

# Quantum GIS User Guide

Version 0.5 'Bandit'

Gary E. Sherman  
Tim Sutton (Rasters)  
Radim Blazek (GRASS)

October 1, 2004

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Whats New in 0.5</b>	<b>3</b>
<b>3</b>	<b>Major Features</b>	<b>3</b>
<b>4</b>	<b>Getting Started</b>	<b>3</b>
4.1	Installation	3
4.2	Sample Data	4
4.3	Starting QGIS	4
4.4	The QGIS Main Window	4
4.4.1	The QGIS menu bar	4
4.4.2	Toolbars	4
4.4.3	The QGIS map legend	5
4.4.4	The QGIS map view	5
4.4.5	The QGIS map overview	6
4.4.6	The QGIS map status bar	6
<b>5</b>	<b>Working with Vector Data</b>	<b>6</b>
5.1	Shapefiles	6
5.1.1	Loading a Shapefile	6
5.2	PostGIS Layers	8
5.2.1	Creating a Stored Connection	8
5.2.2	Loading a PostGIS Layer	8
5.2.3	Importing Data into PostgreSQL	9
5.3	The Vector Properties Dialog	9
5.3.1	Vector Symbology	9
<b>6</b>	<b>Working with Raster Data</b>	<b>10</b>
6.1	What is raster data?	10
6.2	Raster formats supported in QGIS	10
6.3	Loading raster data in QGIS	10
6.4	Raster Properties	11
6.4.1	Symbology Tab	12
6.4.2	General Tab	12
6.4.3	Metadata Tab	12
6.4.4	Pyramids Tab	13

<b>7 GRASS</b>	<b>13</b>
7.1 Starting QGIS with GRASS	13
7.1.1 From GRASS shell	13
7.1.2 Outside GRASS shell	13
7.2 Loading GRASS Data	14
7.3 Vector Data Model	14
7.4 Digitizing and Editing Tools	14
7.4.1 Toolbar	15
7.4.2 Category Tab	15
7.4.3 Settings Tab	16
7.4.4 Symbology Tab	16
7.4.5 Table	16
7.4.6 Region Tool	16
<b>8 Using Plugins</b>	<b>16</b>
8.1 Finding and Installing a Plugin	17
8.2 Managing Plugins	17
8.3 Data Providers	18
8.4 Core Plugins	18
<b>A Supported OGR Formats</b>	<b>19</b>
<b>B GDAL Raster Formats</b>	<b>19</b>

## QGIS Tips

1 USING COMMAND LINE ARGUMENTS	4
2 VIEWING THE LAYER MENU	5
3 ZOOMING THE MAP WITH THE MOUSE WHEEL	6
4 LAYER COLORS	7
5 QGIS USER SETTINGS AND SECURITY	8
6 USING A DEFINITION QUERY WITH POSTGIS LAYERS	9
7 IMPORTING SHAPEFILES CONTAINING POSTGRES SQL RESERVED WORDS	9
8 GATHERING RASTER STATISTICS	12
9 GRASS DATA LOADING	14
10 LEARNING THE GRASS VECTOR MODEL	14
11 GRASS EDIT PERMISSIONS	16
12 CRASHING PLUGINS	18

## 1 Introduction

Quantum GIS (QGIS) is designed to be a Geographic Information System (GIS) built for Linux/Unix, Mac OS X, and Windows. QGIS currently offers basic support for vector (both disk-based and database) and raster formats.

QGIS is released under the GNU Public License. You should have received a full copy of the license with your copy of QGIS.

*Note - The latest version of this document can always be found at <http://qgis.sourceforge.net/docs/userguide.html>*

## 2 Whats New in 0.5

New features in version 0.5 include:

1. Windows version
2. Feature labeling with optional buffering
3. Preliminary digitizing support for shapefiles
4. GRASS digitizing support
5. GPS SVG icons
6. Unique value renderers
7. User interface improvements
8. Specify a query when loading a PostGIS layer
9. Italian translation

## 3 Major Features

QGIS has many common GIS features and functions. The major features are listed below.

1. Support for spatially enabled PostgreSQL tables using PostGIS
2. Support for ESRI shapefiles and other vector formats support by the OGR library, including MapInfo files
3. Identify features
4. Display attribute table
5. Select features
6. Label features
7. Persistent selections
8. Save and restore projects
9. Support for raster formats supported by the GDAL library
10. Change vector symbology (single, graduated, unique value, and continuous)
11. SVG markers symbology (single, unique value, and graduated)
12. Display raster data such as digital elevation models, aerial photography or landsat imagery
13. Change raster symbology (grayscale, pseudocolor and multiband RGB)
14. Export to Mapserver map file
15. Preliminary digitizing support
16. Map overview
17. Plugins

## 4 Getting Started

This section gives you a quick overview of running QGIS and examining data available on the QGIS web page.

### 4.1 Installation

Installation of QGIS is documented in the Installation Guide. The Installation Guide is distributed with the QGIS source code and is also available at <http://qgis.org>.

## 4.2 Sample Data

If you do not have any GIS data handy, you can obtain a dataset for Alaska from the QGIS web site at <http://qgis.org>. The Alaska data set will be used as the basis for the examples and screenshots provided in this document.

## 4.3 Starting QGIS

Assuming that QGIS is installed in the PATH, you can start QGIS by typing: `qgis` at a command prompt or by double clicking on the QGIS application link (or shortcut) on the desktop.

