

Building and Installing QGIS

Gary Sherman Tim Sutton

June 18, 2005

Contents

1 Introduction	1
1.1 Installing Windows Version	2
1.2 Installing Mac OS X Version	2
1.3 Building from Source	2
2 Getting QGIS	2
3 PostgreSQL	3
4 GEOS	4
5 PostGIS	5
6 GRASS	5
7 Proj4	5
8 SQLite	6
9 GDAL/OGR	6
10 Qt	7
11 Building QGIS	8
11.1 Quick and Dirty	8
11.2 Configuring QGIS the Right Way	8
11.2.1 Qt	8
11.2.2 GDAL	8
11.2.3 PostgreSQL	9
11.2.4 GRASS	9
11.2.5 Example Use of Configure	9
11.2.6 Compiling and Installing QGIS	9
12 Building Plugins	9

1 Introduction

This document briefly describes how to build QGIS 0.7 (*Seamus*) from the source distribution. These instructions are for Linux/Unix and other POSIX systems which have the required build environment.

The latest version of this document can always be found at <http://qgis.org/docs/install.html>.

1.1 Installing Windows Version

Installing the Windows version of QGIS is simply a matter of running the user friendly setup wizard. See the README.WIN32 file for additional information regarding the Windows version of QGIS. At version 0.7, the GRASS plugin is not available in Windows.

NOTE - At 0.7pre1, there is no Windows version available.

1.2 Installing Mac OS X Version

To install the compressed disk image containing the OSX version of QGIS, double-click to expand and mount the image, then drag QGIS application to your hard drive. If you want to build from source on Mac OS X, see <http://wiki.qgis.org/qgiswiki/BuildingOnMacOsX>.

1.3 Building from Source

The remainder of this document deals with compiling and installing QGIS from the source code. Specifically this applies to Linux/Unix systems.

At 0.7, there are two new requirements: SQLite and Proj4. These must be built and installed prior to configuring QGIS.

QGIS can be installed with three levels of support for data stores:

1. Basic raster and vector support (GDAL and OGR formats)
2. PostgreSQL/GEOS/PostGIS
3. GRASS raster and vector support

Basic support uses the GDAL/OGR libraries and supports many raster and vector formats. For more information on the available formats, see http://www.remotesensing.org:16080/gdal/formats_list.html and http://www.remotesensing.org:16080/gdal/ogr/ogr_formats.html.

PostgreSQL/PostGIS support allows you to store spatial data in a PostgreSQL database. GRASS support provides access to GRASS mapsets.

Note: - If you plan to build QGIS with GRASS support, version 1.2.3 or higher of GDAL must be used.

Each of the requirements are discussed below. Note that the information given below is abstracted from the installation documentation for each of the libraries. See the install information for each library to get detailed instructions. In the documentation below, the file names and versions used are examples.

If you are building QGIS without PostgreSQL or GRASS support, skip to the section on Installing GDAL/OGR.

2 Getting QGIS

QGIS is available in both source and package format from <http://qgis.org>.

In addition, packages for many Linux distributions are independently maintained in various locations. See the *Download* section on <http://qgis.org> for the latest information on package locations.

Packages for most of the software/libraries discussed below can be found for almost all Linux distributions. While it is possible to mix compiling from source and installing packages to meet the requirements for QGIS, sometimes this becomes tricky. Following the steps below will generally ensure a successful installation. If you are using SuSE 9.1, the LinGIS distribution <ftp://ftp.lingis.org> is a good choice for installing QGIS and its dependencies.

3 PostgreSQL

QGIS uses the latest features of PostgreSQL. For this reason, version 7.4.x or higher is recommended with QGIS version 0.5. If you choose to add PostgreSQL, you must also install PostGIS and the GEOS library (see below).

1. Download PostgreSQL source from www.postgresql.org

2. Extract the source

```
tar -xzf postgresql-7.4.1.tar.gz
```

3. Change to the source directory

```
cd postgresql-7.4.1
```

4. Configure PostgreSQL:

```
./configure --prefix=/usr/local/pgsql
```

5. Build

```
make
```

6. Install

```
make install
```

7. As root, create the postgres user and setup the database (following taken from PostgreSQL INSTALL file with modification)

- Create the postgres user

```
adduser postgres
```

- Create the directory for the PostgreSQL database

```
mkdir /usr/local/pgsql/data
```

- Change ownership of the data directory to the postgres user

```
chown postgres /usr/local/pgsql/data
```

- su to the postgres user (or login as postgres)

```
su - postgres
```

- Change to the PostgreSQL install directory

```
cd /usr/local/pgsql
```

- Initialize the database

```
./bin/initdb -D /usr/local/pgsql/data
```

- Start the PostgreSQL daemon

```
./bin/pg_ctl start -o "-i" -D /usr/local/pgsql/data -l /home/postgres/serverlog
```

- Create the test database

```
./bin/createdb test
```

8. PostgreSQL should now be running. Logon as the postgres user (or use `su - postgres`). You should be able to connect to the test database and execute a test query with the following commands:

```
psql test
select version();
version
-----
PostgreSQL 7.4.1 on i686-pc-linux-gnu, compiled by GCC gcc (GCC) 3.3.1 (SuSE Linux)
(1 row)

\q
```

9. PostgreSQL install is done

4 GEOS

Note: As of version 0.6, GEOS is a requirement in order to build QGIS.

