QGIS Application - Bug report #6913
Python interpreter starts hardwired to search buildSourcePath when run from build directory

2012-12-22 03:31 PM - Larry Shaffer

Status:	Closed			
Priority:	Normal			
Assignee:				
Category:	Python plugins			
Affected QGIS version:	master	**Regression?:**	No	
Operating System:	All	**Easy fix?:**	No	
Pull Request or Patch supplied:	No	**Resolution:**	end of life	
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	16036	

Description

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.

Related issues:

Related to QGIS Application - Bug report # 5879: running from build directory...	**Closed**	**2012-06-27**

History

#1 - 2017-05-01 01:09 AM - Giovanni Manghi

- Easy fix? set to No

- Regression? set to No

#2 - 2019-03-09 04:04 PM - Giovanni Manghi

- Resolution set to end of life

- Status changed from Open to Closed

End of life notice: QGIS 2.18 LTR

Source:

http://blog.qgis.org/2019/03/09/end-of-life-notice-qgis-2-18-ltr/