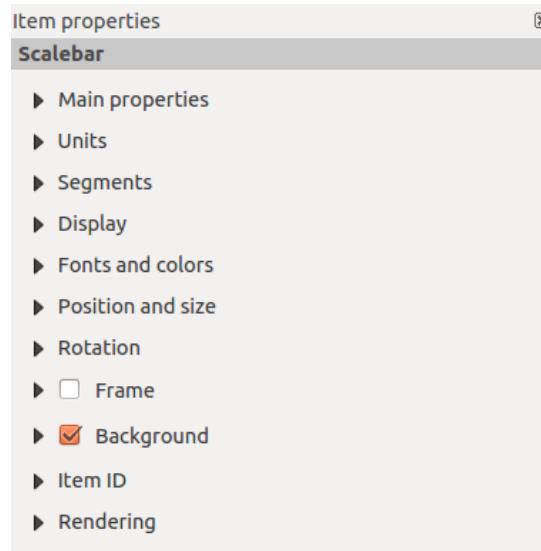


Figura 19.19: Guia propriedades do item da Barra de Escala 

Propriedades principais

O diálogo *Propriedades principais* da guia Barra de escala *Propriedades do item* apresenta as seguintes funcionalidades (veja [figure_composer_scalebar_2](#)):

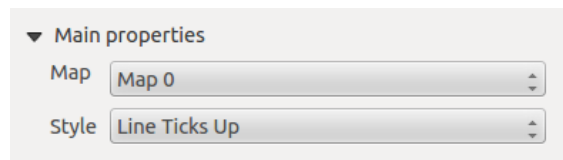


Figura 19.20: Diálogo Propriedades do item da Barra de Escala 

- Primeiro, escolha o mapa a que a barra de escala será ligada.
- Em seguida, escolha o estilo da barra de escala. Seis estilos estão disponíveis:
 - Estilos **Box simples** e **Box duplo**, que contêm uma ou duas linhas de caixas de cores alternadas.
 - Linhas de atração **Meio**, **Acima** ou **Abaixo**.
 - **Numérico**, onde a proporção de escala será impressa (ex., 1:50000).

Unidades e Segmentos

Os diálogos *Unidades* e *Segmentos* da guia Barra de escala *Propriedades do item* apresenta as seguintes funcionalidades (veja [figure_composer_scalebar_3](#)):

Nestas duas caixas de diálogo, você pode definir a forma como a barra de escala será representada.

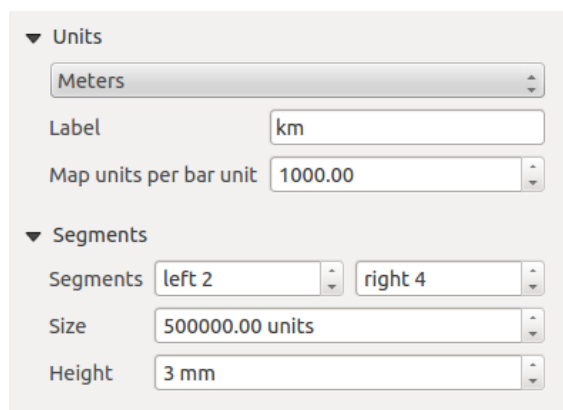



Figura 19.21: Diálogos Segmentos e Unidades da Barra de Escala 

- Selecione a unidade usada no mapa. Você tem quatro possibilidades: **Unidades do mapa** será automaticamente a unidade selecionada; **Metros**, **Pés** ou **Milhas Nauticas** forçam a conversão de unidade.
- O campo *Rótulo* define o texto utilizado para descrever a unidade na barra de escala.
- A *Unidade do mapa por unidade da barra* permite você fixar a taxa entre uma unidade do mapa e sua representação na barra de escala.
- Você pode definir quantos *Segmentos* serão desenhados no lado esquerdo e no lado direito da barra de escala, e qual dimensão cada segmento terá no campo (*Tamanho*). *Altura* também poderá ser definido.

Tela

O diálogo *Mostrar* da guia Barra de escala *Propriedades do item* apresenta as seguintes funcionalidades (veja [figure_composer_scalebar_4](#)):

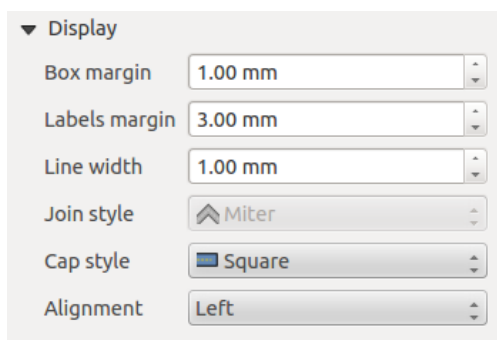



Figura 19.22: Tela da Barra de escala 

Você pode definir agora a barra de escala que será mostrada no quadro.

- *Margem da caixa*: espaço entre o texto e as bordas do quadro
- *Margens dos rótulos*: espaço entre o texto e o desenho da barra de escala
- *Espessura da linha*: espessura da linha desenhada na barra de escala
- *Estilo da união*: Cantos e acabamentos da barra de escala no estilo Cifrado, Pontagudo e Arredondado (disponível apenas para escala de barra estilo caixa simples e caixa dupla)
- *Estilo de cobertura*: Acabamento de todas as linhas no estilo Quadrado, Redondo ou Plano (disponível apenas para Barra de escala estilo Linha tracejadas em cima, em baixo e no meio)
- *Alinhamento*: Posicione o texto na esquerda, meio ou direita do quadro (utilizado apenas para barras de escala estilo Numérico)

Fontes e cores

The *Fonts and colors* dialog of the scale bar *Item Properties* tab provide the following functionalities (see [figure_composer_scalebar_5](#)):

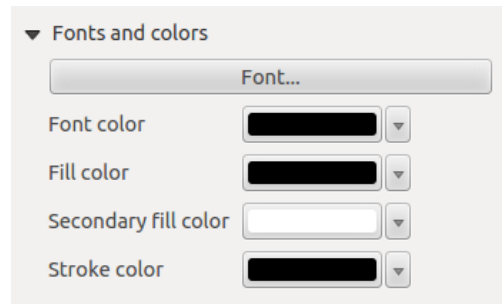





Figura 19.23: Diálogos Cores e Fontes da Barra de Escala 

Voce pode definir a fonte e as cores a usar na barra de escala.

- Use o botão **[Fonte]** para escolher a fonte
- *Cores da Fonte*: escolha a cor da fonte
- *Cor do preenchimento*: define a primeira cor de preenchimento
- *Cor secundária de preenchimento*: define a cor secundária de preenchimento
- *Cor do traço*: define a cor das linhas da barra de escala

Cor de preenchimento são usadas somente para caixa de escala estilos simples e duplo. Para seleccionar uma cor que você pode usar a opção da lista usando a seta para baixo para abrir uma opção de seleção de cores simples ou a opção de seleção de cores mais avançado, que é iniciado quando você clicar na caixa de cor na caixa de diálogo.

19.3.6 Ítems Básicos do Shape

Para adicionar uma forma básica (elipse, retângulo, triângulo), clique no ícone  Adicionar elipse ou no ícone  Adicionar Seta, coloque o elemento mantém pressionado o botão esquerdo do mouse. Personalize a aparência na guia *Propriedades do item*.

Quando você também mantenha a tecla `Shift` ao colocar a forma básica, você pode criar um quadrado, círculo ou triângulo, perfeitos.

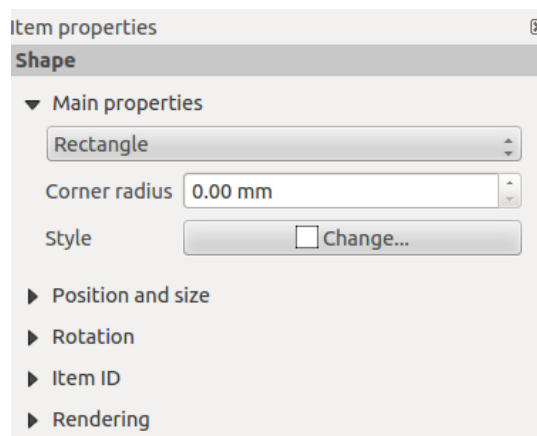



Figura 19.24: Separador de propriedades da Forma 


O item guilabel:*Formato* da guia Propriedades do item permite que você selecione se você quer desenhar uma elipse, retângulo ou triângulo dentro do quadro dado.

Você pode definir o estilo da forma usando o diálogo avançado de estilos de símbolos com o qual você pode definir sua cor de contorno e preenchimento, padrão de preenchimento, usar marcadores etc.

Para a forma de retângulo, você pode definir o valor do raio dos cantos ao redor dos cantos.

Nota: Ao contrário de outros itens, você não pode estilizar o quadro ou a cor do quadro de fundo.

19.3.7 Ítem de Seta

Para adicionar uma seta, clique no ícone  Adicionar seta, para colocar o elemento mantenha pressionado o botão esquerdo do mouse e arraste uma linha para desenhar a seta na tela do compositor de impressão componha e personalize a aparência na guia barra de escala *Propriedades do item*.

Quando você também mantenha a tecla *Shift* ao colocar a seta, ele é colocada em um ângulo de exatamente **45 graus**.

O item seta pode ser utilizado para adicionar uma linha ou uma seta simples que pode ser utilizada, por exemplo, para mostrar a relação entre outros itens do compositor de impressão. Para criar uma seta de norte, o item de imagem deve ser considerado em primeiro lugar. QGIS tem um conjunto de flechas Norte no formato SVG. Além disso, você pode conectar um item de imagem com um mapa para que ele possa rodar automaticamente com o mapa (ver [the_image_item](#)).

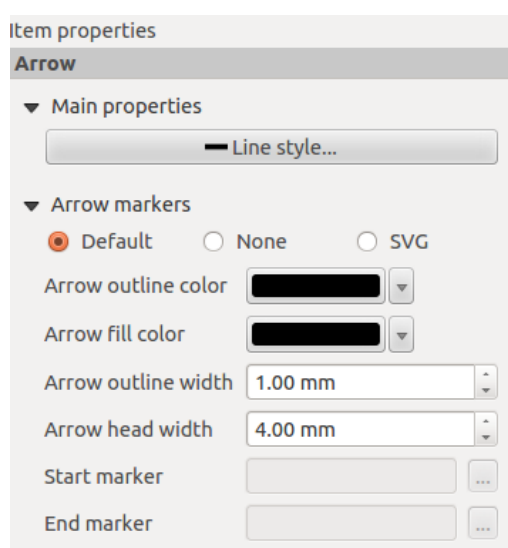



Figura 19.25: Separador das propriedades da Seta 

Propriedades do ítem

O item *Seta* da Propriedades do item permite voce configurar o item seta.

O botão **[Estilo de linha ...]** pode ser usado para selecionar o estilo de linha usando o selecionador de símbolos.

Em *Marcadores de seta* voce pode selecionar um dos três botões.

- *Padrão*: Para desenhar um seta regular, dá-lhe opções para estilizar a ponta da seta
- guilabel:*Nenhum*: Permite desenhar uma linha de seta sem ponta
- *SVG* : Permite desenhar uma linha em SVG *Marcador inicial* e *o/ou Marcador final*

Por *Padrão* marcadores de Seta voce pode usar as seguintes opções de estilo de ponta de seta.


- *Cor de contorno da seta* : Defina a cor de contorno da cabeça de seta
- *Cor de preenchimento da seta*: Defina a cor de preenchimento da seta
- *Espessura do contorno da seta* : Defina a espessura do contorno da seta
- *Espessura da ponta da seta*: Defina a espessura da seta

Para marcadores *SVG* voce pode usar as seguintes opções.

- *Marcadores iniciais* : Escolha uma imagem SVG para desenho no início da linha
- *Marcador final* : Escolha uma imagem SVG para desenhar no final da linha
- *Espessura da ponta da seta*: Defina a espessura da ponta e/ou contorno da seta

Imagens *SVG* são automaticamente rodadas com a linha. A cor da imagem *SVG* não pode ser alterada.

19.3.8 Ítens da Tabela de Atributos

É possível adicionar partes de uma tabela de atributos vetorial na tela do compositor de impressão: Clique no ícone  Adicionar tabela de atributo, coloque o elemento com o botão esquerdo do mouse sobre a tela do compositor de impressão, posicione e personalize a aparência na guia *Propriedades do item*.

A *propriedades do Item* da janela ítem da tabela de atributos apresenta as seguintes funcionalidades (ver [figure_composer_table_1](#)):

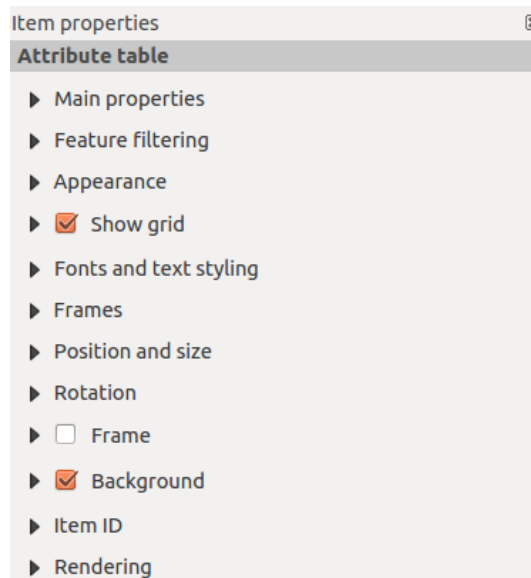



Figura 19.26: Tabela Propriedades do ítem da tabela de atributos 

Propriedades principais

O diálogo *Propriedades principais* da guia da tabela de atributos :guilabel: 'Propriedades do item' apresenta as seguintes funcionalidades (veja [figure_composer_table_2](#)):

- Para *Fonte* voce pode normalmente selecionar uma 'Feição de camada'.
- Com *Camada* voce pode escolher qual camada vetorial carregará no projeto.
- O botão **[Atualizar os dados da tabela]** pode ser usado para atualizar a tabela com o conteúdo atual da tabela após modificada.

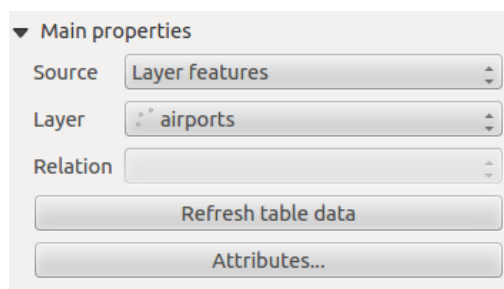



Figura 19.27: Diálogo propriedades principais da Tabela de Atributos 

- O botão **[Atributos...]** inicia o menu *Selecione atributos*, ver [figure_composer_table_3](#), ele pode ser usado para alterar o conteúdo visível na tabela. Ao final das alterações use o botão **[OK]** para aplicar as alterações na tabela.

Na seção *Colunas* voce pode:

- Remover um atributo, basta selecionar uma linha do atributo clicando em qualquer lugar na fila e pressione o botão de menos para remover o atributo selecionado.
- Adicionar um novo atributo usando o botão mais. No final, uma nova linha vazia aparece e você pode seleciona célula vazia da coluna *Atributo*. Você pode selecionar um atributo de campo a partir da lista ou você pode selecionar para construir um novo atributo usando uma expressão regular.
- Use as setas para acima e para abaixo para alterar a ordem dos atributos na tabela.
- Selecione uma célula no cabeçalho da coluna para alterar o cabeçalho, entre um novo nome.
- Selecione uma célula na linha da coluna e voce pode alterar o alinhamento entre Esquerda, centro e direita.
- Selecione uma célula na espessura da coluna e voce pode alterar automaticamente a espessura em mm, entrando um número. Voce pode desfazer a alteração automática, usando o voltar.
- O botão **[Redefinir]** pode ser usado para restaurar as configuração original do atributo.

Na seção *Classificar* voce pode:

- Adicionar um atributo para classificação na tabela. Selecionando um atributo e definindo a ordem de classificação para ‘Crescente’ ou ‘Decrescente’ e pressionando o botão mais. Uma nova linha será adicionada na lista da ordem de classificação.
- selecione uma linha na lista e use o botão acima e abaixo para alterar a prioridade de classificação em nível de atributo.
- use o botão menos para remover um atributo da lista de ordem de classificação.

filtragem de feição

O diálogo *Filtragem de elementos* da guia tabela de atributos *Propriedades do item* apresenta as seguintes funcionalidades (veja [figure_composer_table_4](#)):

Voce pode:

- Definir o *Máximo de linhas* para serem visualizadas.
- Ative **lcaixal** *remover linhas duplicadas na tabela* para mostrar elementos sem repetição.
- Ative **lcaixal** *Mostrar apenas feições visíveis dentro de um mapa* e selecione o correspondente *Compor mapa* para mostrar os atributos das feições visíveis no mapa selecionado.
- Ative **lcaixal** *Mostrar apenas feições visíveis dentro do atlas* apenas se tiver habilitado a **lcaixal** *Geração de Atlas*. Quando ativado, ele irá mostrar uma tabela com apenas as características mostradas no mapa da página especial do atlas.

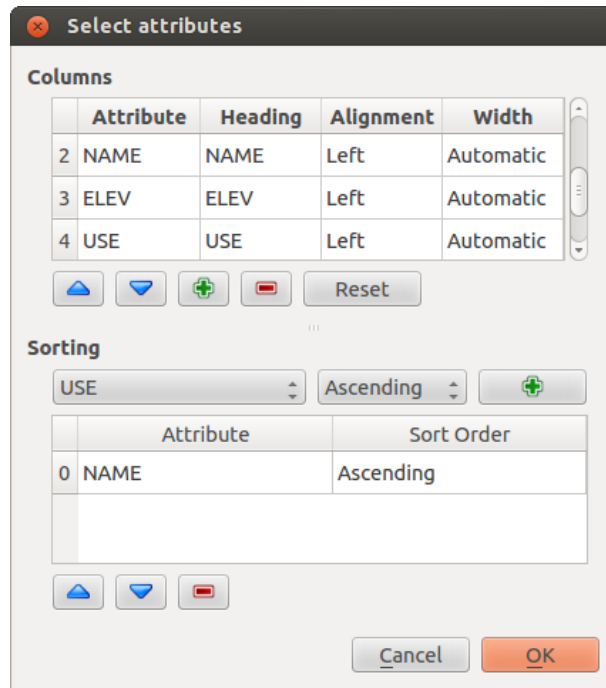


Figura 19.28: Janela Seção de atributos da tabela de atributos 🐧

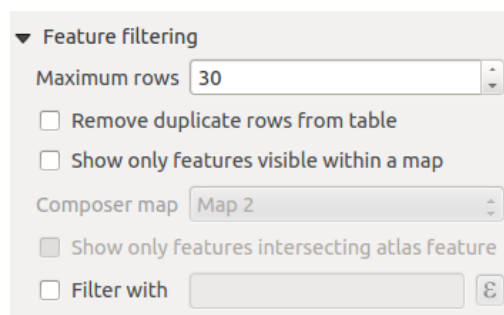


Figura 19.29: Diálogo Filtragem de elementos da tabela de atributos 🐧

- Ative **lcaixal** *Filtrar com* e provendo um filtro por tipo na entrada de linha ou inserindo uma expressão regular usando o botão de expressão disponível. Alguns exemplos de declarações de filtragem você pode usar quando você carregou a camada aeroportos do conjunto de dados de exemplo:

- ELEV > 500
- NAME = 'ANIAK'
- NAME NOT LIKE 'AN%
- regexp_match(atributo(\$currentfeature, 'USE') , '[i]')

A última expressão regular irá incluir apenas os arpoirts que têm uma letra 'i' no campo atributo 'USE'.

Aparência

o Diálogo *Aparancia* da guia da tabela de atributos *Propriedades do item* fornecer as seguintes funcionalidades (veja [figure_composer_table_5](#)):

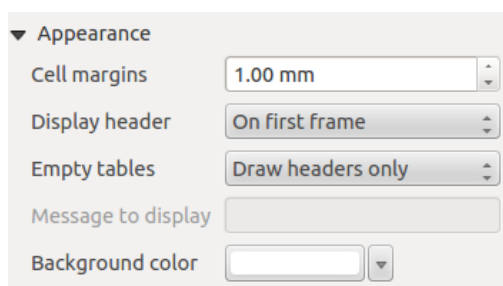


Figura 19.30: Diálogo aparência da tabela de atributos 

- Com *Celulas marginais* você pode definir a margem em torno de texto em cada célula da tabela.
- Com *leitura em tela* você pode selecionar a partir de uma lista de um dos 'Iniciar primeiro quadro', 'Em todos os quadros' opção padrão, ou 'Não leitura'.
- A opção *Mensagens da tabela* controla o que será exibida quando resultado da seleção está vazio.
 - **Desenhar apenas cabeçalhos**, só vai desenhar o cabeçalho, exceto se você tiver escolhido 'Não leitura' para :guilabel: *leitura em tela*.
 - **Ocultar tabela inteira**, só vai desenhar o fundo da tabela. Você pode ativar **lcaixal**: :guilabel:*Não pintar o fundo se o quadro estiver vazio* em *Molduras* para esconder completamente a tabela.
 - **Desenhar apenas cabeçalhos**, irá preencher a tabela de atributos com células vazias, esta opção também pode ser usado para fornecer células vazias adicionais quando você tem um resultado para mostrar!
 - **Mostrar mensagem conjunta**, vai chamar o cabeçalho e adiciona uma célula que abrange todas as colunas e exibir uma mensagem do tipo "Nenhum resultado", que pode ser fornecido na opção *Mensagens na tela*
- A opção *Mensagens na tela* só é ativada quando você tiver selecionado **Mostrar mensagem conjunta** para *Mensagem da tabela*. A mensagem fornecida será mostrada na tabela na primeira linha, quando o resultado for uma tabela vazia.
- Com *Cor do fundo da tabela* você pode definir a cor de fundo da tabela.

Mostrar Grid

A guia *Mostrar grade* do diálogo da tabela de atributo :guilabel:*Propriedades do item* fornecer as seguintes funcionalidades (veja [figure_composer_table_6](#)):

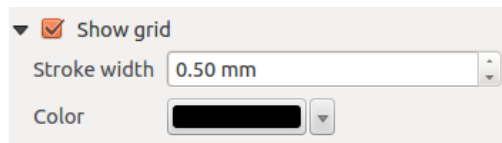




Figura 19.31: Diálogo Mostrar grade da tabela de atributos 

- Ative  *Mostrar grade* quando você quiser exibir a grade, os contornos das células da tabela.
- Com *Largura do traço* você pode definir a espessura das linhas utilizadas na grade.
- A *Cor* da grade pode ser definida usando a caixa de diálogo de seleção de cores.

Estilo de fonte e texto

O diálogo *Estilo da fonte e do texto* da guia da tabela de atributo *Propriedades do item* fornecer as seguintes funcionalidades (veja [figure_composer_table_7](#)):

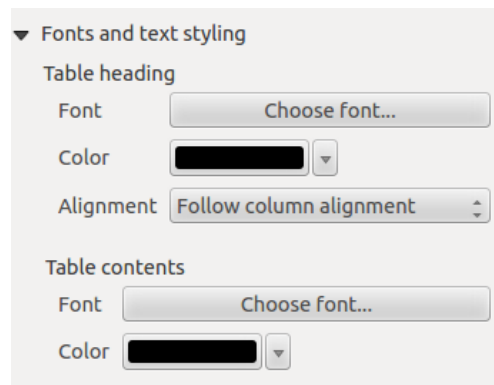



Figura 19.32: Diálogo Estilo da fonte e do texto da tabela de atributos 

- Você pode definir *Fonte* e *Cor* para *Cabeçalho da tabela* e *Tabela de conteúdos*.
- Para *Cabeçalho da tabela* você também poderá definir o *Alinhamento* e escolher entre *Seguir o alinhamento da coluna*, 'Esquerda', *Centro* ou *Direita*. O alinhamento da coluna é definido usando o diálogo *Seleção de atributos* (ver [Figure_composer_table_3](#)).

Quadros

O diálogo *Molduras* da guia da tabela de atributos: `guiabel:Propriedades` item fornecer as seguintes funcionalidades (veja [figure_composer_table_8](#)):

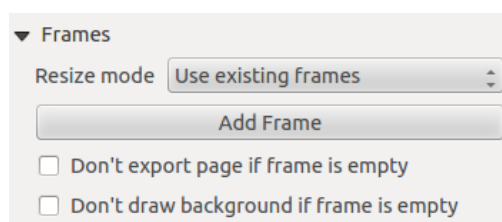



Figura 19.33: Quadros de Diálogo da Tabela de Atributos 

- Com *modo redimensionar* você pode selecionar a forma a transformar o conteúdo da tabela atributo:
 - *Usar molduras existentes* exibe o resultado no primeiro quadro e apenas quadros adicionados.
 - *Estender para a próxima página* vai criar tantos quadros (e páginas correspondentes) como necessárias para exibir a seleção completa de tabela de atributos. Cada quadro pode ser movida no layout. Se você redimensionar um quadro, a tabela resultante será dividido entre os outros quadros. O último quadro será cortado para caber na mesa.
 - *Repetir até terminar* também vai criar tantos quadros como a opção *Estender para próxima página*, exceto todos os quadros terão o mesmo tamanho.
- Use o botão **[Adicionar moldura]** para adicionar outro quadro com o mesmo tamanho do quadro selecionado. O resultado da tabela que não vai caber no primeiro quadro vai continuar no próximo quadro quando você reajusta usa o modo *Usar molduras existentes*.
- Ative **lcaixal** *Não exportar a página se o quadro estiver vazio* impede que a página seja exportada quando o quadro de tabela não tem conteúdo. Isso significa que todos os outros itens compositor, mapas, barra de escala, legendas etc. não serão visíveis no resultado.
- Ative **lcaixal** *Não pintar o fundo se o quadro estiver vazio* impede que o fundo seja desenhado quando o quadro de tabela não possui conteúdo.

19.3.9 Quadro itens HTML

É possível adicionar um quadro que exibe o conteúdo de um site ou até mesmo criar e estilo a sua própria página HTML e exibi-la!

Clique no ícone  Adicionar moldura HTML, coloque o elemento arrastando um retângulo mantém pressionado o botão esquerdo do mouse sobre a tela do compositor de impressão e posicionando e personalizando a aparência na guia *Propriedades do item* (veja [figure_composer_html_1](#)).

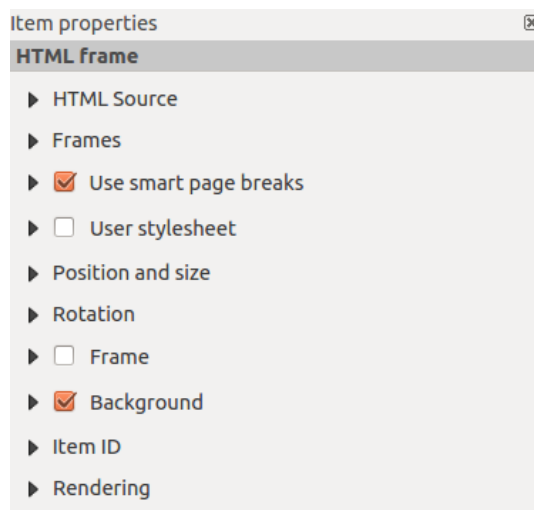



Figura 19.34: Quadros HTML, e ítems da propriedades da Tabela 

Código HTML

Como uma fonte de HTML, você pode definir uma URL e ativar o botão redondo URL ou digitar o código HTML diretamente na caixa de texto fornecida e ativar o botão redondo da opção Fonte.

O diálogo *Fonte HTML* da guia *Moldura HTML Propriedades do item* fornece as seguintes funcionalidades (veja [figure_composer_html_2](#)):

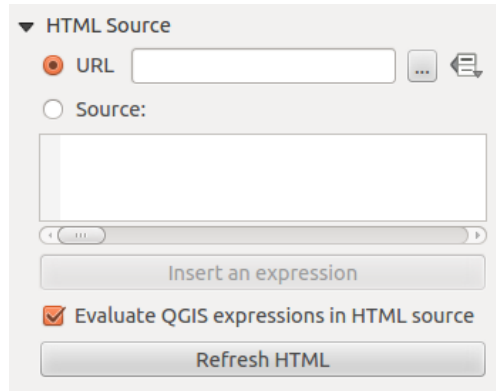




Figura 19.35: Quadro HTML, e propriedades do código HTML 

- Em *URL* você pode digitar a URL de uma página web que você copiou do seu navegador de internet ou selecionar um arquivo HTML utilizando o botão Procurar . Há também a opção de usar o botão de substituição de dados definido, para fornecer uma URL a partir do conteúdo de um campo de atributo de uma tabela ou usando uma expressão regular.
- Em *Fonte* você pode digitar o texto na caixa de texto com algumas tags HTML ou fornecer uma página HTML completo.
- O botão **[inserir uma expressão]** pode ser usado para inserir uma expressão como [%Year (\$now) %] na caixa de texto Fonte para exibir o ano em curso. Este botão só é ativado quando botão redondo *Fonte* estiver selecionado. Depois de inserir a expressão clique em algum lugar na caixa de texto antes de atualizar a estrutura HTML, caso contrário você vai perder a expressão.
- Ative **caixa** *Avaliar expressões QGIS na fonte HTML* para ver o resultado da expressão que foi incluída, caso contrário, você vai ver a expressão em seu lugar.
- Use o botão **[Atualizar HTML]** para atualizar o quadro (s) HTML para ver o resultado das mudanças.

Quadros

O diálogo *Molduras* da guia da Moldura HTML *Propriedades do item* fornece as seguintes funcionalidades (veja [figure_composer_html_3](#)):

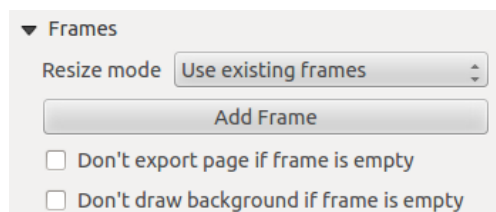



Figura 19.36: Quadro HTML, e propriedades dos quadros 

- Com *Modo de redimensionamento* você pode selecionar a forma de tornar o conteúdo HTML:
 - *Usar molduras existentes* exibe o resultado no primeiro quadro e apenas quadros adicionados.
 - *Estender para a próxima página* vai criar tantos quadros (e páginas correspondentes) como necessárias para tornar a altura da página web. Cada quadro pode ser movida no layout. Se você redimensionar um quadro, a página será dividido entre os outros quadros. O último quadro será cortado para caber na página web.
 - *Repetir em todas as paginas* irá repetir o canto superior esquerdo da página web em todas as páginas em quadros do mesmo tamanho.

- *Repetir até terminar* também vai criar tantos quadros como a opção *Estender para próxima página*, exceto todos os quadros terão o mesmo tamanho.
- Use o botão **[Adicionar moldura]** para adicionar outro quadro com o mesmo tamanho de quadro selecionado. Se a página HTML que não vai caber no primeiro quadro que vai continuar no próximo quadro quando você usar *Modo de redimensionamento* ou *Usar molduras existente*.
- Ative **lcaixal** *Não exportar página se o quadro estiver vazio* impede o layout do mapa de ser exportado quando o quadro não tem conteúdo HTML. Isso significa que todos os outros itens compositor, mapas, barra de escala, legendas etc. não será visível no resultado.
- Ative **lcaixal** *Não pintar o fundo se o quadro estiver vazio* impede que o quadro HTML sendo desenhado se o quadro está vazio.

Use quebras de página inteligentes e estilo de folha do usuário

O diálogo *Use quebra de página inteligente* e diálogo *Arquivos de estilo do usuário* da guia *Moldura HTML*: `guilabel:Propriedades do item` fornece as seguintes funcionalidades (veja `figure_composer_html_4`):

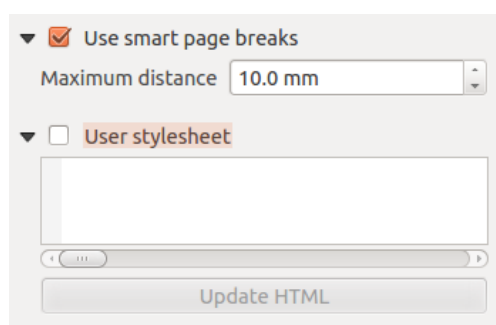


Figura 19.37: Propriedades de estilo do usuário, Moldura HTML, Use de quebra de página inteligente 🐧


- Ative **lcaixal** *Use quebra de página inteligente* para evitar que o conteúdo do quadro html quebre no meio do caminho uma linha de texto por isso continua agradável e suave no próximo quadro.
- Defina a *Distância máxima* permitida no cálculo onde colocar as quebras de página do html. Esta distância é a quantidade máxima de espaço vazio permitido na parte inferior de um quadro após o cálculo da localização ideal. A definição de um valor maior resultará em uma melhor escolha do local de quebra de página, mas mais espaço desperdiçado na parte inferior dos quadros. Isso só pode ser usado quando *Use quebra de página inteligente* estiver ativada.
- Ative **lcaixal** *Arquivo de estilos do usuário* para aplicar estilos de HTML, que muitas vezes é fornecido em folhas de estilos em cascata. Um exemplo de código de estilo será fornecido abaixo para definir a cor verdadeira para etiqueta do cabeçalho `<h1>` e definir a fonte e tamanho de texto incluído em marcas de parágrafo `<p>`.

```
h1 {color: #00ff00;
}
p {font-family: "Times New Roman", Times, serif;
font-size: 20px;
}
```

- Use o botão **[Atualizar HTML]** para ver o resultado das configurações de estilo do usuário.


19.4 Gerenciar ítems

19.4.1 Tamanho e posição

Cada item dentro do Compositor pode ser movido/redimensionado para criar um layout perfeito. Para ambas as operações o primeiro passo é ativar a ferramenta  Selecionar mover item e clicar sobre o item; então você pode movê-lo usando o mouse, mantendo o botão esquerdo. Se você precisa de restringir os movimentos com a horizontal ou o eixo vertical, apenas segure o `Shift` enquanto move o mouse. Se precisar de uma precisão melhor, você pode mover um item selecionado usando o Teclas de Setas no teclado; se o movimento é muito lento, você pode acelerar-lo por exploração `Shift`.

