

# QuantumGIS: The easy way

**T**oday there are a number of projects making life easier for the user. Among these, besides a number of minor projects, several are in use, eg uDIG, gvSIG, OpenJUMP, and especially QuantumGIS.

The good thing is that interoperability among different pieces of free software is very high, as it normally happens with open source programs.

## WHY QUANTUMGIS

QuantumGIS (in short QGIS) has its base development libraries been developed upon popular KDE Linux desktop environment.

## A SHORT STORY

Quantum GIS was created in June 2002 by Gary Sherman. After 1.5 years, in which the basis of the structure was laid out, the development pace increased considerably, from early 2004 (Fig 1). The project has been incubated in the popular SourceForge framework for open source

projects (www.sourceforge.net), but has now its own structure, with a web site (www.qgis.org), where anybody can know more, get the program and the documentation, news, tips and tricks, interviews, etc. QGIS is now an incubating project in Open Source Geospatial Foundation (OSGEO: www.osgeo.org)

## BASE FEATURES

QGIS has good integration



Fig. 1 : Development of QGIS

with other free GIS software, in particular GRASS, PostGIS, and UMN MapServer.

## Multi-platform

QGIS has been translated in several different languages, and more can be added with a minimal effort, thanks to code openness. It is available for Linux (where it is mainly developed), but also for all other popular

platforms, including Windows and MacOSX.

## Main functions

The interface (Fig. 2) looks rather familiar to most GIS users. QGIS can read rasters (50+formats) and vectors (20+ formats). One important feature is the capability of reading and writing geodatabase (PostGIS) data, both local and remote. It can also read Web Services (WMS and WFS, standard OGC). This allows the user to bring on common ground a variety of different data, thus acting as a powerful integrator.

It is possible to manipulate RGB colour bands of rasters, visualizing them in greyscale or pseudocolor and set a transparency level;

loading data can be made faster by pyramiding the original rasters. It is also possible to plot the color histogram (Fig.3). Symbology management for vectors is extensive: colours and patterns for filling and outlining can be selected, also in relation to table values; labels can be shown, and attributes can be shown and queried from the tabular data (with a graphical front-end to SQL queries) and from the map; selected records are highlighted both in the table and in the map.

Vector editing is possible, in various ways: shapefiles can be modified (with some limitation), as well as PostGIS geodatabase layers (PostGIS advanced digitizing is under heavy development

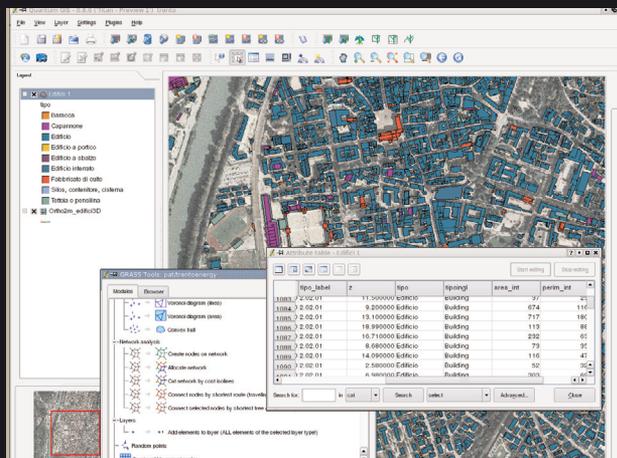


Fig. 2: QGIS user interface

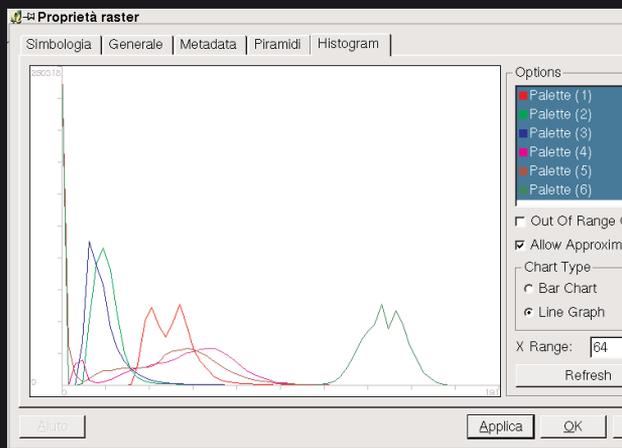


Fig. 3 Histogram

in this period). For heavy duty digitizing, however, the best available option is the GRASS plugin; this allows to display the layers in the various Location and Mapset (the GRASS

directory structure) and to edit vectors with an interface similar to that of GRASS, though more intuitive. With the GRASS plugin, digitalization is fully and natively topological, which

prevents errors (self-intersections, dangles, etc.) to creep in your precious maps.

In QGIS it is possible to save projects (essentially, a list of vector and raster data, with associated visualization options such as colours etc.) for later work; I find it very handy also the “spatial bookmarks”, a simple system of tagging specific portions of the working area, to zoom quickly to one of a series of sub-areas. The layout (colours, fills etc. for each layer) can be saved as a configuration file (.map) for the powerful free web mapping engine, UMN MapServer. The same data can thus be published on the web,

keeping the desired layout, with minimal effort.

