GFOSS Blog | GRASS GIS Courses

Blog & Inhouse training offers by Markus Neteler

Compiling QGIS 2.0.1 for Fedora 19 in a few steps

Thanks to Volker Fröhlich's efforts, a source code RPM package (SRPM) of **QGIS 2.0.1 is now available for Fedora**. If you are not yet F20 user (like me), you can just take the F20 package and compile it for F19 (or even F18) since there will be no backport of QGIS 2 to F19 (it comes with QGIS 1.8). But: we do want QGIS 2 on Fedora19!

Solution: compile it yourself.

1. Preparations

The best way is to use "mock" which is used to recompile SRPMS in a separate local environment ("ch-root") without cluttering the system with extra packages needed for the compilation (run as "root"):

```
su
yum install mock
```

2. Get the source code

Next download the SRPM package from the *Koji* server:
QGIS: http://koji.fedoraproject.org/koji/buildinfo?buildID=467757 (–> src – download) or check here for more recent versions.

3. Compile it locally as RPM package

The general compilation command ("mock") would be:

```
mock -r my_fedora_version_config --rebuild my_source_rpm.src.rpm
```

So, check for *Fedora version config name* which is suitable for your system ("*my_fedora_version_config*")

```
ls /etc/mock/
```

In my case of a 64bit machine, it is "fedora-19-x86_64". Hence we can compile QGIS 2.0.1 directly from the SRPM file:

```
mock -r fedora-19-x86_64 --rebuild qgis-2.0.1-2.fc20.src.rpm
```

Note: the compilation takes 40min on my tiny core i3 laptop (ASUS X202). Use the time to donate some coins to the QGIS project 😃

4. Install and enjoy

Once the compilation job is done, i.e. the binary RPM files are available for installation. To install the freshly compiled QGIS 2.0.1 RPMs, run:
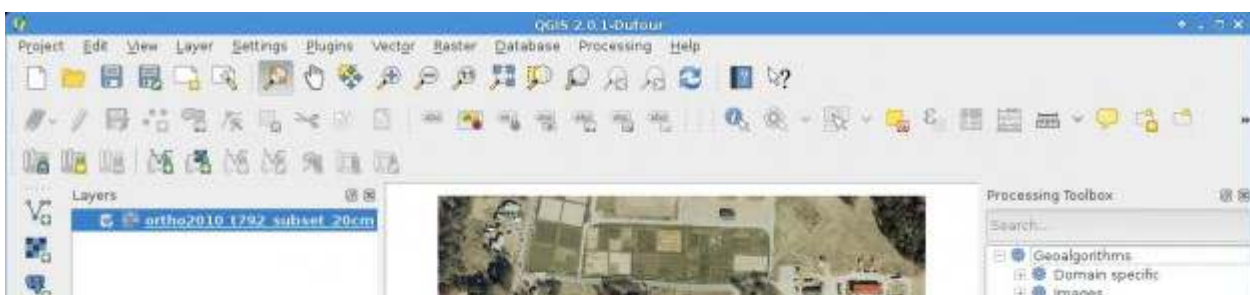
```
cd /var/lib/mock/fedora-19-x86_64/result/

# an existing QGIS1.8 installation will be replaced:
yum localinstall qgis-2.0.1-2.fc19.x86_64.rpm \
qgis-grass-2.0.1-2.fc19.x86_64.rpm qgis-python-2.0.1-2.fc19.x86_64.rpm

# consider to cleanup (or keep it for the next update, it is about 1.5GB):
rm -rf /var/lib/mock/fedora-19-x86_64/
# leave the "root" shell
exit
```

Now we can **happily use QGIS 2.0.1 on Fedora 19**!

```
qgis
```

This entry was posted in Blog, Fedora, Linux, QGIS on September 28, 2013 [http://courses.neteler.org
/compiling-qgis-2-0-1-for-fedora-19-in-a-few-steps/] .

10 COMMENTS ON "COMPILING QGIS 2.0.1 FOR FEDORA 19 IN A FEW STEPS"

dlangenhaun on **October 17, 2013 at 12:23 am** said:

Thank you for this workaround. Everything worked fine except I
had to remove qgis-1.8 manually (yum – remove qgis).