Um item selecionado irá mostrar quadrados em suas fronteiras; movendo um deles com o mouse, vai redimensionar o item na direção correspondente. Enquanto que o redimensionamento, segurando `:kbd: Shift` vai manter a proporção. Segurando `Alt` irá redimensionar a partir do centro de item.

A posição correta para um item pode ser obtida usando atração a grade ou guias inteligentes. Guias são definidas clicando e arrastando nas régua. Guia são movidas clicando na régua, nível com o guia e arrastando para um lugar novo. Para excluir uma guia de movê-lo para fora da tela. Se você precisa desativar a atração on the fly apenas segurar `Ctrl` enquanto move o mouse.

Você pode escolher vários itens com o botão  Selecionar / Mover item. Basta segurar o botão `Shift` e clique em todos os itens que você precisa. Você pode, então, redimensionar / mover este grupo como um único item.


Depois de ter encontrado a posição correta para um item, você pode bloqueá-lo usando os itens na barra de ferramentas ou marcando a caixa ao lado do item no guia *ítems*. Itens bloqueados ficarão **não** selecionáveis na tela.


Itens bloqueados podem ser desbloqueados, selecionando o item na guia *ítems* e desmarcar a caixa de seleção ou você pode usar os ícones na barra de ferramentas.

Para desmarcar um item, basta clicar sobre ele segurando o botão `Shift`.

Dentro do menu *Editar*, você pode encontrar ações para selecionar todos os itens, para limpar todas as seleções ou para inverter a seleção atual.

19.4.2 Alinhamento

A funcionalidade de levantar ou baixar os elementos que estão dentro do menu suspenso  Levantar item selecionado. Escolha um elemento da tela do Compositor de impressão e selecione a funcionalidade correspondente para levantar ou baixar o elemento selecionado em relação aos outros elementos (ver [table_composer_1](#)). Esta ordem é mostrada na guia *Ítems*. Você também pode levantar os objetos mais abaixo na guia *Ítems* clicando e arrastando o rótulo de um objeto na lista.

Existem várias funcionalidades de alinhamento disponíveis dentro do menu suspenso  Alinhar itens selecionados (ver [table_composer_1](#)). Para usar uma funcionalidade de alinhamento, você primeiro seleciona alguns elementos e, em seguida, clique no ícone de alinhamento correspondente. Todos os elementos selecionados serão então alinhados dentro de sua caixa delimitadora comum. Ao mover itens da tela Compositor, linhas auxiliares de alinhamento aparecem quando as fronteiras, centros ou cantos estiverem alinhados.

19.4.3 Copiar/Cortar e Colar ítems

O compositor de impressão inclui ações para utilizar a funcionalidade comum copiar/cortar/colar para os itens no layout. Como de costume, primeiro você precisa selecionar os itens usando uma das opções vistas acima; Neste ponto, as ações podem ser encontradas no menu *Editar*. Ao usar a ação Colar, os elementos serão colados de acordo com a posição atual do mouse.

Nota: Itens HTML não podem ser copiados para esta forma. Para contornar o problema, use o botão **[Adicionar moldura]** na guia *Propriedades do item*.

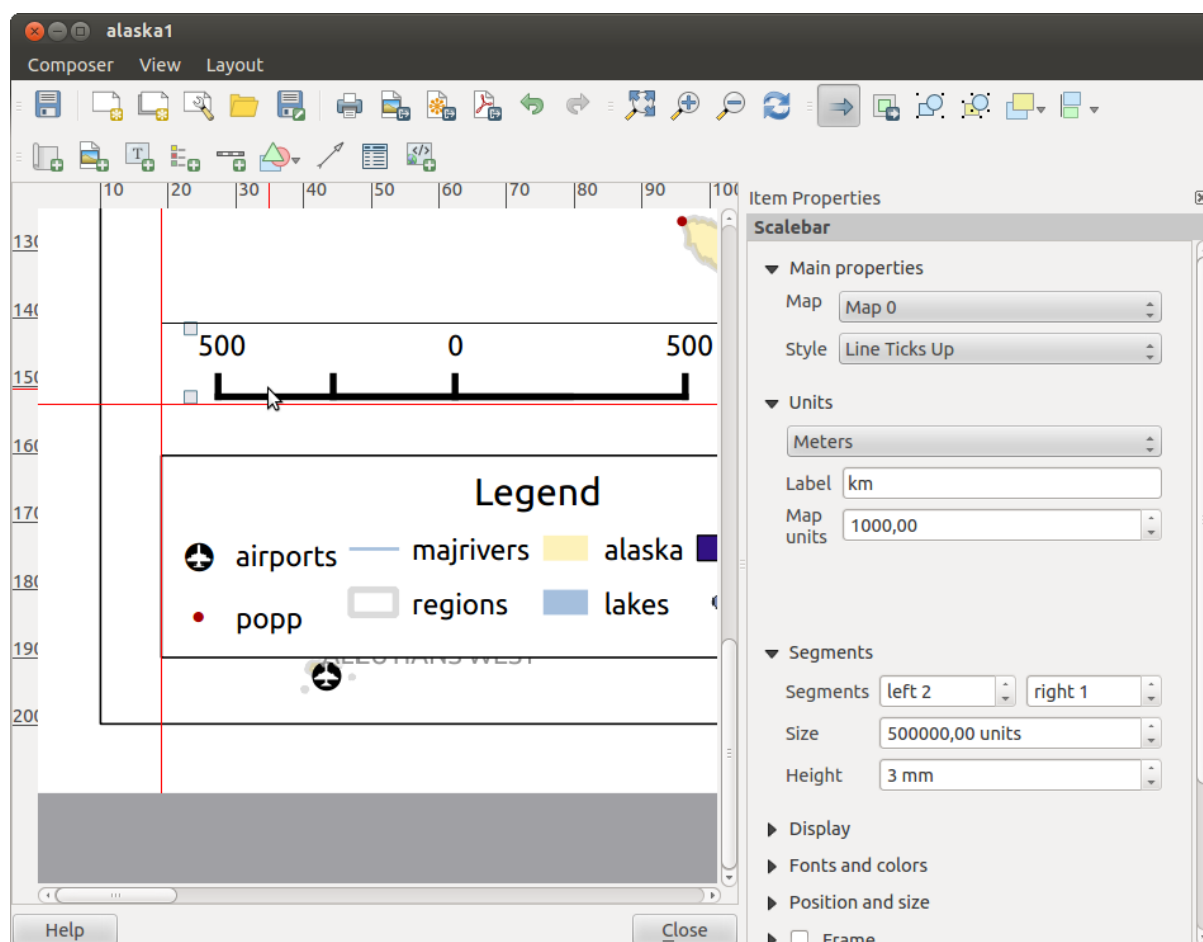





Figura 19.38: Linhas Guia de alinhamento no Compositor de Impressão 


19.5 Ferramentas de Reverter e Restaurar

Durante o processo de layout, é possível reverter e restaurar as alterações. Isto pode ser feito com a reversão e restaurar ferramentas:

-  Reverter as últimas alterações
-  Restaura as últimas alterações

Isso também pode ser feito por clique do mouse dentro da guia *Histórico de comandos* (veja [figure_composer_29](#)).



Figura 19.39: Histórico de comandos no Compositor de Impressão 

19.6 Geração de Atlas

O Compositor de impressão inclui funções de geração que permitem que você crie livros mapa de forma automatizada. O conceito consiste em utilizar uma camada de cobertura, que contém geometrias e campos. Para cada geometria na camada de cobertura, uma nova saída será gerada em que o conteúdo de alguns mapas da tela serão movidos para destacar a geometria atual. Os campos associados com esta geometria pode ser utilizado dentro de legendas de texto.

Cada página será gerada com cada feição. Para ativar a geração de um atlas e de acesso parâmetros de geração, consulte o guia *Geração de Atlas*. Esta guia contém os seguintes widgets (ver [Figure_composer_atlas](#)):

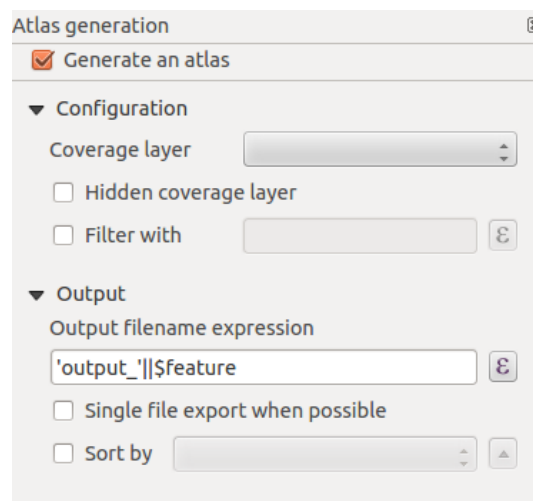




Figura 19.40: Separador de Geração de Atlas 

- *Gerar um atlas*, que habilita ou desabilita a geração atlas.
- A caixa de combinação *Camada de Cobertura*  que permite que você escolha a camada (vetor) que contém as geometrias para iterar.
- Um opcional *Camada cobertura Invisível* que, se marcada, irá esconder a camada de cobertura (mas não as outras) durante a geração.
- Uma opcional área de texto *Filtrar com* que permite que você especifique uma expressão para feições de filtragem da camada de cobertura. Se a expressão não está vazia, apenas as feições que avaliam para Verdadeiro será selecionada. O botão à direita permite que você exiba o construtor de expressões.
- A caixa de texto *Expressão do nome do arquivo de saída* que é usada para criar o nome do arquivo para cada geometria se necessária. É baseado em expressões. Este campo é significativo apenas para a renderização de múltiplos arquivos.
- A *Exportar arquivo único quando possível* que permite forçar a geração de um único arquivo, se isso for possível com o formato escolhido saída (PDF, por exemplo). Se este campo estiver marcado, o valor do campo *Expressão nome do arquivo de saída* não tem sentido.
- Um opcional *Ordenar por* que, se marcado, permite classificar as feições das camadas de cobertura. A caixa de combinação associada permite que você escolha qual coluna será usada como a chave de classificação. A ordem de classificação (crescente ou decrescente) é definida por um botão de dois estados que exibe uma seta para acima ou para baixo.

Você pode usar vários itens de mapa com a geração de atlas; cada mapa será processado de acordo com as feições de cobertura. Para ativar a geração de atlas para um item do mapa específico, você precisa verificar *Controlado pelo Atlas* sob as propriedades do item do mapa. Uma vez selecionada, você pode definir:

- Uma caixa de entrada *Margem em torno da feição* que permite que você selecione a quantidade de espaço adicional em torno de cada geometria dentro do mapa alocado. Seu valor é significativo somente quando utilizar o modo auto-escala.
- A *Escala fixa* que permite que você alterne entre auto-escala e modo de escala fixa. No modo de escala fixa, o mapa só será traduzido para cada geometria que será centralizada. No modo de auto-escala, as extensões do mapa são calculados de tal maneira que cada geometria aparece na sua totalidade.

19.6.1 Rótulos


A fim de adaptar rótulos para a feição do complemento atlas itera longo, você pode incluir expressões. Por exemplo, para uma camada da cidade com campos CITY_NAME e ZIPCODE, você pode inserir o seguinte:

```
The area of [% upper(CITY_NAME) || ', ' || ZIPCODE || ' is ' format_number($area/1000000,2) %] km2
```

A informação [% upper(CITY_NAME) || ', ' || ZIPCODE || ' é ' format_number(\$area/1000000,2) %] é uma expressão usada no interior do rótulo. Isso resultaria no atlas gerado como:


A área de PARIS,75001 é 1.94 km2

19.6.2 Botões substitutos de Dados Definidos


Há vários lugares onde você pode usar o botão  *Substituir dados definidos* para substituir a configuração selecionada. Estas opções são particularmente úteis para Geração de Atlas.

Para os exemplos a seguir a camada *Regions* do banco de dados de amostra do QGIS será utilizada e selecionada para Geração do Atlas. Assumimos também o formato de papel A4 (210x297) sendo selecionado na guia *Composição* para o campo *Predefinidos*.


Com um botão *Orientação* você pode ajustar dinamicamente a orientação do papel. Quando a altura (norte-sul) das extensões de uma região é maior do que sua largura (leste-oeste), você prefere querer usar *retrato* ao invés da orientação *paisagem* para otimizar o uso do papel.

Na *Composição* você pode definir o campo *Orientação* e selecionar *Paisagem* ou *Retrato*. Queremos definir a orientação dinamicamente usando uma expressão em função da geometria da região. pressione o botão  do campo *orientação*, selecione: *menuselection:Editar ...* para que o diálogo *Contrutor de expressão de texto* seja aberto. Dê seguinte expressão:


```
CASE WHEN bounds_width($atlasgeometry) > bounds_height($atlasgeometry) THEN 'Landscape' ELSE 'Portrait'
```

Agora, o papel orienta-se automaticamente em cada Região que você precisa de reposicionamento para localização do item do compositor. Para o item de mapa que você pode usar o botão  do campo *Largura* para defini-la de forma dinâmica usando seguinte expressão:

```
(CASE WHEN bounds_width($atlasgeometry) > bounds_height($atlasgeometry) THEN 297 ELSE 210 END) / 2
```

Use o botão  no campo *Altura* para apresentar a seguinte expressão:

```
(CASE WHEN bounds_width($atlasgeometry) > bounds_height($atlasgeometry) THEN 210 ELSE 297 END) / 2
```

Quando você quer dar um título acima mapa no centro da página, inserir um item do rótulo acima do mapa. Primeiro use as propriedades do item do item rótulo para definir o alinhamento horizontal para : *guilabel: Centro*. Depois ative a partir da caixa superior do meio *Ponto de Referência*. Você pode fornecer seguinte expressão para o campo *X*:

```
(CASE WHEN bounds_width($atlasgeometry) > bounds_height($atlasgeometry) THEN 297 ELSE 210 END) / 2
```

Para todos os outros itens do compositor você pode definir a posição de uma forma semelhante para que eles estejam corretamente posicionados quando a página é rodada automaticamente em modo retrato ou paisagem.

As informações fornecidas são derivadas do excelente blog (em Inglês e Português) sobre as opções de Orientação [Multiple_format_map_series_using_QGIS_2.6](#).

Este é apenas um exemplo de como você pode usar a Orientação.

19.6.3 Pré visualização

Uma vez que as configurações do atlas foram configuradas e itens do mapa selecionado, você pode criar uma pré-visualização de todas as páginas, clicando em *Atlas* → *Pré-visualização do Atlas* e usando as setas, no mesmo menu, para navegar por todas as feições.

19.6.4 Criação

A geração de atlas pode ser feita de diferentes maneiras. Por exemplo, com *Atlas* → *Imprimir Atlas*, você pode imprimi-lo diretamente. Você também pode criar um PDF usando *Atlas* → *Exportar Atlas como PDF*: O usuário será solicitado por um diretório para salvar todos os arquivos PDF gerados (exceto se a *Exportar arquivo único quando possível* foi selecionada). Se você precisar imprimir apenas uma página do atlas, basta iniciar a função de visualização, selecione a página que você precisa e clique em *Compositor* → *Imprimir* (ou criar um PDF).

19.7 Criando um arquivo de Saída

[Figure_composer_output](#) mostra o compositor de impressão com um exemplo de layout de impressão, incluindo cada tipo de item do mapa descrito nas seções anteriores.

O Compositor de impressão permite criar vários formatos de saída, e é possível definir a resolução (qualidade de impressão) e tamanho do papel:

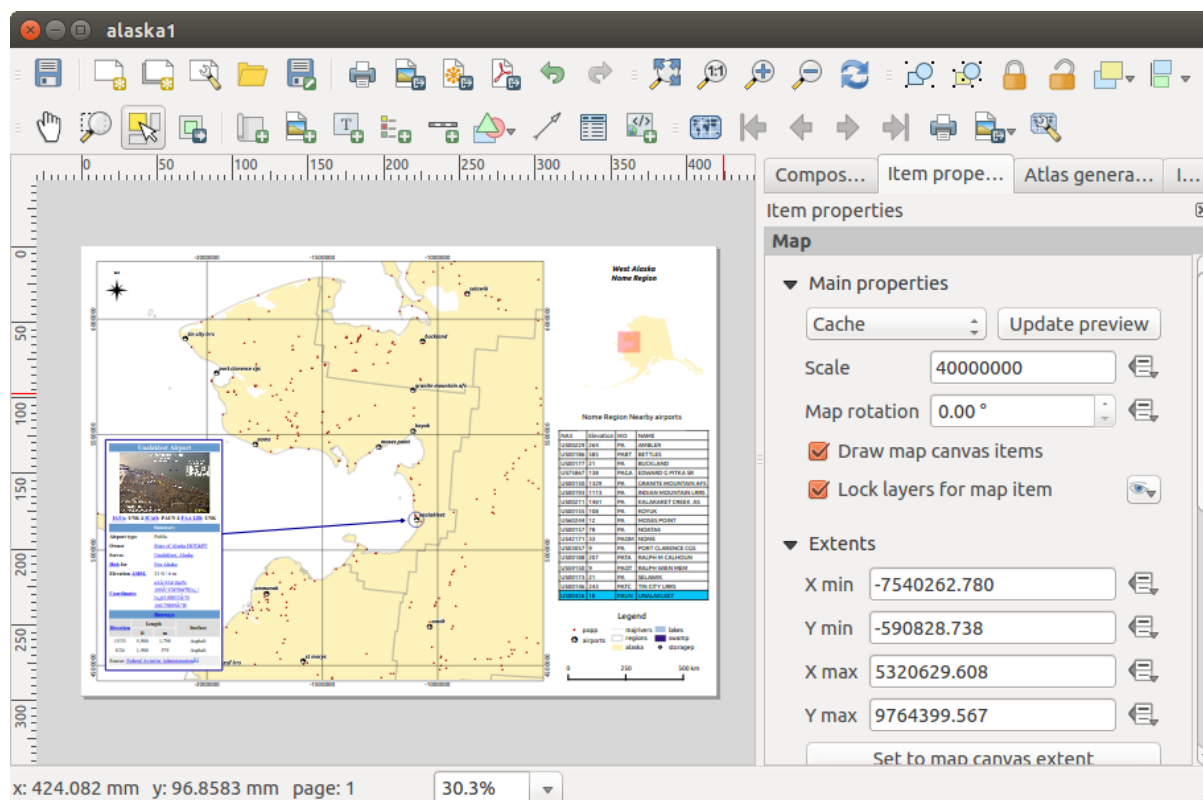






Figura 19.41: Compositor de Impressão com visualização do mapa, legenda, imagem, barra de escala, coordenadas, quadro de texto e HTML adicionados 🐧



- O ícone  **Imprimir** permite imprimir o layout para uma impressora conectada ou um arquivo PostScript, dependendo de drivers de impressora instalados.
- O ícone  **Exportar como imagem** exporta a tela do Compositor em diversos formatos de imagem, tal como PNG, BMP, TIF, JPG,...
- O ícone  **Exportar como PDF** salva a definida a tela do Compositor de impressão diretamente com um PDF.
- O ícone  **Exportar como SVG** salva a tela do Compositor de Impressão como um SVG (Scalable Vector Graphic).


Se você precisa exportar seu layout como uma **imagem georreferenciada** (ou seja, para carregar para dentro do QGIS), você precisa ativar esse recurso na guia Composição. Verifique **lcaixal**:guilabel:'Arquivo Mundial em' e escolha o mapa que pretende usar. Com esta opção, a ação 'Exportar como imagem' também irá criar um arquivo mundo.

Nota:

- Atualmente, a saída SVG é muito básica. Este não é um problema do QGIS, mas um problema com a biblioteca Qt subjacente. Esperamos que possa ser resolvido em versões futuras.
- Exportação rasters grandes às vezes pode falhar, mesmo que parece haver memória suficiente. Isto é também um problema com a administração Qt subjacente do rasters.

19.8 Gerir o Compositor

Com os ícones  Salvar como modelo e , você pode salvar o estado atual de uma sessão do Compositor de impressão como modelo `.qpt` e carregar o modelo novamente em outra sessão.

O botão  Gerenciador do Compositor na barra de ferramentas do QGIS e em *Compositor* → *Gerenciador do Compositor* permite que você adicione um novo modelo de compositor, criar uma nova composição baseada em um modelo salvo anteriormente ou para gerenciar os modelos já existentes.

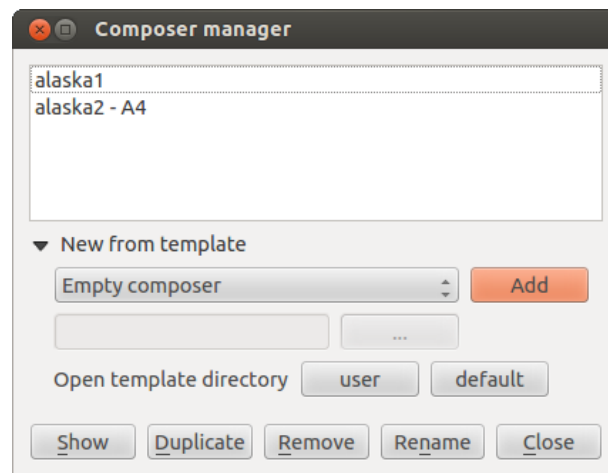






Figura 19.42: O gerenciador do Compositor de Impressão 

Por padrão, o gerenciador do compositor pesquisa por modelos de usuários em `~/.qgis2/composer_template`.

Os botões  Novo Compositor e  Duplicar Compositor na barra de ferramentas do QGIS e em *Compositor* → *Novo Compositor* e *Compositor* → *Duplicar Compositor* permitem que você abra uma nova caixa de diálogo do Compositor, ou duplicar uma composição existente de um criado anteriormente.

Finalmente, você pode salvar a sua composição de impressão com o botão :sup: *Salvar Projeto*. Esta é a mesma feição como na janela principal do QGIS. Todas as alterações serão salvas em um arquivo de projeto do QGIS.

Complementos

20.1 Complementos QGIS

QGIS foi projetado com uma arquitetura de complementos. Isto permite que muitos recursos e funções possam ser facilmente adicionados à aplicação. Muitos recursos no QGIS são atualmente implementados como complementos.

Voce pode utilizar os complementos no menu de complemento que pode abrir com *Complementos > Gerenciar e Instalar complementos*

Quando um complemento necessita ser atualizado, e se a configuração de complementos tenha sido criada em consequência. A interface principal do QGIS pode mostrar um link na barra de status para dizer-te que existe algumas aplicações de complementos disponíveis para serem atualizadas.

20.1.1 Diálogo de Complementos

Os menus do diálogo Complementos permitem ao usuário instalar, desinstalar e atualizar complementos em diferentes formas. Cada complemento tem alguns metadados visualizáveis no painel direito:

- informação se o complemento é experimental
- descrição
- voto(s) pontuação (pode votar em seu complemento preferido!)
- etiquetas
- alguns links úteis como página de inicio, rastreador e código do repositório.
- autor(es)
- versão disponível

Voce pode usar o filtro para pesquisar um complemento específico.



Todos

Aqui, todos os complementos disponíveis são apresentados, incluindo os complementos nativos e externos. Utilize **[Atualizar tudo]** para procurar por novas versões dos complementos. Além disso, você pode usar **[Instalar complemento]**, se um complemento estiver listado, mas não instalado, e **[Desinstalar complemento]** assim como **[Reinstalar complemento]**, se um complemento já estiver instalado. Se um complemento estiver instalado, ele pode ser ativado e desativado utilizando a caixa de marcação.



Instalados

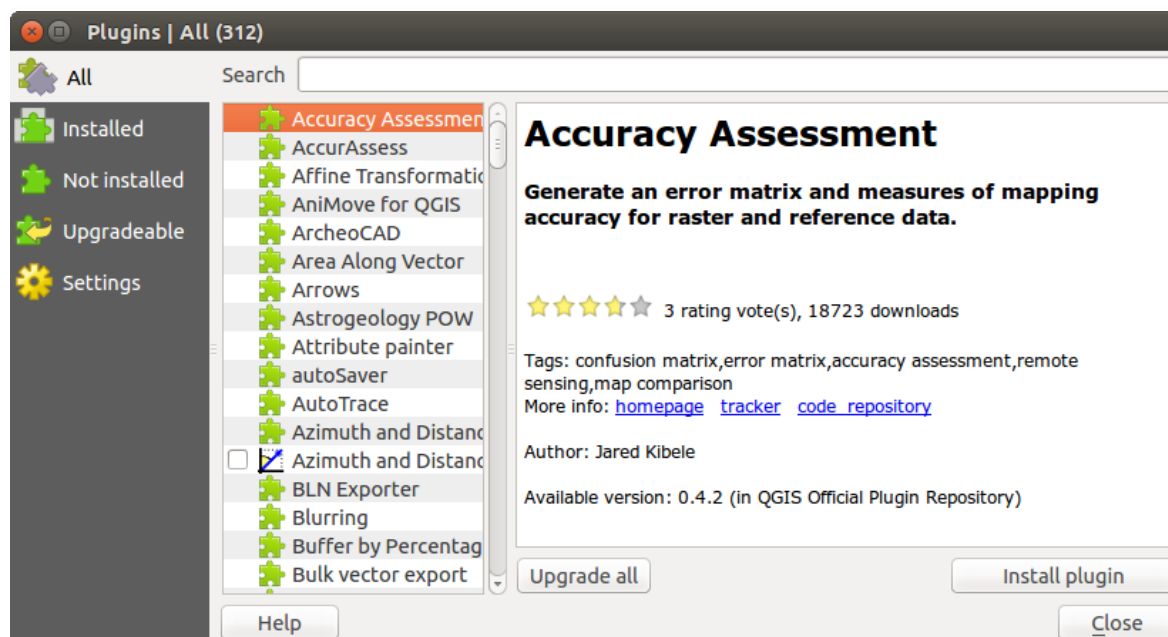





Figura 20.1: O  :guilabel: menu ‘Todos’ 

Neste menu, você pode encontrar somente os complementos instalados. Os complementos externos podem ser desinstalados e reinstalados utilizando os botões **[Desinstalar complemento]** e **[Reinstalar complemento]**. Você também pode **[Atualizar todos]**.

Não instalados

Este menu lista todos os complementos disponíveis que não estão instalados. Você pode usar o botão **[Instalar complemento]** para implementar um complemento no QGIS.

Atualizável


Se você ativar *Também mostrar complementos experimentais* em  *Opções*, você pode utilizar este menu para procurar por versões mais recentes dos complementos. Isto pode ser feito com os botões **[Atualizar complemento]** ou **[Atualizar todos]**.

Opções

Neste menu, você pode definir as seguintes opções:

- *Verificar por atualizações quando iniciar*. Sempre que um novo complemento ou uma atualização de algum estiver disponível, o QGIS informará a você ‘toda vez ao iniciar o QGIS’, ‘uma vez por dia’, ‘a cada 3 dias’, ‘a cada semana’, ‘a cada 2 semanas’ ou ‘a cada mês’.
- *Mostrar também os complementos experimentais*. O QGIS mostrará todos os complementos nos estágios iniciais de desenvolvimento, que geralmente são inadequados para uso em produção
- *Também mostrar complementos obsoletos*. Estes complementos são obsoletos e geralmente inadequados para uso em produção.

Para adicionar repositórios externos, clique em **[Adicionar...]** na seção de *Repositórios de complementos*. Se você não quer um ou mais repositórios adicionados, eles podem ser desabilitados através do botão **[Editar...]** ou **remover completamente com o botão **[Excluir]****.

A função *Buscar* está disponível em quase todos os menus (exceto em  *Opções*). Aqui, você pode procurar por complementos específicos

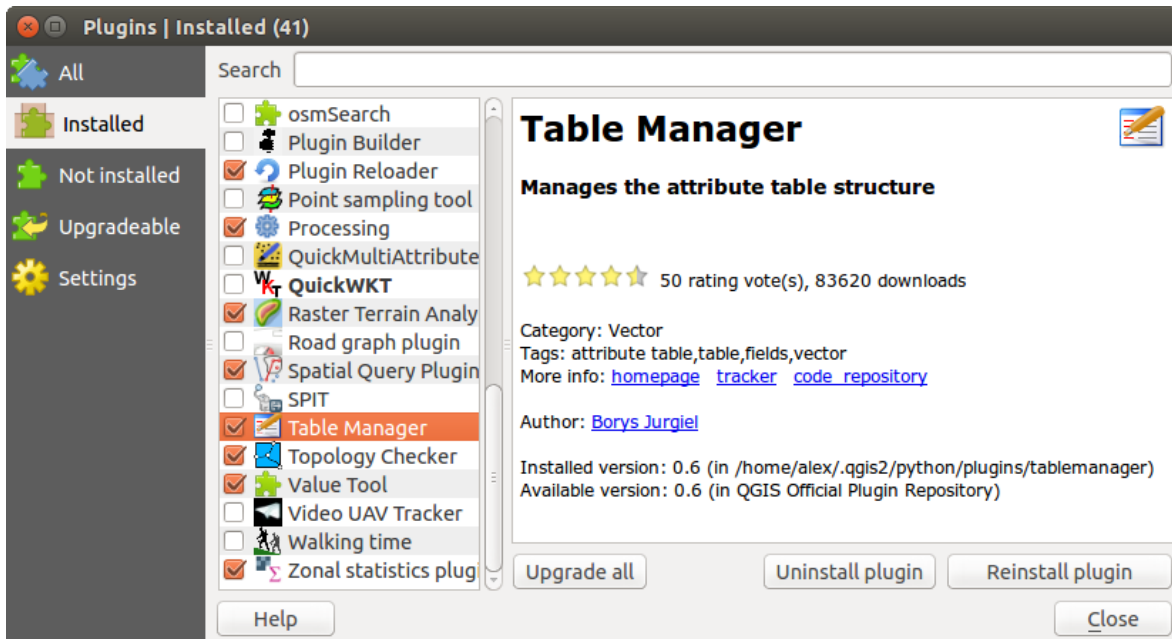






Figura 20.2: O  :guilabel: menu *Instalados* 



Figura 20.3: O  :guilabel: menu *Não instalados* 

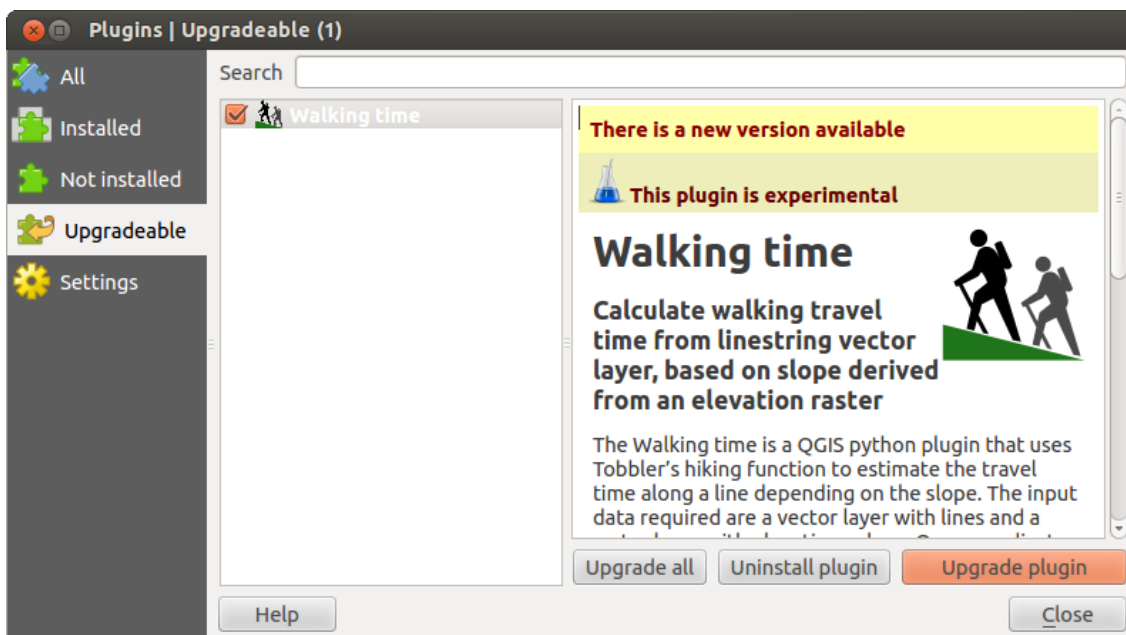




Figura 20.4: O  :guilabel: menu *Atualizável* 

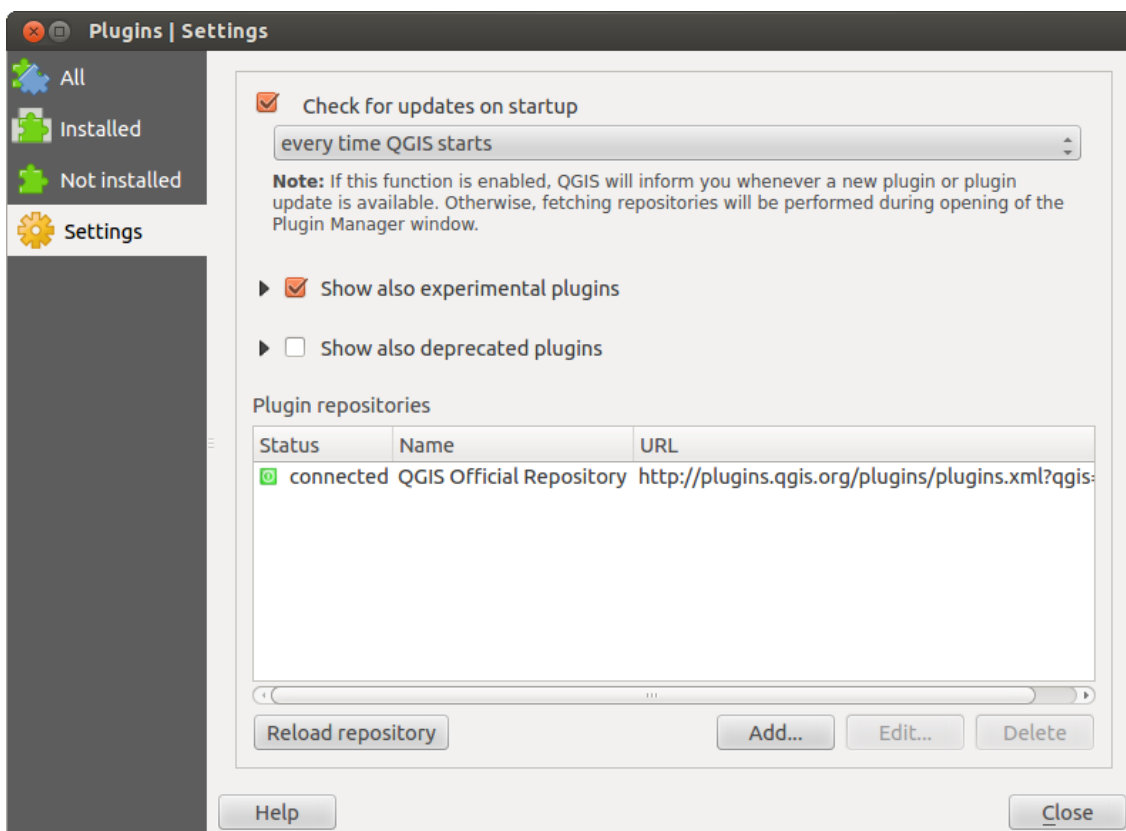




Figura 20.5: O  :guilabel: menu *Opções* 

Dica: Complementos nativos e externos

Complementos do QGIS são implementados como **Complementos nativos** ou **Complementos externos**. **Complementos nativos** são mantidos pelo grupo de desenvolvedores do QGIS e faz parte automaticamente da distribuição de cada versão do QGIS. Eles são desenvolvidos em uma das duas linguagens: C++ ou Python. **Complementos externos** são atualmente desenvolvidos em Python. Eles são armazenados em repositórios externos e mantidos por seus autores

Documentação detalhada sobre o uso, versão mínima do QGIS, home page, autores e outras informações importantes são fornecidas pelo Repositório 'Oficial' do QGIS em <http://plugins.qgis.org/plugins/>. Para repositórios externos, a documentação deve estar disponível com os autores dos mesmos. No geral, não está incluída neste guia.