QGIS uses GEOS to properly fetch features from the the underlying datastore when doing an identify or select operation.

To install GEOS:

1. Download GEOS source from <http://geos.refractorions.net>
2. Untar GEOS

```
tar -xzf geos-2.0.0.tar.gz
```

3. Change to the GEOS source dir

```
cd geos-2.0-.0
```

4. Follow the instructions in the GEOS README file to complete the installation. Typically the install goes like this:

```
./configure
make
make install
```

5 PostGIS

NOTE - You must edit the PostGIS Makefile and make sure that USE_GEOS=1 is set. Also adjust GEOS_DIR to point to your GEOS installation directory.

1. Download PostGIS source from <http://postgis.refractions.net>
2. Untar PostGIS into the contrib subdirectory of the postgresql build directory. The contrib subdirectory is located in the directory created in step 3 of the PostgreSQL installation process.
3. Change to the postgis subdirectory
4. Edit the Makefile to enable GEOS support (see the note above)
5. PostGIS provides a manual in the doc/html subdirectory that explains the build process (see the Installation section)
6. The quick and dirty steps to install PostGIS are:

```
cd contrib
gunzip postgis-0.8.0.tar.gz
tar xvf postgis-0.8.0.tar
cd postgis-0.8.0
make
make install
createlang plpgsql yourtestdatabase
psql -d yourtestdatabase -f postgis.sql
psql -d yourtestdatabase -f spatial_ref_sys.sql
```

The **better way** to install PostGIS is to carefully follow the instructions in the PostGIS manual in the doc/html subdirectory or the online manual at <http://postgis.refractions.net/docs>

6 GRASS

If you want QGIS to support GRASS vector and raster layers, you must build GRASS prior to proceeding. Follow the directions on the GRASS website carefully to build version 6.0.x. Additional information and the build instructions can be found at <http://grass.itc.it>.

The GRASS software is available for download at <http://grass.itc.it/download.html>.

7 Proj4

Proj4 provides the functions needed for on the fly projection of map layers in version 0.7. To build and install Proj4, download the latest version from <http://proj.maptools.org>, untar the distribution and:

```
./configure
make
make install
```

8 SQLite

Sqlite is used to manage the projections database and store persistent data such as spatial bookmarks. Download the latest (3.x) version of SQLite from <http://www.sqlite.org/>. Untar the distribution and:

```
./configure
make
make install
```

Note - SQLite 3.x is included on Mac OS X 10.4.

9 GDAL/OGR

The GDAL and OGR libraries provide support for raster and vector data formats. QGIS makes use of both of these libraries (which come bundled in one distribution).

Note: A Linux binary of GDAL is available at <http://www.remotesensing.org/gdal>. If you choose to install the binary you will also need to download and unpack the source tree since QGIS needs the header files in order to compile.

To install GDAL/OGR from source:

1. Download the GDAL distribution from <http://www.remotesensing.org/gdal>. You should use version 1.1.9 or higher. Versions prior to 1.1.9 contained a bug that caused problems when a null feature was encountered. If you want to build vector support for GRASS, you must use GDAL 1.2.3.
2. Untar the distribution

```
tar xfvz ../path/./gdal-x.x.x.tar.gz
```

3. Change to the gdal-x.x.x subdirectory that was created by step 2

```
cd gdal-x.x.x
```

4. Configure GDAL

```
./configure
```

or if you want GRASS support

```
./configure --with-grass=<full path to grass install>
```

Depending on the GDAL version you are building, it may be necessary to specify `--without-ogdi` when running configure if you don't have the OGD I library available on your system.

5. Build and install GDAL:

```
make
su
make install
```

6. In order to run GDAL after installing it is necessary for the shared library to be findable. This can often be accomplished by setting `LD_LIBRARY_PATH` to include `/usr/local/lib`. On Linux, you can add `/usr/local/lib` (or whatever path you used for installing GDAL) to `/etc/ld.so.conf` and run `ldconfig` as root.
7. Make sure that `gdal-config` (found in the `bin` subdirectory where GDAL was installed) is included in the `PATH`. If necessary, add the path to `gdal-config` to the `PATH` environment variable.

```
export PATH=../path/..gdal-config:$PATH
```

8. Check the install by running:

```
gdal-config --prefix
```

If you've had problems during the installation, refer to this manual, where the whole process is described with some more detail: http://www.remotesensing.org/gdal/gdal_building.html

10 Qt

Qt 3.1.2 or higher is required in order to compile QGIS. You may already have Qt on your system. If so, check to see if you have version 3.1.2 or later. You can check the Qt version using the `find` command:

```
find ./ -name qglobal.h 2>/dev/null | xargs grep QT_VERSION_STR
```

If you have the `locate` utility installed you can do the same more quickly using:

```
locate qglobal.h | xargs grep QT_VERSION_STR
```

In either case the result should look something like this:

```
#define QT_VERSION_STR    "3.3.1"
```

In the example above, Qt 3.3.1 is installed.