Pingback: Compilar QGIS2 en Fedora 19 | GeoSabio

Pingback: Compiling QGIS 2.0.1 for Fedora 19 in a few ste...

suresh on **October 28, 2013 at 6:36 am** said:

why fedora 19 not supporting lex and yacc programs for
execution.Is there any additional libraries required for executing
these programs.Can anyone let me know regarding this issue.
Thanking to you all

neteler

on **October 28, 2013 at 8:10 am** said:

To have "flex" and "yacc" on your system, you have to
install the related RPM packages with: "yum install flex
byacc". Then you can use them.

rahul on **November 9, 2013 at 7:11 pm** said:

i tried compiling a normal c-program in fedora 19 terminal...bt
its showing cc command not found...why is it so? and solution?

neteler
on **November 9, 2013 at 8:51 pm** said:

You need to install the gcc compiler package (then it is
called "gcc" rather than "cc") or another compiler.

Flo on **December 3, 2013 at 9:19 am** said:

Thank you for your guideline – after installing it like described I
get the following error when starting QGIS 2.0.1 – see below.
After clicking OK QGIS starts normally. Some functions like
"Raster – Clipper" also report errors (eg. missing gdal_translate).
Couldn't load plugin 'processing' from ['/usr/share/qgis/python',
'/home/ue000049/.qgis2/python', '/home/ue000049/.qgis2
/python/plugins', '/usr/share/qgis/python/plugins', '/usr/lib64
/python27.zip', '/usr/lib64/python2.7', '/usr/lib64/python2.7/plat-
linux2', '/usr/lib64/python2.7/lib-tk', '/usr/lib64/python2.7
/lib-old', '/usr/lib64/python2.7/lib-dynload', '/usr/lib64/python2.7
/site-packages', '/usr/lib64/python2.7/site-packages/gtk-2.0',
'/usr/lib64/python2.7/site-packages/wx-2.8-gtk2-unicode',
'/usr/lib/python2.7/site-packages', '/usr/lib/python2.7/site-
packages/setuptools-0.6c11-py2.7.egg-info']
Traceback (most recent call last):
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 182,
in loadPlugin
__import__(packageName)
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing/__init__.py", line
20, in
from processing.tools.general import runalg, runandload,
alghelp, alglist, algoptions, load, extent, getobject
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing/tools
/general.py", line 29, in
from processing.core.Processing import Processing
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing
/core/Processing.py", line 52, in
from processing.admintools.AdminToolsAlgorithmProvider
import AdminToolsAlgorithmProvider
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing/admintools
/AdminToolsAlgorithmProvider.py", line 19, in
from processing.admintools.PostGISExecuteSQL import
PostGISExecuteSQL
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing/admintools
/PostGISExecuteSQL.py", line 32, in
from processing.admintools import postgis_utils
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,

```
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
File "/usr/share/qgis/python/plugins/processing/admintools
/postgis_utils.py", line 39, in
import psycopg2
File "/usr/lib64/python2.7/site-packages/qgis/utils.py", line 453,
in _import
mod = _builtin_import(name, globals, locals, fromlist, level)
ImportError: No module named psycopg2
Python version:
2.7.5 (default, Oct 8 2013, 12:19:40)
[GCC 4.8.1 20130603 (Red Hat 4.8.1-1)]
QGIS version:
2.0.1-Dufour Dufour, exported
Python path: ['/usr/share/qgis/python', '/home/ue000049/.qgis2
/python', '/home/ue000049/.qgis2/python/plugins', '/usr/share
/qgis/python/plugins', '/usr/lib64/python27.zip', '/usr/lib64
/python2.7', '/usr/lib64/python2.7/plat-linux2', '/usr/lib64
/python2.7/lib-tk', '/usr/lib64/python2.7/lib-old', '/usr/lib64
/python2.7/lib-dynload', '/usr/lib64/python2.7/site-packages',
'/usr/lib64/python2.7/site-packages/gtk-2.0', '/usr/lib64
/python2.7/site-packages/wx-2.8-gtk2-unicode', '/usr/lib
/python2.7/site-packages', '/usr/lib/python2.7/site-packages
/setuptools-0.6c11-py2.7.egg-info']
```

neteler

on **December 3, 2013 at 2:24 pm** said:

Flo, this is very easy to solve: You need to install one
more package:
yum install python-psycopg2

Pierre Stévens on **December 16, 2013 at 2:56 pm** said:

Nicely done !
Thank you.