20.2 Usando os Complementos Core QGIS

Ícone	Complemento	Descrição	Referência do Manual
	Captura de Coordenadas	Capture coordenadas com o mouse em diferentes SRC	<i>Complemento de Captura de Coordenadas</i>
	Gerenciador BD	Faça o gerenciamento das suas bases de dados dentro do QGIS	<i>Complemento Gerenciador BD</i>
	Conversor DXF2Shape	Converte do DXF para o formato de arquivo SHP	<i>Complemento dxf2shp</i>
	eVis	Ferramenta de Visualização de Eventos	<i>Complemento eVis</i>
	fTools	Um conjunto de ferramentas vetoriais	<i>Complemento fTools</i>
	Ferramentas GPS	Ferramentas para carregamento e importação de dados GPS	<i>Complemento GPS</i>
	GRASS	Funcionalidade GRASS	<i>Integração com SIG GRASS</i>
	Ferramentas GDAL	Funcionalidade matricial GDAL	<i>Complemento Ferramentas GDAL</i>
	Georeferenciador GDAL	Georeferenciar rasters com GDAL	<i>georeferenciar</i>
	Mapa de Densidade	Criar rasters heatmap para entrar pontos vetoriais	<i>Complemento Mapa de Densidade</i>
	Complemento de Interpolação	Interpolação baseada nos vértices da camada vetorial	<i>Complemento de Interpolação</i>
	Edição Offline	Edição Offline e sincronização com a base de dados	<i>Complemento Edição Offline</i>
	Oracle Spatial GeoRaster	Acesso ao Oracle Spatial GeoRasters	<i>Complemento GeoRaster Espacial Oracle</i>
	Gerenciador de complementos	Gerenciar complementos core e externos	<i>Diálogo de Complementos</i>
	Análise do Terreno Matricial	Computar características geomorfológicas a partir de MDE	<i>Complemento Análise do Terreno</i>
	Complemento caminho mais curto	Análise do Caminho mais curto	<i>Complemento Menor Distância</i>
	Complemento SQL Anywhere	Acesso a BD SQL anywhere	<i>Complemento SQL Anywhere</i>
	Interrogação Espacial	Interrogação espacial nos vetoriais	<i>Complemento de Consulta Espacial</i>
	SPIT	Ferramenta de importação Shapefile para PostgreSQL/PostGIS	<i>Complemento SPIT</i>
	Estatísticas Zonal	Calcular estatísticas matriciais a partir de polígonos vetoriais	<i>Complemento Estatística Zonal</i>
	MetaSearch	Interage com os serviços de catálogo de metadados (CSW)	<i>Catálogo do Cliente MetaBusca</i>

20.3 Complemento de Captura de Coordenadas

O complemento de captura de coordenadas é fácil de usar e oferece a possibilidade de visualizar as coordenadas na tela de mapa para dois sistemas de referência de coordenadas selecionados (SRC).

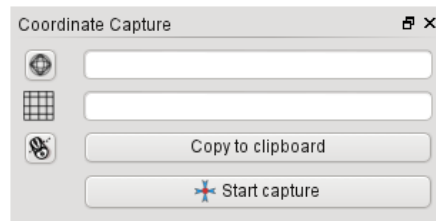









Figura 20.6: Complemento de Captura de Coordenadas 

1. Iniciar QGIS, selecionar  *Propriedades do Projeto* a partir do menu *Configurações* (KDE, Windows) ou *Arquivo* (Gnome, OSX) e clique no `:guilabel: Aba Projecção`. Alternativamente, você pode clicar no ícone  `Status do SRC` no canto inferior direito da barra de status.
2. Clique na caixa de verificação  *Ativar projeção "on the fly"* e selecione o sistema de coordenadas projetado à sua escolha (veja também *Trabalhando com Projeções*).
3. Ative o complemento captura de coordenadas no Gerenciador de Complementos (veja: ref: *managing_plugins*) e garanta-se que o diálogo seja visível, vá para *Exibir* → *Painéis* e assegure-se que a **lcaixal** `:guilabel: Captura de Coordenadas` esteja habilitada. A caixa de diálogo de captura de coordenar aparece como mostrado na Figura `figure_coordinate_capture_1`. Alternativamente, você também pode ir para `:menuselection: Vetor -> Captura de Coordenadas` e ver se a **lcaixal** *Captura de Coordenadas* esteja habilitada.
4. Clique no ícone  `Clique para selecionar o SRC para a exibição de coordenadas` e selecione um SRC diferente a partir do que selecionou acima.
5. Para iniciar a capturar coordenadas, clique em **[Iniciar captura]**. Pode clicar agora em qualquer sítio do enquadramento do mapa e o módulo irá mostrar as coordenadas em ambos os SRC selecionados.
6. Para habilitar coordenar de rastreamento do mouse, clique no  `:sup: ícone rastreamento do mouse`.
7. Pode também copiar as coordenadas selecionadas para a área de transferência.

20.4 Complemento Gerenciador BD

O complemento Gerenciador BD é uma parte oficial do núcleo do QGIS e pretende substituir o Complemento SPIT e adicionalmente integrar todas os outros formatos de base de dados suportados pelo QGIS numa única interface para o utilizador. O módulo  `Gerenciador BD` fornece várias características. Pode arrastar camadas do QGIS Browser para o Gerenciador BD e irá importar a sua camada para a sua base de dados espacial. Pode arrastar e largar tabelas entre bases de dados espaciais e elas serão importadas. Pode também usar o Gerenciador DB para executar interrogações SQL em relação a sua base de dados espacial e de seguida ver a saída espacial para as consultas adicionando os resultados ao QGIS como uma camada de consulta.

O menu *Base de Dados* permite ligar a uma base de dados existente, para começar a janela de SQL e sair do complemento Gerenciador BD. Quando estiver ligado a uma base de dados existente os menus *Esquema* e *Tabela* apareceram adicionalmente.

O menu *Esquema* inclui ferramentas para criar e apagar esquemas (vazias) e, se estiver topologia disponível (ex.: PostGIS 2), para iniciar o *TopoViewer*.

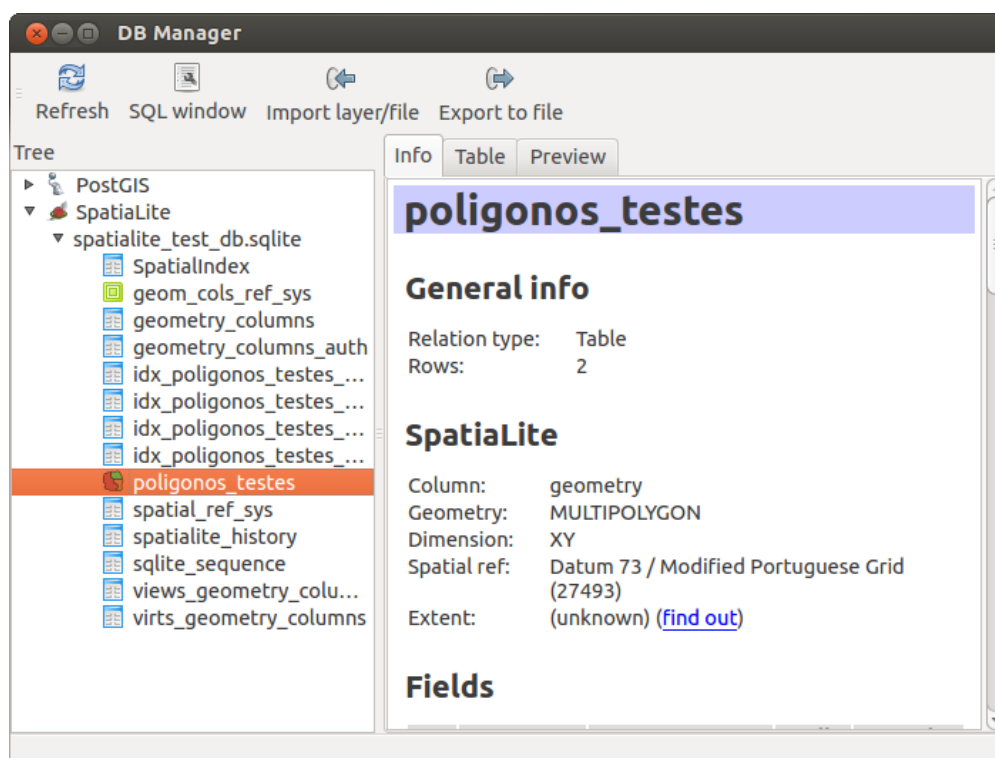



Figura 20.7: Janela do Gerenciador BD 

O menu *Tabela* permite criar e editar tabelas e apagar as tabelas e as vistas. É também possível esvaziar tabelas e mover tabelas de um esquema para outro. Como mais uma funcionalidade pode executar o comando VÁCUO e depois o comando ANÁLISE para cada tabela selecionada. O VÁCUO recupera simplesmente o espaço e torna-o disponível para re-utilizar e ANÁLISE de atualização de estatísticas para determinar o uma maneira mais eficiente de executar uma interrogação. Finalmente, pode importar camadas / arquivos, se eles forem carregados no QGIS ou existente no sistema de arquivos. E pode exportar as suas tabelas das base de dados para Shape com a característica de Exportação de Arquivo.

A janela *Árvore* lista todas as bases de dados existentes suportadas pelo QGIS. Com o duplo clique pode ligar à base de dado. Com o botão direito do rato pode renomear e apagar esquemas e tabelas existentes. As tabelas também podem ser adicionadas ao enquadramento do QGIS com o menu de contexto.

Se estiver ligado à base de dados, a janela **principal** do Gerenciador BD oferece três separadores. O separador *Informação* fornecem informação sobre a tabela e a sua geometria assim como os campos existentes, restrições e índices. Permite também correr a Análise de Vácuo e criar um índice espacial na tabela selecionada, se não estiver atualmente feita. O separador *Tabela* mostra todos os atributos e o separador *Pré-visualização* renderiza as geometrias como pré-visualização.

20.5 Complemento dxf2shp

O complemento conversor dxf2shape pode ser usado para converter dados vetoriais DXF para o formato shapefile. Ele requer os seguintes parâmetros a serem especificados antes de executar:

- **Entrada de arquivo DXF:** Digite o caminho para o arquivo DXF que será convertido.
- **Saída de arquivo Shp:** Digite o nome do arquivo shapefile que será criado.
- **Tipo de arquivo de saída:** Especifique o tipo de geometria do shapefile de saída. Os tipos suportados atualmente são linha, polígono, e ponto.

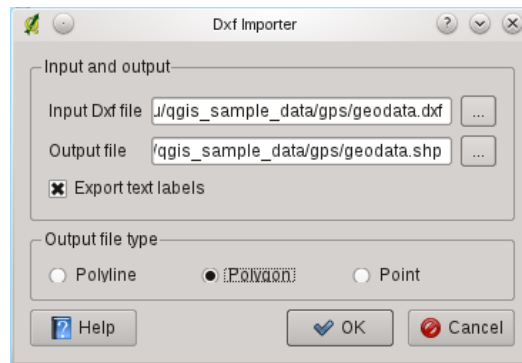



Figura 20.8: Complemento Conversor Dxf2Shape

- **Exportar rótulos de texto:** Se esta opção for ativada, uma camada adicional shapefile tipo ponto será criada, e a tabela DBF associada, irá conter informações sobre os campos “Texto” encontrados no arquivo DXF, e as próprias frases do texto.

20.5.1 Usando o Complemento

1. Inicie QGIS, carregue o complemento Dxf2Shape no Gerenciador de complementos (ver *Diálogo de Complementos*) e clique no ícone  Converter Dxf2Shape, surgirá na barra de menu QGIS. Aparecerá a janela do complemento Dxf2Shape, como mostrado na *Figura_dxf2shape_1*.
2. Digite o arquivo de entrada, um nome para o shapefile de saída e o tipo do shapefile.
3. Marque a caixa *Export text labels* se você deseja criar uma camada extra do tipo pontos com os rótulos.
4. Clique [OK].

20.6 Complemento eVis

(Esta seção é derivada do Horning, N., K. Koy, P. Ersts. 2009. eVis (v1.1.0) Guia do usuário. Museu Americano de História Natural, Centro para Biodiversidade e Conservação. Disponível em <http://biodiversityinformatics.amnh.org/>, e liberada sob a licença GNU FDL.)

O Mecanismo de Informática da Biodiversidade no Museu Americano de História Natural (MAHN) Centro para Biodiversidade e Conservação (CBC) desenvolveram a Ferramenta de Visualização de Eventos (eVis), outra programa de ferramenta para adicionar ao conjunto de monitoramento de conservação e ferramenta de apoio à decisão para orientar a área de proteção e planejamento da paisagem. Este plug-in permite aos usuários ligar facilmente fotografias georreferenciadas (ou seja, referenciado com latitude e longitude ou coordenadas X e Y), e outros documentos comprovativos, para dados vetoriais em OGIS.

eVis agora é automaticamente instalado e habilitado em novas versões do OGIS, e como com todos os complementos, pode ser ativado e desativado utilizando o Gerenciador de Documentos (ver: *Diálogo de Complementos*).

O Complemento eVis é composto de três módulos: a “ferramenta de Conexão de Banco de Dados”, “ferramenta de Identificação de Evento”, e o “Navegador de Evento”. Estes trabalham em conjunto para permitir a visualização de fotografias geocodificadas e outros documentos que estão ligados aos recursos armazenados em arquivos vetoriais, banco de dados ou planilhas

20.6.1 Navegador de Evento

O módulo Navegador de Evento tem a funcionalidade de exibir fotografias georreferenciadas que são ligadas às características do vetor exibidas na janela do mapa QGIS. Dados pontuais, por exemplo, podem vir do arquivo

vetorial que pode ser introduzido usando QGIS ou pode ser resultado da consulta ao banco de dados. O recurso vetorial deve ter informações atribuídas associadas com ele para descrever a localização e o nome do arquivo que contém a fotografia e, opcionalmente, a direção que estava apontada a bússola da câmera quando a imagem foi adquirida. Sua camada vetorial deve ser carregada no QGIS antes de executar o Navegador de Evento

Iniciando o módulo Navegador de Evento

Para iniciar o módulo Navegador de Eventos, clique em *Banco de Dados* → *eVis* → *eVis Navegador de Evento*. Isto abrirá o *Navegador genérico de Evento* janela.

A janela *Navegador de Evento* tem três guias exibidas no topo da janela. A aba *Exibir* é usado para ver a fotografia e é associado aos dados atribuídos. A aba *Opções* fornece um número de ajustes que pode ser ajustado para controlar o comportamento to plug-in eVis. Por fim a aba *Configurar Aplicações Externas* é usado para permitir que o eVis exiba imagens de outros documentos.

Entendendo a Janela de Exibição

Para ver a janela *Exibir*, clique na aba *Exibir* na janela *Navegado de Evento*. A janela *Exibir* é usado para ver fotografias georreferenciadas e suas informações atribuídas associadas.

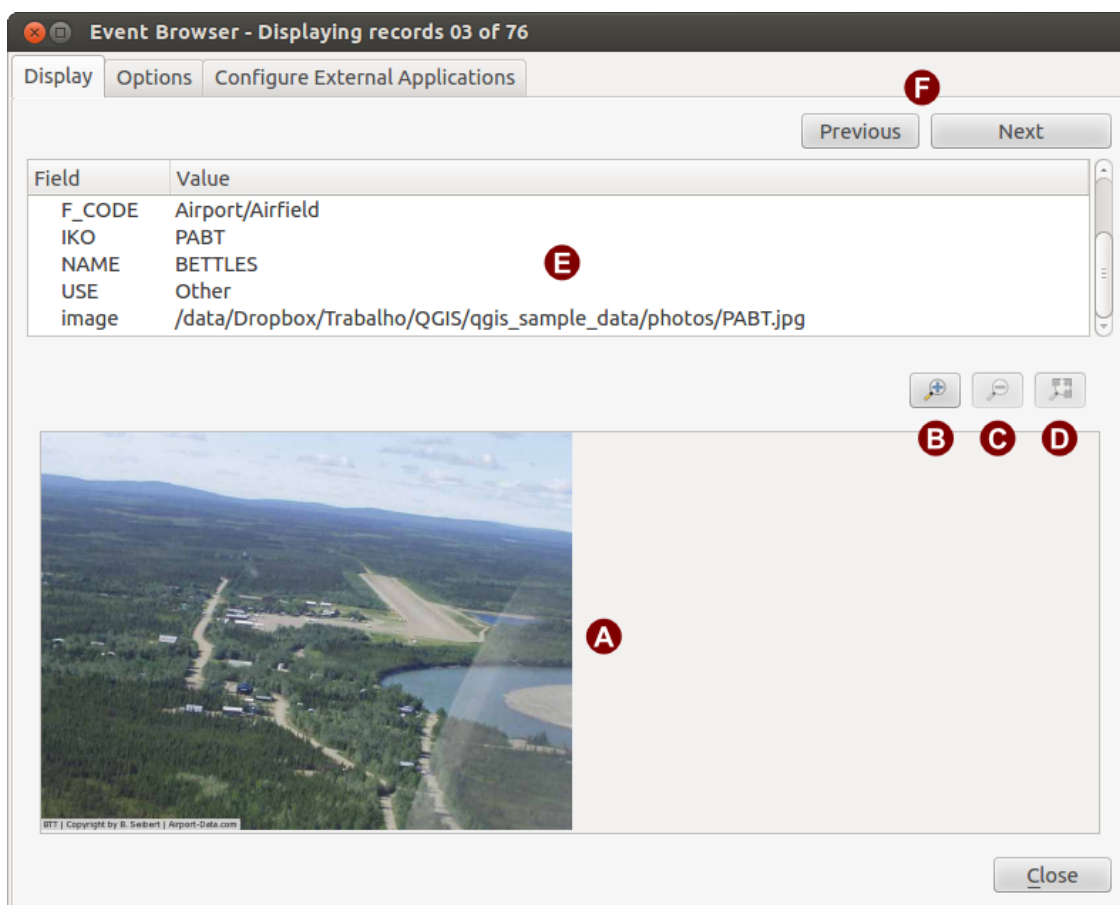


Figura 20.9: A janela de exibição do “eVis”

1. **Display window:** A window where the photograph will appear.
2. **Botão mais Zoom:** Dê Zoom para ver mais detalhes. Se a imagem inteira não pode ser exibida na janela de exibição, barras de rolagem aparecerão no lado esquerdo e inferior da janela para permitir que você se movimente pela imagem
3. **Botão menos Zoom:** Diminua o Zoom para ver mais área

4. Botão **Zoom para extensão total**: Mostra toda a extensão da fotografia.
5. **Janela de informações de atributo**: Toda as informações de atributo para o ponto associado com a fotografia que está sendo visualizada é apresentada aqui. Se o tipo do arquivo que está sendo referenciado no registro exibido não é uma imagem mas é um tipo de arquivo definido na aba *Configurar Aplicações Externas* , então quando você clica duas vezes sobre o campo de valor contendo o caminho para o arquivo, o aplicativo para abrir o arquivo será iniciado para ver ou ouvir o conteúdo do arquivo. Se as extensões do arquivo são reconhecidas, os dados do atributo serão exibidos em verde.
6. **Navigation buttons**: Use the Previous and Next buttons to load the previous or next feature when more than one feature is selected.

Entendendo a janela Opções

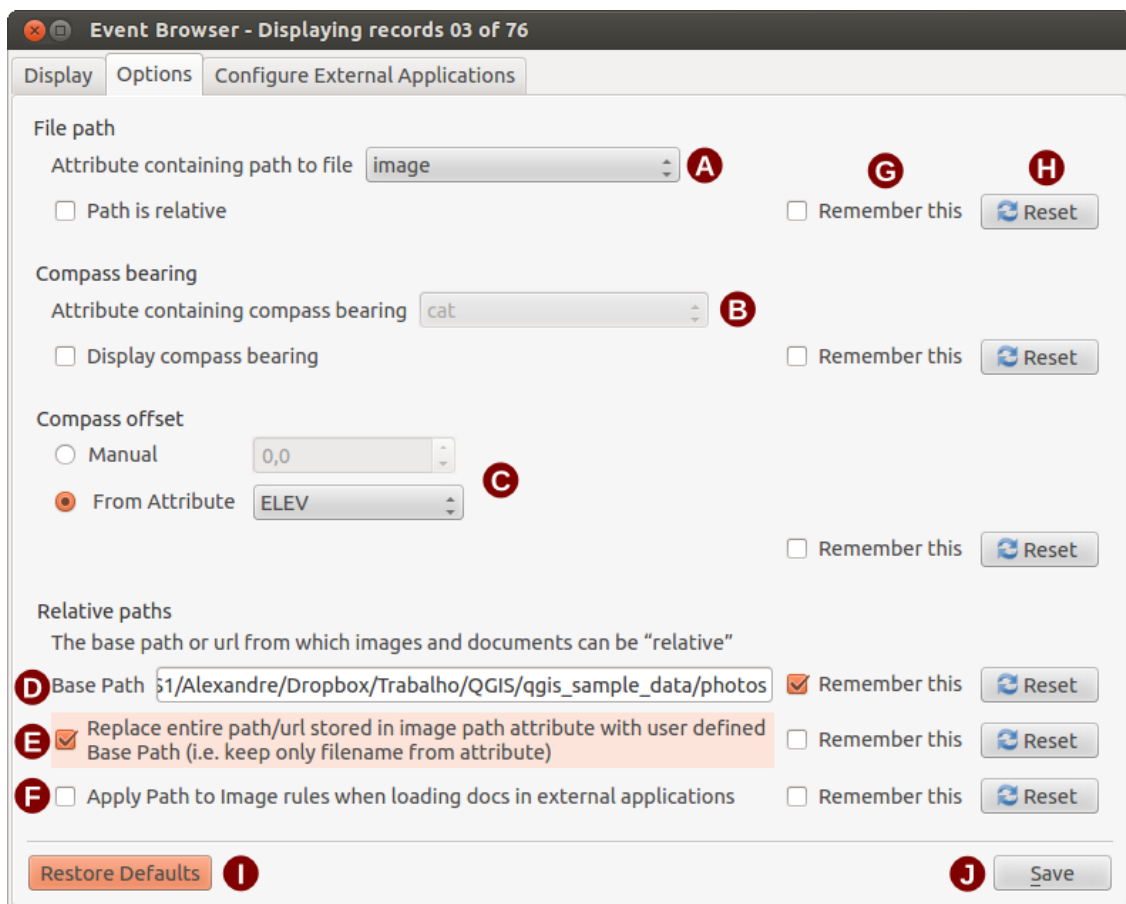


Figura 20.10: A janela de Opções do “eVis”

1. **Caminho de arquivo**: Uma lista suspensa para especificar o campo de atributo que contém o caminho do diretório ou URL para as fotografias ou outros documentos que estão sendo exibidos. Se a localização é um caminho relativo, então a caixa de seleção deve ser clicada. O caminho de base para um caminho relativo pode ser inserido na caixa de texto abaixo *Caminho Base* . Informação sobre diferentes opiniões para especificar a localização são anotadas na seção abaixo *Especificando o local e o nome de uma fotografia* .
2. **Bússola**: Uma lista suspensa para especificar o campo de atributo que contém a bússola associada com a fotografia que está sendo exibida. Se a informação da bússola estiver disponível, é necessário clicar na caixa de seleção abaixo do título do menu suspenso
3. **Bússola compensada**: Bússola compensada pode ser utilizado para compensar a declinação (para ajustar as direções coletadas usando direções magnéticas para a verdadeira direção norte). Clique no botão rádio *Manual* para inserir a compensação na caixa de texto ou clique no botão rádio *Atributo De* para

selecionar o campo de atributo contendo a compensação. Para ambas as opções, declinações leste devem ser inseridas usando valores positivos, e declinações oeste valores negativos.

4. **Caminho Diretório base:** O caminho de base para que o caminho relativo definido no [Figure_eVis_2](#) (A) será anexado.
5. **Substituir caminho:** Se esta caixa de seleção está selecionada, apenas o nome do arquivo a partir de A será anexado ao caminho de base.
6. **Aplicar regra a todos os documentos:** Se marcado, as mesmas regras de caminho que são definidas para as fotografias serão usadas para os documentos não-imagem, como filmes, documentos de texto e arquivos de som. Se não marcado as regras de caminho só se aplica a fotografias e os outros documentos irão ignorar o parâmetro do caminho base.
7. **Salve as configurações:** Se esta opção estiver marcada, os valores dos parâmetros associados serão salvos para a próxima sessão, quando a janela é fechada ou quando o botão **[Salvar]** abaixo é pressionado.
8. **Redefinir valores:** Redefine os valores nesta linha para a configuração padrão.
9. **Restaurar padrões** Isto redefinirá todos os campos para suas configurações originais. Tem o mesmo efeito que clicar no botão **[Redefinir]**
10. **Salvar:** Isto salvará as configurações sem fechar o painel *Opções*

Entendendo a janela Configurar Aplicações Externas

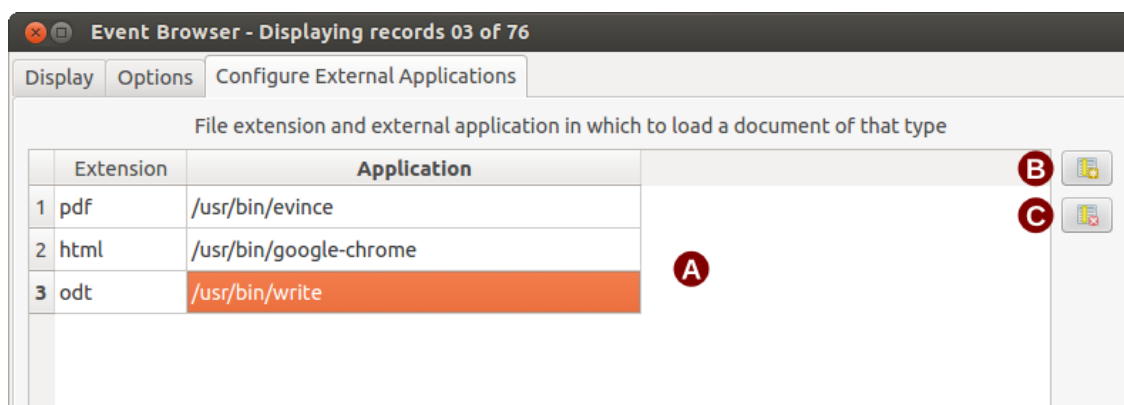


Figura 20.11: A janela de Aplicações Externas do “eVis”

1. ****Arquivo tabela de referência**:** Uma tabela contendo os tipos de arquivos que podem ser abertos usando eVis. Cada tipo de arquivo tem uma extensão de arquivo e o caminho para um aplicativo poder abrí-lo. Isso fornece a capacidade de abrir uma ampla gama de arquivos como filmes, gravações sonoras, e documentos de texto em vez de apenas imagens.
2. **Adicionar novo tipo de arquivo:** Adiciona um novo tipo de arquivo com uma extensão única e o caminho para o aplicativo poder abrí-lo.
3. **** Excluir linha atual **:** Exclui o tipo de arquivo em destaque na tabela e definido por uma extensão de arquivo e um caminho para um aplicativo associado.

20.6.2 Especificando o local e o nome de uma fotografia

O local e o nome da fotografia podem ser armazenados usando um caminho absoluto ou relativo, ou um URL se a fotografia está disponível em um servidor web. Exemplos das diferentes abordagens estão listadas na Tabela eVis_examples “_”.

X	Y	FILE	BEARING
780596	1784017	C:\Workshop\eVis_Data\groundphotos\DSC_0168.JPG	275
780596	1784017	/groundphotos/DSC_0169.JPG	80
780819	1784015	http://biodiversityinformatics.amnh.org/\ evis_testdata/DSC_0170.JPG	10
780596	1784017	pdf:http://www.testsite.com/attachments.php?\ attachment_id-12	76

20.6.3 Especificando o local e o nome de outros documentos suportados

Documentos comprovativos tais como documentos de texto, videos e clipes de som podem ser exibidos ou executados pelo eVis. Para fazer isso, é necessário adicionar uma entrada na tabela de referência do arquivo que pode ser acessada do :guilabel'Configurar Aplicação Externa' janela no :guilabel'Navegador de Evento Genérico' que corresponde a extensão da aplicação que pode ser utilizado para abrir o arquivo. É preciso também ter o caminho da URL ou do arquivo que contém a tabela de atributos para a camada vetorial. Uma regra adicional pode ser utilizada para URLs que não contém uma extensão de arquivo para o documento. O formato é `extension:URL`. A URL é precedida pela extensão do arquivo e dois pontos; isto é particularmente útil para acessar documentos de wikis e outros sites da web que usem banco de dados para administrar páginas da web (veja Tabela [evis_examples](#)).

20.6.4 Usando o Navegador de Evento

Quando a janela *Navegador de Evento* abre, uma fotografia aparecerá na janela de exibição se o documento referenciado no arquivo vetor tabela de atributos é uma imagem e se a informação do local do arquivo na janela :guilabel'Opções' está configurado corretamente. Se a fotografia é esperada e não aparece, será necessário ajustar os parâmetros na janela :guilabel'Opções'.

Se um documento de apoio (ou uma imagem que não tem uma extensão de arquivo reconhecida pelo eVis) está referenciada na tabela de atributos, o campo contendo a extensão do arquivo estará destacado em verde na janela de informação de atributo se esta extensão do arquivo é definido na tabela de referência do arquivo localizado na janela *Configurar Aplicativos Externos*. Para abrir o documento, dois cliques na linha destacada em verde na janela de atributo da informação. Se um documento de apoio é referenciado na janela de informação de atributo e o arquivo não está destacado em verde, então será necessário adicionar uma entrada para a extensão de arquivo do arquivo na janela *Configurar Aplicativos Externos*. Se o caminho do arquivo está destacado em verde mas não abre quando é clicado duas vezes, será necessário ajustar os parâmetros na janela *Opções* então o arquivo pode ser localizado pelo eVis


Se nenhuma bússola é fornecida pela janela *Opções*, um asterisco vermelho será exibido no topo do vetor característico que está associado com a fotografia exibida. Se uma bússola é fornecida, então uma seta aparecerá apontando na direção indicada pelo valor da bússola exibida no campo na janela *Navegador de Evento*. A seta será centrada sobre o ponto que é associada com a fotografia ou outro documento.

Para fechar o *Event Browser* window, clique no botão **[Fechar]** na *Exibir* janela.

20.6.5 Ferramenta ID de Evento

O módulo 'Identificador de Evento' permite que você exiba uma fotografia clicando numa característica exibida na janela do mapa QGIS. A característica do vetor deve ter informações de atributo associadas a ele para descrever a localização e o nome do arquivo contendo a fotografia e, opcionalmente, a direção que a bússola da câmera estava apontada quando a imagem foi adquirida. Esta camada deve ser carregada no QGIS antes de executar a ferramenta 'Identificador de Evento'

Iniciando o módulo Identificação de Evento

Para iniciar o módulo 'Evento ID', ou clique no ícone do : `sup:Event ID` or clique em :menuselection: 'Banco de Dados -> eVis -> Ferramenta de Eventos ID'. Isso fará com que o cursor se transforme em uma seta com um 'i' em cima dela o que significa que a ferramenta ID está ativa.


Para ver as fotografias ligadas às características do vetor na camada ativa do vetor exibida na janela do mapa QGIS, mova o cursor ID do Evento sobre o recurso e então clique o mouse. Depois de clicar no recurso, a janela *Navegador de Evento* é aberta e as fotografias em ou perto da localidade clicada estão disponíveis para visualização no navegador. Se mais de uma fotografia está disponível, você pode percorrer os diferentes recursos usando os botões [Anterior] e [Próximo]. Os outros controles estão descritos na seção *ref:evis_browser* deste guia

20.6.6 Conexão com o Banco de Dados


O módulo ‘Conexão de Banco de Dados’ fornece ferramentas para conectar e consultar o banco de dados ou outro recurso ODBC, como uma planilha

eVis pode realizar conexões diretas com os seguintes tipos de bancos de dados: PostgreSQL, MySQL e SQLite; ele também pode ler a partir de conexões ODBC (por ex., MS Access). Ao ler a partir de um banco de dados ODBC (como uma planilha do Excel), é necessário configurar o driver ODBC para o sistema operacional que você está usando.