---

**Tip 1** USING COMMAND LINE ARGUMENTS

---

You can also start QGIS by specifying one or more datafiles on the command line. For example, assuming you are in your data directory, you could start QGIS with two shapefiles and a raster file set to load on startup using the following command: `qgis ak_shade.tif alaska.shp majrivers.shp`

---

## 4.4 The QGIS Main Window

When QGIS starts, an empty window is displayed as shown below.

textscNote - Your window decorations (title bar, etc.) may appear different depending on your operating system and window manager The QGIS main window is divided into five areas:

1. The menu bar
2. The tool bar
3. The map legend
4. The map view
5. The map overview
6. The status bar

These six components of the QGIS interface are described in more detail in the following sections

### 4.4.1 The QGIS menu bar

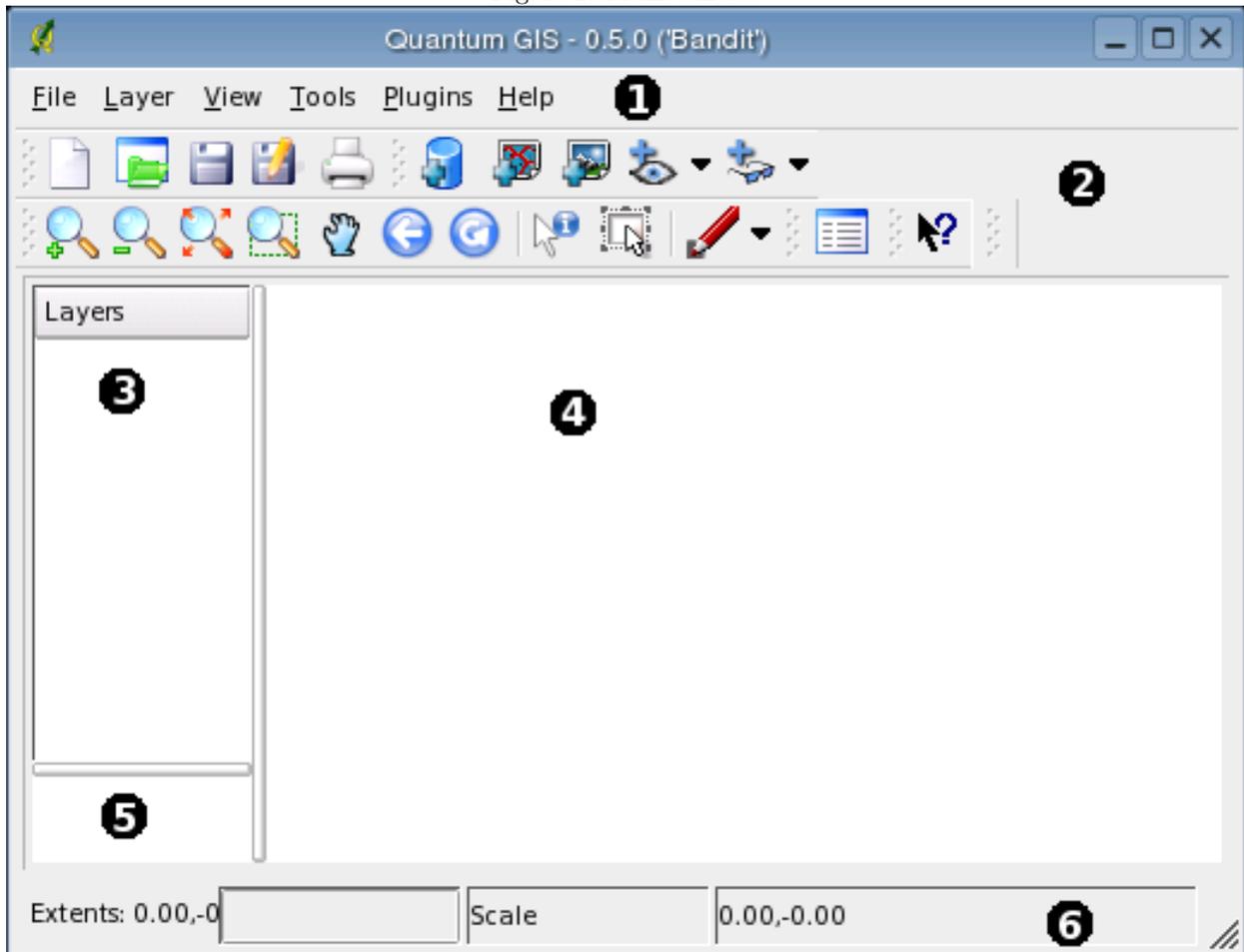
The menu bar provides access to various QGIS features using a standard windows heirarchical menu. The top-level menus and a summary of some of the functions provided are:

- File (project open, save, export image, properties)
- Layer (add, show, hide layers)
- View (zoom, refresh)
- Tools (plugin manager, preferences)
- Plugins (menus added by plugins as they are loaded)
- Help (documentation and web links)

### 4.4.2 Toolbars

The toolbars provide access to most of the same functions as the menus, plus additional tools for interacting with the map. Each toolbar item has popup help available. Hold your mouse over the item and a short description of the tool's purpose will be displayed.

Figure 1: Main window



#### 4.4.3 The QGIS map legend

The map legend area is used to set the visibility and z-ordering of layers. Z-ordering means that layers listed nearer the top of the legend are drawn over layers listed lower down in the legend. The checkbox in each legend entry can be used to show/hide that layer.

---

#### **Tip 2** VIEWING THE LAYER MENU

You can display the context menu for any layer in the legend by right-clicking on the layer name. The context menu contains items for working with the layer and viewing its properties.

---

#### 4.4.4 The QGIS map view

This is the 'business end' of QGIS - maps are displayed in this area! The map displayed in this window will depend on the vector and raster layers you have chosen to load (see sections that follow for more info on this). The map view can be panned (shifting to focus of the map display to another region), zoomed in and out, and supports various other actions as described in the toolbar description above. The map view and the legend are tightly bound to each other - the maps in view reflect changes you make in the legend area.

---

**Tip 3 ZOOMING THE MAP WITH THE MOUSE WHEEL**

---

You can use the mouse wheel to zoom in and out on the map. Place the mouse cursor inside the map area and roll it forward (away from you) to zoom in and backwards (towards you) to zoom out.

