

QGIS Application - Bug report #10487

pythonplugin error under qgis 64bit - import wx module failed

2014-06-06 03:57 AM - Jan Lippmann

Status: Closed	
Priority: Normal	
Assignee: Jürgen Fischer	
Category: Python plugins	
Affected QGIS version: master	Regression?: No
Operating System: Windows	Easy fix?: No
Pull Request or Patch supplied: No	Resolution: up/downstream
Crashes QGIS or corrupts data: No	Copied to github as #: 18899

Description

hello,

if i import the wx module in any python plugin under win 7 64bit and qgis 64 bit (master and 2.2 64bit), qgis throw the error below.
can anybody reproduce this error with my testplugin?how can i solve this problem?
there are no problems with qgis 32 bit (same master codeversion and qgis 2.2 32 bit).

thanks a lot :-)

Konnte Erweiterung TestPlugin nicht laden wegen eines Fehler beim Aufruf seiner classFactory()-Methode

Traceback (most recent call last):

```
File "C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python\qgis\utils.py", line 204, in startPlugin
    plugins[packageName] = package.classFactory(iface)
File "C:/Users/Jan/.qgis2/python/plugins\TestPlugin\__init__.py", line 26, in classFactory
    from testplugin import TestPlugin
File "C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python\qgis\utils.py", line 453, in _import
    mod = _builtin_import(name, globals, locals, fromlist, level)
File "C:/Users/Jan/.qgis2/python/plugins\TestPlugin\testplugin.py", line 33, in
    import wx
File "C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python\qgis\utils.py", line 453, in _import
    mod = _builtin_import(name, globals, locals, fromlist, level)
File "C:\\PROGRA~1\\QGISWE~1\\apps\\Python27\\lib\\site-packages\\wx-2.8-msw-unicode\\wx\\__init__.py", line 45, in
    from wx._core import *
File "C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python\qgis\utils.py", line 453, in _import
    mod = _builtin_import(name, globals, locals, fromlist, level)
File "C:\\PROGRA~1\\QGISWE~1\\apps\\Python27\\lib\\site-packages\\wx-2.8-msw-unicode\\wx\\_core.py", line 4, in
    import _core_
File "C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python\qgis\utils.py", line 453, in _import
    mod = _builtin_import(name, globals, locals, fromlist, level)
ImportError: DLL load failed: Das angegebene Modul wurde nicht gefunden.
```

Python-Version:

2.7.5 (default, May 15 2013, 22:44:16) [MSC v.1500 64 bit (AMD64)]

QGIS-Version:

2.3.0-Master Master, ca6d4d9

Python-Pfad: ['C:/Users/Jan/.qgis2/python/plugins\\processing', 'C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python',
u'C:/Users/Jan/.qgis2/python', u'C:/Users/Jan/.qgis2/python/plugins']

```
'C:/PROGRA~1/QGISWE~1/apps/qgis-dev/.python/plugins', 'C:\\\\PROGRA~1\\\\QGISWE~1\\\\bin\\\\python27.zip',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\DLLs', 'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\plat-win',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\lib-tk', 'C:\\\\PROGRA~1\\\\QGISWE~1\\\\bin',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27', 'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\PIL',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\jinja2-2.7.2-py2.7.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\markupsafe-0.23-py2.7-win-amd64.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\pytz-2012j-py2.7.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\win32',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\win32\\\\lib',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\Pythonwin',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\Shapely-1.2.18-py2.7-win-amd64.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\six-1.3.0-py2.7.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\wx-2.8-msw-unicode',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\xlrd-0.9.2-py2.7.egg',
'C:\\\\PROGRA~1\\\\QGISWE~1\\\\apps\\\\Python27\\\\lib\\\\site-packages\\\\xlwt-0.7.5-py2.7.egg',
'C:\\\\Users\\\\Jan\\\\.qgis2\\\\python\\\\plugins\\\\DigitizingTools\\\\tools',
'C:\\\\Users\\\\Jan\\\\.qgis2\\\\python\\\\plugins\\\\mmqgis/forms',
'C:\\\\Users\\\\Jan\\\\.qgis2\\\\python\\\\plugins\\\\QuickMultiAttributeEdit/forms']
```

Related issues:

Related to QGIS Application - Feature request # 18693: include python3-pypiwi...