Inicia o módulo Conexão com a Base de Dados

Para iniciar o módulo ‘Conexão de Banco de Dados’, clique no ícone apropriado  eVis Conexão de Banco de Dados ou clique em *Banco de Dados -> eVis -> Conexão de Banco de Dados*. Isto irá abrir a :guilabel: janela *Conexão de Banco de Dados*. A janela tem três abas: *Consultas Pré-definidas*, *Conexão de Banco de Dados* e *Consulta SQL*. A janela :guilabel: *Console de saída* na parte inferior da tela exibe o status de ações iniciadas pelas diferentes seções deste módulo.

Conectar a um banco de dados

Clique na aba: :guilabel: *Conexão de Banco de Dados* para abrir a interface de conexão com o banco. Em seguida, utilize a caixa de combinação *Tipo de Banco de Dados*  para selecionar o tipo de banco de dados que você deseja se conectar. Se for necessária uma senha ou nome de usuário, essa informação pode ser inserida nas caixas de texto :guilabel: ‘Nome de Usuário’ e *Senha*.

Digite o local hospedado do banco de dados na caixa de texto :guilabel: ‘Local do Banco de Dados’. Esta opção não estará disponível se você tiver selecionado ‘MS Access’ como o tipo de banco de dados. Se o banco de dados reside em seu desktop, você deve digitar “localhost”.

Entre o nome do banco de dados na caixa de texto *Nome Banco de Dados*. Se você selecionou ODBC como tipo de banco de dados, você precisa introduzir o nome da fonte de dados.

Quando todos os parâmetros são preenchidos, clique no botão **** [Connect] ****. Se a conexão for bem sucedida, uma mensagem será gravada na janela :guilabel: *Console de saída* informando que a conexão foi estabelecida. Se uma conexão não foi estabelecida, você precisará checar os parâmetros que foram inseridos anteriormente.

1. **Tipo de Base de dados:** Uma lista suspensa para especificar o tipo de base de dados que será usado.
2. **Banco de dados hospedeiro:** O nome do banco de dados hospedeiro.
3. **Porta:** O numero da porta se uma base de dados MySQL ou PostgreSQL for selecionada.
4. **Nome do banco de dados:** O nome do banco de dados.
5. **Conectar** Um botão para conectar-se a banco de dados usando os parâmetros definidos acima.
6. **Console de Saída** A janela do console onde as mensagens relacionadas ao processamento são exibidos.
7. **Nome de Usuário:** Nome de usuário para uso quando uma base de dados é protegida por senha.
8. **Senha:** Senha para uso quando um banco de dados é protegida por senha.
9. **Consultas pré-definidas:** Guia para abrir a janela “Consultas pré-definidas”.
10. ****Conexão com o banco de dados **:** Guia para abrir a janela “Conexão com o Banco de dados”.

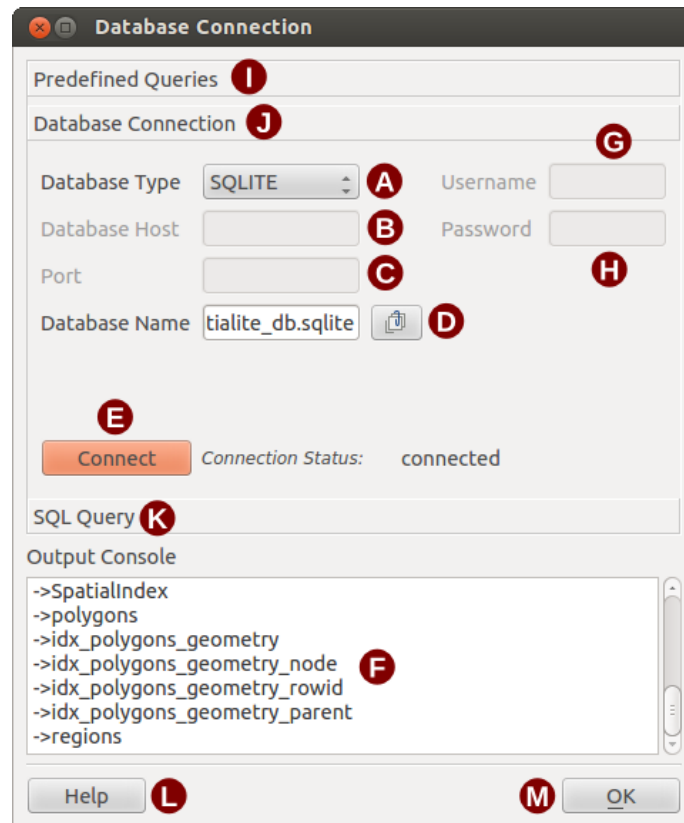


Figura 20.12: A janela de conexão do banco de dados do “eVis”

11. **Consulta SQL:** Guia para abrir a janela “Consulta SQL”.
12. **Ajuda:** Mostra a ajuda online.
13. **OK:** Fecha a janela principal do “Conexão com o Banco de Dados”.

Execução de consultas SQL

Consultas SQL são usadas para extrair informações de uma base de dados ou de recursos ODBC. Em eVis a saída dessas consultas é uma camada vetorial adicionado à janela de mapa do QGIS. Clique na guia *Consultas SQL* para exibir a interface de consulta SQL. Comandos SQL podem ser inseridos nesta janela de texto. Um tutorial útil em comandos SQL está disponível em <http://www.w3schools.com/sql>. Por exemplo, para extrair todos os dados de uma planilha em um arquivo do Excel, “ Selecione * de [Plan1 \$] “, onde “ Plan1 “ é o nome da planilha.

Clique no botão **** [Executar Consulta] **** para executar o comando. Se a consulta for bem sucedida, a janela *Seleção de Arquivo de Banco de Dados* será exibida. Se a consulta não for bem sucedida, uma mensagem de erro aparecerá na janela *Console de Saída*.

No *Seleção de arquivo do banco de dados* janela, insira o nome da camada que será criada a partir dos resultados da consulta na caixa de texto :*guilabel: 'Nome da Nova Camada* .

1. **Janela de Texto de Consulta SQL:** Uma tela para escrever consultas SQL.
2. **Executar consulta:** Botão que executa a consulta no *SQL Janela de Consulta*.
3. **Janela Console:** A janela console onde as mensagens relacionadas ao processamento são exibidas.
4. **Ajuda:** Mostra a ajuda online.
5. **OK:** Fecha o principal *Conexão banco de dados* janela.

Utilize as caixas de combinação :*guilabel: Coordenada X* e *Coordenada Y* para selecionar os campos do banco de dados que armazena as coordenadas X (ou longitude) e Y (ou latitude). Clicando no botão **** [OK]**

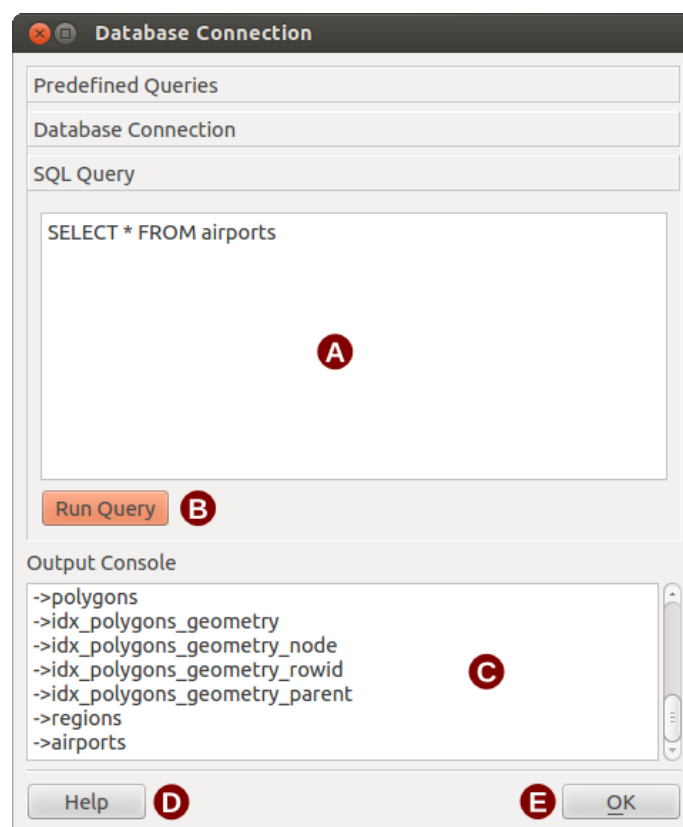


Figura 20.13: A tabela de consulta SQL do eVis

** faz com que a camada vetorial criada a partir da consulta SQL seja exibida na janela do QGIS.



Para salvar este arquivo vetor para uso futuro, você pode usar o comando QGIS ‘Salvar como...’ que é acessado clicando com o botão direito na camada nome na legenda QGIS do mapa e em seguida seccionando ‘Salvar como...’

Dica: Criando uma camada vetorial a partir de uma planilha do Microsoft Excel

Ao criar uma camada vetorial a partir de uma planilha do Microsoft Excel, você pode visualizar que zeros indesejados (“0”) foram inseridos nas linhas da tabela abaixo dos dados válidos. Isso pode ser causado deletando os valores dessas células no Excel usando o tecla Backspace. Para corrigir esse problema, você precisa abrir o arquivo do Excel (você vai precisar fechar o QGIS se você estiver conectado ao arquivo, para permitir que você edite o arquivo) e em seguida vá em :menuselection:’Editar -> Excluir’ para remover as linhas em branco do arquivo. Para evitar esse problema, você pode simplesmente apagar várias linhas na planilha do Excel acessando :menuselection:’Editar -> Excluir’ antes de salvar o arquivo.

Executando consultas pré-definidas

Com consultas predefinidas você pode selecionar consultas previamente escritas e armazenadas no formato XML em um arquivo. Isto é particularmente útil se você não estiver familiarizado com os comandos SQL. Clique na guia :guilabel:’Consultas Predefinidas’ para exibir a interface de consulta pré definida.

Para carregar um conjunto de consultas pré-definidas, clique no ícone  Abrir arquivo. Irá abrir a janela *Abrir Arquivo*, que é utilizada para localizar o arquivo que contém as consultas SQL. Quando as consultas são carregadas, seus títulos são definidos no arquivo XML que será exibido no menu suspenso localizado logo abaixo do ícone  Abrir Arquivo. A descrição completa das consultas são exibidas na janela de texto no menu suspenso.

Selecione a consulta que você deseja executar no menu suspenso e, então clique em aba *Consulta SQL* para ver se a consulta foi carregada na janela Consulta. Se for a primeira vez que você estiver executando uma consulta predefinida ou estiver migrando bancos de dados, você precisa ter certeza de estar conectado ao banco de dados.

Clique no botão **[Executar Consulta]** na aba *SQL Consulta* para executar o comando. Se a consulta é bem sucedida, a janela *Seleção do Banco de Dados do Arquivo* será exibida. Se a consulta não for bem sucedida, uma mensagem de erro aparecerá na janela *Console de Saída*

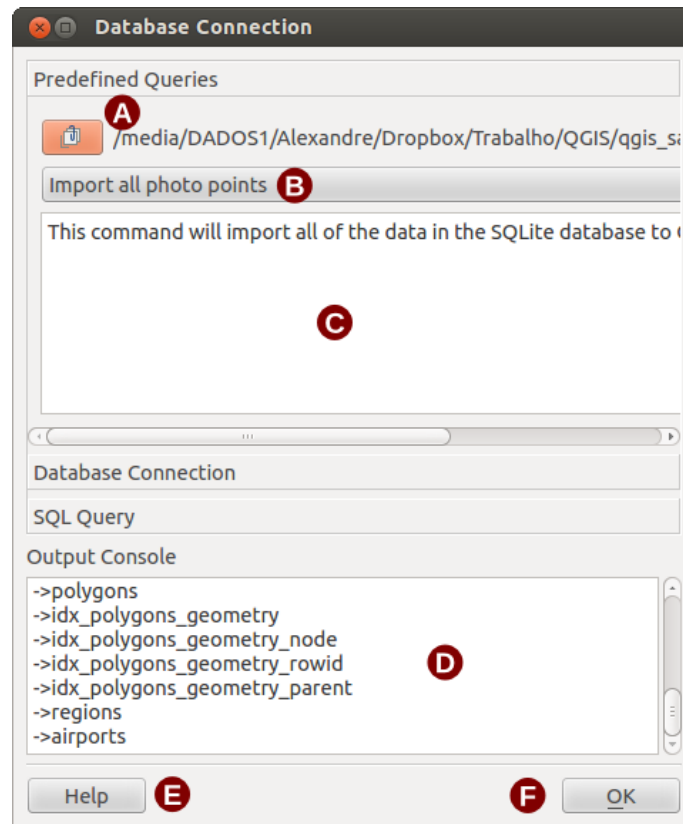


Figura 20.14: A guia de Consultas Pré-definidas do “eVis”

1. **Abrir Arquivo:** Inicia o navegador de arquivos “Abrir Arquivo” para procurar pelo arquivo XML mantendo as consultas pré-definidas.
2. **Consultas Pré-definidas:** Uma lista suspensa com todas as consultas definidas pelo arquivo XML de consultas pré-definidas
3. **Consulta descrição:** Uma breve descrição da consulta. Esta descrição é do arquivo consultas prédefinida XML.
4. **Janela Console:** A janela console onde as mensagens relacionadas ao processamento são exibidas.
5. **Ajuda:** Mostra a ajuda online.
6. **OK:** Fecha a janela principal do “Conexão com o Banco de Dados”.

Formato XML para consultas predefinidas eVis

As marcas XML lidas por eVis

Marcas	Descrição
Consulta	Define o início e o fim de uma instrução de consulta.
Descrição Breve	Uma pequena descrição da consulta que aparece no menu suspenso do eVis
Descrição	Uma descrição mais detalhada da consulta exibida na janela de texto Consulta Predefinida.
Tipo de Base de Dados	O tipo de banco de dados, definido no menu suspenso Tipo de Banco de Dados na guia Conexão Banco de Dados
Porta da Base de Dados	O porto tal como definido na caixa de texto Porto na aba Conexão Banco de Dados
Nome da Base de Dados	O nome do banco de dados como definido na caixa de texto Nome Banco de Dados na aba Conexão Banco de Dados
Nome de Usuário da Base de Dados	O nome de usuário do banco de dados como definido na caixa de texto na aba Conexão de Banco de Dados
Senha de Base de Dados	A senha do banco de dados como definida na caixa de texto Senha na aba Conexão Banco de dados
Instrução SQL autoconexão	O comando SQL. Uma bandeira (“verdadeira” ou “falsa”) para especificar as classificações acima deverá ser usada para conectar automaticamente o banco de dados sem executar a conexão de rotina na aba banco de Dados

Um exemplo de arquivo XML completo com três consultas é exibido abaixo:

```
<?xml version="1.0"?>
<doc>
  <query>
    <shortdescription>Import all photograph points</shortdescription>
    <description>This command will import all of the data in the SQLite database to QGIS
      </description>
    <databasetype>SQLITE</databasetype>
    <databasehost />
    <databaseport />
    <databasename>C:\textbackslash Workshop\textbackslash
eVis\_Data\textbackslash PhotoPoints.db</databasename>
    <databaseusername />
    <databasepassword />
    <sqlstatement>SELECT Attributes.*, Points.x, Points.y FROM Attributes LEFT JOIN
      Points ON Points.rec_id=Attributes.point_ID</sqlstatement>
    <autoconnect>>false</autoconnect>
  </query>
  <query>
    <shortdescription>Import photograph points "looking across Valley"</shortdescription>
    <description>This command will import only points that have photographs "looking across
      a valley" to QGIS</description>
    <databasetype>SQLITE</databasetype>
    <databasehost />
    <databaseport />
    <databasename>C:\Workshop\eVis_Data\PhotoPoints.db</databasename>
    <databaseusername />
    <databasepassword />
    <sqlstatement>SELECT Attributes.*, Points.x, Points.y FROM Attributes LEFT JOIN
      Points ON Points.rec_id=Attributes.point_ID where COMMENTS='Looking across
      valley' </sqlstatement>
    <autoconnect>>false</autoconnect>
  </query>
  <query>
    <shortdescription>Import photograph points that mention "limestone"</shortdescription>
    <description>This command will import only points that have photographs that mention
      "limestone" to QGIS</description>
    <databasetype>SQLITE</databasetype>
    <databasehost />
    <databaseport />
    <databasename>C:\Workshop\eVis_Data\PhotoPoints.db</databasename>
```

```

<databaseusername />
<databasepassword />
<sqlstatement>SELECT Attributes.*, Points.x, Points.y FROM Attributes LEFT JOIN
    Points ON Points.rec_id=Attributes.point_ID where COMMENTS like '%limestone%'
</sqlstatement>
<autoconnect>>false</autoconnect>
</query>
</doc>

```

20.7 Complemento fTools

O objetivo do módulo python fTools é fornecer um recurso único para várias tarefas SIG comuns aos dados vetoriais, sem a necessidade de software adicional, bibliotecas, ou trabalho complexo. Fornece um conjunto em crescimento de formas de gestão de dados espaciais e funções de análise que são rápidas e funcionais.

O fTools está automaticamente instalado e ativado nas novas versões do QGIS, juntamente com todos os módulos, e pode ser desativado e ativado através do Gestor de Módulos (Veja Seção *Diálogo de Complementos*). Quando ativado, o módulo fTools adiciona o menu *Vetor* ao QGIS, fornecendo funções, desde Ferramentas de Análise e Investigação a Ferramentas de Geometria e Geoprocessamento, assim como várias ferramentas úteis de Gestão de Dados.

20.7.1 Ferramentas de análise









Ícone	Ferramenta	Finalidade
	Matriz de Distância	Medir distâncias entre dois pontos de uma camada, os resultados saem como a) matriz de distâncias quadrada, b) matriz de distância linear, ou c) Síntese de distâncias. Pode limitar distâncias com as características mais próximas de k.
	Soma de comprimento de linhas	Calcular a soma total dos comprimentos das linhas para cada polígono de uma camada vetorial do tipo polígono.
	Pontos no polígono	Conta o número de pontos que existem em cada polígono de uma camada de entrada do tipo vetorial poligonal.
	Lista valores únicos	Origina uma lista de valores únicos num campo de uma camada vetorial.
	Estatísticas básicas	Calcular estatísticas básicas (média, std dev, N, soma, CV) em um campo de entrada.
	Análise de Vizinhança	Calcula as estatísticas de vizinho mais próximo avaliando o nível de agrupamento numa camada de pontos vetorial
	Coordenada(s) média(s)	Calcular o centro médio ponderado normal ou inteiro de uma camada vetorial, ou de várias feições com base em um campo exclusivo de identificação.
	Interseções de linhas	Localizar cruzamentos entre as linhas, e as feições de uma outra shapefile. Útil para localizar cruzamentos ou fluxo, ignora cruzamentos de linha com comprimento > 0.

Table Ftools 1: Ferramentas de análise fTools

20.7.2 Ferramentas de pesquisa

Ícone	Ferramenta	Finalidade
	Seleção aleatória	Seleciona aleatoriamente n números de elementos, ou n percentagem de elementos.
	Seleção aleatória dentro de subconjuntos	Selecionar aleatoriamente feições dentro de subconjuntos com base em um campo exclusivo de identificação.
	Pontos aleatórios	Gerar pseudo-pontos aleatórios sobre uma camada de entrada.
	Pontos regulares	Gerar uma grade regular de pontos ao longo de uma região determinada e exportá-los como um shapefile de pontos.
	Grade vetorial	Gera uma grelha de linhas ou polígonos baseada num espaçamento específico dado pelo utilizador.
	Selecionar pela localização	Selecionar feições com base na sua localização em relação a uma outra camada, para formar uma nova seleção, adicionar ou subtrair da seleção atual.
	Polígono a partir da extensão da camada	Criar uma camada com um único polígono retangular a partir da extensão de uma camada de entrada raster ou vetorial.

Tabela fTools 2: Ferramentas de Pesquisa fTools

20.7.3 Ferramentas de geoprocessamento

Ícone	Ferramenta	Finalidade
	Forma(s) convexa(s)	Criar o mínimo de forma(s) convexas em uma camada de entrada, ou com base em um campo de ID.
	Borda(s)	Criar borda(s) em torno de feições com base em distâncias, ou em um campo de distância.
	Interseção	Intersectar sobreposição de camadas de forma que a saída contém áreas onde ambas as camadas se cruzam.
	União	Sobreposição de camadas de forma que a saída contém a interseção e não-interseção de áreas.
	Diferença simétrica	Sobreposição de camadas de forma que a saída contém tanto as áreas que não se intersectam da entrada e das camadas de diferença.
	Cortar	Sobreposição de camadas de forma que a saída contém áreas que se cruzam a camada de corte.
	Diferença	Sobreposição de camadas de forma que a saída contém áreas que não cruzam a camada de corte.
	Dissolver	Mesclar feições com base no campo de entrada. Todas as feições com valores idênticos de entrada são combinadas para formar uma única feição.
	Elimine polígonos prateados	Mescla feições selecionadas com o polígono vizinho com a maior área ou maior fronteira comum.

Tabela fTools 3: Ferramentas de Geoprocessamento fTools

20.7.4 Ferramentas de Geometria

Ícone	Ferramenta	Finalidade
	Verificar validade da geometria	Verifica os polígonos para interseções, buracos fechados, e corrige ordenação de nós.
	Exportar/Adicionar geometrias de colunas	Adicionar Informações de geometrias vetoriais em camadas do tipo ponto (xCoord, yCoord), linha (comprimento), ou polígono (área, perímetro).
	Centróides de polígonos	Calcular os centróides reais para cada polígono em uma camada de entrada do tipo polígono.
	Triangulação de Delaunay	Calcula e produz uma saída baseada na triangulação de Delaunay (como polígonos) de uma camada de pontos vetorial.
	Polígonos de Voronoi	Calcula polígonos de Voronoi de uma camada de pontos vetoriais.
	Simplificar geometrias	Generalizar linhas ou polígonos com o algoritmo de Douglas-Peucker.
	Densificação de geometria	Adensa linhas ou polígonos através de adição de vértices.
	Densificar geometrias	Converter feições de várias partes para feições de partes individuais . Cria polígonos e linhas simples.
	Multipartes para partes simples	Mesclar várias feições para uma única feição multipartes baseada em um campo exclusivo de identificação.
	Polígonos para linhas	Converte polígonos para linhas, polígonos multipartes para várias linhas de partes únicas.
	Linhas para polígonos	Converte linhas para polígonos, linhas multipartes para polígonos de partes simples.
	Extração de nós	Extrair nós de camadas do tipo linhas e polígonos, de forma a resultar camadas de pontos.

Tabela fTools 4: ferramentas de Geometrias fTools

Nota: A ferramenta *Simplificar geometrias* pode ser usada para remover nós duplicados em geometrias de linhas e polígonos, faça este truque definindo o parâmetro *Tolerância de generalização* para 0.

20.7.5 Ferramentas de gerenciamento de dados

Ícone	Ferramenta	Finalidade
	Definir a projeção atual	Especificar o SRC para o arquivo shape cujo SRC não foi definido.
	Unir atributos por localização	Unir atributos adicionais a camada vetorial com base na relação espacial. Os atributos de uma camada vetorial são anexados à tabela de atributos de uma outra camada e exportados como um shapefile.
	Dividir camada vetorial	Dividir uma camada de entrada em várias outras camadas distintas com base em um campo de entrada.
	Juntar em um shapefile	Mesclar vários shapefiles dentro de uma pasta para um novo shapefile com base no tipo de camada (ponto, linha, área).
	Criar índices espaciais	Cria um índice espacial para os formatos OGR suportados.

Tabela fTools 5: ferramentas de gerenciamento de dados fTools

20.8 Complemento Ferramentas GDAL

20.8.1 O que são as Ferramentas GDAL?

O complemento Ferramentas GDAL oferece uma coleção GUI de ferramentas da Biblioteca de Abstração de Dados Geoespaciais, <http://gdal.osgeo.org>. Estas são ferramentas de gerenciamento raster para consultar, reprojetar, torcer e unir um conjunto variado de formatos raster. Inclui também ferramentas para criar camadas de contornos (vetor), ou relevos sombreados a partir de MDT matriciais, e para fazer um vrt (Virtual Raster Tile em formato XML) a partir de uma coleção de um ou mais arquivos raster. Estas ferramentas estão disponíveis quando o complemento está instalado e ativado.

Biblioteca GDAL

A biblioteca GDAL consiste num conjunto de programas da linha de comandos, cada um com uma lista cheia de opções. Os utilizadores que sabem usar a linha de comandos podem preferir a execução dos comandos no terminal, com acesso a todo o conjunto de opções. O complemento Ferramentas GDAL oferece uma interface fácil para as ferramentas, expondo apenas as opções mais populares.

20.8.2 Lista das ferramentas GDAL

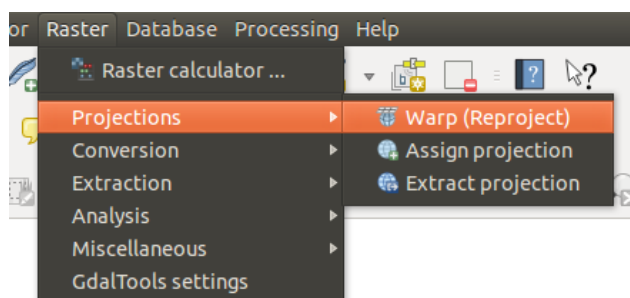










Figura 20.15: Lista do menu *Ferramentas GDAL*



Projeções

 <p><i>Torcer</i> (<i>Reprojetar</i>)</p>	<p>Este utilitário serve para os mosaicos de imagem, reprojeção e torções. O programa pode reprojetar para qualquer projeção suportada, e pode também ser aplicado a PC armazenados na imagem se a imagem tiver dados em “bruto” com controlo da informação. Para mais informação pode ler no sítio na internet do GDAL http://www.gdal.org/gdalwarp.html.</p>
 <p><i>Atribuir projeção</i></p>	<p>Esta ferramenta permite definir uma projeção para os rasters que estão georeferenciados mas não têm a informação da projeção. Com isto também ajuda na possibilidade de alterar a definição de uma projeção atual. Tanto o modo arquivo único como o modo batch são suportados. Para mais informações, visite a página do utilitário no sítio na internet do GDAL, http://www.gdal.org/gdalwarp.html.</p>
 <p><i>Extrair projeção</i></p>	<p>Este utilitário ajuda-o a extrair a informação da projeção de um arquivo de entrada. Se quiser extrair a projeção do diretório pode usar o modo Batch. Irá criar os arquivos <code>.prj</code> e <code>.wld</code>.</p>







Conversão

 <i>Digitalizar</i>	<p>Este programa torna geometrias vetoriais (pontos, linhas e polígonos) em banda(s) de uma imagem raster. Os vetores são lidos a partir de formatos OGR suportados. Repare que os dados do vetor devem ser do mesmo sistema de coordenadas que os dados rasters; a projeção “on-the-fly” não é fornecida. Para mais informação veja http://www.gdal.org/gdal_rasterize.html.</p>
 <i>Vectorizar</i>	<p>Este utilitário cria polígonos vetoriais para todas as regiões de pixels ligadas num matricial que partilha um valor comum de pixel. Cada polígono é criado com um atributo que indica o valor do pixel desse polígono. O utilitário irá criar um vetor de saída se não existir, por defeito em formato ESRI shapefile. Veja também http://www.gdal.org/gdal_polygonize.html.</p>
 <i>Traduzir</i>	<p>Este utilitário pode ser usado para converter matriciais em diferente formatos, potencialmente executa algumas operações como subconfigurações, amostragem, reescalonamento de pixels no processo. Para mais informações pode ler em http://www.gdal.org/gdal_translate.html</p>
 <i>RGB para PCT</i>	<p>Este utilitário irá computar uma tabela pseudo-cor otimizada para uma dada imagem RGB usando um algoritmo da mediana cortado num histograma RGB de resolução reduzida. De seguida converte a imagem numa imagem de pseudo-cor usando tabelas de cor. Esta conversão utiliza o pontilhado de Floyd-Steinberg (erro de difusão) para maximizar a qualidade visual da imagem de saída. Este utilitário é também descrito em http://www.gdal.org/rgb2pct.html</p>
 <i>PCT para RGB</i>	<p>Este utilitário irá converter uma banda de pseudo-cor num arquivo de entrada para um arquivo RGB de saída do formato desejado. Para mais informação veja http://www.gdal.org/pct2rgb.html</p>






Extração

 <i>Con-torno</i>	<p>Este programa gera um arquivo de contornos vetoriais a partir de um modelo digital do terreno (MDT) matricial. Pode encontrar mais informação em http://www.gdal.org/gdal_contour.html .</p>
 <i>Corta-dor</i>	<p>Este utilitário permite o corte (extração de um subconjunto) usando um enquadramento selecionado ou baseado no limite de um vetor. Mais informação pode ser encontrado em http://www.gdal.org/gdal_translate.html.</p>

Análise

 <i>Crivo</i>	<p>Este utilitário remove polígonos rasters mais pequenos que o tamanho de limiar (em pixels) fornecido e substitui-os com o valor do pixel mais alto do vizinho mais próximo. O resultado pode ser escrito na banda raster existente, ou copiado para um novo arquivo. Para mais informação veja http://www.gdal.org/gdal_sieve.html .</p>
 <i>Próximo ao Preto</i>	<p>Este utilitário irá digitalizar a imagem e tentar definir todos os pixels que existem perto do preto (ou perto do branco) à volta do limite para exatamente preto (ou branco). Isto é usado usualmente para “corrigir” perdas em fotos áreas comprimidas para que esses pixels de cor possam ser tratadas como transparentes nas operações de mosaico. Veja também http://www.gdal.org/nearblack.html.</p>
 <i>Preencher sem dados</i>	<p>Este utilitário preenche as regiões raster selecionadas (usualmente conhecidas com áreas sem valor) por interpolação de de pixels validados à volta das bordas da área. Pode encontrar mais informação em http://www.gdal.org/gdal_fillnodata.html .</p>
 <i>Proximidade</i>	<p>Este utilitário gera um mapa de proximidade raster indicando a distância desde o centro de cada pixel para o centro do pixel mais próximo identificado como pixel alvo. Os pixels alvo são os que estão presentes no raster inicial em que cada valor do pixel é definido como valores pixels alvo. Para mais informação veja http://www.gdal.org/gdal_proximity.html .</p>
 <i>Grade (Interpolação)</i>	<p>Este utilitário cria um grade regular (raster) a partir da leitura de dados dispersos de um fonte de dados OGR. Os dados de entrada serão interpolados para preencher os nós da grade com valores, pode escolher vários métodos de interpolação. O utilitário também é descrito no sítio na internet do GDAL http://www.gdal.org/gdal_grid.html .</p>
 <i>MDE (Modelos de Elevação)</i>	<p>Ferramentas para analisar e visualizar MDT. Podem ser criados, relevos sombreados, declives, exposições, relevo colorido, índice de rugosidade do terreno, índice de posição topográfica e mapas de irregularidades a partir de elevação raster GDAL suportadas. Para mais informação poderá ler em http://www.gdal.org/gdaldem.html</p>

Diversos

 <p><i>Construir Raster Virtual (Catálogo)</i></p>  <p><i>Juntar</i></p>	<p>Este programa constrói um VRT (Conjunto de Dados Virtual) que é um mosaico da lista dos conjuntos de dados do GDAL. Veja também http://www.gdal.org/gdalbuildvrt.html .</p> <p>Este utilitário irá criar mosaicos de imagem a partir de um conjunto. Todas as imagens devem ter o mesmo sistema de coordenadas e ter o mesmo número de bandas correspondentes, mas podem ser sobrepostas, e em diferentes resoluções. Nas áreas de sobreposição, a última imagem será copiada sobre as mais recentes. O utilitário é também descrito em http://www.gdal.org/gdal_merge.html .</p>
 <p><i>Informação</i></p>	<p>Este utilitário cria uma lista com várias informações sobre o conjunto de dados raster GDAL suportado. Pode encontrar mais informação em http://www.gdal.org/gdalinfo.html .</p>
 <p><i>Construir Reduções</i></p>	<p>O utilitário gdaladdo pode ser usado para construir ou reconstruir imagem de visualização para a maioria dos formatos suportados com um dos algoritmos de redução de escala. Para mais informação veja http://www.gdal.org/gdaladdo.html .</p>
 <p><i>Índice de Quadrículas</i></p>	<p>Este utilitário constrói um shapefile com o registo de cada arquivo raster de entrada, um atributo contendo um nome do arquivo, e a geometria do polígono do limite do raster. Veja também http://www.gdal.org/gdaltindex.html .</p>

configurações das Ferramentas GDAL

Use este diálogo para incorporar suas variáveis GDAL.

20.9 Complemento Georreferenciador

O complemento Georreferenciador é uma ferramenta para gerar world files para rasters. Ele possibilita que você possa referenciar rasters para sistemas de coordenadas projetadas ou geográficas através da criação de um novo GeoTiff ou adicionando um world file à imagem existente. A georreferenciação do raster passa por uma abordagem simples de localização de pontos no raster para que possa com precisão determinar as suas coordenadas.

Recursos

Ícone	Propósito	Ícone	Propósito
	Abrir Raster		Iniciar georreferenciamento
	Gerar scrip GDAL		Carregar pontos GCP
	Salvar pontos GCP como		Configurações de transformação
	Adicionar ponto		Excluir ponto
	Mover ponto GCP		Movimentar
	Aproximar		Afastar
	Ver a camada		Última visualização
	Próxima visualização		Ligar Georeferenciador ao QGIS
	Link QGIS ao Georeferenciador		Esticar totalmente o histograma
	Esticar localmente o histograma		

Tabela Georreferenciador 1: Ferramentas do georreferenciador.

20.9.1 Procedimento comum

Dois procedimentos alternativos podem ser usados, para adicionar as coordenadas X e Y (DMS (dd mm ss.ss), DD (dd.dd)) ou coordenadas projetadas (mmmm.mm) que correspondem aos pontos selecionados na imagem:

- O raster as vezes apresenta cruces, marcas fiduciais, com coordenadas “escritas” na imagem. Neste caso, você pode introduzir as coordenadas manualmente.
- Usando camadas já georreferenciadas. Elas podem conter informação vectorial ou raster que contenham os mesmos objetos/elementos que esteja na imagem que queira georreferenciar e com a projeção que você deseja para a sua imagem. Neste caso, você pode digitar as coordenadas, clicando sobre o conjunto de dados de referência carregados nos | qg | tela do mapa.