---

#### 4.4.5 The QGIS map overview

The map overview area provides a full extent view of layers added to it. Within the view is a rectangle showing the current map extent. This allows you to quickly determine which area of the map you are currently viewing.

#### 4.4.6 The QGIS map status bar

The status bar shows you your current position in map coordinate (e.g. meters or decimal degrees) as the mouse pointer is moved across the map view.

## 5 Working with Vector Data

QGIS supports vector data in a number of formats, including shapefiles, MapInfo mif, and PostGIS layers in a PostgreSQL database. Support for additional data types is provided by plugins, for example delimited text.

This section describes how to work with two common formats: shapefiles and PostGIS layers. Many of the features available in QGIS work the same regardless of the vector data source. This is by design and includes the identify, select, labeling, and attributes functions.

### 5.1 Shapefiles

Shapefile support is provided by a library of functions (OGR <http://www.remotesensing.org/gdal/ogr>). See Appendix A for a list of supported formats.

A shapefile actually consists of a minimum of three files:

1. .shp file containing the feature geometries
2. .dbf file containing the attributes in dBase format
3. .shx index file

The technical specification for the shapefile format can be found at <http://www.esri.com/software/opengis/openpdf.html>.

#### 5.1.1 Loading a Shapefile



To load a shapefile, start QGIS and click on the *Add a vector layer* toolbar button. This same tool can be used to load any of the formats supported by the OGR library.

Clicking on the tool brings up a standard open file dialog (Figure 2) which allows you to navigate the file system and load a shapefile (or other supported data source). Selecting a shapefile from the list and clicking Ok loads it into QGIS. Figure 3 shows QGIS after loading the alaska.shp file.

Figure 2: Open OGR Data Source Dialog

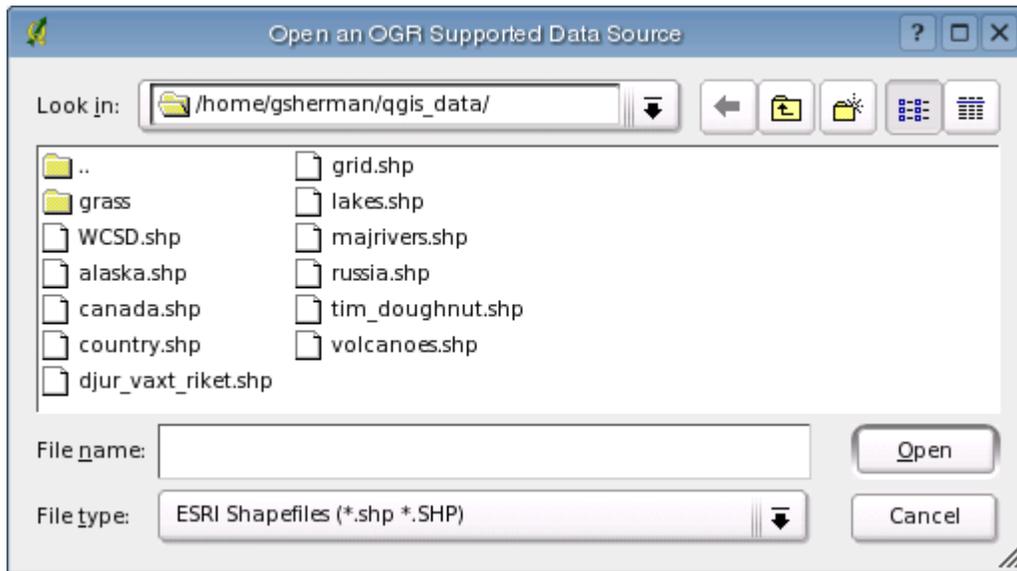
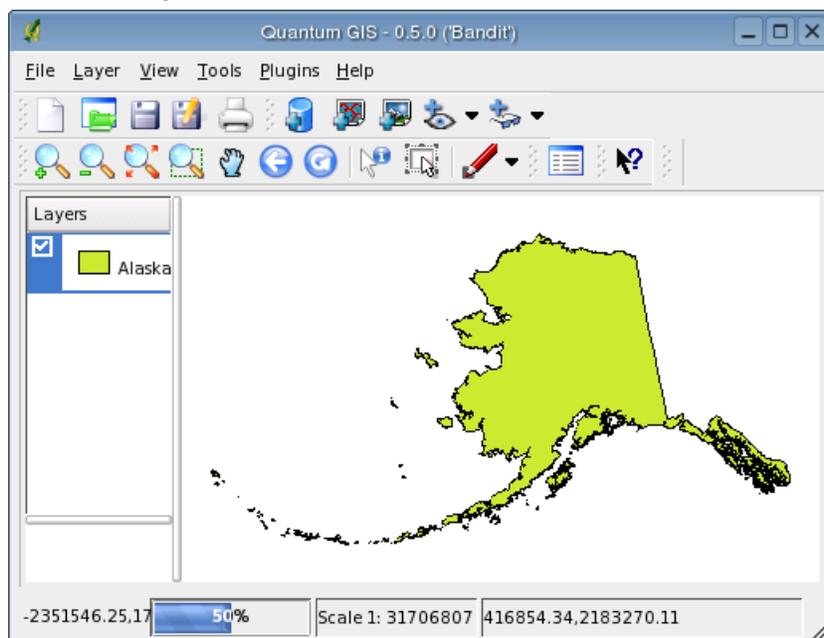


Figure 3: QGIS with the Alaska Shapefile Loaded




---

**Tip 4 LAYER COLORS**

When you add a layer to the map, it is assigned a random color. When adding more than one layer at a time, different colors are assigned to each.

---

Once loaded, you can zoom around the shapefile using the map navigation tools. To change the symbology of a layer, open the layer properties dialog by double clicking on the layer name or by right-clicking on the name in the legend and choosing *Properties* from the popup menu. See Section 5.3.1 for more information

on setting symbology of vector layers.