Closed

2018-04-10

History

#1 - 2014-06-06 06:09 AM - Jürgen Fischer

- Priority changed from High to Normal

#2 - 2014-06-21 06:48 AM - Jürgen Fischer

what do you need wx for? wx is only included for the grass gui frontend - which QGIS doesn't even use.

#3 - 2014-06-21 06:49 AM - Giovanni Manghi

- Status changed from Open to Feedback

#4 - 2014-06-21 08:05 AM - Jürgen Fischer

- Category set to Python plugins

- Tag set to windows

#5 - 2014-06-23 12:36 AM - Jan Lippmann

I use wx for my communication interface (WM_COPYDATA) between qgis and other windows applications.

WM_COPYDATA Infos:

<http://msdn.microsoft.com/en-us/library/windows/desktop/ms649011%28v=vs.85%29.aspx>

Here is a wiki python sample which i use to create my "communicationwindow":

```
import wx
```

```
class TestFrame(wx.Frame):
```

2024-04-27

2/4

```
def __init__(self, parent, title):
    wx.Frame.__init__(self, parent, wx.ID_ANY, title=title)
    text = wx.StaticText(self, label="Hello World!")

app = wx.App(redirect=False)
frame = TestFrame(None, "A little example")
frame.Show()
app.MainLoop()
```

#6 - 2014-06-23 01:10 AM - Jürgen Fischer

- Resolution set to up/downstream
- Status changed from Feedback to Closed

Wouldn't it be more handy to use Qt for that (ie. a window with winEvent in place)?

#7 - 2014-06-23 01:20 AM - Jan Lippmann

hello juergen,

do you have an example link for the solution with QT or are there alternative example code for communication interfaces between qgis and other windows applications?

but generally it's not possible to use wx under 64bit? why it works under 32 bit?

thank you

#8 - 2014-06-23 01:29 AM - Jürgen Fischer

Jan Lippmann wrote:

do you have an example link for the solution with QT or are there alternative example code for communication interfaces between qgis and other windows applications?

Sorry, I don't have a link. But I recently did it in a training.

but generally it's not possible to use wx under 64bit? why it works under 32 bit?

Obviously it is - GRASS does it. Not sure if it's a good idea to mix toolkits and also why it doesn't work in 64bit, while it does in 32bit. But still it's sounds more like a edge-easy packaging problem - as said QGIS doesn't use wx.

thank you

#9 - 2015-01-21 05:11 AM - Jan Lippmann

- Status changed from Closed to Reopened
- Assignee set to Jürgen Fischer
- Target version changed from Future Release - Lower Priority to Version 2.8

Hello Jürgen,

The problem is now under qgis bit 32 in the current master on. You may be the reason the update of Qt to version 4.8.6? Is there a way the previous communication technology (WM_COPYDATA) to continue to use ?:

@import wx

```
class TestFrame(wx.Frame):
    def __init__(self, parent, title):
        wx.Frame.__init__(self, parent, wx.ID_ANY, title=title)
        text = wx.StaticText(self, label="Hello World!")
```

```
app = wx.App(redirect=False)
frame = TestFrame(None, "A little example")
frame.Show()
app.MainLoop()@
```

32 bit

QGIS-Version 2.7.0-Master QGIS-Codeversion f0447db
 Kompiliert gegen Qt 4.8.6 Laufendes Qt 4.8.6
 Kompiliert mit GDAL/OGR 1.11.1 Läuft mit GDAL/OGR 1.11.1
 Kompiliert mit GEOS 3.4.2-CAPI-1.8.2 Läuft mit GEOS 3.4.2-CAPI-1.8.2 r3921
 PostgreSQL-Client-Version 8.3.10 SpatiaLite-Version 4.1.1
 QWT-Version 5.2.1 PROJ.4-Version 480
 QScintilla2-Version 2.8.4 Diese QGIS-Kopie schreibt Debugausgaben.

#10 - 2015-01-21 06:22 AM - Jürgen Fischer

- Status changed from Reopened to Closed

Still not a qgis issue.

Why insist on using wx inside of a Qt application if you can use Qt for the same thing (hint: derive from QDialog and implement winEvent; the incoming message can be decoded in python with ctypes; ctypes can also used to invoke sendMessageW)?

#11 - 2018-04-21 05:50 PM - Jürgen Fischer

- Related to Feature request #18693: include python3-pypwin32 in windows standalone installer added

Files

TestPlugin.zip	23.7 KB	2014-06-06	Jan Lippmann
----------------	---------	------------	--------------