O procedimento habitual para o georreferenciamento de uma imagem consiste em selecionar múltiplos pontos no raster, especificar suas coordenadas e escolher o tipo de transformação mais apropriado para o arquivo. Baseado nos dados e parâmetros de entrada, o complemento irá computar os parâmetros do arquivo world ou então criar um novo GeoTIFF. Quanto mais pontos de controle (coordenadas) forem informados, melhor será o resultado do processo.

O primeiro passo é iniciar o QGIS, carregar o Módulo Georreferenciador (veja *Diálogo de Complementos*) e clique no ícone **Imselecionl:Raster -> Georreferenciador** que aparece no menu da barra de ferramentas do QGIS. A janela do módulo do Georreferenciador aparece como demonstra a *Figura_de_georeferenciador_1*.

Para este exemplo, estamos a usar uma carta militar do Sul de Dakota do SDGS. Pode ser visualizada mais tarde juntamente com a informação proveniente da localização do GRASS `spearfish60`. Você pode baixaarr a carta militar aqui: http://grass.osgeo.org/sampledata/spearfish_toposheet.tar.gz.

Entrando com pontos de controle GCPs (Ground Control Points)

1. Para iniciar o georreferenciamento de um raster sem georeferencia, necessitamos de carregá-lo usando o botão . O raster será mostrado na janela principal da área de trabalho. Uma vez carregado o raster, podemos começar a introduzir pontos de referência.
2. O botão Adicionar Ponto é usado para adicionar pontos na área de trabalho principal e introduzir as suas coordenadas (veja Figura *figura_do_georeferenciador_2*). Para este procedimento tem três opções:
 - Clique em determinado ponto da imagem raster e entre com suas coordenadas X e Y manualmente.

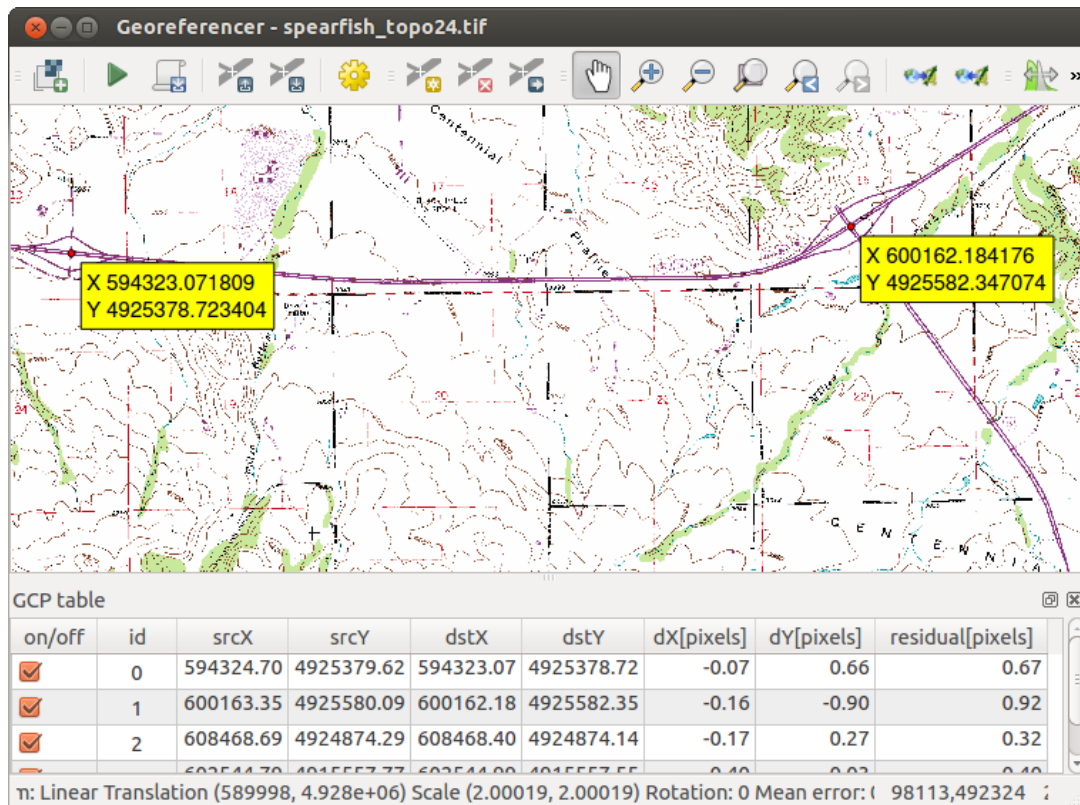


Figura 20.16: Janela do Módulo Georeferenciador

- Clique no ponto da imagem matricial e escolha o botão A partir do mapa na tela para adicionar as coordenadas X e Y com a ajuda do mapa georeferenciado que já se encontra carregado no enquadramento do mapa do QGIS.
 - Com o botão , pode mover os GCP em ambas as janelas, se estiverem no lugar errado.
3. Continue inserindo os pontos de controle. São necessários no mínimo 4 pontos ,espalhados pelas 4 pontas da imagem,, e quanto mais pontos ,coordenadas, forem adicionados, melhor será o resultado obtido. Existem ferramentas adicionais na caixa de diálogo do complemento para usar o zoom e pan na tela a fim de que seja possível localizar um conjunto de possíveis GCPs.

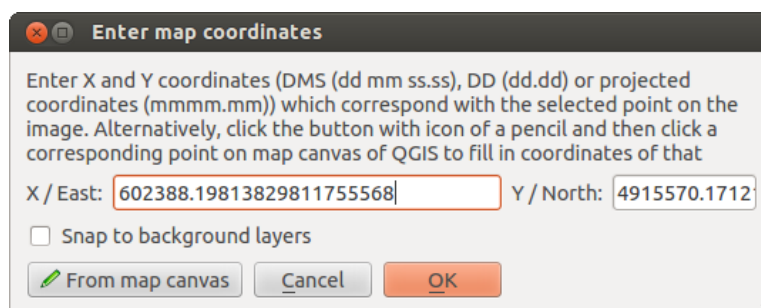


Figura 20.17: Adicionar pontos à imagem raster

Os pontos que adicionar ao mapa serão guardados num arquivo de texto separado ([filename].points) normalmente junto com a imagem raster. Isto permite que possamos reabrir o módulo do Georeferenciador mais tarde e adicionar novos pontos ou apagar existentes para otimizar o resultado. O arquivo de pontos contem valores na forma de: mapX, mapY, pixelX, pixelY. Pode usar o Carregar pontos GCP e o Guardar pontos GCP como para gerir os arquivos.

Definindo as configurações de transformação

Depois que os pontos GCP estão devidamente adicionados à imagem raster, é necessário definir as configurações de transformação para o processo de georeferenciamento.

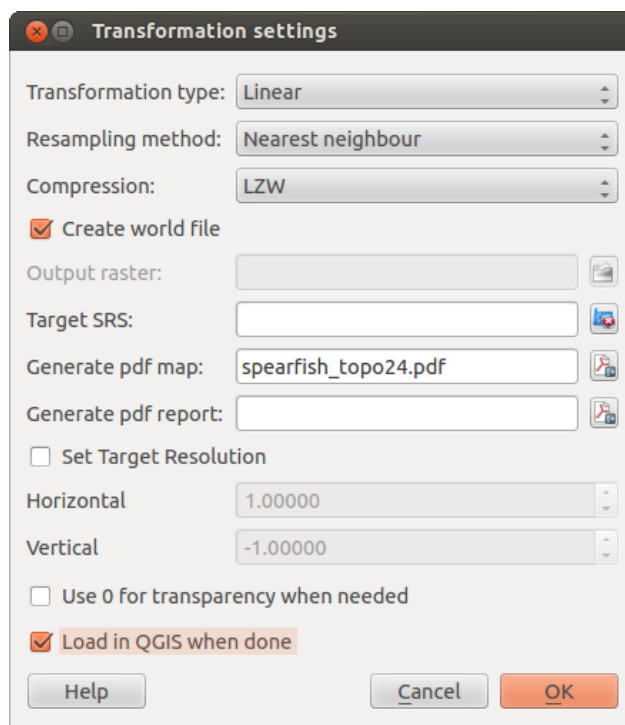


Figura 20.18: Definindo as configurações de transformação do georeferenciador 🐧

Algoritmos de transformação disponíveis

Dependendo da quantidade de pontos de controle capturados, será possível a utilização de diferentes algoritmos de transformação. A escolha de um desses algoritmos também depende do tipo e da qualidade dos dados de entrada inseridos e a quantidade de distorção geométrica que se está disposto a introduzir no resultado final.

Atualmente, os seguintes *Tipos de transformação* estão disponíveis:

- O algoritmo **Linear** é usado para criar o world-file, e é diferente dos outros algoritmos, e não transforma verdadeiramente o raster. Este algoritmo provavelmente não será suficiente se estiver trabalhando com material digitalizado.
- A transformação de **Helmert** executa um simples escalonamento e transformações de rotação.
- O algoritmo **Polinomial** 1-3 estão entre os algoritmos mais utilizados introduzidos para coincidir com a origem e o destino dos pontos de controle. O algoritmo polinomial mais amplamente utilizado é a transformação polinomial de segunda ordem, o que permite alguma curvatura. A transformação polinomial de primeira ordem (afim) preserva a colinearidade e permite apenas o escalonamento, translação e rotação.
- O algoritmo **Suavizador em Lâminas Finas (TPS)** é o método mais moderno de georeferenciamento, que permite introduzir deformações locais nos dados. Este algoritmo é útil quando originais de baixa qualidade estão a ser georeferenciados.
- A transformação **Projectiva** é uma rotação linear e de translação de coordenadas.

Definindo o método de reamostragem

O tipo de amostragem que escolhe irá depender dos seus dados de entrada e do objetivo do exercício. Se não quiser mudar as estatísticas da imagem, deverá escolher 'Vizinho mais próximo', uma vez que que a 'Amostragem cúbica' irá fornecer um resultado mais suavizado.

No QGIS, é possível escolher entre 5 diferentes métodos de reamostragem:

1. Vizinho mais próximo
2. Linear
3. Cúbico
4. Cúbico suavizado
5. Lanczos

Definindo as configurações de transformação

Existem várias opções que devem ser definidas para o arquivo raster (georreferenciado) de saída.

- A caixa de verificação *Criar world file* está apenas disponível se decidir usar o tipo de transformação linear, pois significa que a sua imagem raster atualmente não será transformada. Nesse caso, o campo *Raster de Saída* não está ativo, porque apenas será criado um novo world-file.
- Para outro tipo de transformação você necessita definir um *Raster de Saída*. Por padrão um novo arquivo ([filename]_modified) será criado na mesma pasta junto da imagem raster original.
- Como próximo passo, necessitamos definir *SRC* (Sistema de Referência Espacial) para o raster georreferenciado (veja seção *Trabalhando com Projeções*).
- Também é possível **gerar um mapa em pdf** e também um **Relatório pdf**. O relatório inclui informações sobre os parâmetros de transformação utilizados, além de uma imagem dos resíduos e uma lista com todos os GCPs e seus erros RMS.
- Além disso, é possível ativar a caixa de diálogo *Acertar a definição de saída* e definir a resolução dos pixels do raster de saída. Como padrão, as resoluções horizontal e vertical são iguais a 1.
- A caixa *Use 0 para transparência quando necessário* pode ser ativada, caso os pixels com valor 0 devam ser visualizados como transparentes.
- Finalmente, :guilabel:'Carregar no QGIS quando concluído' carrega o raster de saída automaticamente para o enquadramento do mapa do QGIS depois de ser feita a transformação.


Mostrando e adaptando as propriedades do raster

Clicando no menu *Propriedades do raster* dialog in the *Settings* abrirá a caixa de diálogo com as propriedades do raster que será georreferenciado.

Configurando o georreferenciador


- É possível definir a visualização ou não das coordenadas GCPs e/ou IDs.
- Podem ser definidas como unidades residuais pixels e unidades do mapa.
- Para o relatório PDF podem ser definidas as margens direita e esquerda e também o tamanho do papel para o mapa PDF.
- Por último, também pode-se ativar a caixa *Mostrar a janela do Georreferenciador ancorada*

Iniciando a transformação

Depois de recolher todos os GPCs e as configurações de transformação definidas, pressione o botão  Iniciar georreferenciamento para criar o novo raster georreferenciado.

20.10 Complemento de Interpolação

O complemento interpolação pode ser usado para gerar uma interpolação TIN ou IDW a partir de camada vetorial de pontos. É muito simples de manusear e oferece uma interface de usuário gráfica intuitiva para a criação de camadas raster interpoladas (ver [Figure_interpolation_1](#)). O complemento requer que os seguintes parâmetros sejam especificados antes de executar:

- **Entrada Camada vetorial:** Especifique a camada(s) de pontos vetoriais de entrada a partir da lista de camadas de ponto carregadas. Se várias camadas forem especificadas, os dados de todas as camadas serão usados na interpolação. Nota: É possível inserir linhas ou polígonos como restrições para a triangulação, especificando tanto “pontos”, “linhas de estrutura” ou “linhas de quebrar” na caixa combinada *Tip* .
- **Atributo de interpolação:** Selecione a coluna do atributo que será usado para interpolação ou habilite a caixa de seleção *Use Z-Coordenada*
- **Método de interpolação:** Selecione o método de interpolação. Ele pode ser ‘Triangulated Irregular Network (TIN)’ or ‘Inverse Distance Weighted (IDW)’.
- **Número de colunas/linhas:** Especifique o número de linhas e colunas para o arquivo raster de saída.
- **Arquivo de Saída:** Define o nome do arquivo raster de saída.
- *guiabel:* ‘Adicionar resultado ao projeto’ para carregar o resultado na tela do mapa.

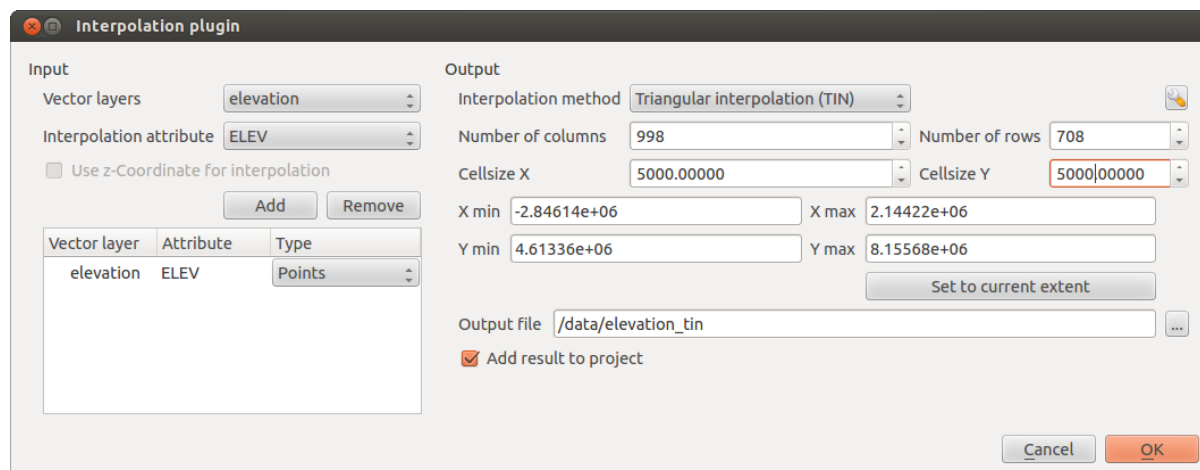




Figura 20.19: Complemento Interpolação 


20.10.1 Usando o complemento

1. Inicie QGIS e coloque uma camada vetorial de ponto (por exemplo, `elevp.csv`).
2. Carregue o complemento Interpolação no Gerenciador de complementos (ver [Diálogo de Complementos](#)) e clique em *Raster* → *Interpolação* →  *Interpolação*, que aparecerá na barra de menu QGIS. A janela do complemento interpolação aparecerá como mostrada na [Figure_interpolation_1](#).



3. Selecione uma camada de entrada (p. ex., *elevp* |selectstring|) e a coluna (p. ex., ‘ELEV’) para interpolação.
4. Selecione um método de interpolação (ex.: ‘Triangulated Irregular Network (TIN)’), e especifique um tamanho de célula de 5000 e um bom nome de arquivo raster de saída (ex.: *elevation_tin*).
5. Clique [OK].

20.11 Complemento Edição Offline


Para a coleta de dados, é comum a situação de trabalhar com um laptop ou um telefone celular offline no campo. Ao retornar para a rede, as mudanças precisam ser sincronizadas com a fonte de dados mestre (por exemplo, um banco de dados PostGIS). Se várias pessoas estão trabalhando simultaneamente no mesmo conjunto de dados, é difícil de fundir as edições concomitantes, mesmo que as pessoas não mudam as mesmas características.

O complemento  Edição Offline automatiza a sincronização pela cópia do conteúdo da fonte de dados (usualmente PostGIS ou WFS-T) para uma base de dados SpatiaLite e armazena as edições offline em tabelas dedicadas. Depois de estarem ligadas outra vez à rede, é possível aplicar as edições offline no conjunto de dados principal.

20.11.1 Usando o complemento

- Abra algumas camadas de vetoriais (por exemplo, de base de dados PostGIS ou WFS-T).
- Salve isto como um projeto.
- Vá para *Base de dados* → *Edição Offline* →  Converter para projeto offline e selecionar a camada para salvar. O conteúdo das camadas é salvo em tabelas SpatiaLite.
- Edite as camadas offline.
- Quando voce estiver conectado, poderá carregar os dados usando *Base de dados* → *Edição Offline* →  Sincronizar.

20.12 Complemento GeoRaster Espacial Oracle


Em base de dados Oracle, dados raster podem ser armazenados em objetos SDO_GEORASTER válidos com a extensão Espacial Oracle. No QGIS, o complemento  Oracle Spatial GeoRaster é suportado pelo GDAL e depende de bases de dados Oracle produto que está sendo instalado e funcionando em sua máquina. Enquanto a Oracle é um software proprietário, eles fornecem o software livre para fins de desenvolvimento e teste. Aqui apresentaremos um exemplo simples de como carregar imagens raster em GeoRaster:

```
$ gdal_translate -of georaster input_file.tif geor:scott/tiger@orcl
```

Isto irá carregar o raster para a tabela padrão GDAL_IMPORT table, como coluna designada de RASTER.

20.12.1 Gerenciando conexões

Em primeiro lugar, o Complemento Oracle GeoRaster deve ser ativado usando o Gerenciador de Complementos (veja :ref: *managing_plugins*). A primeira vez que você carregar um GeoRaster em QGIS, você deve criar uma

conexão com o banco de dados Oracle que contém os dados. Para fazer isso, comece clicando no botão  Adicionar Camada Oracle GeoRaster barra de ferramentas – isso vai abrir a janela *Selecione Oracle Spatial GeoRaster* de

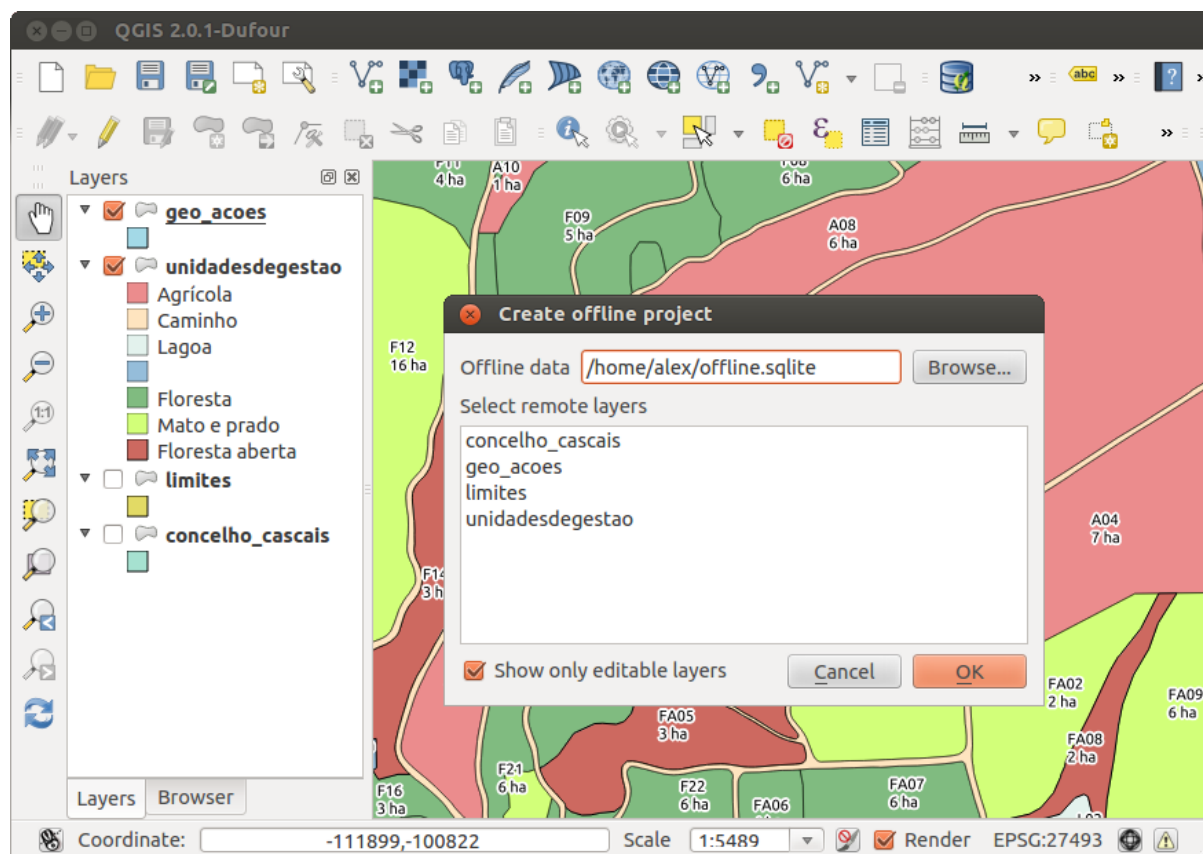


Figura 20.20: Criando de um projeto offline de camadas PostGIS ou WFS

diálogo. Clique em **** [Novo]**** para abrir a janela de diálogo, e especificar os parâmetros de conexão (Ver [Figure_oracle_raster_1](#)):

- **Nome:** Entre um nome para a conectar a base de dados.
- **Instância base de dados:** Entre o nome da base de dados que você irá conectar.
- **Nome do usuário:** Especifique o seu próprio nome de usuário que você usará para acessar o banco de dados.
- **Senha:** Fornecer a senha associada ao seu nome de usuário que é necessário para acessar o banco de dados.

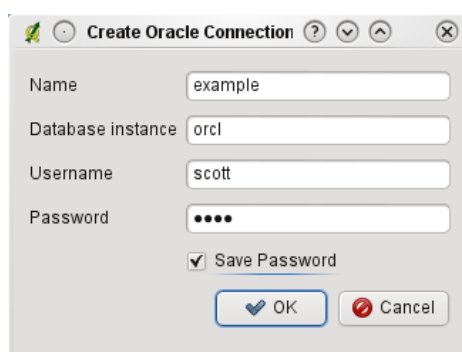


Figura 20.21: Janela de Criação de Conexão Oracle

Agora, volte à janela principal *Oracle Spatial GeoRaster* (veja [Figure_oracle_raster_2](#)), use a lista drop-down para escolher uma conexão, e use o botão **[Ligar]** para estabelecer a conexão. Pode também **[Editar]** a conexão, abrindo a janela anterior e efetuar alterações na informação da ligação, ou usar o botão **[Apagar]** para remover a conexão da lista drop-down.

20.12.2 Selecionando um GeoRaster

Uma vez que uma conexão foi estabelecida, a janela subconjuntos vai mostrar os nomes de todas as tabelas que contêm colunas GeoRaster nesse banco de dados no formato de um nome de subconjunto GDAL.

Clique num dos subconjuntos de dados listados em seguida clique em **[Selecionar]** para escolher o nome da tabela. Agora, outra lista de subconjunto de dados irá mostrar os nomes das colunas GeoRaster nessa tabela. Geralmente costuma ser uma lista pequena, uma vez que a maioria dos usuários não irá ter mais de uma ou duas colunas GeoRaster na mesma tabela.

Clique em algum da lista e subconjuntos e depois clique em **[Selecione]** para escolher uma combinação de tabela/coluna. A caixa de diálogo irá agora mostrar todas as linhas que contêm objetos GeoRaster. Note que a lista subconjuntos irá agora mostrar a Tabela de Dados Raster e pares Id Raster.

A qualquer momento, a entrada de seleção podem ser editadas, a fim de ir diretamente para um GeoRaster conhecido ou para voltar ao início e selecionar um outro nome da tabela.

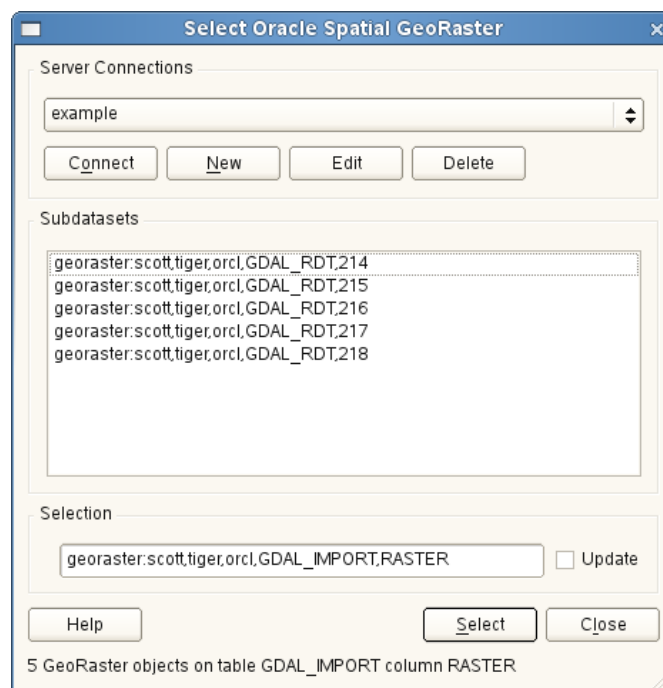


Figura 20.22: Janela de Seleção Oracle GeoRaster

A entrada dos dados de seleção também pode ser usado para inserir uma cláusula ONDE no final da sequência de identificação (ex.: `geor:scott/tiger@orcl,gdal_import,raster,geoid=`). Ver http://www.gdal.org/frmt_georaster.html para mais informações.

20.12.3 Exibindo o GeoRaster

Finalmente, selecionando um GeoRaster da lista de tabelas de dados Raster e Raster Ids, a imagem raster será carregada no QGIS.

A janela *Select Oracle Spatial GeoRaster* pode ser fechada agora e da próxima vez que se abrir, ela irá manter a mesma conexão e irá mostrar a mesma lista de subconjuntos anterior, tornando-o muito fácil de abrir uma outra imagem do mesmo contexto.

Nota: GeoRasters que contêm pirâmides permitirá a exibição muito mais rápida, mas as pirâmides precisam ser geradas fora do QGIS utilizando Oracle PL/SQL ou `gdaladdo`.

A seguir está um exemplo usando `gdaladdo`:

```
gdaladd georaster:scott/tiger@orcl,georaster\_table,georaster,georid=6 -r
nearest 2 4 6 8 16 32
```

Este é um exemplo usando PL/SQL:

```
$ sqlplus scott/tiger
SQL> DECLARE
  gr sdo_georaster;
BEGIN
  SELECT image INTO gr FROM cities WHERE id = 1 FOR UPDATE;
  sdo_geor.generatePyramid(gr, 'rLevel=5, resampling=NN');
  UPDATE cities SET image = gr WHERE id = 1;
  COMMIT;
END;
```

20.13 Complemento Análise do Terreno



O Complemento Análise do Terreno pode ser utilizado para calcular a declividade, aspecto, sombreamento, índice de rugosidade e relevo para modelos digitais de elevação (MDE). É muito simples de manusear e oferece uma interface de usuário gráfica intuitiva para a criação de novas camadas raster (ver [Figure_raster_terrain_1](#)).

Descrição das análises:

- **Declividade:** Calcula o ângulo de inclinação para cada célula em graus (com base na estimativa derivada de primeira ordem).
- **Aspecto:** Exposição (começando com o 0 para direção norte, em graus anti-horário).
- **Sombreamento:** Cria um mapa sombreado usando luz e sombra para proporcionar uma aparência mais tridimensional para um mapa de relevo sombreado.
- **Índice de rugosidade:** Uma medição quantitativa da heterogeneidade do terreno, tal como descrito por Riley et al. (1999). É calculado para cada local, resumindo a mudança de altitude dentro da grade de pixels 3x3.
- **Relevo:** Cria um mapa de relevo sombreado a partir de dados digitais de elevação. Implementado é um método para escolher as cores de elevação através da análise da distribuição de frequência.

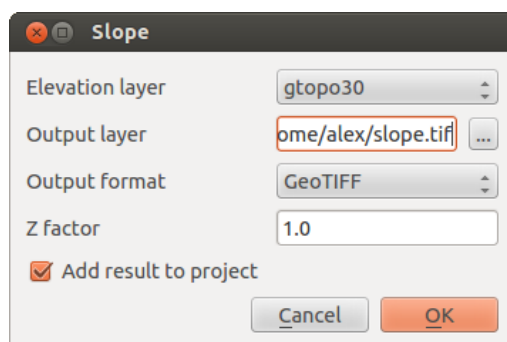


Figura 20.23: Complemento Modelagem Raster do Terreno (cálculo de declividade)

20.13.1 Usando o complemento


1. Inicie o QGIS e carregue a camada raster `gtopo30` da localização amostra do GRASS.
2. Carregue o complemento Análise Raster do Terreno no Gerenciador de Complementos (ver *Diálogo de Complementos*).

3. Selecione o método de análise a partir do menu (ex.: *Raster* → *Análise de Terreno* → *Declividade*). O diálogo *Declividade* irá aparecer como mostra a [Figure_raster_terrain_1](#).
4. Especifique um caminho de saída do arquivo, e o tipo de arquivo.
5. Clique [OK].

20.14 Complemento Mapa de Densidade

O complemento *Mapa de Densidade* usa a Estimativa de Densidade Kernel para criar um raster de densidade (mapa de temperatura) a partir de uma camada de pontos vetorial. A densidade calculada é baseada no número de pontos numa localização, com um conjunto largo de pontos agrupados resultando em valores altos. Os mapas de temperatura permitem facilmente identificar “pontos quentes” e agrupar pontos.

20.14.1 Ativar o complemento Mapa de Densidade


Primeiro esse complemento nativo precisa ser ativado usando o Gerenciador de Complementos (veja: ref: *managing_plugins*). Após a ativação, o ícone heatmap  pode ser encontrado na barra de ferramentas Raster, no menu *Raster* → *Mapa de calor*.

Selecione o menu *Exibir* → *Barra de Ferramentas* → *Raster* para exibir a Barra de Ferramentas Raster se não estiver visível.

20.14.2 Usando o complemento Mapa de Densidade

Ao clicar no botão da ferramenta **heatmap** *Mapas de Densidade* abre a janela do complemento Mapas de Densidade (veja [figure_heatmap_2](#)).

A janela tem as seguintes opções:

- **Camada de entrada de pontos:** Apresenta todas as camadas de pontos vetoriais no projeto atual e é usado para selecionar a camada que vai ser analisada.
- **Arquivo raster de saída:** Usa o botão  para selecionar a pasta e o nome do arquivo para o raster de saída que o complemento de mapas de densidade vai gerar. A extensão do arquivo não é requerida.
- **Formato de Saída:** seleciona o formato de saída. Embora todos os formatos suportados pelo o GDAL poderem ser escolhidos, na maioria dos casos o GeoTIFF é o melhor formato de escolha.
- **Raio:** usado para especificar o raio de pesquisa (ou largura do kernel) do mapa de densidade em metros ou em unidades de mapa. O raio especifica a distância em torno de um ponto no qual se fará sentir a influência do ponto. Os valores altos resultam em maior suavização, mas valores pequenos podem mostrar detalhes finos e a variação da densidade de pontos.

Quando a caixa de verificação *Avançada* é ativada, opções adicionais serão disponibilizadas:

- **Linhas e Colunas:** usada para mudar as dimensões do raster de saída. Esses valores estão também ligados aos valores **Tamanho de célula X** e **Tamanho de célula Y**. Aumentando o número de linhas ou colunas irá diminuir o tamanho de célula e aumenta o tamanho do arquivo de saída. Os valores nas Linhas e Colunas também estão ligados, portanto duplicando o número de linhas irá automaticamente duplicar o número de colunas e o tamanho da célula irá passar para metade. A área geográfica do raster de saída irá ser o mesmo!
- **Tamanho da célula X e Tamanho da célula Y:** controle o tamanho geográfico de cada pixel para o raster de saída. Alterando estes valores irá também mudar o número de Linhas e Colunas do raster de saída.
- **Forma do kernel:** A forma do kernel controla a taxa que influencia o ponto decrescente como a distância proveniente do ponto de crescente. Diferentes kerneis enfraquecem em diferentes taxas, portanto um kernel triweight dá elementos de maior peso para distâncias perto do ponto que o kernel de Epanechnikov dá.

Conseqüentemente, o resultado triweight resulta em pontos quentes “nítidos”, e Epanechnikov resulta em pontos quentes “suavizados”. Um número de funções kernel padrão estão disponíveis no QGIS, que são descritos e ilustrados no [Wikipedia](#).

- **Taxa de decaimento:** pode ser usado com kernels triangulares para maior controle de como a densidade a partir dos elementos decrescem com a distância a partir do elemento.
 - O valor de 0 (=mínimo) indica que a densidade irá ser concentrada no centro do raio dado e será extinto no borda.
 - O valor de 0.5 indica que esses pixels da borda do raio serão atribuídos metade da densidade dos pixels que estão no raio do centro de pesquisa.
 - O valor de 1 refere que a densidade espalha-se por todo o raio do círculo de pesquisa. (Isto é equivalente ao kernel ‘Uniforme’).
 - Um valor maior que 1 indica que a densidade é mais alta para a borda do raio de pesquisa, e em seguida, no centro.

