A Testimony to the Power of Open Source

Quantum GIS 1.0

Quantum GIS (QGIS) is a user-friendly Geographic Information System (GIS). The current stable version 1.0 was released in January 2009. Although the initial goals were modest, QGIS has become a mature and extensible tool for viewing, editing, and performing GIS analysis. While it took nearly seven years to get to version 1.0, the process is a testimony to the power of open source in bringing the talents and ideas of many individuals together to create a tool used by thousands in academia, government, and private industry around the world.

By Otto Dassau, Gary Sherman, Tim Sutton, Marco Hugentobler and Paolo Cavallini

2. Graphical User Interface

Working with QGIS is simple and intuitive as you are presented with a modern and friendly GUI based on Qt4. All functions are clearly separated (see Figure 2).

A menu bar provides access to QGIS features using a standard hierarchical menu, with icons of the corresponding tools as they appear on the tool bar and with keyboard shortcuts. The tool bar icons provide direct access to functions of the menu bar, plus additional tools for interacting with the map view. To make the GUI appear simpler, tool bar icons can be switched on and off. The “business end” of QGIS is the map view. Various operations can be performed on the map such as pan, zoom-in, zoom-out, select or query. It is tightly bound to the map legend, where layer visibility is managed and set to a z-order, meaning layers listed nearer the top of the legend are drawn over layers listed lower down. The map overview area provides a full extent view of selected layers with a rectangle showing the current map extent in the map view. And finally, the status bar shows the current mouse pointer position in map coordinates, view extents of the map view, the progress of rendering or analysis activities, the current map scale.
depending on the defined Coordinate Reference System (CRS), and information about available external plugin updates.

3. Functionality
QGIS offers a growing array of common GIS functionality provided by core features and plugins, and at a glance provides the following features:

- view and overlay vector and raster layer in different formats and projections without conversion to an internal or common format. Supported are PostgreSQL/PostGIS, GDAL/OGR supported vector and raster layers such as ESRI Shapefile, MapInfo, GML, GeoTiff and Erdas Img, GRASS rasters, vectors, and locations, and OGC-compliant WMS and WFS;
- interactively explore data, including features such as on-the-fly (OTF) projection, identify/select geometries, view, select and search attributes, change vector and raster symbology;
- compose print layouts adding map canvas, legend, scale bar, images and text labels in a print composer plugin;
- create, edit, manage and export vector layers into several formats. Raster layer have to be imported into GRASS GIS to be edited and exported;
- perform spatial geo-processing on PostgreSQL/PostGIS and other OGR supported vector layers including overlay, buffer, sampling, geometry and database management. The integrated GRASS plugin allows easy access to more than 260 GRASS modules, allowing complex GIS raster and vector analyses, including raster algebra, hydrological modeling, interpolating surfaces, network analyses, database operations, and much more.

4. Plugin Architecture
QGIS has been designed with a plugin architecture and therefore new customized features and functions can easily be added to the application. Many of the features in QGIS are actually implemented as core or external plugins.

Core plugins are maintained by the QGIS Development Team. They are written in C++ or Python, are automatically part of every QGIS distribution and can be enabled with the Plugin Manager. There are currently 17 core plugins available, including GRASS GIS integration, Georeferencer, Mapserver Export, Shapefile to PostGIS Import Tool, OGR Layer Converter, GPS Tools, Add Delimited Text Layer and WFS support.

External plugins are all written in Python and divided into official and user-contributed plugins. The user can easily add those plugins to QGIS with the Python Plugin Installer (See Figure 5).

- Official external Python plugins are stored in an official, moderated repository at http://pyqgis.org/repo/official as part of the official QGIS release and maintained by their respective author
- User-contributed external Python plugins are stored in an unofficial repository at http://pyqgis.org/repo/contributed and contain plugins that are not yet mature enough but are on the way to the official repository

In addition to these two repositories, a number of QGIS developers provide and maintain their own repositories. These can be added to the repository list of the Python Plugin Installer.

5. Development
Since QGIS is open source software, participation in the development process and writing new applications that use the libraries of the QGIS project are not only possible but encouraged. Development with QGIS can be done either in the existing classes of QGIS, as plugin extensions or in the form of custom applications that make use of the QGIS libraries. All code in QGIS is licensed under the GNU GPL www.fsf.org/licensing/licenses/gpl.html. That means that for all three cases, published software must be distributed under the terms of the GPL, too. QGIS 1.0 provides a stable API which provides an assurance that plugins and applications developed against the 1.0 API will work against future releases in the 1.X release series.

