QGIS Installation Guide

For Version 0.1 *Moroz*

by Gary E. Sherman, Tim Sutton

1. Introduction

This document briefly describes how to build QGIS from the source distribution. The information below is for **version 0.1 *Moroz*** of QGIS.

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

QGIS requires that a number of libraries be installed on your system:

- Qt 3.1.2 or higher
- GDAL/OGR
- PostgreSQL and PostGIS (Optional)

Each of these 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 used are examples.

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

2. PostgreSQL

QGIS uses the latest features of PostgreSQL. For this reason, version 7.4.x is recommended with QGIS version 0.1. 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 serverlog ./bin/createdb
     ```
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.

3. GEOS

QGIS uses GEOS to properly fetch features from the database when doing an identify or select. You can still view PostGIS layers without GEOS, but you will not be able to create a selection or identify a feature.

If you choose to proceed without GEOS support, QGIS will warn you each time you load a layer from the database.

To install GEOS:

1. Download GEOS source from geos.refractions.net

2. Untar GEOS
   ```
   tar -xzf geos-1.0.0.tar.gz
   ```
3. Change to the GEOS source dir
   ```
   cd geos-1.0-.0
   ```
4. Follow the instructions in the GEOS README file to complete the installation. Typically the install goes like this:
   - ./configure
   - make
   - make install

4. 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 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
7. The **better way** 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

5. 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.
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 without PostgreSQL support (necessary):
   ```
   ./configure --without-pg
   ```
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 running ldconfig.
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

6. 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. If not already 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)

7. 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*.

7.1. 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
```

> **Note:**
>
> This assumes that the gdal-config program is in your PATH

See the next section for the full configuration instructions.

7.2. Configuring QGIS

To see the configure options available, change the 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)
```

7.2.1. 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.

7.2.2. 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
```

7.2.3. 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
```

7.2.4. Example Use of Configure

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

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

Note:

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

7.3. Compiling and Installing QGIS

Once properly configured simply issue the following commands:

```
make
make install
```

Note:

As of this version, you can no longer run QGIS from the src directory. You must do a make install and start QGIS from the installed location

SourceForge.net Logo