A camada de pontos de entrada pode ter também campos de atributos que podem afetar como influencia o mapa de densidade:



- **Usar o raio a partir do campo:** define o raio de pesquisa para cada elemento da camada de entrada.
- **Usar o peso a partir do campo:** permite a introdução de elementos para ser pesado por um campo de atributo. Isto pode ser usado para aumentar a influência de certos elementos existente no resultado de mapa de densidade.

Quando o nome do arquivo raster de saída é especificado, o botão [OK] pode ser usado para criar o mapa de densidade.



20.14.3 Tutorial: Criando um Mapa de Densidade

Para o seguinte exemplo nós iremos usar a camada de pontos vetorial `airports` do conjunto de dados amostra do QGIS (veja *Amostra de Dados*). Outro excelente tutorial QGIS de produzir mapas de densidade podem ser encontrados em <http://qgis.spatialthoughts.com>.

Na [Figure_Heatmap_1](#) estão demonstrados os aeroportos do Alaska.

1. Selecione o botão da ferramenta **hetmap1** *Mapa de Densidade* para abrir a janela do complemento Mapas de Densidades (veja [Figure_heatmap_2](#)).
2. No campo *Introduzir camada de pontos*  seleccione `airports` da lista de camadas de pontos carregadas no projeto atual.
3. Especifique um arquivo de saída clicando o botão  perto do campo *Raster de saída* . Introduza o nome do arquivo `heatmap_airports` (não é necessário extensão).
4. Deixe o *Formato de Saída* o formato `GeoTIFF` como padrão.
5. Altere o *Raio* para 1000000 metros.
6. Clique em [OK] e carregue o mapa de temperatura dos aeroportos (veja [Figure_Heatmap_3](#)).

O QGIS irá gerar o mapa de densidade e adicionar os resultados à sua janela do mapa. Por defeito, o mapa de densidade é sombreado a cinzento, com as áreas mais brilhantes concentradas nos aeroportos. O mapa de densidade pode ser personalizado no QGIS para melhorar a aparência.

1. Abra a janela de propriedades da camada do `heatmap_airports` (selecione a camada `heatmap_airports`, abra o menu de contexto com o botão direito do rato e selecione *Propriedades*).
2. Selecione o separador *Estilo*.
3. Alterar o *Tipo de Renderização*  para ‘Banda única pseudo cor’.
4. Selecione um suitable *mapa de cor* adequado  , por exemplo `YlOrRed`.

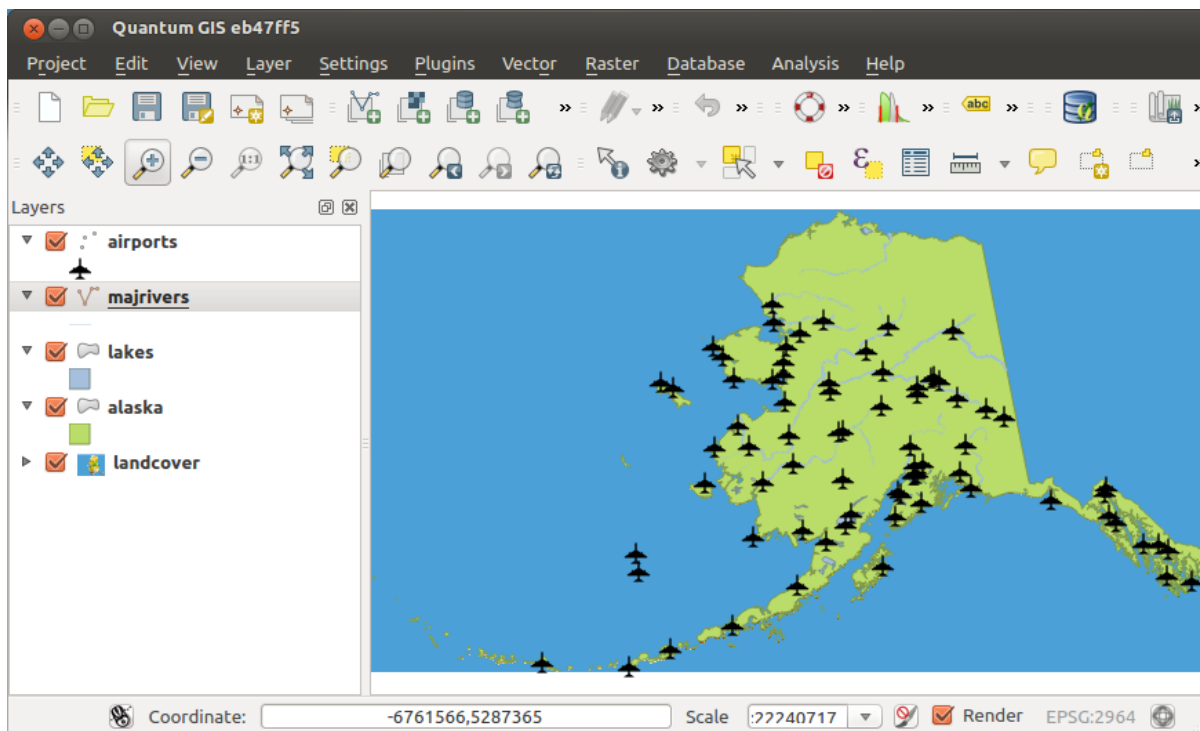


Figura 20.24: Aeroportos do Alaska 

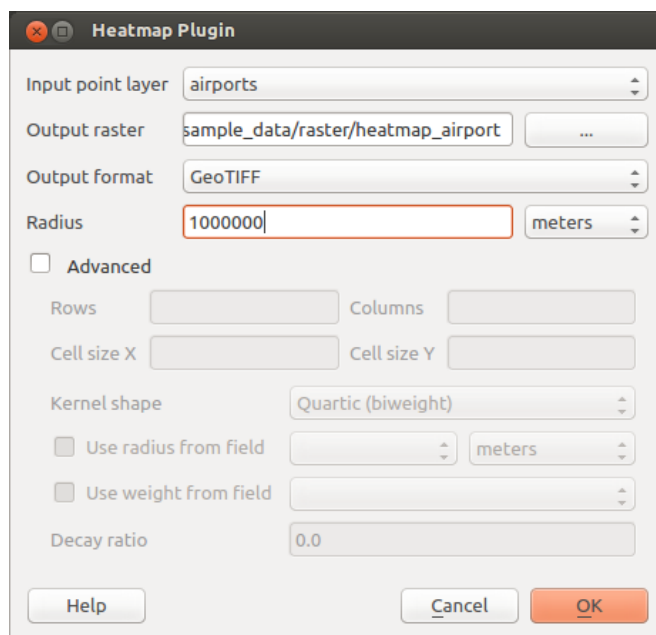



Figura 20.25: Janela do Mapa de Densidade 

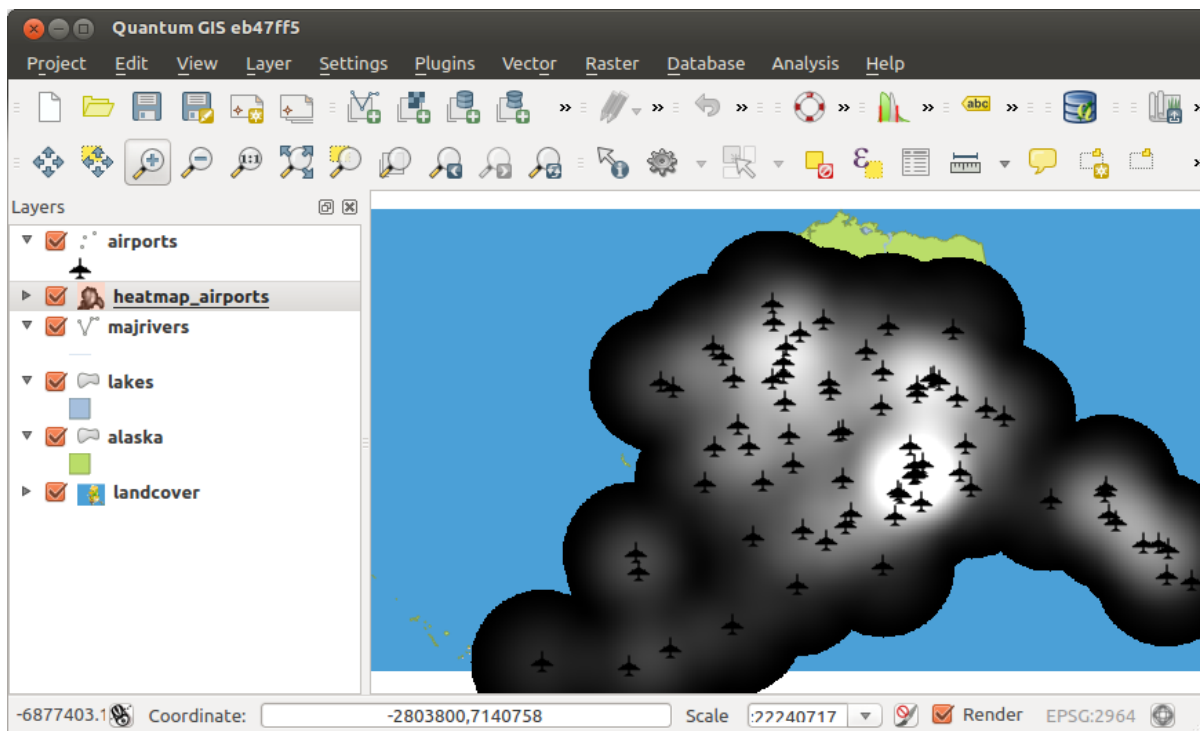
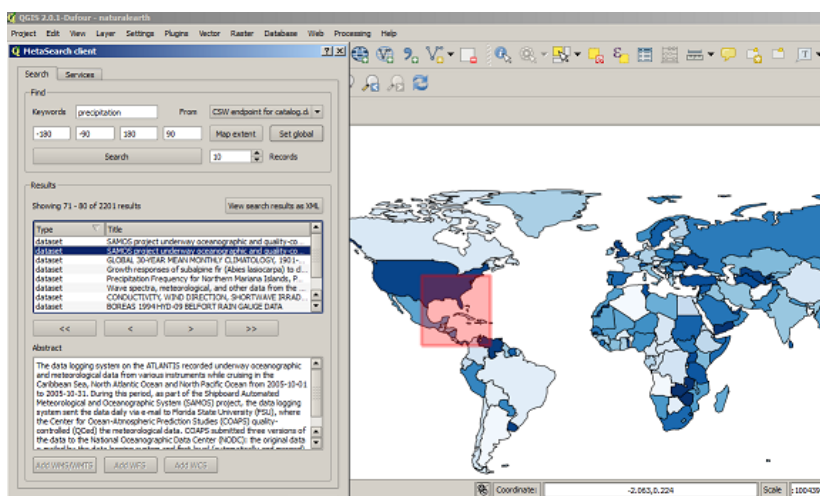


Figura 20.26: O mapa de densidade após ser carregado fica com uma superfície cinzenta 🐧

5. Clique no botão **[Carregar]** para recolher os valores mínimos e máximos para cada raster, e depois clique no botão **[Classificar]**.
6. Pressione **[OK]** para atualizar a camada.

O resultado final é demonstrado na Figure_Heatmap_4.

20.15 Catálogo do Cliente MetaBusca



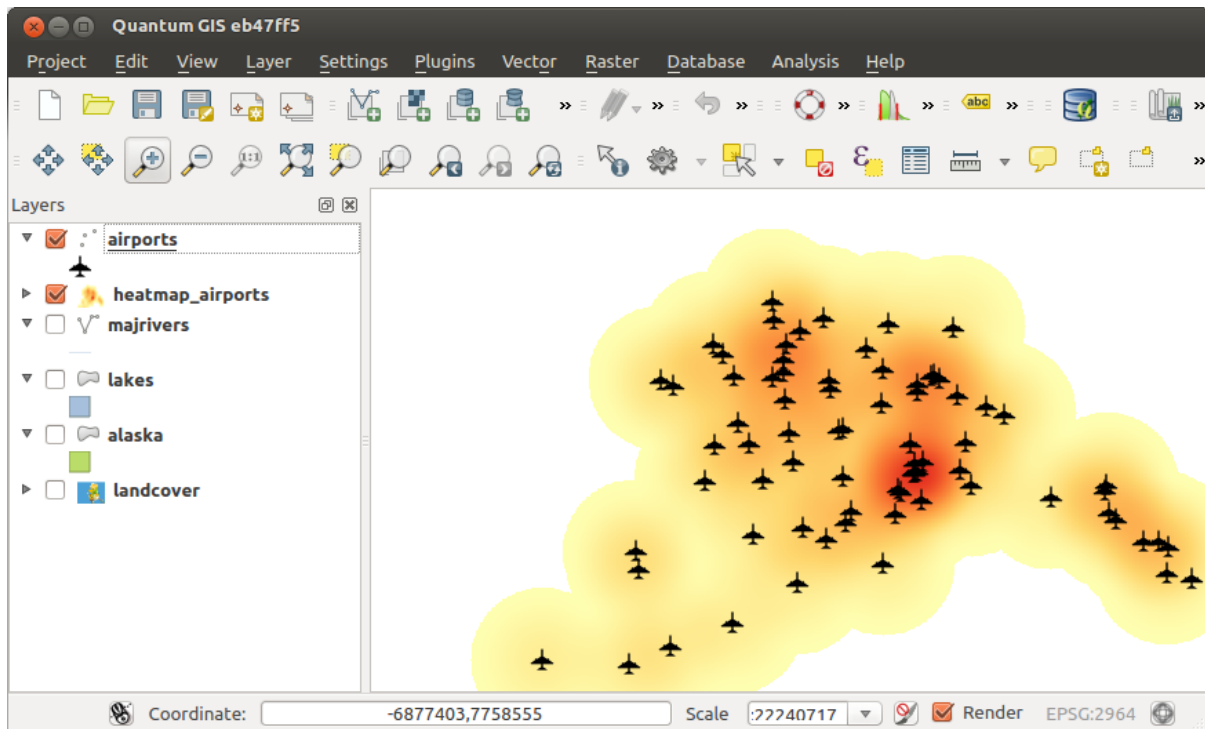


Figura 20.27: Mapa de Densidade decorado dos aeroportos do Alaska 🐧

20.15.1 Introdução

MetaBusca é um complemento QGIS para interação com serviços de catálogos de metadados, suportando Serviços de Catálogo OGC para Web padrão (CSW).

MetaBusca provem um fácil e intuitivo acesso e interface usuário-amiga para pesquisar catálogos de metadados através do QGIS.

20.15.2 Instalação

MetaBusca foi incluído por padrão com o QGIS 2.0 e superiores. Todas as dependências foram incluídas através do MetaBusca.

Instale o MetaBusca através do Gerenciador e Instalador de Complementos do QGIS, ou manualmente pelo <http://plugins.qgis.org/plugins/MetaSearch>.

20.15.3 Trabalhando com Catálogos Metadado no QGIS

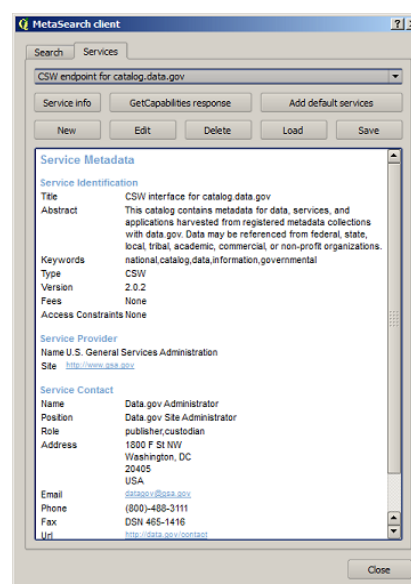
CSW (Catalogue Service for the Web)

CSW (Catalogue Service for the Web) é um OGC (Open Geospatial Consortium) especificação, que define as interfaces comuns para descobrir, pesquisar, e consultar metadados sobre os dados, serviços e outros recursos potenciais.

Iniciando

Para iniciar a MetaBusca, clique no ícone MetaBusca ou selecione Web / MetaBusca / Metabusca através do menu principal QGIS. O diálogo MetaBusca aparecerá. A GUI principal consiste em duas guias: 'Buscar' e 'Serviços'.

Gerenciador de serviços do Catálogo



A guia “Serviços” permite ao usuário gerenciar todos os serviços de catálogo disponíveis. MetaBusca fornece uma lista padrão de Catálogo de Serviços, que pode ser adicionado ao pressionar o botão ‘Adicionar serviços padrão’.

Para todas as entradas de Serviços do Catálogo listadas, clique para Selecioná-la na caixa suspensa.

Para adicionar uma entrada de serviço de catálogo, clique no botão “Novo”, e digite um nome para o serviço, bem como a URL / ponto final. Note que apenas a URL base é necessária (e não uma GetCapabilities URL completa). Ao clicar ok vai adicionar o serviço à lista de entradas.

Para editar uma entrada de serviço de catálogo existente, selecione o item que deseja editar e clique no botão “Editar”, e modifique os valores nome ou URL, depois clique ok.

Para excluir uma entrada de serviço de catálogo, selecione o item que deseja excluir e clique no botão “Excluir”. Você será solicitado para confirmar a exclusão da entrada.