## 5.2 PostGIS Layers

PostGIS layers are stored in a PostgreSQL database. The advantage of PostGIS is the spatial indexing, filtering, and query capability. Using PostGIS, vector functions such as select and identify work more accurately than with OGR layers in QGIS. To use PostGIS layers you must:

1. Create a stored connection in QGIS to the PostgreSQL database (if one is not already defined)
2. Connect to the database
3. Select the layer to add to the map
4. Optionally provide a SQL where clause to define which features to load from the layer
5. Load the layer

### 5.2.1 Creating a Stored Connection



The first time you use a PostGIS data source, you must create a connection to the PostgreSQL database that contains the data. Begin by clicking on the *Add a PostGIS Layer* toolbar button. The *Add PostGIS Table(s)* dialog will be displayed. To access the connection manager, click on the *New* button to display the *Create a New PostGIS Connection* dialog. The parameters required for a connection are shown in Table 1. Once the parameters have been filled in, you can test the connection by clicking on the

Table 1: PostGIS Connection Parameters

Name	A name for this connection. Can be the same as <i>Database</i>
Host	Name of the database host. This must be a resolvable host name the same as would be used to open a telnet connection or ping the host
Database	Name of the database
Port	Port number the PostgreSQL database server listens on. The default port is 5432.
Username	User name used to login to the database
Password	password used with <i>Username</i> to connect to the database

*Test Connection* button. To save the password with the connection information, check the *Save Password* option.

---

#### Tip 5 QGIS USER SETTINGS AND SECURITY

Your customized settings for QGIS are stored based on the operating system. On Linux/Unix, the settings are stored in your home directory in `.qt/qgisrc`. On Windows, the settings are stored in the registry. Depending on your computing environment, storing passwords in your QGIS settings may be a security risk.

---

### 5.2.2 Loading a PostGIS Layer



Once you have one or more connections defined, you can load layers from the PostgreSQL database. Of course this requires having data in PostgreSQL. See Section 5.2.3 for a discussion on importing data into the database.

To load a layer from PostGIS, perform the following steps:

1. If the PostGIS layer dialog is not already open, click on the *Add a PostGIS Layer* toolbar button

2. Choose the connection from the drop-down list and click *Connect*
3. Find the layer you wish to add in the list of available layers
4. Select it by clicking on it. You can select multiple layers by holding down the shift key while clicking
5. Click on the *Add* button to add the layer to the map

---

**Tip 6 USING A DEFINITION QUERY WITH POSTGIS LAYERS**

---

If you wish to further define the features added from a PostGIS layer, double-click on the layer to open the SQL editor. Enter a conditional statement to be used to select features from the layer for display. For example, if you have a towns layer with a population field you could select only larger towns by entering *population > 100000*. Currently the SQL editor does not provide a list of field names to aid you in forming the conditional statement. The editor will be improved in a future release. Field names and values can be examined by loading the layer without a conditional statement and using the identify tool or by opening the attribute table.

---

### 5.2.3 Importing Data into PostgreSQL

Data can be imported into PostgreSQL using a number of methods. PostGIS includes a utility called *shp2pgsql* that can be used to import shapefiles into a PostGIS enabled database.



QGIS comes with a plugin named SPIT (Shapefile to PostGIS Import Tool). SPIT can be used to load multiple shapefiles at one time and includes support for schemas. To use SPIT, open the Plugin Manager from the Tools menu and load the plugin by checking the box next to the SPIT plugin and click Ok. The SPIT icon will be added to the plugin toolbar.

To import a shapefile, click on the SPIT tool in the toolbar to open the dialog. You can add one or more files to the queue by clicking on the *Add* button. To process the files, click on the *Import* button. The progress of the import as well as any errors/warnings will be displayed as each shapefile is processed.

---

**Tip 7 IMPORTING SHAPEFILES CONTAINING POSTGRESQL RESERVED WORDS**

---

SPIT does not handle importing of shapefiles that contain field names that are reserved words in the PostgreSQL database. This issue will be addressed in a future release of the plugin

---

## 5.3 The Vector Properties Dialog

The vector properties dialog provides information about a layer, symbology settings, and labeling options. To access the properties dialog, double-click on a layer in the legend or right-click on the layer and select Properties from the popup menu.

### 5.3.1 Vector Symbology

QGIS supports a number of symbology renderers to control how vector features are displayed. Currently the following renderers are available:

**Single symbol** - a single style is applied to every object in the layer.

**Graduated symbol** - objects within the layer are displayed with different symbols classified by the values of a particular field.

**Continuous colour** - objects within the layer are displayed with a spread of colours classified by the numerical values within a specified field.

**Unique value** - objects are classified by the unique values within a specified field with each value having a different symbol.

For layers containing point features, additional renderers are available that use SVG icons:

**Single marker** - a single specified icon is used for every point within the layer.

**Graduated marker** - points within the layer are displayed with different icons classified by values within a particular field

**Unique value marker** - points are classified by unique values within a specified field with each value having a different icon.

## 6 Working with Raster Data

QGIS supports a number of raster data formats. This section describes how to work with raster data in QGIS.

### 6.1 What is raster data?

Raster data in GIS are matrices of discrete cells that represent features on, above or below the earth's surface. Each cell in the raster grid is the same size, and cells are usually rectangular (in QGIS they will always be rectangular). Typical raster datasets include remote sensing data such as aerial photography or satellite imagery and modelled data such as an elevation matrix.

Unlike vector data, raster data typically do not have an associated database record for each cell.

In GIS, a raster layer would have georeferencing data associated with it which will allow it to be positioned correctly in the map display to allow other vector and raster data to be overlaid with it. QGIS makes use of georeferenced rasters to properly display the data.

### 6.2 Raster formats supported in QGIS

QGIS supports a number of different raster formats. Currently tested formats include:

- Arc/Info Binary Grid
- Arc/Info ASCII Grid
- Grass Raster
- GeoTIFF
- Spatial Data Transfer Standard Grids (with some limitations)
- USGS ASCII DEM
- Erdas Imagine

Because the raster implementation in QGIS is based on the GDAL library, other raster formats implemented in GDAL are also likely to work, but have not yet been tested. See Appendix B for more details.