If Qt is not installed, you will have to install the Qt development package for your distribution. If you are not able to install the required Qt packages, you will have to build from source.

To install Qt from source:

1. Download Qt from <http://www.trolltech.com/developer> (choose the Qt/X11 Free Edition)
2. Unpack the distribution
3. Follow directions provided in the distribution directory (`doc/html/install-x11.html`)
4. Use whatever configure options you like but make sure you include `-thread` for use with QGIS. You can configure Qt with minimal options:

```
./configure -thread
```

5. Complete the installation per the instructions provided in the Qt documentation (see step 3)

11 Building QGIS

After you have installed the required libraries, you are ready to build QGIS. Download and untar the QGIS distribution and change to the QGIS source directory. You have two options for building and installing QGIS: **Quick and Dirty** and the **right way**.

11.1 Quick and Dirty

If you don't need PostgreSQL support and have installed GDAL, you can configure and build QGIS by changing to the distribution directory and typing:

```
./configure
make
make install
```

The above assumes that the `gdal-config` program is in your `PATH`. See the next section for the full configuration instructions.

11.2 Configuring QGIS the Right Way

To see the configure options available, change to the QGIS directory and enter:

```
./configure --help
```

Among other options, there are three that are important to the success of the build:

```
--with-qtdir=DIR           Qt installation directory default=$QTDIR
--with-gdal=path/gdal-config Full path to 'gdal-config' script,
                           e.g. '--with-gdal=/usr/local/bin/gdal-config'
--with-pg=path/pg_config   PostgreSQL (PostGIS) Support
                           (full path to pg_config)
--with-grass=DIR           GRASS Support (full path to GRASS binary package)
```

11.2.1 Qt

The configure script will detect Qt, unless it is installed in a non-standard location. Setting the `QTDIR` environment variable will make ensure that the detection succeeds. You can also specify the path using the `--with-qtdir` option.

11.2.2 GDAL

If the `gdal-config` script is in the `PATH`, configure will automatically detect and configure GDAL support. If not in the path, you can specify the full path to `gdal-config` using the `--with-gdal` option. For example:

```
./configure --with-gdal=/usr/mystuff/bin/gdal-config
```


11.2.3 PostgreSQL

If the `pg_config` script is in the `PATH`, `configure` will automatically detect and configure PostgreSQL support. If not, you can use the `--with-pg` option to specify the full path to `pg_config`. For example:

```
./configure --with-pg=/usr/local/psql/bin/pg_config
```

11.2.4 GRASS

To build QGIS with GRASS support you must specify the full path to the installed GRASS binary package:

```
./configure --with-grass=/usr/local/grass-6.0.0
```

This assumes that GRASS is installed in the default location. Change the path to match the location of your GRASS installation.

11.2.5 Example Use of Configure

An example of use of `configure` for building QGIS with all options:

```
./configure --prefix=/usr/local/qgis \  
--with-gdal=/usr/local/gdal/bin/gdal-config \  
--with-pg=/usr/local/psql/bin/pg_config \  
--with-grass=/usr/local/grass-6.0.0
```

This will configure QGIS to use GDAL, GRASS, and PostgreSQL. QGIS will be installed in `/usr/local/qgis`.

If `QTDIR` is set and `gdal-config` and `pg_config` are both in the `PATH`, there is no need to use the `--with-gdal` and `--with-pg` options. The `configure` script will properly detect and configure GDAL and PostgreSQL. You must still use the `--with-grass` option if building with GRASS support.

11.2.6 Compiling and Installing QGIS

Once properly configured simply issue the following commands:

```
make  
make install
```

NOTE - As of version 0.1, you can no longer run QGIS from the `src` directory. You must do a `make install` and start QGIS from the installed location. In the case of the example above, the QGIS binary resides in the `bin` subdirectory of the directory specified with the `prefix` option (`/usr/local/qgis/bin`).

For information on using QGIS see the QGIS User Guide.

12 Building Plugins

The QGIS source distribution contains a number of "core" plugins. These are built along with QGIS using the instructions above. Additional external plugins are available from the QGIS community

website at <http://community.qgis.org>. Instructions for building an external plugin can be found at <http://wiki.qgis.org/qgiswiki/StepByStepBuildInstructions>. Some external plugins may include instructions on building. If so, follow the instructions provided with the plugin rather than those provided in the wiki.