5.1 Development in the Core Classes of QGIS
Changes to existing classes may be submitted as patches using the QGIS Project bug tracker https://trac.osgeo.org/qgis. The code maintainers of the QGIS project, each responsible for a certain part of the code base, regularly check the tracker.

5.2 Development of Extensions as C++ or Python Plugins
The plugin interface allows extensions to access the running QGIS instance and to use and extend the objects in the core of QGIS. Plugins may be written in C++ or in Python. The QGIS documentation contains

Figure 3: Shapefile to PostGIS Import Tool

Figure 4: One of the QGIS Core Plugins (GRASS GIS Integration)
5.3 Custom Applications that Use the QGIS Libraries

It is also possible to write new applications that provide their own user interface and use the QGIS core library for the GIS logic, data access and map rendering.

An example using this approach is the QGIS map server project [http://karlinapp.ethz.ch/qgis_wms](http://karlinapp.ethz.ch/qgis_wms) that provides a WMS-compatible map server on top of the QGIS core library. This software has no graphical user interface. It is a FastCGI application that waits until called by a web server. It parses the request parameters and uses QGIS to render a map into an off-screen buffer. The content is then returned as a binary image back to the client.

Another context where this approach would make sense is to provide a mapping application for mobile devices. Applications for mobile devices usually need different user interfaces to desktop computer applications and laptops. The QGIS libraries offer the potential to be used as a GIS back end for applications targeting mobile devices.

6. Who uses QGIS?

QGIS is now widely used by professionals, government and local agencies, universities, students, and amateurs alike for a large variety of tasks, from simply viewing raster and vector data (especially useful is the capability of dealing with PostGIS layers) to running complex and custom analyses through GRASS modules. Often QGIS is used to replace the capability of dealing with PostGIS layers) to running complex and custom analyses through GRASS modules. Often QGIS is used to replace the GIS libraries offer the potential to be used as a GIS back end for applications targeting mobile devices.

7. Perspective / Conclusion

Quantum GIS began as a one-developer application that was met with skepticism by many who asked, “Why another open source GIS?”. Although the initial goals were modest, QGIS has become a mature and extensible tool for viewing, editing, and performing GIS analysis. Creating a feature-complete GIS from scratch is a tremendous undertaking and at the outset was not really a goal of the project. With the GRASS integration and the extensibility possible through plugins, QGIS is positioned to grow into an even more robust toolset for the GIS user. Early in life, the QGIS community was small and grew quite slowly. With the addition of several key developers, the features and capabilities expanded rapidly and with it, the community. QGIS now has an established community providing peer support, testing, and new features via plugins.

At version 1.0, QGIS provides a rich, stable API from which developers can create custom solutions in Python or C++. As the project moves forward, there are many exciting developments underway in both the core application and plugins.

While it took nearly seven years to get to version 1.0, the process is a testimony to the power of open source in bringing the talents and ideas of many individuals together to create a tool used by thousands in academia, government and private industry around the world.

The authors of this article are the QGIS Project Steering Committee Members:

- Otto Dassau dassau@nature-consult.de lives and works in Hannover, Germany. His topics are FOSS GIS and applied remote sensing.
- Gary Sherman gsherman@mrcc.com lives and works in Alaska and has been torturing computers and programming languages for well over two decades.
- Paolo Cavallini cavallini@faunalia.it lives and works in Italy. He heads a small and active group of professionals (biologists and agronomists) under the name Faunalia.

For more information, have a look at the following websites:

- Quantum GIS project: [http://qgis.org](http://qgis.org)
- QGIS Forum: [http://forum.qgis.org](http://forum.qgis.org)
- QGIS Blog: [http://blog.qgis.org](http://blog.qgis.org)
- QGIS User Mailing List: [http://lists.osgeo.org/mailman/listinfo/qgis-user](http://lists.osgeo.org/mailman/listinfo/qgis-user)
- QGIS IRC: Channel #qgis port 6667 at irc.freemode.net
- QGIS Map Server Project: [http://karlinapp.ethz.ch/qgis_wms](http://karlinapp.ethz.ch/qgis_wms)
- GNU GPL: [www.fsf.org/licensing/licenses/gpl.html](http://www.fsf.org/licensing/licenses/gpl.html)
- Open Source Geospatial Foundation: [www.osgeo.org](http://www.osgeo.org)