### 6.3 Loading raster data in QGIS



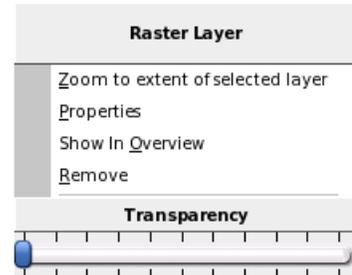
Raster layers are loaded either by clicking on the Load Raster icon or by selecting the View->Add Raster Layer menu option. More than one layer can be loaded at the same time by holding down the

Control key and clicking on multiple items in the file dialog.

## 6.4 Raster Properties

To view and set the properties for a raster layer, right click on the layer name. This displays the raster layer context menu that includes a number of items that allow you to:

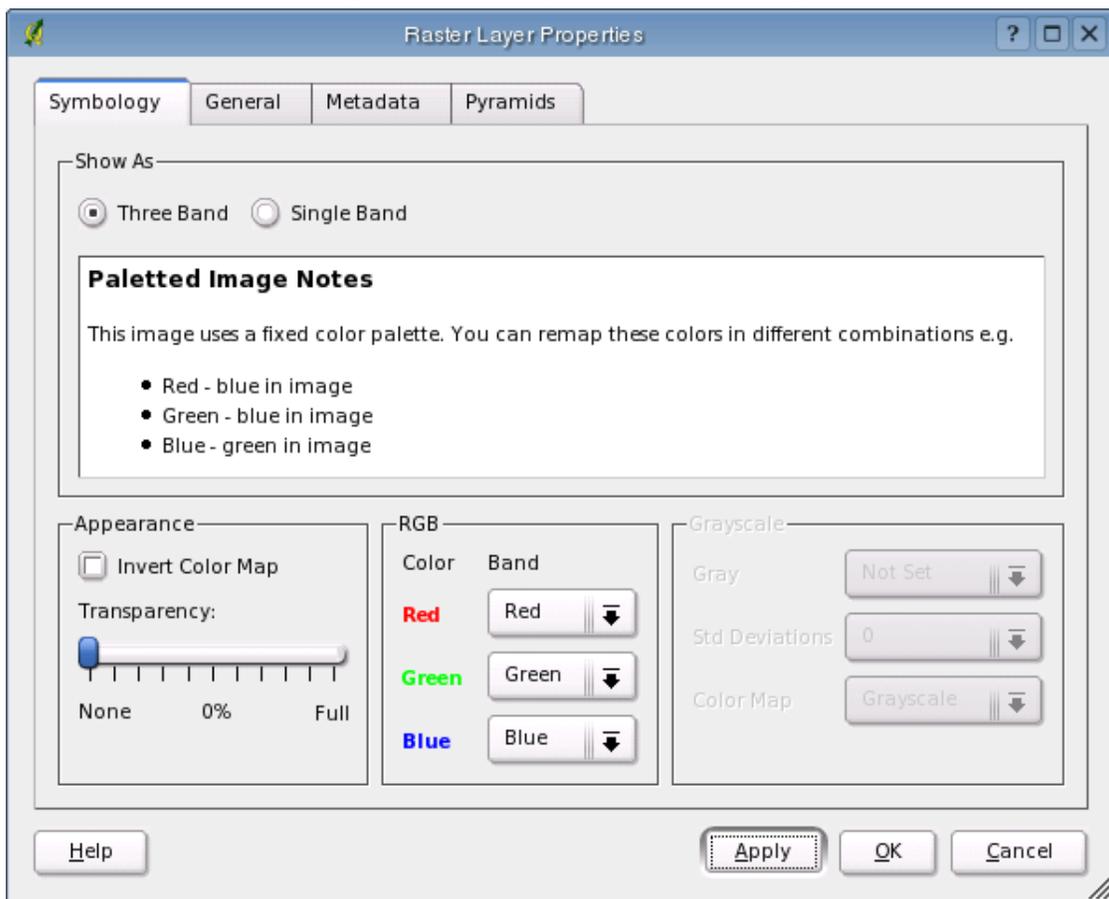
- Zoom to the full extent of the raster
- Show the raster in the map overview window
- Open the properties dialog (of course)
- Remove the layer from the map
- Set the transparency using a slider control



Choose *Properties* from the context menu to open the raster properties dialog for the layer.

Figure 4 shows the properties dialog. There are four tabs on the dialog: *Symbology*, *General*, *Metadata*, and *Pyramids*.

Figure 4: Raster Layers Properties Dialog



### 6.4.1 Symbology Tab

QGIS supports three forms of raster layers:

- Single Band Grayscale Rasters
- Palette Based RGB Rasters
- Multiband RGB Rasters

From these three basic layer types, eight forms of symbolised raster display can be used:

1. Single Band Grayscale
2. Single Band Pseudocolor
3. Paletted Grayscale (where only the red, green or blue component of the image is displayed)
4. Paletted Pseudocolor (where only the red, green or blue component of the image is displayed, but using a pseudocolor algorithm)
5. Paletted RGB
6. Multiband Grayscale (using only one of the bands to display the image)
7. Multiband Pseudocolor (using only one of the bands shown in pseudocolor)
8. Multiband RGB (using any combination of three bands)

QGIS can invert the colors in a given layer so that light colors become dark (and dark colors become light). Use the *Invert Color Map* checkbox to enable / disable this behavior.

QGIS has the ability to display each raster layer at varying transparency levels. Use the transparency slider to indicate to what extent the underlying layers (if any) should be visible through the current raster layer. The transparency can also be set using the transparency slider in the layer context menu which is accessible by right-clicking on the layer in the legend.

