

QGIS Application - Bug report #15311

Issue with QgsVectorLayerUndoCommand class (QGIS crashes)

2016-07-19 01:06 PM - gcarrillo -

Status: Open	
Priority: Normal	
Assignee:	
Category: Vectors	
Affected QGIS version: 3.7(master)	Regression?: No
Operating System:	Easy fix?: No
Pull Request or Patch supplied: No	Resolution:
Crashes QGIS or corrupts data: Yes	Copied to github as #: 23243
Description	
<p>I've discovered an issue with QgsVectorLayerUndoCommand class (QgsVectorLayerUndoCommandChangeAttribute.undo() method) when updating an attribute value after featureAdded SIGNAL is emitted.</p>	
<p>Steps to reproduce:</p>	
<ol style="list-style-type: none">1. Open QGIS, load a sample vector layer like this (http://downloads.tuxfamily.org/tuxgis/tmp/test_point.zip) and select it in the Layer List Widget (ToC).2. Run the following code snippet into the QGIS Python Console to connect featureAdded SIGNAL to a custom SLOT that changes an attribute value:	
<pre>myLayer = iface.activeLayer() field = 0 # Please use the test Shapefile provided, if not, adjust the field index def updateFeatureAttrs(fld): f = myLayer.getFeatures(QgsFeatureRequest(fld)).next() geom = f.geometry() myLayer.changeAttributeValue(fld, field, geom.asPoint().x()) myLayer.featureAdded.connect(updateFeatureAttrs)</pre>	
<ol style="list-style-type: none">3. Start editing. Add a new point. (If a form opens click on OK). Don't save anything.4. Stop editing and choose "Close without saving."5. QGIS crashes and prints the following error message:	
<pre>%% Fatal: ASSERT: "it != mBuffer->mAddedFeatures.end()" in file /tmp/build/qgis-2.14.2+20trusty/src/core/qgsvectorlayerundocommand.cpp, line 255 Stacktrace (piped through c++filt): /usr/bin/qgis.bin[0x40a440] /usr/bin/qgis.bin[0x40a548] /usr/lib/x86_64-linux-gnu/libQtCore.so.4(qt_message_output(QtMsgType, char const*)+0x21)[0x7fd508939bb1] /usr/lib/x86_64-linux-gnu/libQtCore.so.4(+0x70ff9)[0x7fd508939ff9] /usr/lib/x86_64-linux-gnu/libQtCore.so.4(qFatal(char const*, ...)+0x94)[0x7fd50893a804] /usr/lib/libqgis_core.so.2.14.2(QgsVectorLayerUndoCommandChangeAttribute::undo()+0x659)[0x7fd5092e2c09] /usr/lib/x86_64-linux-gnu/libQtGui.so.4(QUndoCommand::undo()+0x50)[0x7fd5084144e0]</pre>	

```
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QUndoStack::setIndex(int)+0xc8)[0x7fd508414e58]
/usr/lib/libqgis_core.so.2.14.2(QgsVectorLayerEditBuffer::rollBack()+0x23)[0x7fd5092c4d43]
/usr/lib/libqgis_core.so.2.14.2(QgsVectorLayer::rollBack(bool)+0x46)[0x7fd5092b81e6]
/usr/lib/libqgis_app.so.2.14.2(QgisApp::toggleEditing(QgsMapLayer*,
bool)+0xb4b)[0x7fd50a1b677b]
/usr/lib/libqgis_app.so.2.14.2(+0x478d41)[0x7fd50a4edd41]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QMetaObject::activate(QObject*,
QMetaObject const*, int, void**)+0x2da)[0x7fd508a5a87a]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QAction::triggered(bool)+0x32)[0x7fd507dd9a62]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QAction::activate(QAction::ActionEvent)+0x63)[0x7fd507ddb433]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(+0x57cb02)[0x7fd508192b02]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QAbstractButton::mouseReleaseEvent(QMouseEvent*)+0x6c)[0x7fd508192c2c]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QToolButton::mouseReleaseEvent(QMouseEvent*)+0xa)[0x7fd508249a4a]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QWidget::event(QEvent*)+0xaca)[0x7fd507e2f50a]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QApplicationPrivate::notify_helper(QObject*,
QEvent*)+0x8c)[0x7fd507ddfe2c]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QApplication::notify(QObject*,
QEvent*)+0x3ad)[0x7fd507de65dd]
/usr/lib/libqgis_core.so.2.14.2(QgsApplication::notify(QObject*,
QEvent*)+0x5b)[0x7fd509049c8b]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QCoreApplication::notifyInternal(QObject*,
QEvent*)+0x6d)[0x7fd508a464dd]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QApplicationPrivate::sendMouseEvent(QWidget*,
QMouseEvent*, QWidget*, QWidget*, QWidget**, QPointer<QWidget>&,
bool)+0x153)[0x7fd507de5d93]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(+0x2449cb)[0x7fd507e5a9cb]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(QApplication::x11ProcessEvent(XEvent*)+0x1559)[0x7fd507e5a269]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(+0x26bb02)[0x7fd507e81b02]
/lib/x86_64-linux-gnu/libglib-2.0.so.0(g_main_context_dispatch+0x254)[0x7fd500ceee04]
/lib/x86_64-linux-gnu/libglib-2.0.so.0(+0x49048)[0x7fd500cef048]
/lib/x86_64-linux-gnu/libglib-2.0.so.0(g_main_context_iteration+0x2c)[0x7fd500cef0ec]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>)+0x71)[0x7fd508a737a1]
/usr/lib/x86_64-linux-gnu/libQtGui.so.4(+0x26bbb6)[0x7fd507e81bb6]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QEventLoop::processEvents(QFlags<QEventLoop::ProcessEventsFlag>)+0x2f)[0x7fd508a45af]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>)+0x175)[0x7fd508a453a5]
/usr/lib/x86_64-linux-gnu/libQtCore.so.4(QCoreApplication::exec()+0x89)[0x7fd508a4ab79]
/usr/bin/qgis.bin[0x406fd7]
/lib/x86_64-linux-gnu/libc.so.6(_libc_start_main+0xf5)[0x7fd507358ec5]
/usr/bin/qgis.bin[0x409d48]
Aborted
%%%%%%%%%
```