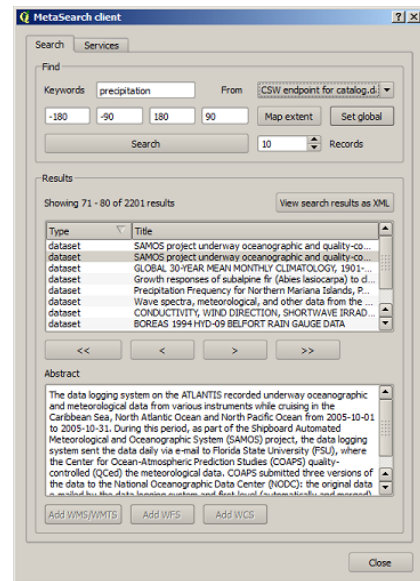
MetaBusca permite carregar e salvar conexões de um arquivo XML. Isso é útil quando você precisa compartilhar configurações entre as aplicações. Abaixo está um exemplo do formato de arquivo XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<qgsCSWConnections version="1.0">
  <csw name="Data.gov CSW" url="http://catalog.data.gov/csw-all"/>
  <csw name="Geonorge - National CSW service for Norway" url="http://www.geonorge.no/geonetwork/eng/inspire-csw-service-for-norway">
  <csw name="Geoportale Nazionale - Servizio di ricerca Italiano" url="http://www.pcn.minambiente.it/geoportale/eng/inspire-csw-service-for-italy">
  <csw name="LINZ Data Service" url="http://data.linz.govt.nz/feeds/csw"/>
  <csw name="Nationaal Georegister (Nederland)" url="http://www.nationaalgeoregister.nl/geonetwork/eng/inspire-csw-service-for-netherlands">
  <csw name="RNDT - Repertorio Nazionale dei Dati Territoriali - Servizio di ricerca" url="http://www.rndt.it/eng/inspire-csw-service-for-italy">
  <csw name="UK Location Catalogue Publishing Service" url="http://csw.data.gov.uk/geonetwork/eng/inspire-csw-service-for-uk">
  <csw name="UNEP/GRID-Geneva Metadata Catalog" url="http://metadata.grid.unep.ch:8080/geonetwork/eng/inspire-csw-service-for-unep">
</qgsCSWConnections>
```

Para carregar uma lista de entradas, clique no botão ‘Carregar’. Uma nova janela irá aparecer; clique no botão “Buscar” e navegue até o arquivo XML de entradas que você deseja carregar e clique em “Abrir”. A lista de entradas será exibida. Selecione as entradas que você deseja adicionar na lista e clicar em ‘Carregar’.

O botão ‘Informação do serviço’ exibe informações sobre o Serviço de Catálogo selecionado, como a identificação de serviço, provedor do serviço e informações de contato. Se você gostaria de ver a resposta XML bruta, clique no botão “Resposta GetCapabilities”. Uma janela separada será aberta exibindo Solicitação de XML e Resposta.

Buscar Catálogo de Serviços



A janela ‘Buscar’ permite ao usuário consultar os Catálogo de dados e serviços, estabelecendo vários parâmetros de pesquisa e visualizando os resultados.

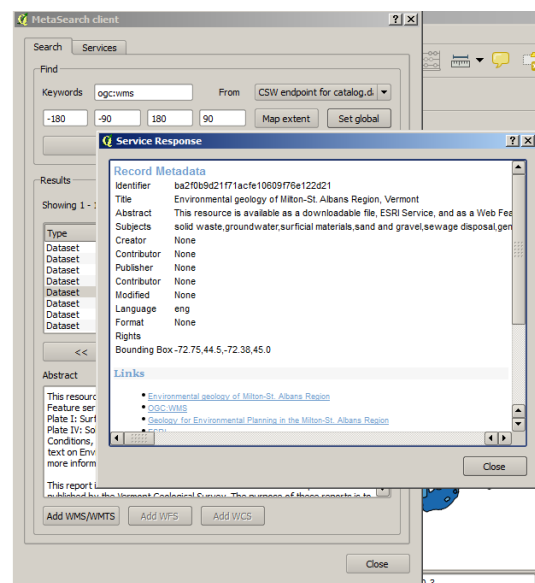
Os seguinte parâmetros de pesquisa estão disponíveis:

- **Palavras-Chave:** texto livre buscando palavras chave
- **De:** o Serviço de catálogo para realizar a consulta de relação
- **Caixas limites:** a área espacial de interesse para filtrar. A caixa delimitadora padrão é a visualizacao do mapa / tela. Clique em ‘Conjunto global’ para fazer uma pesquisa global, ou inserir valores personalizados como desejado
- **Resultados:** o número de registros que retornam ao pesquisar. O padrão é 10 resultados por página

Clicando no botão ‘Buscar’ irá procurar os Catálogos de Metadados selecionados. O resultado da pesquisa será exibido na lista e podem ser classificados, clicando na cabeçalho da coluna título. Você pode navegar através dos resultados da pesquisa com os botões direcionais abaixo dos resultados da pesquisa. Clicando no botão ‘Ver resultados da pesquisa como XML’ abrirá uma janela com a resposta do serviço em formato XML puro.

Ao clicar no resultado mostrará resumo do registro na janela ‘Registro de Metadados’ e oferece as seguintes opções:

- se o registro de metadados tem uma caixa delimitadora associada, uma pegada de caixa delimitadora será exibida no mapa
- duplo clique nos resultados abrirá a tela com o registro de medados com seus links de acesso. Clicando nos links abrirá o link em seu navegador web
- se o registro é um serviço web OGC (WMS / WMTS, WFS, WCS), os apropriados botões a “Adicionar ao WMS / WMTS | WFS | WCS’ serão habilitados para o usuário adicionar ao QGIS. Ao clicar neste botão, MetaBusca irá verificar se este é um OWS válido. O OWS será então adicionado à lista de conexão QGIS adequada, e os diálogo adequados WMS/WMTS|WFS|WCS irão aparecer



Configurações

Você pode fazer sintonia fina no MetaBusca com as seguintes configurações:

- **Resultados de paginação:** quando procurar catálogos de metadados, o número de resultados mostrados por página
- **Tempo de espera:** quando procurar catálogos de metadados, o número de segundos para o bloqueio de tentativa de conexão. O valor padrão é 10

20.16 Complemento Menor Distância

O Complemento Menor Distância é um complemento em C++ para o QGIS que calcula o caminho mais curto entre dois pontos em qualquer camada poligonal e contextualiza este caminho através da rede rodoviária.

Características principais:

- Calcula caminho, assim como o comprimento e tempo de viagem.
- Otimiza pelo comprimento ou tempo de viagem.
- Exporta o caminho para camada vetorial.
- Destaca estradas direções (isto pode ser lento e usado principalmente para propósitos para o teste de configurações).

Como uma camada estradas, você pode usar qualquer camada poligonal vetorial em qualquer formato suportado pelo QGIS. Duas linhas com um ponto em comum são consideradas conectadas. Por favor, note que é necessário usar um SRC para camada como SRC do projeto durante a edição de uma camada de estradas. Isto é devido ao fato de recálculo das coordenadas entre diferentes SRCs introduz alguns erros que podem resultar em descontinuidades, mesmo quando “aproximação” for usado.

Na tabela de atributos da camada, o campo a seguir pode ser usado:

- Velocidade no percurso rodoviário (campo número).
- Direção (qualquer tipo que possa ser convertido para texto). Frente e direções inversas correspondem a uma via de única mão, ambas as direções indicam uma via de mão dupla.

Se alguns campos não têm qualquer valor ou não existem, os valores padrão serão usados. Você pode alterar os padrões e algumas configurações do complemento na janela de configuração do complemento.

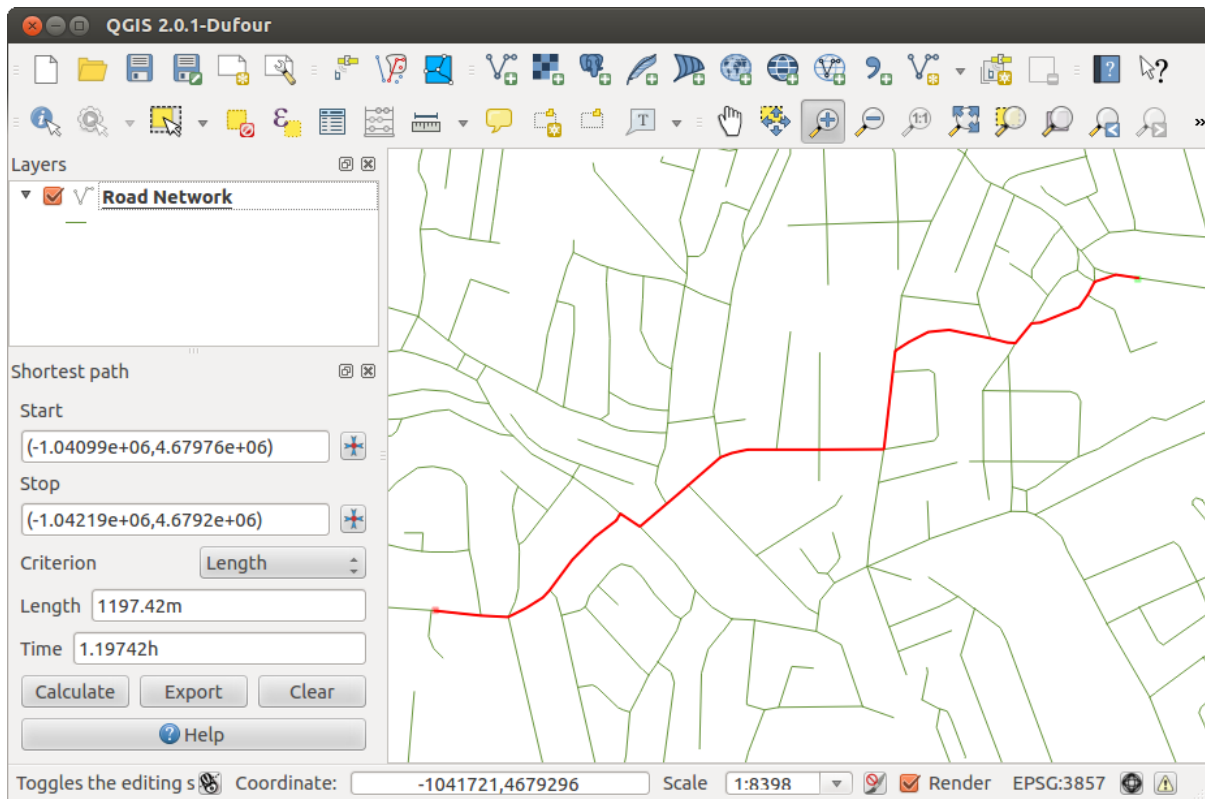


Figura 20.28: Complemento Menor Distância 🐧


20.16.1 Usando o complemento

Depois da ativação do complemento, você vai ver um painel adicional no lado esquerdo da janela do QGIS. Agora, entre com os parâmetros na janela *Configurações do complemento Menor Distância* no menu *Vetor* → *Menor Distância* (ver [figure_road_graph_2](#)).

Depois de configurar a *Unidade de tempo*, *Unidade de distância* e *Tolerância da topologia*, você pode escolher a camada vetorial na aba *Camada de transporte*. Aqui você também pode escolher o *Campo de direção* e *Campo de velocidade*. Na aba *Configurações padrão*, você pode definir a *Direção* para o cálculo.

Finalmente, no painel *Caminho mais curto*, selecione um ponto de início e de fim na camada de rede de caminhos e clique em [**Calcular**].

20.17 Complemento de Consulta Espacial

O complemento  Complemento de Consulta Espacial permite que faça consultas espaciais (ex.: selecionar elementos) numa camada alvo com referência a outra camada. A funcionalidade é baseada na biblioteca GEOS e depende de uma camada fonte de elementos selecionada.

Operações possíveis:

- Contém
- Igual
- Sobrepõe
- Cruza
- Intersecta

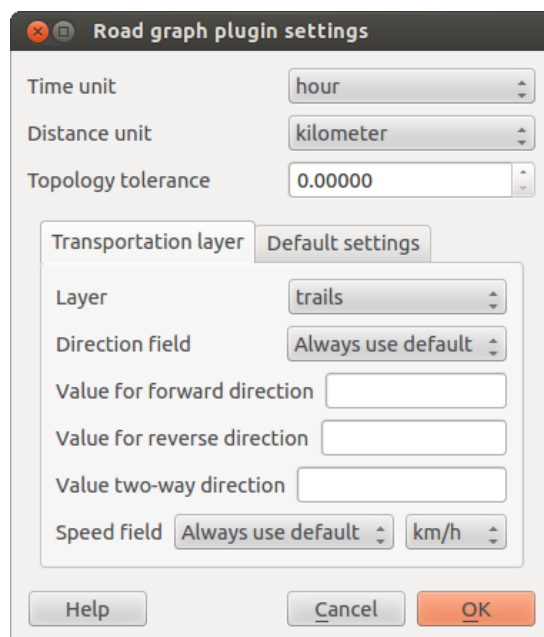




Figura 20.29: Configurações do complemento Menor Distância 




- Disjunto
- Toca
- Dentro

20.17.1 Usando complemento

Como exemplo, nós queremos encontrar as regiões no conjunto de dados do Alaska que contém os aeroportos. Os seguintes passos são necessários:

1. Iniciar QGIS e carregar as camadas de vetoriais :file: *regions.shp* e :file: *airports.shp*.
2. Carregue o complemento Consulta Espacial no Gerenciador de Complementos (ver *Diálogo de Complementos*) e clique no ícone  Consulta Espacial que aparece no menu de ferramentas do QGIS. A janela do complemento aparecerá.
3. Selecione a camada *regions* como camada fonte e *airports* como camada de elementos referência.
4. Selecione ‘Contém’ como operador e clique [Aplicar].

Agora obtém uma lista de elementos ID da consulta e tem várias opções como é mostrado na *figure_spatial_query_1*.

- Clique em  Criar camada com a lista de itens
- Selecione um ID da lista e clique em  Criar uma camada com os selecionados .
- Selecione ‘Remover da seleção atual’ no campo *E uso resultado para* .
- Adicionalmente pode *Ampliar ao item* ou exibir *Registro de mensagens*.

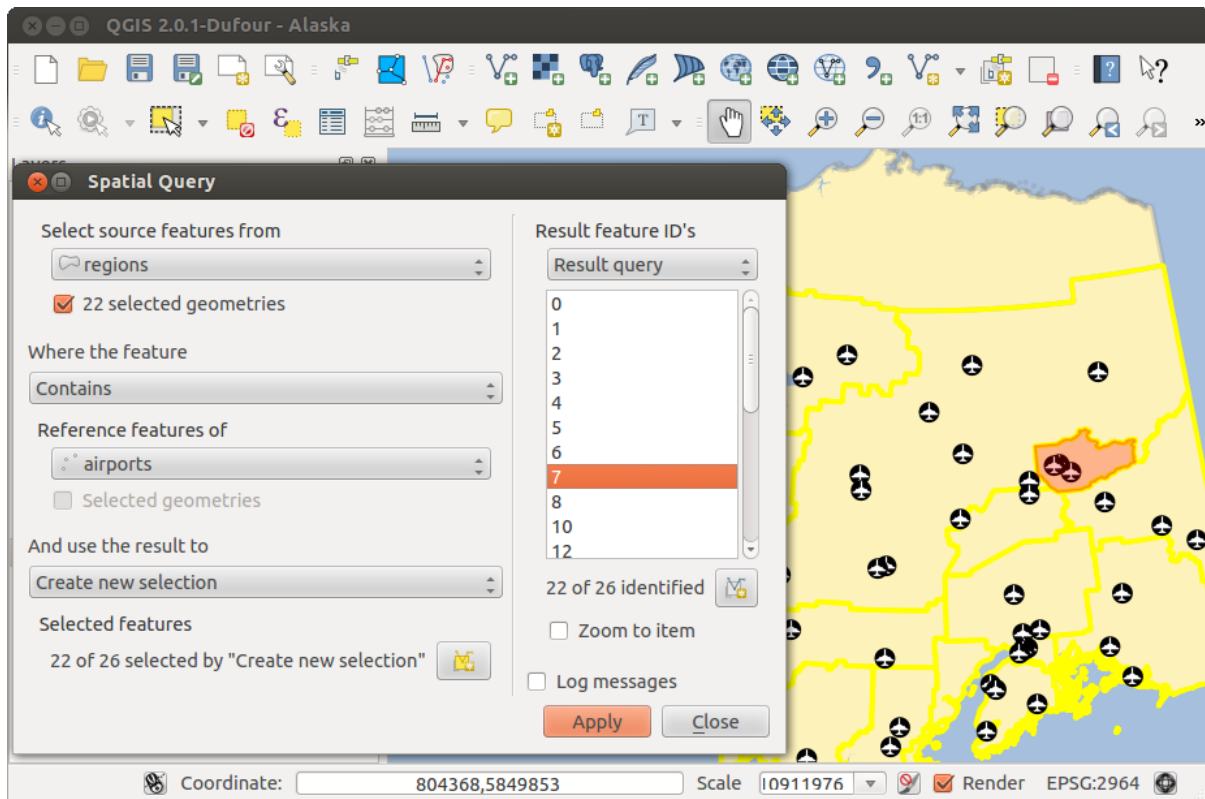


Figura 20.30: Análise de consulta espacial - regiões contêm aeroportos 🐧

20.18 Complemento SPIT

O QGIS vem com um complemento chamado SPIT (Ferramenta de Importação de Shapefile para PostGIS). O SPIT pode ser usado para carregar múltiplas shapefiles de uma só vez e inclui o suporte para esquemas. Para usar o SPIT, abra o Gerenciador de Complementis do menu *Complementos*, no menu *Instalado* e marque a caixa perto do :guilabel:'SPIT' e clique [OK].

Para importar uma shapefile, use *Base de dados* → *Spit* → *Importar Shape para PostgreSQL* da barra de menu para abrir a janela *SPIT - Ferramenta de Importação de Shape para PostGIS*. Selecione a base de dados PostGIS que quer conectar e clique em [Conectar]. Se quiser, pode definir ou alterar opções de importação. Agora pode adicionar um ou mais arquivos para a fila clicando no botão [Adicionar]. Para processar os arquivos, clique no botão [OK]. O progresso de importação assim como algum tipo de erros/avisos serão exibidos em cada shapefile processado.

20.19 Complemento SQL Anywhere

O SQL Anywhere é um sistema de gerenciamento de bancos de dados com propriedade relacional (RDBMS) do Sybase. O SQL Anywhere possui suporte espacial, incluindo OGC, shapefiles e funções internas para exportar para os formatos KML, GML e SVG.

SQL Anywhere permite que você se conecte a bancos de dados do SQL Anywhere espacialmente habilitado. A janela *Add SQL Anywhere layer* é similar em funcionalidade com a janela do PostGIS e SpatiaLite.

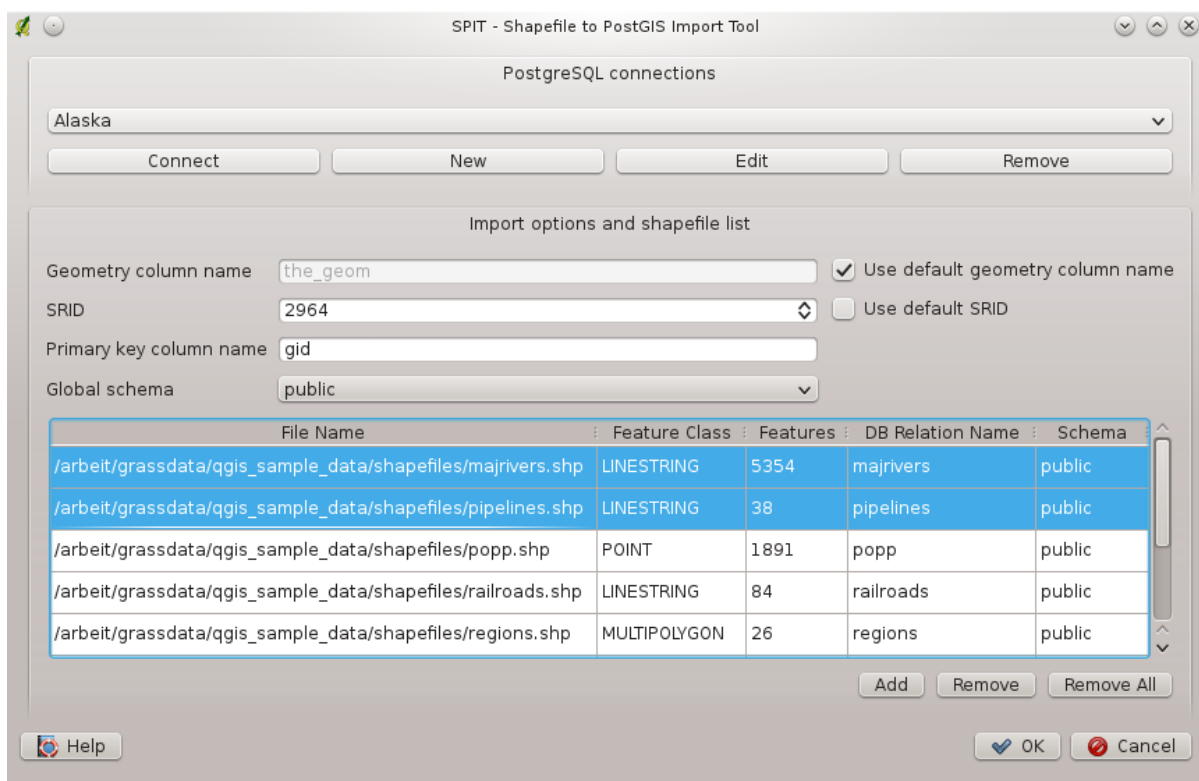


Figura 20.31: Usando Complemento SPIT para importar arquivos shapefile do PostGIS 🐧

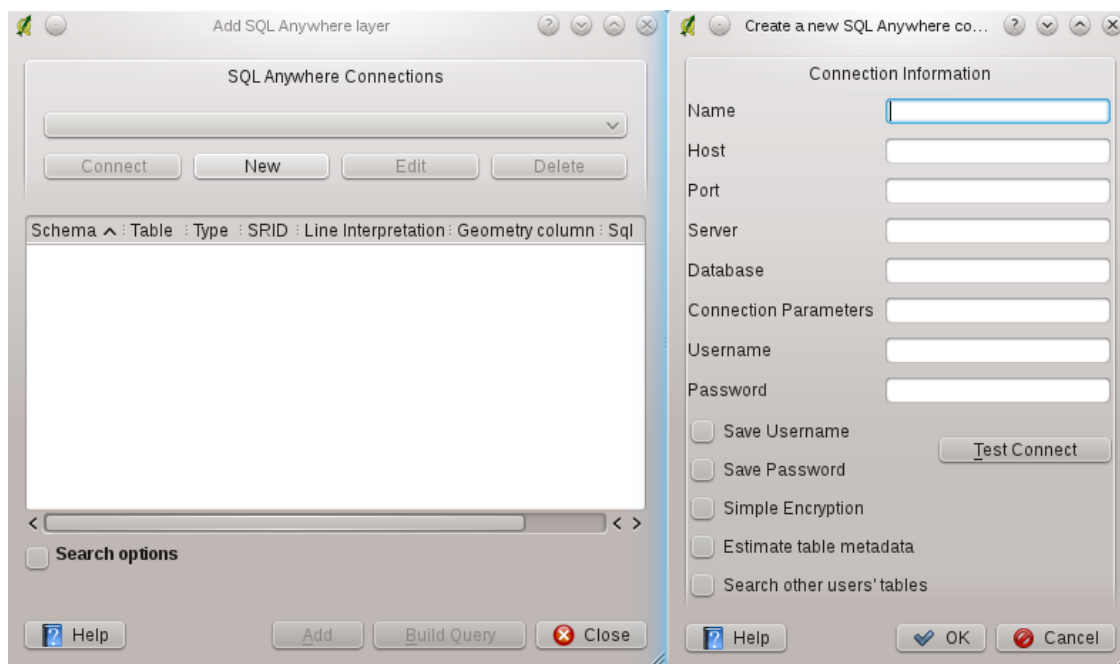


Figura 20.32: Janela SQL Anywhere (KDE) 🐧

20.20 Complemento Verificador de Topologia

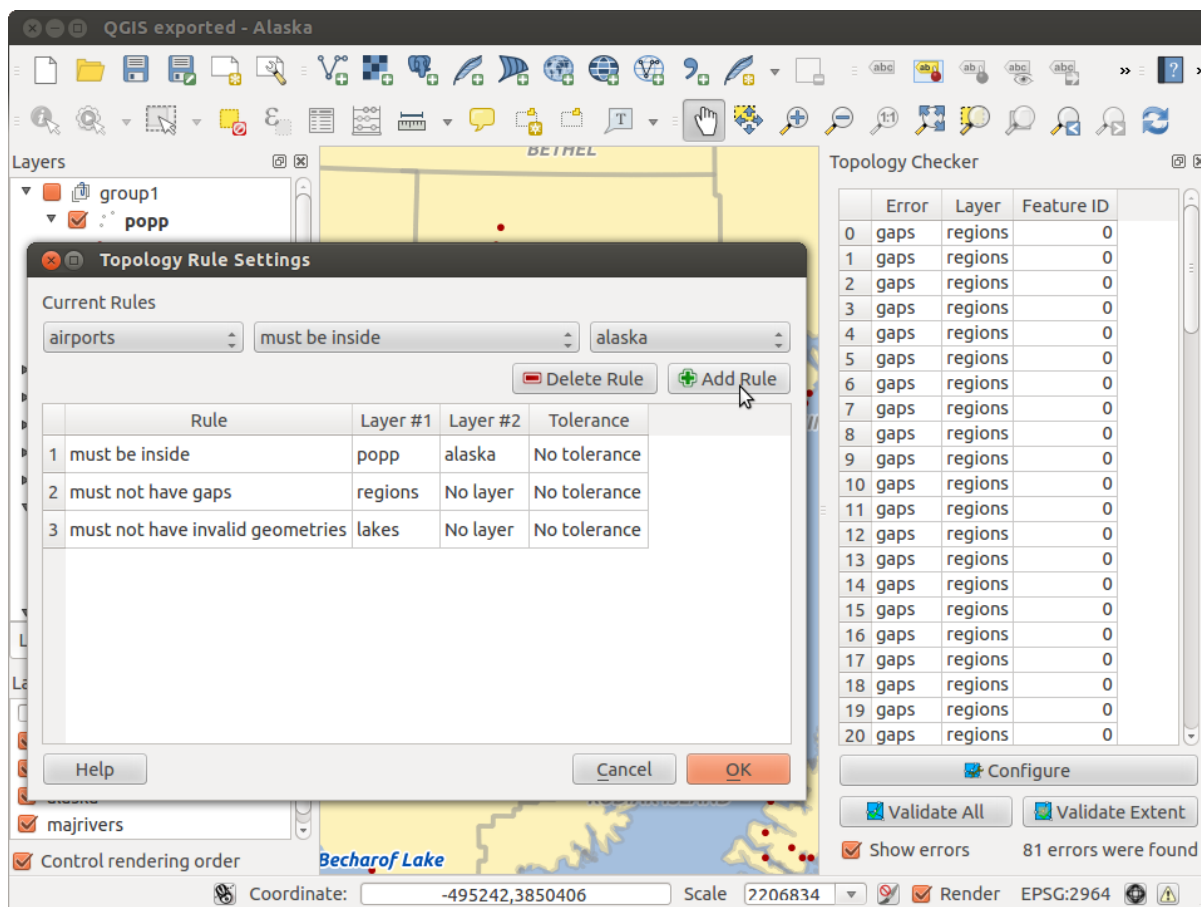


Figura 20.33: O Complemento Verificador de Topologia

Topologia descreve as relações entre pontos, linhas e polígonos que representam as feições de uma região geográfica. Com o complemento Verificador de Topologia, você pode olhar sobre seus arquivos vetoriais e verificar a topologia com várias regras de topologia. Estas regras verificam com as relações espaciais se suas características 'igual', 'contém', 'copia', são 'copiar para', 'Cruzar', são 'Disjunção', 'Interseção', 'Sobreposição', 'Toque' ou são 'Dentro' entre si. Tudo depende das suas questões individuais que regras topológicas que você aplicar a seus dados vetoriais (por exemplo, normalmente você não vai aceitar superação da camadas de linha, mas se eles retratam ruas sem saída que você não vai removê-los de sua camada vetorial).

QGIS tem uma feição de edição topológica embutida, o que é ótimo para a criação de novas feições sem erros. Mas os erros de dados existentes e os erros induzidos pelo usuário são difíceis de encontrar. Este complemento ajuda a encontrar tais erros através de uma lista de regras.

É muito simples para criar regras topológicas com o complemento Verificador de Topologia

Nas **camadas do tipo vetor** as seguintes regras estão disponíveis:

- **Deve ser coberto por:** Aqui você pode escolher uma camada vetorial de seu projeto. Pontos que não forem cobertos pela camada de dados vetorial ocorrem no campo 'Erro'.
- **Deve ser coberto por pontos finais:** Aqui você pode escolher uma camada de linha de seu projeto.
- **Deve estar dentro:** Aqui você pode escolhe uma camada de polígono a partir de seu projeto. Os pontos devem estar dentro de um polígono. Caso contrário, o QGIS escreve um 'Erro' para o ponto.
- **Não deve haver duplicados:** Sempre que um ponto é representado duas vezes ou mais, vai ocorrer no campo 'Erro'.
- **Não deve haver geometrias inválidas:** Verifica se as geometrias são válidas.

- **Não deve haver geometrias multi partes:** Todos os pontos de multi partes estão escritos no campo 'Erro'.


Em **camadas de linha**, as seguintes regras estão disponíveis:

- **Pontos finais devem ser cobertos por:** Aqui você pode selecionar uma camada de ponto a partir de seu projeto.
- **Não deve haver oscilação:** Isto irá mostrar as superações na camada de linha.
- **Não deve ter duplicações:** Sempre que uma feição de linha for representada duas vezes ou mais, irá acusar no campo 'Erro'.
- **Não deve haver geometrias inválidas:** Verifica se as geometrias são válidas.
- **Não deve haver geometrias multi partes:** Às vezes, a geometria é realmente uma coleção de geometrias simples (partes simples). Essa geometria é chamada geometria multi-parte. Se ela contém apenas um tipo de geometria simples, chamamos isso de multi-ponto, multi-cadeia de linha ou multi-polígono. Todas as linhas de várias partes estão escritas no campo 'Erro'.
- **Não deve ter pseudos:** Extremidade de uma geometria linha deve ser conectada aos extremidade de outras duas geometrias. Se a extremidade estiver ligada a apenas uma extremidade de outra geometria, a extremidade é chamada de pseudo nó.

Em **camadas de polígono**, as seguintes as regras estão disponíveis:

- **Deve conter:** camada de polígono deve conter geometria em pelo menos um ponto da segunda camada.
- **Não deve ter duplicações:** Polígonos a partir da mesma camada não deve ter geometrias idênticas. Sempre que um polígono é representado duas vezes ou mais irá ocorrer no campo 'Erro'.
- **Não deve ter lacunas:** polígonos adjacentes não deve formar lacunas entre eles. Limites administrativos poderiam ser mencionado como um exemplo (polígonos estaduais do BRASIL não possuem lacunas entre eles...).
- **Não deve ter geometrias inválidas:** Verifica se as geometrias são válidas. Algumas das regras que definem uma geometria válida são:
 - Os anéis do polígono devem estar fechados.
 - Anéis que definem buracos devem estar dentro de anéis que definem os limites exteriores.
 - Os anéis não se podem intersectar a si mesmo (nem mesmo tocar ou cruzar um no outro).
 - Os anéis não podem tocar outros anéis, à exceção de um ponto.
- **Não deve ter geometrias multi partes:** Às vezes, a geometria é realmente uma coleção de geometrias simples (partes simples). Essa geometria é chamada geometria multi-parte. Se ele contém apenas um tipo de geometria simples, chamamos isso de multi-ponto, multi-cadeia de linha ou multi-polígono. Por exemplo, um campo constituído por múltiplas ilhas pode ser representado como um sistema multi-polígono.
- **Não devem se sobrepor:** polígonos adjacentes não devem compartilhar áreas em comum.
- **Não deve sobrepor-se:** polígonos adjacentes de uma camada não devem compartilhar áreas comuns com polígonos de outra camada.

20.21 Complemento Estatística Zonal

Com o  Complemento Estatística Zonal pode-se analisar os resultados de uma classificação temática. Permite calcular vários valores de pixels de um raster com ajuda de uma camada poligonal (veja [figure_zonal_statistics](#)). Pode calcular a soma, valor médio e contagem total dos pixels que estão dentro do polígono. Este complemento gera uma coluna de saída na camada vetorial com um prefixo definido pelo usuário.

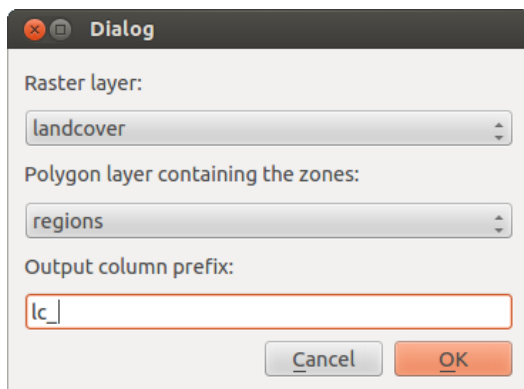


Figura 20.34: Janela Estatística Zonal (KDE) 🐧

Ajuda e Suporte

21.1 Listas de Discursão

O QGIS está ativo no seu desenvolvimento e como tal poderá não apresentar alguns erros. A melhor forma para obter ajuda será juntar-se na lista de discussão qgis-users. As suas questões irão chegar a uma audiência mais ampla e as respostas irão beneficiar outros.

21.1.1 Usuários QGIS

Esta lista de discussão é usada para uma discussão geral do QGIS, assim como as questões específicas relacionadas com a instalação e o uso. Pode assinar a lista de discussão do qgis-users visitando o seguinte URL: <http://lists.osgeo.org/mailman/listinfo/qgis-user>

21.1.2 Lista fossgis-talk

Para os utilizadores de língua alemã o FOSSGIS e.V. alemão fornece uma lista de discussão fossgis-talk. Esta lista de discussão é usada para a discussão dos SIG de código aberto incluindo o QGIS. Pode assinar a lista de discussão fossgis-talk visitando o seguinte URL: <https://lists.fossGIS.de/mailman/listinfo/fossGIS-talk-liste>

21.1.3 Desenvolvedor QGIS

Se você é um desenvolvedor enfrentando problemas de natureza mais técnica, você pode querer se juntar a lista de discussão qgis-desenvolvedor aqui: <http://lists.osgeo.org/mailman/listinfo/qgis-developer>

21.1.4 Atualizações QGIS

Cada vez que um commit é feito no repositório de código QGIS um email é publicado nesta lista. Se pretende estar ocorrente de todas as alterações ao código base actual, pode subscrever-se nesta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-commit>

21.1.5 QGIS-trac

Esta lista fornece uma notificação de email relacionado a gerenciamento de projetos, incluindo relatórios de bugs, tarefas e solicitações de recursos. Você pode se inscrever para esta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-trac>

21.1.6 Equipe da Comunidade QGIS

Esta lista trata de temas como a documentação, ajuda de contexto, guia do usuário, experiência online incluindo sites, blog, listas de discussão, fóruns e esforços de tradução. Se você gosta de trabalhar com o guia do usuário, esta lista é um bom ponto de partida para fazer suas perguntas. Você pode se inscrever para esta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-community-team>

21.1.7 Equipe de lançamento QGIS

Esta lista inclui tópicos como o processo de lançamento, pacotes binários para vários Sistemas Operacionais e anúncios de novos lançamentos para o mundo em geral. Você pode se inscrever nesta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-release-team>

21.1.8 Tradução do QGIS

Esta lista trata dos esforços de tradução. Se você gosta de trabalhar na tradução dos manuais ou na interface gráfica do usuário (GUI), esta lista é um bom ponto de partida para fazer suas perguntas. Você pode se inscrever nesta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-tr>

21.1.9 Aprendizado QGIS

Esta lista lista contém os esforços na área de educação do QGIS. Se quiser trabalhar nos materiais de educação, esta lista é boa para começar a fazer perguntas. Pode assinar esta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-edu>

21.1.10 Comitê Diretor QGIS

Esta lista é usada para discutirmos assuntos do Comitê de Direção relacionados com a boa gestão e direção do GIS. Podem assinar esta lista em: <http://lists.osgeo.org/mailman/listinfo/qgis-psc>

Você está convidado a se inscrever em qualquer uma das listas. Por favor, lembre-se de contribuir com a lista respondendo a perguntas e compartilhando suas experiências. Note que os links das listas Atualizações QGIS e QGIS-trac são projetados apenas para notificação e não destinado a postagens de usuários.

21.2 IRC

Nós também marcamos presença no IRC - visite-nos acessando o canal # qgis em irc.freenode.net. Por favor, aguarde pela resposta de sua pergunta. Como muitas pessoas no canal estão fazendo outras atividades isso pode levar um tempo para perceberem sua pergunta. Se perder a discussão no IRC, não tem problema! Nós registamos toda a discussão, portanto é fácil estar atualizado. Vá a <http://qgis.org/irclogs> and read the IRC-logs.

Suporte comercial para o QGIS também está disponível. Verifique o website <http://qgis.org/en/commercial-support.html> for more information.

21.3 Rastreador de Erros

Enquanto que a lista de discussão do [qgis-users](http://lists.osgeo.org/mailman/listinfo/qgis-users) é útil para as típicas questões ‘como eu faço o XYZ no QGIS’, você pode notificar-nos sobre erros existentes no QGIS. Você pode submeter erros usando o rastreador de erros QGIS em <http://hub.qgis.org/projects/quantum-gis/issues>. Quando cria uma nova notificação para erro, por favor forneça um endereço email onde poderemos pedir informação adicional.

Por favor tenha em atenção que o seu erro pode nem sempre ter a prioridade que deseja (vai depender da severidade). Alguns erros podem requer esforços significativos de programadores para remediar e a mão-de-obra nem sempre está disponível para isso.

Solicitações de recursos também podem ser enviadas usando o mesmo sistema de mensagem de erros. Por favor, certifique-se de ter selecionado o tipo de recurso.

Se encontrou um erro e corrigiu-o você pode submeter esse patch. Mais uma vez, o adorável sistema de avisos redmine do <http://hub.qgis.org/wiki/quantum-gis/issues> funciona bem para isso. Ative a caixa de verificação `Patch fornecido` e anexe o seu patch antes de submeter o erro. Alguém dos programadores irá revê-lo e aplicá-lo no QGIS. Por favor não fique alarmado se o seu patch não for logo aplicado — os programadores podem estar ocupados com outros compromissos.

21.4 Blog

A comunidade QGIS também possui um blog em <http://planet.qgis.org/planet/> que tem alguns artigos interessantes para utilizadores e programadores assim como outros blogs da comunidade. Você está convidado para contribuir com o seu blog QGIS!

21.5 Plugins

A página web <http://plugins.qgis.org> fornece o portal oficial de complementos QGIS. Aqui você pode encontrar uma lista de todos os módulos estáveis e experimentais do QGIS através do ‘Repositório Oficial de Complementos QGIS’.

21.6 Wiki

E finalmente, nós mantemos uma página WIKI em <http://hub.qgis.org/projects/quantum-gis/wiki> onde você pode encontrar uma variedade de informações úteis relacionada com o desenvolvimentos do QGIS, planos de lançamento, links para sites de transferência, mensagens de dicas de tradução entre outros. Visite, lá temos boas explicações!

22.1 Licença Pública Geral GNU

Versão 2, Junho 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

É permitido a todos copiar e distribuir uma cópia desta documento da licença, porém não é permitido alterar esta.

Preâmbulo

As licenças de muitos software são desenvolvidas para restringir sua liberdade de compartilhá-lo e mudá-lo. Contrária a isso, a Licença Pública Geral GNU pretende garantir sua liberdade de compartilhar e alterar software livres – garantindo que o software será livre e gratuito para os seus usuários. Esta Licença Pública Geral aplica-se à maioria dos software da Free Software Foundation e a qualquer outro programa cujo autor decida aplicá-la. (Alguns outros software da FSF são cobertos pela Licença Pública Geral de Bibliotecas, no entanto.) Você pode aplicá-la também aos seus programas.

Quando falamos em softwares livres, nos referimos a liberdade, não preço. Nossa Licença Pública Geral é projetada para garantir: que você tenha liberdade de compartilhar cópias do software livre (e cobrar por este serviço, se desejar) que você recebeu o código fonte ou pode adquirir se quiser; que você possa modificar o software ou usar partes dele em novos softwares livres; que você saiba que pode fazer tais coisas.

Para proteger os seus direitos nós precisamos fazer restrições que proibem qualquer pessoa a negar a você tais direitos ou a pedir que se renda a eles. Estas restrições se traduzem em certas responsabilidades para você, se você distribuir ou modificar cópias do software.

Por exemplo, se você distribuir cópias de determinado programa, seja ele grátis ou por uma taxa, você deve dar aos beneficiários todos os direitos que você tem. Você deve garantir que eles também recebam ou possam acessar o código fonte. E você deve mostrar a eles estes termos para que eles saibam seus direitos.

Nós protegemos seus direitos com dois passos: (1) direitos autorais do software, e (2) oferecemos a você esta licença que lhe dá a permissão legal para copiar, distribuir e/ou modificar o software.

Também, para a proteção nossa e de cada autor, nós queremos garantir que todos compreendam que não há garantia para este software livre. Se o software for modificado por outra pessoa e passado adiante, os beneficiários devem saber que o que eles tem não é o software original, para que, qualquer problema introduzido por outros não reflita na reputação do autor original.

Por fim, qualquer programa livre é constantemente ameaçado por patentes de softwares. Nós queremos evitar o risco de que redistribuidores de um programa livre obtenham licenças individuais, fazendo assim do programa proprietário. Para prevenir isso, nós deixamos claro que qualquer patente deve ser licenciada para o uso livre de todas as pessoas.

Seguem os precisos termos e condições para cópia, distribuição e modificação. **TERMOS E CONDIÇÕES PARA CÓPIA, DISTRIBUIÇÃO E MODIFICAÇÃO**

0. Esta licença se aplica a qualquer programa ou outro trabalho que contenha um aviso inserido pelo detentor dos direitos autorais informando que o mesmo pode ser distribuído sob as condições desta Licença Pública

Geral. O “Programa” abaixo refere-se a qualquer programa ou trabalho, e “trabalho baseado no Programa” significa tanto o Programa em si como quaisquer trabalhos derivados, de acordo com a lei de direitos autorais: isto quer dizer um trabalho que contenha o Programa ou parte dele, tanto originalmente ou com modificações, e/ou tradução para outros idiomas. (Doravante o processo de tradução está incluído sem limites no termo “modificação”.) Cada licenciado é mencionado como “você”.

Atividades outras que a cópia, a distribuição e modificação não estão cobertas por esta Licença; elas estão fora de seu escopo. O ato de executar o Programa não é restringido e o resultado do Programa é coberto apenas se seu conteúdo contenha trabalhos baseados no Programa (independentemente de terem sido gerados pela execução do Programa). Se isso é verdadeiro depende do que o programa faz.

1. Você pode copiar e distribuir cópias fiéis do código-fonte do Programa da mesma forma que você o recebeu, usando qualquer meio, desde que você conspicua e apropriadamente publique em cada cópia um aviso de direitos autorais e uma declaração de inexistência de garantias; mantenha intactas todos os avisos que se referem a esta Licença e à ausência total de garantias; e forneça a outros recebedores do Programa uma cópia desta Licença, junto com o Programa.

Você pode cobrar uma taxa pelo ato físico de transferir uma cópia e pode, opcionalmente, oferecer garantia em troca de pagamento.

2. Você pode modificar sua cópia ou cópias do Programa, ou qualquer parte dele, assim gerando um trabalho baseado no Programa, e copiar e distribuir essas modificações ou trabalhos sob os termos da seção 1 acima, desde que você também se enquadre em todas estas condições:
 - (a) Você tem que fazer com que os arquivos modificados levem avisos proeminentes afirmando que você alterou os arquivos, incluindo a data de qualquer alteração.
 - (b) Você tem que fazer com que quaisquer trabalhos que você distribua ou publique, e que integralmente ou em partes contenham ou sejam derivados do Programa ou de suas partes, sejam licenciados, integralmente e sem custo algum para quaisquer terceiros, sob os termos desta Licença.
 - (c) Se qualquer programa modificado normalmente lê comandos interativamente quando executados, você tem que fazer com que, quando iniciado tal uso interativo da forma mais simples, seja impresso ou mostrado um anúncio de que não há qualquer garantia (ou então que você fornece a garantia) e que os usuários podem redistribuir o programa sob estas condições, ainda informando os usuários como consultar uma cópia desta Licença. (Exceção: se o Programa em si é interativo mas normalmente não imprime estes tipos de anúncios, seu trabalho baseado no Programa não precisa imprimir um anúncio.)

Estas exigências aplicam-se ao trabalho modificado como um todo. Se seções identificáveis de tal trabalho não são derivadas do Programa, e podem ser razoavelmente consideradas trabalhos independentes e separados por si só, então esta Licença, e seus termos, não se aplicam a estas seções quando você distribui-las como trabalhos em separado. Mas quando você distribuir as mesmas seções como parte de um todo que é trabalho baseado no Programa, a distribuição como um todo tem que se enquadrar nos termos desta Licença, cujas permissões para outros licenciados se estendem ao todo, portanto também para cada e toda parte independente de quem a escreveu.

Desta forma, esta seção não tem a intenção de reclamar direitos ou contestar seus direitos sobre o trabalho escrito completamente por você; ao invés disso, a intenção é a de exercitar o direito de controlar a distribuição de trabalhos, derivados ou coletivos, baseados no Programa.

Adicionalmente, a mera adição ao Programa de outro trabalho não baseado no Programa (ou de trabalho baseado no Programa) em um volume de armazenamento ou meio de distribuição não faz o outro trabalho parte do escopo desta Licença.

3. Você pode copiar e distribuir o Programa (ou trabalho baseado nele, conforme descrito na Seção 2) em código-objeto ou em forma executável sob os termos das Seções 1 e 2 acima, desde que você faça um dos seguintes:
 - (a) O acompanhe com o código-fonte completo e em forma acessível por máquinas, que tem que ser distribuído sob os termos das Seções 1 e 2 acima e em meio normalmente utilizado para o intercâmbio de software; ou,
 - (b) O acompanhe com uma oferta escrita, válida por pelo menos três anos, de fornecer a qualquer um, com um custo não superior ao custo de distribuição física do material, uma cópia do código-fonte completo

e em forma acessível por máquinas, que tem que ser distribuído sob os termos das Seções 1 e 2 acima e em meio normalmente utilizado para o intercâmbio de software; ou,

- (c) O acompanhe com a informação que você recebeu em relação à oferta de distribuição do código-fonte correspondente. (Esta alternativa é permitida somente em distribuição não comerciais, e apenas se você recebeu o programa em forma de código-objeto ou executável, com oferta de acordo com a Subseção b acima.)

O código-fonte de um trabalho corresponde à forma de trabalho preferida para se fazer modificações. Para um trabalho em forma executável, o código-fonte completo significa todo o código-fonte de todos os módulos que ele contém, mais quaisquer arquivos de definição de “interface”, mais os “scripts” utilizados para se controlar a compilação e a instalação do executável. Contudo, como exceção especial, o código-fonte distribuído não precisa incluir qualquer componente normalmente distribuído (tanto em forma original quanto binária) com os maiores componentes (o compilador, o “kernel” etc.) do sistema operacional sob o qual o executável funciona, a menos que o componente em si acompanhe o executável.

Se a distribuição do executável ou código-objeto é feita através da oferta de acesso a cópias de algum lugar, então ofertar o acesso equivalente a cópia, do mesmo lugar, do código-fonte equivale à distribuição do código-fonte, mesmo que terceiros não sejam compelidos a copiar o código-fonte com o código-objeto.

4. Você não pode copiar, modificar, sub-licenciar ou distribuir o Programa, exceto de acordo com as condições expressas nesta Licença. Qualquer outra tentativa de cópia, modificação, sub-licenciamento ou distribuição do Programa não é válida, e cancelará automaticamente os direitos que lhe foram fornecidos por esta Licença. No entanto, terceiros que de você receberam cópias ou direitos, fornecidos sob os termos desta Licença, não terão suas licenças terminadas, desde que permaneçam em total concordância com ela.
5. Você não é obrigado a aceitar esta Licença já que não a assinou. No entanto, nada mais o dará permissão para modificar ou distribuir o Programa ou trabalhos derivados deste. Estas ações são proibidas por lei, caso você não aceite esta Licença. Desta forma, ao modificar ou distribuir o Programa (ou qualquer trabalho derivado do Programa), você estará indicando sua total aceitação desta Licença para fazê-los, e todos os seus termos e condições para copiar, distribuir ou modificar o Programa, ou trabalhos baseados nele.
6. Cada vez que você redistribuir o Programa (ou qualquer trabalho baseado nele), os recebedores adquirirão automaticamente do licenciador original uma licença para copiar, distribuir ou modificar o Programa, sujeitos a estes termos e condições. Você não poderá impor aos recebedores qualquer outra restrição ao exercício dos direitos então adquiridos. Você não é responsável em garantir a concordância de terceiros a esta Licença.
7. Se, em consequência de decisões judiciais ou alegações de infringimento de patentes ou quaisquer outras razões (não limitadas a assuntos relacionados a patentes), condições forem impostas a você (por ordem judicial, acordos ou outras formas) e que contradigam as condições desta Licença, elas não o livram das condições desta Licença. Se você não puder distribuir de forma a satisfazer simultaneamente suas obrigações para com esta Licença e para com as outras obrigações pertinentes, então como consequência você não poderá distribuir o Programa. Por exemplo, se uma licença de patente não permitirá a redistribuição, livre de “royalties”, do Programa, por todos aqueles que receberem cópias direta ou indiretamente de você, então a única forma de você satisfazer a ela e a esta Licença seria a de desistir completamente de distribuir o Programa.

Se qualquer parte desta seção for considerada inválida ou não aplicável em qualquer circunstância particular, o restante da seção se aplica, e a seção como um todo se aplica em outras circunstâncias.