QGIS can restrict the data displayed to only show cells whose values are within a given number of standard deviations of the mean for the layer. This is useful when you have one or two cells with abnormally high values in a raster grid that are having a negative impact on the rendering of the raster. This option is only available for pseudocolor images.

### 6.4.2 General Tab

The General tab displays basic information about the selected raster, including the layer source and display name in the legend (which can be modified). This tab also shows a thumbnail of the layer, its legend symbol, and the palette.

### 6.4.3 Metadata Tab

The Metadata tab displays a wealth of information about the raster layer, including statistics about each band in the current raster layer. Statistics are gathered on a 'need to know' basis, so it may well be that a given layer's statistics have not yet been collected.

---

#### **Tip 8** GATHERING RASTER STATISTICS

To gather statistics for a layer, select pseudocolor rendering and click the *Apply* button. Gathering statistics for a layer can be time consuming. Please be patient while QGIS examines your data!

---

### 6.4.4 Pyramids Tab

Large resolution raster layers can slow navigation in QGIS. By creating lower resolution copies of the data (pyramids), performance can be considerably improved as QGIS selects the most suitable resolution to use depending on the level of zoom.

You must have write access in the directory where the original data is stored to build pyramids.

Please note that building pyramids may alter the original data file and once created they cannot be removed. If you wish to preserve a 'non-pyramided' version of your raster, make a backup copy prior to building pyramids.

## 7 GRASS

The GRASS plugin adds the following features to QGIS:

- Add GRASS vector layers
- Add GRASS raster layers
- Vector layers digitizing
- Changing of the GRASS region

### 7.1 Starting QGIS with GRASS

When using the GRASS plugin, QGIS can be started in two ways: from the GRASS shell or from a regular shell.

#### 7.1.1 From GRASS shell

If QGIS is started from the GRASS shell (GRASS started by `grass57` command), no additional settings are required.

#### 7.1.2 Outside GRASS shell

If QGIS is not started from the GRASS shell, the environment variables must be properly set before starting QGIS.

The path to GRASS libraries must be added to `LD_LIBRARY_PATH` environment variable. For example (in bash):

```
export LD_LIBRARY_PATH=/usr1/grass57/dist.i686-pc-linux-gnu/lib:$LD_LIBRARY_PATH
```

The `GISBASE` environment variable must be set to the full path of the directory where GRASS is installed (the same as used for `-with-grass=` option). For example (in bash):

```
export GISBASE=/usr1/grass57/dist.i686-pc-linux-gnu
```

## 7.2 Loading GRASS Data

With the GRASS plugin loaded, you can load a vector or raster layer using the appropriate button on the toolbar.

---

**Tip 9** GRASS DATA LOADING

---

If you have problems loading data or QGIS terminates abnormally, check to make sure you have started GRASS properly as described in Section 7.1.

---

## 7.3 Vector Data Model

It is important to understand the GRASS vector data model prior to digitizing. In general, GRASS uses a topological vector model. This means that areas are not represented as closed polygons, but by one or more boundaries. A boundary between two adjacent areas is digitized only once, and it is shared by both areas. Boundaries must be connected without gaps. An area is identified (labeled) by the centroid of the area.

Besides boundaries and centroids, a vector map can also contain points and lines. All these geometry elements can be mixed in one vector.

It is possible to store more 'layers' in one vector dataset. For example, fields, forests and lakes can be stored in one vector. Adjacent forest and lake can share the same boundary, but they have separate attribute tables. It is also possible to attach attributes to boundaries. For example, the boundary between lake and forest is a road with different attribute table.

The 'layer' of the feature is defined by 'field' (sorry for this name). 'Field' is the number which defines if the geometry is forest or lake. For now, it can be only a number, in the future GRASS will also support names as fields in the user interface.

Attributes are stored in external database tables, for example DBF, PostgreSQL, etc.

Attributes in database tables are linked to geometry elements using 'category'. 'Category' (key, ID) is an integer attached to geometry primitives, and it is used as the link to one column in the database table.

---

**Tip 10** LEARNING THE GRASS VECTOR MODEL

---

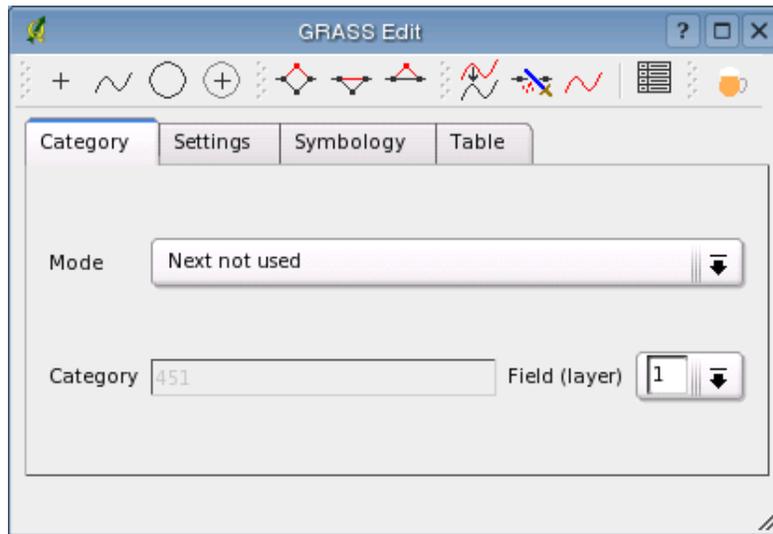
The best way to learn the GRASS vector model and its capabilities is to download the demo mapset from <http://mpa.itc.it/radim/g51/g51test-12-multi.tar.gz>. Extract the mapset, add all layers from vector 'multi' to QGIS, and query attributes. Finally start editing of vector 'multi', to see how those layers are stored.