Printing is done through a dedicated window, where you can easily choose paper characteristics and scale, and inserting texts, legends and scale bars (Fig 4). Maps can be printed, or saved in various formats: encapsulated postscript (eps) and pdf, in raster format (bitmap, jpeg, png etc.) and in scalable vector graphics (svg). Further editing of the map can thus be carried out in different image or vector editors (among the free ones: the GIMP for rasters, Inkscape for vectors).

Projections can be handled gracefully; layers in different reference systems can

# Mest Ad Page 63

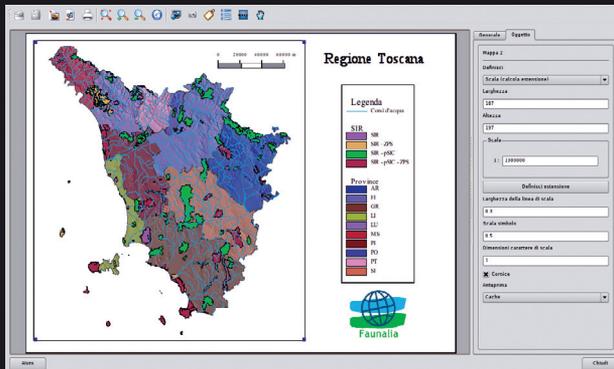


Fig 4. Print layout

be reprojected in the system of choice (over 2,700 are available) on the fly, thus avoiding to duplicate data, converting them in different systems.

### A modular structure: the plugins

QGIS has a modular structure (Fig. 5), which allows to add new functionality for specific tasks.

Each plugin can be activated (as usual, just a click on the menu is necessary) at the user's choice.

The simplest plugins allow the user to add to the map user-defined grids, scale bar, north arrow, copy-right labels and more. Others allow the import and export of geographic data from various sources:

- a text file can be loaded as geo-

graphical data (points), provided it has an X and Y column (plus eventual associated data).

- GPS data (waypoints, routes, and tracks) can also be loaded and shown, either directly or through a text file ; QGIS uses a specialized program for this: GPSBabel ([www.gpsbabel.sf.net](http://www.gpsbabel.sf.net)), also free and open source software; similarly, geographical data can be uploaded on the GPS

- geodatabase layers from

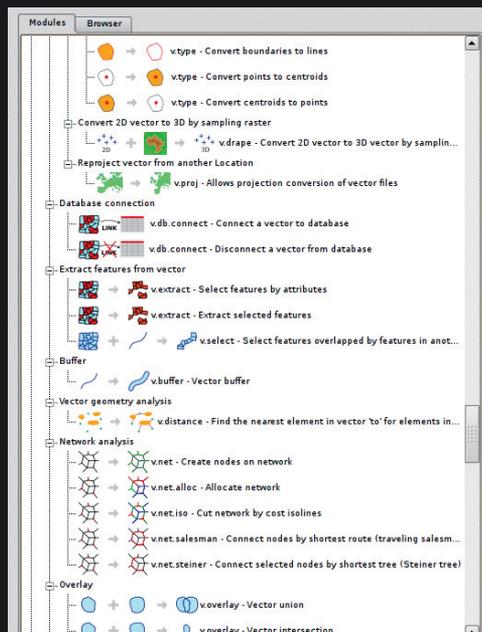


Fig 5. Modular Structure

PostGIS can also be shown, and shapefiles can be imported into the database

The GRASS plugin allows not only to display and edit GRASS layers, but also to use QGIS as a graphical interface for many GRASS commands. Activating the plugin, a window shows the commands, grouped on the basis of the function, with a graphical description. Clicking on one of

the commands, all the relevant informations for the chosen command are shown: one window for the management of the options, one for the output messages, and one for the manual page. If for instance we need the intersection between two polygonal vectors, we just have to select the command "Vector intersection"

from the group "Vector overlay", choose the name of the two vectors to be overlaid, plus the name of the resulting vector. Then clicking on "Run" we obtain the result, which can thus be displayed on the map.

Another plugin allows the georeferencing raster images (both with linear methods, without image stretching, and with Helmert method, based on polynomials. More specialized plugins are available, generally as separate

packages since they are of very specialized use. One of the most exciting features of version 0.9, now under development, is the possibility of writing plugins in python, a very simple yet powerful interpreted language. New plugins can therefore be written and tested with minimal effort.

### CONCLUSIONS

QGIS is, right now, a modern piece of software, easy to use, and comparable to well-known Closed Source alternatives; being a young project, bugs are still present (a complete list on the QGIS web site; you can check yourself if any of these is critical for your work, and in case they are, fixing them is usually cheap and fast, if you wish to invest in the project).

Features are constantly increasing, thanks to the open source development model. As an example, in the development version is now possible to save as shapefile any kind of vector layer that has been loaded into QGIS, as a whole or a subselection of it, with a left-click on the layer name.

Ultimately, as an open source project, it will be the response from developers and users to determine the priorities and the outcomes of the next steps.

**Paolo Cavallini**  
Co-author: Leonardo Lami  
Faunalia Piazza Garibaldi, Italy