O propósito desta seção não é de induzi-lo a infringir quaisquer patentes ou reivindicação de direitos de propriedade outros, ou a contestar a validade de quaisquer dessas reivindicações; esta seção tem como único propósito proteger a integridade dos sistemas de distribuição de software livres, o que é implementado pela prática de licenças públicas. Várias pessoas têm contribuído generosamente e em grande escala para os software distribuídos usando este sistema, na certeza de que sua aplicação é feita de forma consistente; fica a critério do autor/doador decidir se ele ou ela está disposto a distribuir software utilizando outro sistema, e um licenciado não pode impor esta escolha.

Esta seção destina-se a tornar bastante claro o que se acredita ser consequência do restante desta Licença.

8. Se a distribuição e/ou uso do Programa são restringidos em certos países por patentes ou direitos autorais, o detentor dos direitos autorais original, e que colocou o Programa sob esta Licença, pode incluir uma limitação geográfica de distribuição, excluindo aqueles países de forma a tornar a distribuição permitida apenas

naqueles ou entre aqueles países então não excluídos. Nestes casos, esta Licença incorpora a limitação como se a mesma constasse escrita nesta Licença.

9. A Free Software Foundation (Fundação do Software Livre) pode publicar versões revisadas e/ou novas da Licença Pública Geral de tempos em tempos. Estas novas versões serão similares em espírito à versão atual, mas podem diferir em detalhes que resolvem novos problemas ou situações.

A cada versão é dada um número distinto. Se o Programa especifica um número de versão específico desta Licença que se aplica a ele e a “qualquer nova versão”, você tem a opção de aceitar os termos e condições daquela versão ou de qualquer outra versão publicada pela Free Software Foundation. Se o programa não especifica um número de versão desta Licença, você pode escolher qualquer versão já publicada pela Free Software Foundation.

10. Se você pretende incorporar partes do Programa em outros programas livres cujas condições de distribuição são diferentes, escreva ao autor e solicite permissão. Para o software que a Free Software Foundation detém direitos autorais, escreva à Free Software Foundation; às vezes nós permitimos exceções a este caso. Nossa decisão será guiada pelos dois objetivos de preservar a condição de liberdade de todas as derivações do nosso software livre, e de promover o compartilhamento e reutilização de software em aspectos gerais.

AUSÊNCIA DE GARANTIA

11. PELO PROGRAMA SER LICENCIADO SEM ÔNUS, NÃO HÁ QUALQUER GARANTIA PARA O PROGRAMA, NA EXTENSÃO PERMITIDA PELAS LEIS APLICÁVEIS. EXCETO QUANDO EXPRESSADO DE FORMA ESCRITA, OS DETENTORES DOS DIREITOS AUTORAIS E/OU TERCEIROS DISPONIBILIZAM O PROGRAMA “NO ESTADO”, SEM QUALQUER TIPO DE GARANTIAS, EXPRESSAS OU IMPLÍCITAS, INCLUINDO, MAS NÃO LIMITADO A, AS GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E AS DE ADEQUAÇÃO A QUALQUER PROPÓSITO. O RISCO TOTAL COM A QUALIDADE E DESEMPENHO DO PROGRAMA É SEU. SE O PROGRAMA SE MOSTRAR DEFEITUOSO, VOCÊ ASSUME OS CUSTOS DE TODAS AS MANUTENÇÕES, REPAROS E CORREÇÕES.
12. EM NENHUMA OCASIÃO, A MENOS QUE EXIGIDO PELAS LEIS APLICÁVEIS OU ACORDO ESCRITO, OS DETENTORES DOS DIREITOS AUTORAIS, OU QUALQUER OUTRA PARTE QUE POSSA MODIFICAR E/OU REDISTRIBUIR O PROGRAMA CONFORME PERMITIDO ACIMA, SERÃO RESPONSABILIZADOS POR VOCÊ POR DANOS, INCLUINDO QUALQUER DANO EM GERAL, ESPECIAL, ACIDENTAL OU CONSEQÜENTE, RESULTANTES DO USO OU INCAPACIDADE DE USO DO PROGRAMA (INCLUINDO, MAS NÃO LIMITADO A, A PERDA DE DADOS OU DADOS TORNADOS INCORRETOS, OU PERDAS SOFRIDAS POR VOCÊ OU POR OUTRAS PARTES, OU FALHAS DO PROGRAMA AO OPERAR COM QUALQUER OUTRO PROGRAMA), MESMO QUE TAL DETENTOR OU PARTE TENHAM SIDO AVISADOS DA POSSIBILIDADE DE TAIS DANOS.

QGIS Qt exceções a GPL

Em adição, excepcionalmente, a Equipe de Desenvolvimento QGIS dá a permissão de linkar o código deste programa com a biblioteca QT, inclusa, mas não limitada as seguintes versões (ambas gratuitas e comerciais):Qt / Não-Comercial do Windows, Qt / Windows, Qt/X11, Qt / Mac, e Qt / Embedded (ou com versões modificadas do Qt que usam a mesma licença do Qt) e distribuir combinações relacionadas, incluindo os dois. Você deve obedecer a GNU General Public License em todos os aspectos para todo o código usado em outra Qt. Se você modificar este arquivo, você pode estender essa exceção para a sua versão do arquivo, mas você não é obrigado a fazê-lo. Se você não quiser fazer isso, exclua essa declaração exceção de sua versão.

22.2 GNU Licença de Documentação Gratuita

Versão 1.3, 3 de Novembro de 2008

Direitos Autorais 2000, 2001, 2002, 2007, 2008 Free Software Foundation (Fundação do Software Livre)

<<http://fsf.org/>>

Qualquer pessoa tem permissão para copiar e distribuir cópias fiéis deste documento de licença, mas modificá-lo não é permitido.

Preâmbulo

O objetivo desta licença é produzir um manual, compêndio, ou outro documento funcional e útil. “Grátis” no sentido de liberdade: para garantir a todos a efetiva liberdade de copiar e distribuí-la com ou sem modificações, comercialmente ou não. Secundariamente esta licença preserva ao autor e editor uma maneira de ter crédito pelo seu trabalho, embora não sejam considerados responsáveis por modificações feitas por outros.

Esta licença é uma espécie de “copyleft”, o que significa que trabalhos derivativos do documento devem ser livres no mesmo sentido. Ela complementa a GNU (Licença Pública Geral), que é uma licença copyleft projetada para softwares livres.

Criamos esta Licença para que seja usada em manuais para programas livres, porque programas livres precisam de documentação livre: um programa livre deveria vir com manuais que ofereçam as mesmas liberdades que o programa oferece. Mas esta Licença não está limitada a manuais de programas de computador; ela pode ser usada para qualquer trabalho de texto, independentemente do assunto ou se é publicado como um livro impresso. Nós recomendamos esta Licença principalmente para trabalhos cujo propósito é instrução ou referência.

1. APPLICABILITY AND DEFINITIONS

Esta licença se aplica a qualquer manual ou outro trabalho, em qualquer meio, que contenha uma nota introduzida pelo detentor dos direitos autorais dizendo que o documento pode ser distribuído sob os termos desta. Tal nota garante uma licença mundial, livre de royalties, de duração ilimitada, para usar este trabalho sob as condições aqui colocadas. O “Documento”, abaixo, se refere a qualquer tal manual ou trabalho. Qualquer membro do público é um licenciado, e será tratado por “você”. Você aceita a licença se copiar, modificar ou distribuir o trabalho de um modo que necessite de permissão de acordo com a lei de direitos autorais.

Uma “Versão Modificada” do Documento se refere a qualquer trabalho contendo o Documento ou uma parte deste, quer seja copiado sem modificações, quer com modificações e/ou traduzido para outra língua.

Uma “Seção Secundária” é um apêndice com nome ou uma seção inicial do Documento que trata exclusivamente da relação dos editores ou autores do Documento com seu assunto geral (ou temas relacionados) e não contém nada que possa estar diretamente dentro do assunto geral. Assim, se o Documento é em parte um livro-texto de matemática, uma Seção Secundária não pode explicar nada de matemática. Tal relação pode ser uma conexão histórica com o assunto ou com temas relacionados, ou tratar de questões legais, comerciais, filosóficas, éticas ou políticas com relação a eles.

“Seções Invariantes” são certas Seções Secundárias cujos títulos são designados como sendo de Seções invariantes na nota que afirma que o Documento é publicado sob esta Licença. Se uma seção não se encaixa na definição acima de Secundária, então não se permite que seja designada como Invariante. O Documento pode não conter nenhuma Seção Invariante. Se o documento não identificar quaisquer Seções Invariantes, então não há nenhuma.

“Textos de Capa” são certas passagens de texto que são listada como Textos de Capa Frontal ou Texto de Quarta Capa, na nota que afirma que o Documento é publicado sob esta Licença. Um Texto de Capa Frontal pode ter no máximo 5 palavras, e um Texto de Quarta Capa pode ter no máximo 25 palavras.

Uma cópia “Transparente” do Documento significa uma cópia que pode ser lida pelo computador, representada em um formato cuja especificação esteja disponível ao público geral, que seja apropriada para a imediata revisão do documento usando-se editores de texto genéricos ou (para imagens compostas de pixels) programas gráficos genéricos ou (para desenhos) algum editor de desenhos amplamente disponível, e que seja apropriado para inclusão em formatadores de texto ou para tradução automática para uma variedade de formatos apropriados para inclusão em formatadores de texto. Uma cópia feita em outro formato de arquivo Transparente cuja marcação, ou ausência desta, foi manipulada para impedir ou desencorajar modificação subsequente pelos leitores não é Transparente. Um formato de imagem não é Transparente se usado em lugar de qualquer quantidade substancial de texto. Uma cópia que não é “Transparente” é chamada “Opaca”.

Exemplos de formatos apropriados para cópias Transparentes incluem ASCII puro sem marcação, formato de entrada Texinfo, LaTeX, SGML ou XML usando um DTD publicamente disponível, e HTML padrão simples, PostScript ou PDF projetados para modificação por humanos. Exemplos de formatos de imagem transparentes incluem PNG, XCF e JPG. Formatos Opacos incluem formatos proprietários que podem ser lidos e editados somente por processadores de texto proprietários, SGML ou XML para os quais o DTD e/ou ferramentas de

processamento não são largamente disponibilizadas, e HTML, Postscript ou PDF gerados automaticamente com propósito apenas de saída por alguns processadores de texto.

“Página de Título” significa, para um livro impresso, a própria página do título, além das páginas subseqüentes necessárias para conter, de forma legível, o material que esta Licença requer que apareça na página do título. Para trabalhos em formatos que não têm uma página de título assim, “Página de Título” significa o texto próximo à ocorrência mais proeminente do título do trabalho, precedendo o início do corpo do texto.

O “**publicador**” é qualquer pessoa ou entidade que distribua cópias do Documento para o público.

Uma seção “Intitulada XYZ” significa uma sub-unidade com nome do Documento cujo título ou é precisamente XYZ ou contém XYZ em parênteses seguindo o texto que traduz XYZ em outra língua. (Aqui XYZ representa o nome de uma seção específica mencionado acima, tal como “Agradecimentos”, “Dedicatória”, “Apoio”, ou “Histórico”.) “Preservar o Título” de uma seção assim quando você modifica o Documento significa que ela continua sendo uma seção “Intitulada XYZ” de acordo com esta definição.

O Documento pode incluir Notas de Garantia em seguida à nota que afirma que esta Licença se aplica ao Documento. Estas Notas de Garantia são tidas como inclusas por referência nesta Licença, mas somente com relação às notas de garantia: qualquer outra implicação que estas Notas de Garantia possam ter é anulada e não tem efeito algum no conteúdo desta Licença.

2. VERBATIM COPYING

Você pode copiar e distribuir o Documento em qualquer meio, comercialmente ou não-comercialmente, desde que esta licença, as notas de direitos autorais (copyright), e a nota de licença afirmando que esta Licença se aplica ao Documento sejam reproduzidas em todas as cópias, e que você não inclua outras condições, quaisquer que sejam, às condições desta Licença. Você não pode usar de medidas técnicas para obstruir ou controlar a leitura ou cópia futura das cópias que você fizer ou distribuir. Contudo, você pode aceitar compensação em troca das cópias. Se você distribuir um número suficientemente grande de cópias, você deve também respeitar as condições na seção 3.

Você pode também emprestar cópias, sob as mesmas condições acima mencionadas, e você também as pode mostrar publicamente.

3. COPYING IN QUANTITY

Se você publicar cópias impressas (ou cópias em um meio que normalmente tem capas impressas) do documento, em número maior que 100, e a nota de licença do Documento requer Textos de Capa, você deve encadernar as cópias em capas que carreguem, de forma clara e legível, todos estes Textos de Capa: Textos de Capa Frontal na capa frontal, e Textos de Quarta Capa na quarta capa. Ambas as capas devem também identificar, de forma clara e legível, você como o editor das cópias. A capa frontal deve apresentar o título completo com todas as palavras deste igualmente proeminentes e visíveis. Você pode adicionar outro material nas capas. Cópias com mudanças limitadas às capas, desde que preservando o título do Documento e satisfazendo estas condições, podem ser tratadas como cópias literais em outros aspectos.

Se os textos necessários a qualquer uma das capas são demasiado volumosos para serem incluídos de forma legível, você deve colocar os primeiros listados (quantos couberem razoavelmente) na própria capa, e continuar o resto nas páginas adjacentes.

Se você publicar ou distribuir cópias Opacas do Documento em número maior que 100, você deve ou incluir uma cópia Transparente legível por computador juntamente com cada cópia Opaca, ou dizer em, ou juntamente com, cada cópia Opaca um endereço de rede a partir do qual o público geral possa acessar e obter, usando protocolos de rede públicos padrão, uma cópia Transparente completa do Documento, livre de material adicionado. Se você decidir pela segunda opção, você deve seguir passos razoavelmente prudentes, quando começar a distribuir as cópias Opacas em quantidade, para garantir que esta cópia transparente permanecerá acessível no local indicado por pelo menos um ano após a última vez que você distribuir uma cópia Opaca (diretamente ou através de seus agentes ou distribuidor) desta edição ao público.

É solicitado, mas não exigido, que você contate os autores do Documento muito antes de redistribuir qualquer número grande de cópias, para dar a eles uma chance de lhe fornecer uma versão atualizada do Documento.

4. MODIFICATIONS

Você pode copiar e distribuir uma Versão Modificada do Documento sob as condições das seções 2 e 3 acima, desde que você forneça a Versão Modificada estritamente sob esta Licença, com a Versão Modificada no papel de

Documento, permitindo assim a distribuição e modificação da Versão Modificada a quem quer que possua uma cópia desta. Além disso, você deve executar os seguintes procedimentos na Versão Modificada:

1. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
2. Liste na Página de Título, como autores, uma ou mais pessoas ou entidades responsáveis pela autoria ou modificações na Versão Modificada, juntamente com pelo menos cinco dos autores principais do Documento (todos seus autores principais, se houver menos que cinco), a menos que estes lhe desobriguem desta exigência.
3. Mencione na Página de Título o nome do editor da Versão Modificada, como seu editor.
4. Preserve todas as notas de direitos autorais (copyright) do Documento.
5. Adicione uma nota apropriada de direitos autorais para suas modificações, adjacente às outras notas de direitos autorais.
6. Inclua, imediatamente após as notas de direitos autorais, uma nota de licença dando ao público permissão para usar a Versão Modificada sob os termos desta Licença, na forma mostrada no Adendo abaixo.
7. Preserve naquela nota de licença a lista completa de Seções Invariantes e Textos de Capa requeridos dados na nota de licença do Documento.
8. Inclua uma cópia inalterada desta Licença.
9. Preserve a seção intitulada “Histórico”, preserve seu título, e adicione a esta um item mencionando pelo menos o título, ano, novos autores, e editor da Versão Modificada conforme incluído na Página de Título. Se não houver uma seção intitulada “Histórico” no Documento, crie uma mencionando o título, ano, autores e editor do Documento como mostrado na Página de Título, em seguida adicione um item descrevendo a Versão Modificada como mencionado na sentença anterior.
10. Preserve o endereço de rede, se algum, dado no Documento para acesso público a uma cópia Transparente deste e, da mesma maneira, os endereços de rede dados no Documento para versões prévias nas quais este se baseia. Estes podem ser colocados na seção “Histórico”. Você pode omitir um endereço de rede para um trabalho que foi publicado pelo menos quatro anos antes do Documento em si, ou se o editor original da versão à qual o endereço se refere der permissão.
11. Para qualquer seção intitulada “Agradecimentos” ou “Dedicatória”, preserve o título da seção, e preserve dentro da seção toda a substância e tom de cada um dos agradecimentos e/ou dedicatórias lá mencionados.
12. Preserve todas as Seções Invariantes do Documento, inalteradas no seu texto e títulos. Números de seção ou o equivalente não são considerados parte dos títulos das seções.
13. Apague qualquer seção intitulada “Apoio”. Tal seção não ser incluída na Versão Modificada.
14. Não modifique o título de qualquer seção a ser intitulada “Apoio” ou que resulte em conflito com título de qualquer Seção Invariante.
15. Preserve any Warranty Disclaimers.

Se a Versão Modificada incluir novas seções iniciais ou apêndices que sejam qualificados como Seções Secundárias, e não contiver material copiado do Documento, você pode, a seu critério, tornar algumas dessas ou todas essas seções em invariantes. Para fazer isso, adicione seus títulos à lista de Seções Invariantes na nota de licença da Versão Modificada. Estes títulos devem ser distintos de quaisquer outros títulos de seções.

Você pode incluir uma seção intitulada “Apoio”, dado que ela contenha nada além de apoio recebido para sua Versão Modificada por várias fontes – por exemplo, notas do revisor ou de que o texto foi aprovado por uma organização como a definição autoritativa de um padrão.

Você pode adicionar uma passagem de até cinco palavras como Texto de Capa Frontal, e uma passagem de até 25 palavras como Texto de Quarta Capa, ao fim da lista de Textos de Capa na Versão Modificada. Somente uma passagem de Texto de Capa Frontal e uma de Texto de Quarta Capa pode ser adicionado por (ou através de arranjos feitos por) uma entidade qualquer. Se o Documento já incluir um texto de capa para a mesma capa, previamente incluído por você ou por arranjo feito pela mesma entidade em cujo nome você está agindo, você não pode adicionar outro; mas você pode substituir o antigo, com permissão explícita do editor anterior, que o incluiu.

O(s) autor(es) e editor(es) do Documento, por esta Licença, não dão permissão para seus nomes serem usados para publicidade ou defesa ou apoio implícito para qualquer Versão Modificada.

5. COMBINING DOCUMENTS

Você pode combinar o documento com outros documentos publicados sob esta Licença, sob os termos definidos na seção 4 acima para versões modificadas, desde que você inclua na combinação todas as Seções Invariantes de todos os documentos originais, sem modificações, e as liste como Seções Invariantes de seu trabalho combinado, na sua nota de licença, e que você preserve todas as Notas de Garantia.

O trabalho combinado somente precisa conter uma cópia desta Licença, e múltiplas Seções Invariantes idênticas podem ser substituídas por uma única cópia. Se houver múltiplas Seções Invariantes com o mesmo nome, porém com conteúdos diferentes, torne o título de cada uma destas seções único, adicionando ao fim dele, entre parênteses, o nome do autor ou editor original desta seção, se conhecido, ou então um número único. Faça o mesmo ajuste nos títulos de seção na lista de Seções Invariantes na nota de licença do trabalho combinado.

Na combinação, você deve combinar quaisquer seções intituladas “Histórico” nos vários documentos originais, formando uma seção intitulada “Histórico”; do mesmo modo, combine quaisquer seções intituladas “Agradecimentos”, e quaisquer seções intituladas “Dedicatória”. Você deve apagar todas as seções intituladas “Apoio”.

6. COLLECTIONS OF DOCUMENTS

Você pode fazer uma coleção consistindo do Documento e outros documentos publicados sob esta Licença, e substituir as cópias individuais desta Licença, nos vários documentos, por uma única cópia a ser incluída na coleção, desde que você siga as regras desta Licença para cópias literais de cada documento em todos os outros aspectos.

Você pode extrair um único documento desta coleção, e distribuí-lo individualmente sob esta Licença, desde que você insira uma cópia desta Licença no documento extraído, e siga esta Licença em todos os outros aspectos com relação à cópia literal do documento.

7. AGGREGATION WITH INDEPENDENT WORKS

Uma compilação do Documento ou seus derivados com outros documentos ou trabalhos separados e independentes, dentro de ou junto a um volume de um meio de armazenagem ou distribuição, configura um “agregado” se os direitos autorais resultantes da compilação não forem usados para limitar os direitos legais dos usuários desta além do que os trabalhos individuais permitem. Quando o Documento é incluído em um agregado, esta Licença não se aplica aos outros trabalhos no agregado que não forem, por sua vez, derivados do Documento.

Se o requerimento do Texto de Capa da seção 3 for aplicável a estas cópias do documento, então, se o Documento for menor que metade do agregado inteiro, os Textos de Capa do Documento podem ser colocados em capas que encerrem o Documento dentro do agregado, ou o equivalente eletrônico das capas se o Documento estiver em formato eletrônico. Do contrário, eles devem aparecer como capas impressas que envolvam o agregado inteiro.

8. TRANSLATION

Uma tradução é considerada como sendo um tipo de modificação, então você pode distribuir traduções do Documento sob os termos da seção 4. A substituição de Seções Invariantes por traduções requer permissão especial dos detentores dos direitos autorais, embora você possa incluir traduções de algumas ou todas as Seções Invariantes juntamente às versões originais destas. Você pode incluir uma tradução desta Licença, e todas as notas de licença no Documento, e qualquer Nota de Garantia, desde que você também inclua a versão original em Inglês desta Licença e as versões originais das notas de licença e garantia. Em caso de discordância entre a tradução e a versão original desta Licença ou nota de licença ou garantia, a versão original prevalecerá.

Se uma seção no Documento for intitulada “Agradecimentos”, “Dedicatória”, ou “Histórico”, o requerimento (seção 4) de Preservar seu Título (seção 1) tipicamente exigirá a mudança do título em si.

9. TERMINATION

Você não pode copiar, modifica, sub-licenciar, ou distribuir o Documento à exceção do modo expressamente provido por esta Licença. Qualquer outra tentativa de copiar, modificar, sub-licenciar ou distribuir o Documento é anulada, e implicará em término automático de seus direitos sob esta Licença.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b)

permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Além disso, sua licença de um detentor dos direitos autorais é permanentemente reinstalada se o detentor notificar você da violação por motivos razoáveis, esta é a primeira vez que você recebeu notícias da violação desta licença de qualquer detentor e você remedia a violação anteriormente a 30 dias após ter recebido a notícia.

O término dos seus direitos sobre esta seção não finaliza a licença de partes que receberam cópias de você sobre esta licença. Se seus direitos terminaram e não foram permanentemente reinstalados, o recebimento de uma cópia ou parte do mesmo material não dá a você o direito de usá-lo.

10. FUTURAS REVISÕES DESTA LICENÇA

A Fundação do Software Livre pode publicar novas, revisadas versões do GNU de tempos em tempos. A nova versão será similar a presente versão, mas pode diferir em detalhes para encaminhar novos problemas ou preocupações. Veja <http://www.gnu.org/copyleft/>.

Para cada versão da Licença é dado um número de versão distinto. Se o Documento especificar que uma versão particular desta Licença “ou qualquer versão posterior” se aplica a ele, você tem a opção de seguir os termos e condições tanto daquela versão específica, ou de qualquer versão posterior que tenha sido publicada (não como um rascunho) pela Free Software Foundation. Se o Documento não especificar um número de versão desta Licença, você pode escolher qualquer versão já publicada (não como rascunho) pela Free Software Foundation. Se o Documento especificar que um procurador pode decidir quais versões futuras desta licença pode ser usada, uma afirmação pública do procurador de aceitação de uma versão permanentemente autoriza você a escolher essa versão para o documento.

11. RELICENCIAMENTO

“Massive Multiauthor Collaboration Site” (ou “Site MMC”) significa que qualquer servidor da World Wide Web que publica trabalhos protegidos por direitos autorais e também oferece facilidades importantes para qualquer pessoa para editar essas obras. Um wiki público que qualquer um pode editar, está um exemplo de um servidor. A “Massive Multiauthor Collaboration” (ou “MMC”) contidos no site significa que qualquer conjunto de trabalhos protegidos por direitos autorais, assim, publicado no site da MMC.

“CC-BY-SA” significa Creative Commons Attribution-Share Alike 3.0 publicado pela Creative Commons Corporation, uma corporação sem fins lucrativos com sede principal em San Francisco, Califórnia, bem como futuras versões de copyleft da licença publicada pela mesma organização.

“Incorporar” significa publicar ou republicar um Documento, por inteiro ou em partes, como parte de outro documento.

O MMC é “elegível para um novo licenciamento” se está licenciado sob esta Licença, e se todas as obras que foram publicadas pela primeira vez sob esta licença em outro lugar que este MMC e, posteriormente, incorporadas no todo ou em parte para o MMC, (1) não tinha textos cobertos ou seções invariantes, e (2) foram assim incorporados antes de 1 de novembro de 2008.

O operador de um site MMC pode republicar uma MMC contido no site sob CC-BY-SA no mesmo local, a qualquer momento, antes de 1 de agosto de 2009, desde que o MMC é elegível para um novo licenciamento.

ADDENDUM: Como usar esta Licença em seus documentos

Para usar esta Licença num documento que você escreveu, inclua uma cópia da Licença no documento e ponha as seguintes notas de copyright e licenças logo após a página de título:

Copyright © ANO SEU NOME. É concedida permissão para copiar, distribuir e / ou modificar este documento sob os termos da GNU Free Documentation License, Versão 1.3 ou qualquer versão posterior publicada pela Free Software Foundation; sem Seções Invariantes, sem Textos de Capa Frontal, e sem volta Textos de Capa. Uma cópia da licença é incluída na seção intitulada “GNU Free Documentation License”.

Se você tiver Seções Invariantes, Textos de Capa Frontal e Textos de Capa Traseira, substitua a linha”com ... Textos” por esta:

com as Seções Invariantes sendo LISTA SEUS TÍTULOS, com os Textos de Capa sendo LISTA, e com os textos de Contra-Capa sendo LISTA.

Se você tiver Seções Invariantes, sem Textos de Capa, ou alguma outra combinação dos três, mesclar essas duas alternativas para se adequar a situação.

Se o documento contiver exemplos não triviais de código de programas, nós recomendamos a publicação desses exemplos em paralelo sob a sua escolha de licença de software livre, como a GNU General Public License, para permitir seu uso em software livre.

Referências Bibliográficas e Web

GDAL-SOFTWARE-SUITE. Geospatial data abstraction library. <http://www.gdal.org>, 2013

GRASS-PROJECT. Geographic resource analysis support system. <http://grass.osgeo.org> , 2013.

NETELER, M., AND MITASOVA, H. Open source gis: A grass gis approach, 2008.

OGR-SOFTWARE-SUITE. Geospatial data abstraction library. <http://www.gdal.org/ogr> , 2013.

OPEN-GEOSPATIAL-CONSORTIUM. Web map service (1.1.1) implementation specification. <http://portal.opengeospatial.org>, 2002.

OPEN-GEOSPATIAL-CONSORTIUM. Web map service (1.3.0) implementation specification. <http://portal.opengeospatial.org>, 2004.

POSTGIS-PROJECT. Spatial support for postgresql. <http://postgis.refrations.net/> , 2013.

-
-
- Ações, 103
Ações de atributo, 103
ampliação com a roda do mouse, 31
anotação, 40
apache, 159
apache2, 159
Arc/Info_ASCII_Grid, 137
Arc/Info_Binary_Grid, 137
ArcInfo_Binary_Coverage, 68
arquivo de saída salvo como imagem, 20
Atalhos de teclado, 33
Atlas_Generation, 665
Atração, 116
Attribute_Table, 655
Attribute_Table_Selection, 128
Avoid_Intersections_Of_Polygons, 118
- Barra de Escala
 Map_Scalebar, 650
barra de ferramentas do layout, 29
- cálculo da escala, 32
Caixa de ferramenta, 28
Calculadora_Raster, 147
Calculator_Field, 135
carregando_raster, 137
CAT, 149
Categorizada_Edição, 84
CGI, 158
Common_Gateway_Interface, 158
Complemento_deslocamento, 86
Complementos, 671
Compose_Maps, 633
Composer_Manager, 668
Composer_Template, 634
Compositor de cores, 80
Conteúdo da ajuda, 33
Coordinate_Reference_System, 153
Create_Maps, 633
Create_New_Layers, 125
CSV, 68, 121
Current_Edits, 119
- DB_Manager, 75
Debian_Squeeze, 159
- Delocando_pontos_editados, 86
Derived_Fields, 135
desenhar durante a atualização da edição, 35
Digitalização, 118
documentação, 7
documento da licença, 725
- edição de qualidade, 35
Edição_baseada-Regra, 86
Edição_Categorizada, 84
Edição_Graduada, 84
Edição_Símbolo_Simples, 83
Editando_pontos_deslocados, 86
Editar escala dependente, 34
Elements_Alignment, 662
Enquadramento do Mapa, 45
EPSG, 57
Erdas Imagine, 137
Escala, 34
Espacial MSSQL, 75
Espacial Oracle, 75
ESRI, 65
European_Petroleum_Search_Group, 57
Export_as_image, 667
Export_as_PDF, 667
Export_as_SVG, 667
Expressões, 109
- FastCGI, 158
favoritos, 41
favoritos espaciais
 ver favoritos, 41
Ferramenta_Consulta, 133
Ferramentas de análise, 689
Ferramentas de pesquisa, 689
Ferramentas do Georreferenciador, 695
Ferramentas GRASS, 184
 Buscador, 191
 customizar, 192
Field_Calculator, 135
Formato Tiger (Codificação Geográfica Topologica-
mente Referenciada e Integrada), 68
Funções_Calculadora_Campo, 110
- GDAL, 137
GeoTIFF, 137
-

- GML, 149
- Grade
 - Grides
 - Map_Grid, 641
- Graduada_Edição, 84
- GRASS, 175, *veja* Criando novos vetores;editando;criando uma nova camada
 - armazenamento de atributos, 180
 - configurando categorias, 181
 - configurando simbologias, 182
 - editando tabela, 182
 - ferramentas, 188
 - ferramentas de digitalização, 180
 - lincagem de atributos, 180
 - região, 184
 - região editada, 184
 - região visualizada, 184
 - resultados na tela, 186, 187
 - tolerância de aproximação, 182
- Histograma, 145
- HTML_Frame, 660
- Identificar feições, 37
- IGNF, 57
- Igual_Intervalo, 84
- Importar Mapas, 63
- impressão rápida do compositor de impressão, 20
- Imprimindo
 - Export_Map, 667
- Institut_Geographique_National_de_France, 57
- InteProxy, 157
- Interpolação_de_cores, 141
- janela principal, 21
- juntar, 106
- juntar camada, 106
- Layout_Maps, 633
- legenda, 29
- Método Discreto, 141
- Map_Legend, 647
- Map_Navigation, 117
- Map_Template, 634
- Mapa de cores, 141
- MapInfo, 68
- medição, 35
 - áreas, 35
 - ângulos, 35
 - comprimento de linha, 35
- Melhora_do_contraste, 141
- menus, 22
- Merge_Attributes_of_Selected_Features, 125
- Merge_Selected_Features, 125
- Metadados, 145
- Movimentar, 117
- Muitas_Quebras, 84
- Nós, 120
- New_GPX_Layer, 125, 128
- New_Shapefile_Layer, 125
- New_SpatialLite_Layer, 125
- New_Spatialite_Layer, 126
- Node_Tool, 120
- Non_Spatial_Attribute_Tables, 130
- OGC, 149
- OGR, 65
- OGR Simple Feature Library, 65
- ogr2ogr, 73
- opções da linha de comandos, 17
- Open_Geospatial_Consortium, 149
- OpenStreetMap, 70
- OSM, 70
- Pesquisador de Mapas, 63
- pgsql2shp, 73
- Picture_database, 645
- Pirâmides , 144
- PostGIS, 70
- PostgreSQL, 70
- print_composer
 - ferramentas, 633
- Proj.4, 60
- Proj4, 59
- Proj4_texto, 59
- Projeções, 57
- projetos de assentamento, 43
- Proxy, 151
- proxy-server, 151
- QGIS_mapserver, 157
- QGIS_Server, 158
- QSpatialLite, 75
- Quartil, 84
- Quebra_natural_(Jenks), 84
- Rótulos_colidindo , 92
- Rampa_cor_Gridente, 80
- Rampa_cor_personalizada, 80
- Rampa_cores, 80
- Raster, 137
- Raster_banda_única, 139
- Raster_Color_Três_Bandas , 139
- Raster_multibandas, 139
- Regra-baseada_Edição, 86
- Relações, 130
- Rendering_Mode, 638
- Renderização, 33
- Renderizar_Polígono_Invertido, 87
- Revert_Layout_Actions, 663
- Rotate_Point_symbols, 125
- Rotated_North_Arrow, 645
- Símbolo_Simples_Edição, 83
- Search_Radius, 116
- Secured_OGC_Authentication, 157
- Selecionar_usando_Consulta, 135

Serviço de Cobertura Web, 157
 SFS, 149
 Shapefile, 65
 Shapefile_to_Postgis_Import_Tool, 714
 Shared_Polygon_Boundaries, 118
 shp2pgsql, 72
 Simbologia, 91, 139
 Sistema_de_Referência_de_Coordenadas, 57
 SLD, 158
 SLD/SE, 158
 Snapping_On_Intersections, 118
 Snapping_Tolerance, 116
 Spatialite, 75
 Spatialite_Manager, 75
 SPIT, 714
 Split_Features, 124
 SQLite, 75
 SRC, 57, 153
 SRC_padrão, 57
 SRC_personalizado, 60
 SRS, 153
 ST_Shift_Longitude, 74
 Suspend edição, 34

 teclas de direções para mover, 32
 Topological_Editing, 118
 Transformação_datum, 61
 Transparência, 143

 UK_National_Transfer_Format, 68
 US_Census_Bureau, 68

 Vértice, 120
 Vértices, 120
 Valores Separados por Vírgula, 68
 visibilidade da camada, 29

 WCS, 149, 157
 WFS, 149, 157
 WFS-T, 157
 WFS_Transactional, 157
 WKT, 57, 121
 WMS, 149
 WMS-C, 154
 WMS_1.3.0, 157
 WMS_client, 149
 WMS_identify, 155
 WMS_layer_transparency, 153
 WMS_metadata, 155
 WMS_properties, 155
 WMS_tiles, 154
 WMTS, 154
 WMTS_client, 149
 Work_with_Attribute_Table, 128

 Zoom_In Zoom_Out, 117