---

## 7.4 Digitizing and Editing Tools

The digitizing tools for GRASS vector layers are accessed using the *Edit GRASS Vector Layer* tool on the toolbar. Make sure you have loaded a GRASS vector and it is the selected layer in the legend before clicking on the edit tool. In this release, the vector must exist prior to beginning to edit. The ability to create a new "empty" layer will be added in a future version. Figure 5 shows the GRASS Edit dialog that is displayed when you click on the edit tool. The tools and settings are discussed in the following sections.

Figure 5: GRASS Edit Dialog



### 7.4.1 Toolbar

Table 2 lists the digitizing tools provided by the GRASS plugin. These correspond to the tool buttons in the toolbar(s) across the top of the dialog.

Table 2: GRASS Digitizing Tools

Tool	Purpose
New Point	digitize new point
New Line	digitize new line (finish by selecting new tool)
New Boundary	digitize new boundary (finish by selecting new tool)
New Centroid	digitize new centroid (label existing area)
Move vertex	select one vertex of existing line or boundary and identify new position
Add vertex	add a new vertex to existing line
Delete vertex	delete one vertex from existing line (confirm selected vertex by another click)
Move line	select existing line and click on new position
Split line	split an existing line to 2 parts
Delete line	delete existing line (confirm selected line by another click)
Edit attributes	edit attributes of existing element (note that one element can represent more features, see above)
Mug	close digitizing session

### 7.4.2 Category Tab

This tab allows you to set the way in which the category will be assigned to each new feature and/or assign a category to a feature.

- Mode: what category should be attached to geometry
  - Next not used - next category not yet used in vector

- Manual entry - define the category in 'Category entry'
- No category - digitize geometry without category
- Category - a number (ID) attached to digitized feature
- Field - feature (attribute table) identification

### 7.4.3 Settings Tab

This tab allow you to set the snapping in screen pixels. This is the threshold in pixels in which new points or line ends are snapped to existing nodes. This helps prevent gaps or dangles between boundaries

### 7.4.4 Symbology Tab

This tab allows you to view and set symbology for various geometry types and their topological status (e.g. closed / opened boundary).

### 7.4.5 Table

This tab provides the means to view, create, or modify the database table for a given field.

---

**Tip 11 GRASS EDIT PERMISSIONS**

---

You must be the owner of the GRASS mapset you want to edit. It is impossible to edit vectors in mapsets which are not yours, even if you have write permissions.

---

### 7.4.6 Region Tool

The current region (window) in GRASS is very important for all raster modules. All new created rasters have the extension and resolution of the current region, regardless their original region. The region is stored in \$LOCATION/\$MAPSET/WIND file, and it defines north, south, east, west, number of columns, number of rows, horizontal and vertical resolution.

It is possible to switch on/off the grass region in QGIS canvas using the *Display Current GRASS Region* button.

With the *Edit Current GRASS Region* you can open a tool in which you can change the current region and symbology of the rectangle on the QGIS Canvas. When the tool is running, it is also possible to select a new region interactively on the QGIS canvas.

Both tools are available only if QGIS was started from a GRASS shell or if the GISRC enviroment variable pointing to a valid GISRC file was set (i.e. only if you are running GRASS within your mapset).

## 8 Using Plugins

QGIS has been designed with a plugin architecture. This allows new features/functions to be added to the application. Many of the features in QGIS are actually implemented as plugins.

There are two types of plugins in QGIS: core and user-contributed. A core plugin is maintained by the QGIS development team and is part of every QGIS distribution. A user-contributed plugin is an external plugin that is maintained by the individual author. The QGIS Community site (<http://community.qgis.org>) serves as the repository for user contributed plugins.

## 8.1 Finding and Installing a Plugin

When you install QGIS, all of the core plugins are included (these are described below). Additional user-contributed plugins may be available on the QGIS Community site. To see what user-contributed plugins are available, see the plugins page on the Community site (<http://community.qgis.org/plugins>).

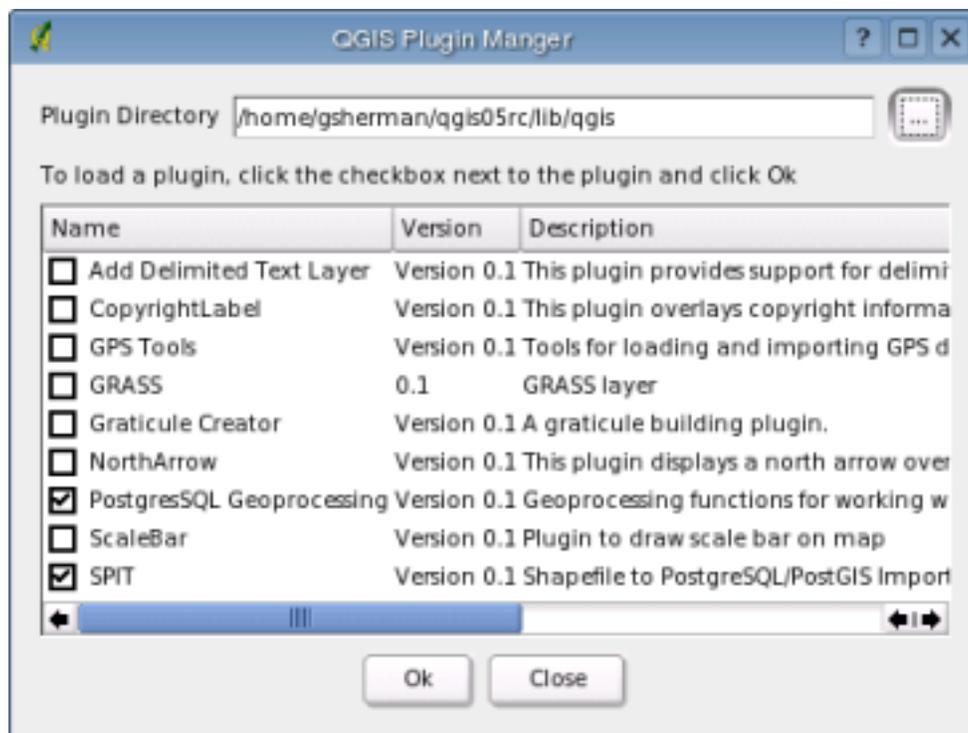
Typically user-contributed plugins are distributed in source form and require compiling. For instructions on building and installing a user-contributed plugin, see the documentation included with the plugin.

