

QGIS Application - Bug report #5879

running from build directory - no python plugins

2012-06-27 05:51 AM - Etienne Tournig

| | | |
|--|-----------------|-------------------------------------|
| Status: | Closed | |
| Priority: | Normal | |
| Assignee: | Sandro Santilli | |
| Category: | Python plugins | |
| Affected QGIS version: | master | Regression?: No |
| Operating System: | Ubuntu | Easy fix?: No |
| Pull Request or Patch supplied: | Yes | Resolution: |
| Crashes QGIS or corrupts data: | No | Copied to github as #: 15338 |
| Description | | |
| <p>When developing qgis, much time can be saved by running from the build directory, but the core plugins are missing because they are not present in the build dir.</p> <p>Perhaps a new "make install-dev" Makefile command could copy the plugins and also other missing things?</p> <p>Also related, "imports should be moved into classFactory() so that nothing happens unless the plugin is explicitly started".</p> <p>More details: http://lists.osgeo.org/pipermail/qgis-developer/2012-June/020744.html</p> <p>Quoting Martin Dobias</p> <ul style="list-style-type: none">- when QGIS is run from build directory, it doesn't copy the internal python plugins to the build output directory - that's why sextante is complaining about missing plugin installer. We should probably fix that in order to provide an environment that is as similar to the installed one as possible- the imports should be moved into classFactory() so that nothing happens unless the plugin is explicitly started. (this problem will go away once we stop using metadata from __init__.py and only use them from metadata.txt) | | |
| Related issues: | | |
| Related to QGIS Application - Bug report # 6913: Python interpreter starts ha... | | Closed 2012-12-22 |

History

#1 - 2012-10-08 06:53 AM - Sandro Santilli

+1, this is very important for core plugin development

#2 - 2012-10-10 01:31 PM - Sandro Santilli

- Pull Request or Patch supplied changed from No to Yes

So I've done some work to install the db_manager plugin to output/python/plugin/* and to have qgis load it.

The result is two pull requests:

<https://github.com/qgis/Quantum-GIS/pull/285>

#3 - 2012-10-10 02:28 PM - Sandro Santilli

Alright, I confirm things work with those two pulls above. I can get db_manager loaded from output dir.
Next stop will be installing all plugins under output/

?

Ideally we'd have a macro for this on the CMake side, as the db_manager install has been very tedious....

?

Anyway please pull those two branches so I can get back to db_manager hacking when I find the time :)

#4 - 2012-10-10 11:12 PM - Sandro Santilli

I was thinking about two other possible ways to fix this:

1. Have the build dir listed in sys.path and plugin_paths, and make sure all sources are also copied to build dir
2. Have both the build dir and the source dir listed in sys.path and plugins_path

?

The second option would make the support automatically available to *all* plugins with no need to maintain anything at the plugin-level.

#5 - 2012-10-19 02:25 PM - Sandro Santilli

- % Done changed from 0 to 30

As of 6461a0125b2f83649d1604cbc11fcb6678490ed5 qgis running from build tree would find any python plugin under output/python/plugins. So next thing to do is find an easy way to get plugins under there...

#6 - 2012-10-19 04:18 PM - Sandro Santilli

- % Done changed from 30 to 70

8ca2236134a8d441803c9bdfdfb5dfcbc5536524 provides a PLUGIN_INSTALL macro and makes db_manager use that.
All plugins that are to be loaded from build dir should switch to use that macro now, in order for this to be closed

#7 - 2012-10-20 01:01 PM - Sandro Santilli

plugin_installer plugin ready to run from build dir as of f93f844867e0bbe461ef571f9dc7a6dfdaf3e6

#8 - 2012-10-23 01:36 PM - Sandro Santilli

- % Done changed from 70 to 80

fTools ready with b892a021af26b39285bebe9199c9534deade135d

mapserver_export ready with 9a0c4ffdb56765893764ca294088cb0967ca03d4

?

Still left: osm, sextante and sextanteexampleprovider

#9 - 2012-10-23 01:44 PM - Sandro Santilli

9f1351b08b957f76a570d80c2338ec691550d1a2 does osm, so only left is sextante

#10 - 2012-10-25 01:24 AM - Larry Shaffer

- Status changed from Open to Feedback

A current issue regarding the loading of plugins (while running from the build directory) is when plugins are restored on launch of the app. There is currently a goofy fix for this with commits `commit:e31fb3c9` and `commit:` where `QgsApplication::pkgDataPath()` is temporarily set to something other than `QgsApplication::buildSourcePath()` when restoring core plugins.

?

The reason for that patch: when `QgsPluginRegistry::restoreSessionPlugins()` is called the Python packages are imported from **`QgsApplication::buildSourcePath()/python/plugins`** even though that path is **NOT** in `sys.path` for the interpreter. If `QgsApplication::pkgDataPath()` is pointed to something other than `QgsApplication::buildSourcePath()`, or an empty `QString`, it works. However, **I could find no means by which the interpreter was assigned that module search path.**

?

I have tried:

?

? - changing the current working directory in C++ and via Python

?? - setting `PYTHONPATH`

?? - setting all kinds of debug output from the interpreter (never shows `buildSourcePath()/python/plugins` in `sys.path`)

?? - giving up

?



To reproduce the issue, run QGIS from the build directory and then launch DB Manager core plugin. You will get an error about a missing `ui_*.py` file, because that 'compiled' version of a `*.ui` file does not exist in source directory, only in the build/output/python/plugins staged version of the plugin.



Now, run QGIS again, but with the `--noplugins` option. This will keep `restoreSessionPlugins()` from being called. After using Plugin Manager to turn back on DB Manager, launch the plugin and you should not get the error: `sys.path` is being honored, and the plugin is imported from build/output/python/plugins staged area, as expected.



While the current patch works, it requires core plugins to not request `QgsApplication::pkgDataPath()` when the plugin loads. A better solution is needed.

#11 - 2012-10-25 02:57 AM - Sandro Santilli

- Assignee set to Sandro Santilli
- % Done changed from 80 to 100

Sextante loads as of commit:6ca7ea987d86251ee051b7d7ee974a1e9d78bd8f

I think this ticket could be closed, and Larry's findings about plugins restore should be in a separate ticket.

#12 - 2012-11-12 01:17 PM - Sandro Santilli

Larry, did you file a ticket for the plugin restore issue ?

#13 - 2012-12-22 03:34 PM - Larry Shaffer

Sandro Santilli wrote:



Larry, did you file a ticket for the plugin restore issue ?



Finally. :^)



See issue #6913

#14 - 2014-06-28 07:42 AM - Jürgen Fischer

- *Target version changed from Version 2.0.0 to Future Release - Lower Priority*

#15 - 2014-06-28 03:12 PM - Sandro Santilli

- *Status changed from Feedback to Closed*

Given confirmation of Larry this ticket can be closed. Python plugins are loaded fine from build dir.