Crash when loading project with custom renderer for vector layers

Status: Closed
Priority: High
Assignee: Martin Dobias
Category: Python plugins
Affected QGIS version: master
Operating System: all
Pull Request or Patch supplied: No
Crashes QGIS or corrupts data: Yes

Description

I have custom renderer for vector layers, and can save project including information about this custom renderer. But when I try to open saved project QGIS crashes. Here is backtrace

```
#0 0x00000000ffffffff in ?? ()
#1 0x00007fffff5426d2 in QgsSymbolV2::mapUnitScale (this=0x7fff5c019040) at /home/alex/devel/cpp/qgis/src/core/symbology-ng/qgssymbolv2.cpp:274
#2 0x00007fffff542cba8 in QgsMarkerSymbolV2::renderPoint (this=0x7fff5c019040, point=..., f=0x7fff63dfdb0, context=..., layerIdx=-1, selected=false) at /home/alex/devel/cpp/qgis/src/core/symbology-ng/qgssymbolv2.cpp:1251
#3 0x00007fffff542929b in QgsSymbolV2::renderFeature (this=0x7fff5c019040, feature=..., context=..., layer=-1, selected=false, drawVertexMarker=false, currentVertexMarkerType=1, currentVertexMarkerSize=3) at /home/alex/devel/cpp/qgis/src/core/symbology-ng/qgssymbolv2.cpp:718
#4 0x00007fffff542a0f6 in QgsFeatureRendererV2::renderFeatureWithSymbol (this=0x8a8bf0, feature=..., symbol=0x7fff5c019040, context=..., layer=-1, selected=false, drawVertexMarker=false) at /home/alex/devel/cpp/qgis/src/core/symbology-ng/qgsrendererv2.cpp:283
#5 0x00007fffff543598 in QgsFeatureRendererV2::renderFeature (this=0x8a8bf0, feature=..., context=..., layer=-1, selected=false, drawVertexMarker=false) at /home/alex/devel/cpp/qgis/src/core/symbology-ng/qgsrendererv2.cpp:277
#6 0x00007fffff52b74e in sipQgsFeatureRendererV2::renderFeature (this=0x8a8bf0, a0=..., a1=..., a2=-1, a3=false, a4=false) at /home/alex/devel/cpp/qgis/build/python/core/sip_corepart1.cpp:37671
#7 0x00007fffff529119 in QgsVectorLayerRenderer::drawRendererV2 (this=0x2fd9780, fit=...) at /home/alex/devel/cpp/qgis/src/core/qgsvectorlayerrenderer.cpp:312
#8 0x00007fffff5289aa in QgsVectorLayerRenderer::render (this=0x2fd9780) at /home/alex/devel/cpp/qgis/src/core/qgsvectorlayerrenderer.cpp:249
#9 0x00007fffff57030c4 in QgsMapRendererCustomPainterJob::doRender (this=0x13e6bd0) at /home/alex/devel/cpp/qgis/src/core/qgsmaprenderercustompainterjob.cpp:251
#10 0x00007fffff570cd81 in QgsMapRendererCustomPainterJob::staticRender (self=0x13e6bd0) at /home/alex/devel/cpp/qgis/src/core/qgsmaprenderercustompainterjob.cpp:214
#11 0x00007fffff5705ab7 in QtConcurrent::StoredFunctorCall1<void, void (*)(QgsMapRendererCustomPainterJob*), QgsMapRendererCustomPainterJob*>::runFunctor (this=0x2fd7320) at /usr/include/qt4/QtCore/qtconcurrentstoredfunctioncall.h:277
#12 0x00007fffff57035a in QtConcurrent::RunFunctionTask<void><>(this=0x2fd7320) at /usr/include/qt4/QtCore/qtconcurrentrunbase.h:134
#13 0x00007fffff4b380 in QThreadPool::run (this=0x346ce90) at concurrent/qthreadpool.cpp:108
#14 0x00007fffff4b0f70 in QThreadPoolPrivate::start (arg=0x346ce90) at thread/qthread_unix.cpp:349
#15 0x00007fffed6f0a4 in start_thread (arg=0x7fff63df700) at pthread_create.c:309
#16 0x00007fffee402d4 in clone () at /sysdeps/unix/sysv/linux/x86_64/clone.S:111
```

Resolution: invalid
Copied to github as #: 22038

2022-01-10

1/2
Test plugin with custom renderer for point layers attached.

Seems there are some issues with interaction between Python bindings and C++ core code.

Related issues:
Related to QGIS Application - Bug report # 13973: Crash when switching from c...
Closed 2015-12-10

Associated revisions
Revision 83eed110 - 2016-01-25 06:16 PM - Martin Dobias

Improve renderer class documentation (refs #14025)

History
#1 - 2015-12-22 07:03 AM - Alessandro Pasotti
- Assignee set to Martin Dobias

Not sure, but this could be the problem:

src/core/symbology-ng/qgssinglesymbolrendererv2.cpp:458

QgsSymbolV2List symbols = const_cast<QgsFeatureRendererV2 *>( renderer )->symbols( context );

the about to be cloned object is a sip object.

#2 - 2016-01-25 08:28 AM - Martin Dobias
- Resolution set to invalid
- Status changed from Open to Closed
- File mytestrenderer.py added

The real issue was that symbols returned by the test renderer have not been initialized before being used in symbolForFeature().

Attached (fixed and simplified) version works fine for me - closing as invalid. I have some ideas how the situation could be improved in QGIS 3 so that custom renderer implementations are less crash-prone...

Files
renderertest.tar.bz2 4.12 KB 2015-12-22 Alexander Bruy
mytestrenderer.py 3.22 KB 2016-01-25 Martin Dobias