## 8.2 Managing Plugins

Managing plugins consists of loading or unloading them from QGIS. Loaded plugins are "remembered" when you exit the application and restored the next time you run QGIS.

To manage plugins, open the *Plugin Manager* from the *Tools* menu. The Plugin Manager displays all the available plugins and their status (loaded or unloaded). Figure 6 shows the Plugin Manager dialog.

Figure 6: Plugin Manager



Typically all QGIS plugins are installed in the same location. This location is shown in the Plugin Directory

text field. You can tell QGIS to load plugins from another location by specifying a different directory.

---

**Tip 12 CRASHING PLUGINS**

If you find that QGIS crashes on startup, a plugin may be at fault. You can stop all plugins from loading by editing your .qt/qgisrc file in your home directory on Linux/Unix (Windows users will have to edit the registry). On Linux/Unix, open the qgisrc file in a text editor and find the [Plugins] section. Set all the plugin values to false to prevent them from loading. For example, to prevent the Delimited text plugin from loading, the entry in qgisrc should look like this: `Add Delimited Text Layer=false`. Do this for each plugin in the [Plugins] section. You can then start QGIS and add the plugins one at a time from the Plugin Manger to determine which is causing the problem.

---

### 8.3 Data Providers

Data Providers are "special" plugins that provides access to a data store. By default, QGIS supports PostGIS layers and disk-based data stores supported by the OGR library (Appendix A). A Data Provider plugin extends the ability of QGIS to use other data sources.

Data Provider plugins are registered automatically by QGIS at startup. They are not managed by the Plugin Manager but are used behind the scenes when a corresponding data type is added as a layer in QGIS.

### 8.4 Core Plugins

QGIS currently contains 9 core plugins that can be loaded using the Plugin Manager. Table 3 lists each of the core plugins along with a description of their purpose. Figure 7 shows the icon for each plugin in the Plugin toolbar (the number corresponds to the Item in Table 3. Note the GRASS plugin is not included below because it installs its own toolbar (see Section 7 for a discussion of available features in GRASS plugin).

Table 3: QGIS Core Plugins

Item	Plugin	Description
1	Copyright Label	Display a copyright label on the map canvas
2	Delimited Text	Load a delimited text file containing x,y coordinates as a point layer
3	GPS Tools	Load and display GPS data
4	Graticule Creator	Create a latitude/longitude grid and save as a shapefile
5	Scalebar	Add a scalebar to the map canvas
6	North Arrow	Add a north arrow to the map canvas
7	PostgreSQL Geoprocessing	Buffer a PostGIS layer
8	SPIT	Shapefile to PostGIS Import Tool - import shapefiles into PostgreSQL

Figure 7: Plugin Toolbar and Icons



## A Supported OGR Formats

At the data of this document, the following formats are supported by the OGR library.

- Arc/Info Binary Coverage
- Comma Separated Value (.csv)
- DODS/OPeNDAP
- ESRI Shapefile
- FMEObjects Gateway
- GML
- IHO S-57 (ENC)
- Mapinfo File
- Microstation DGN
- OGDII Vectors
- ODBC
- Oracle Spatial
- PostgreSQL<sup>1</sup>
- SDTS
- SQLite
- UK .NTF
- U.S. Census TIGER/Line
- VRT - Virtual Datasource

## B GDAL Raster Formats

At the date of this document, the following formats are supported by the GDAL library. Note that not all of these format may work in QGIS for various reasons. For example, some require external commercial libraries. Only those formats that have been well tested will appear in the list of file types when loading a raster into QGIS. Other untested formats can be loaded by selecting the *All other files (\*)* filter. Formats known to work in QGIS are indicated in **bold**.

- **Arc/Info ASCII Grid**
- **Arc/Info Binary Grid (.adf)**
- Microsoft Windows Device Independent Bitmap (.bmp)
- BSB Nautical Chart Format (.kap)
- VTP Binary Terrain Format (.bt)
- CEOS (Spot for instance)
- First Generation USGS DOQ (.doq)
- New Labelled USGS DOQ (.doq)
- Military Elevation Data (.dt0, .dt1)
- ERMapper Compressed Wavelets (.ecw)
- ESRI .hdr Labelled
- ENVI .hdr Labelled Raster
- Envisat Image Product (.n1)

---

<sup>1</sup>QGIS implements its own PostgreSQL functions. OGR should be built without PostgreSQL support

- EOSAT FAST Format
- FITS (.fits)
- Graphics Interchange Format (.gif)
- **GRASS Rasters<sup>2</sup>**
- **TIFF / GeoTIFF (.tif)**
- Hierarchical Data Format Release 4 (HDF4)
- **Erdas Imagine (.img)**
- Atlantis MFF2e
- Japanese DEM (.mem)
- **JPEG JFIF (.jpg)**
- JPEG2000 (.jp2, .j2k)
- JPEG2000 (.jp2, .j2k)
- NOAA Polar Orbiter Level 1b Data Set (AVHRR)
- Erdas 7.x .LAN and .GIS
- In Memory Raster
- Atlantis MFF
- Multi-resolution Seamless Image Database MrSID
- NITF
- NetCDF
- OGD Bridge
- PCI .aux Labelled
- PCI Geomatics Database File
- Portable Network Graphics (.png)
- Netpbm (.ppm,.pgm)
- **USGS SDTS DEM (\*CATD.DDF)**
- SAR CEOS
- **USGS ASCII DEM (.dem)**
- X11 Pixmap (.xpm)

---

<sup>2</sup>GRASS raster support is supplied by the QGIS GRASS data provider plugin