I've tested both Shapefiles and SpatialLite layers, getting the same error.

Do you have an idea of what could be happening?

I'm using QGIS v.2.14.2 on a Linux Mint machine and also confirmed the crash using QGIS v.2.14.1 on Windows 7.

History

#1 - 2016-07-20 01:26 AM - Martin Dobias

Unfortunately currently if you connect to one of the layer "editing" signals, issuing further editing commands will break the undo stack of the layer (because the previous undo command is not fully finished when a new undo command appears). A workaround for the meanwhile is to delay the follow up editing command - e.g. by making the connection "queued" or by using a QTimer with zero timeout or something similar...

#2 - 2016-07-26 03:57 PM - gcarrillo -

Thank you Martin for your comments. I've been trying with 3 layer editing SIGNALs (featureAdded, geometryChanged, and attributeValueChanged) and the only one that seems to fail is featureAdded. The other 2 don't make QGIS crash and allow QGIS to undo the edits properly.

#3 - 2017-02-04 02:50 AM - Giovanni Manghi

- Priority changed from Normal to High

#4 - 2017-03-07 07:52 AM - Giovanni Manghi

- Target version changed from Future Release - High Priority to Version 2.18

- Affected QGIS version changed from 2.14.2 to 2.18.4

#5 - 2017-05-01 01:05 AM - Giovanni Manghi

- Easy fix? set to No

- Regression? set to No

#6 - 2019-01-29 01:23 PM - Peter Petrik

- Status changed from Open to Feedback

- Description updated

Can you please retest with 3.x release, since 2.18 is no longer developed and let me know if this is still the issue? Thanks

#7 - 2019-03-09 03:09 PM - Giovanni Manghi

- Status changed from Feedback to Closed

- Resolution set to end of life

End of life notice: QGIS 2.18 LTR

Source:

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

QGIS 3.4 has recently become our new Long Term Release (LTR) version. This is a major step in our history – a long term release version based on the massive updates, library upgrades and improvements that we carried out in the course of the 2.x to 3x upgrade cycle.

We strongly encourage all users who are currently using QGIS 2.18 LTR as their preferred QGIS release to migrate to QGIS 3.4. This new LTR version will receive regular bugfixes for at least one year. It also includes hundreds of new functions, usability improvements, bugfixes, and other goodies. See the relevant changelogs for a good sampling of all the new features that have gone into version 3.4

Most plugins have been either migrated or incorporated into the core QGIS code base.

We strongly discourage the continued use of QGIS 2.18 LTR as it is now officially unsupported, which means we'll not provide any bug fix releases for it.

You should also note that we intend to close all bug tickets referring to the now obsolete LTR version. Original reporters will receive a notification of the ticket closure and are encouraged to check whether the issue persists in the new LTR, **in which case they should reopen the ticket.**

If you would like to better understand the QGIS release roadmap, check out our roadmap page! It outlines the schedule for upcoming releases and will help you plan your deployment of QGIS into an operational environment.

The development of QGIS 3.4 LTR has been made possible by the work of hundreds of volunteers, by the investments of companies, professionals, and administrations, and by continuous donations and financial support from many of you. We sincerely thank you all and encourage you to collaborate and support the project even more, for the long term improvement and sustainability of the QGIS project.

#8 - 2019-03-18 12:51 AM - gcarrillo -

- Resolution deleted (end of life)
- Status changed from Closed to Reopened
- Priority changed from High to Normal
- Affected QGIS version changed from 2.18.4 to 3.7(master)

Just tested this on master (v3.7) and it's still happening: QGIS crashes when closing the edit session discarding the edits.

#9 - 2019-03-18 01:12 AM - gcarrillo -

Steps to reproduce the issue in QGIS v3.x:

1. Open QGIS, load a sample vector layer (with a numeric field in the field index 0) and select it in the Layer List Widget (ToC).
2. Run the following code snippet into the QGIS Python Console to connect featureAdded SIGNAL to a custom SLOT that changes an attribute value:

```
myLayer = iface.activeLayer()
field = 0

def updateFeatureAttrs( fld ):
    f = myLayer.getFeature( fld )
    geom = f.geometry()
    myLayer.changeAttributeValue( fld, field, geom.asPoint().x() )

myLayer.featureAdded.connect( updateFeatureAttrs )
```

3. Start editing. Add a new point. (If a form opens click on OK). Don't save anything.
4. Stop editing and choose "Close without saving."
5. QGIS crashes and prints this error:

```
src/gui/qgsmaptool.cpp:92 : (activate) [3241ms] Cursor has been set
src/app/qgsmaptooldigitizefeature.cpp:155 : (cadCanvasReleaseEvent) [601ms] savePoint = 269, -100
src/app/qgsfeatureaction.cpp:175 : (addFeature) [0ms] reuseLastValues: 0
src/app/qgsfeatureaction.cpp:293 : (onFeatureSaved) [1838ms] reuseLastValues: 0
src/gui/qgsmapcanvas.cpp:499 : (refresh) [0ms] CANVAS refresh - invalid settings -> nothing to do
Fatal: ASSERT: "it != mBuffer->mAddedFeatures.end()" in file /docs/dev/qgis/QGIS/src/core/qgsvectorlayerundocommand.cpp, line 234
QGIS died on signal -1Could not attach to process. If your uid matches the uid of the target
process, check the setting of /proc/sys/kernel/yama/ptrace_scope, or try
again as the root user. For more details, see /etc/sysctl.d/10-ptrace.conf
ptrace: Operation not permitted.
No thread selected
No stack.
gdb returned 0
```

Aborted (core dumped)

#10 - 2019-03-20 12:37 PM - Giovanni Manghi

- Status changed from Reopened to Open