QGIS Application - Bug report #18504
Qgis crash when info over geometry was cliccked

2018-03-21 02:30 PM - Gerardo Aliberti

Status:	Closed		
Priority:	Normal		
Assignee:			
Category:	C++ Plugins		
Affected QGIS version:	3.0.0	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	26392

Description

User Feedback

Report Details

Crash ID: 5a80ca2d33ad0bea355da4b3f1628b5b29334b24

Stack Trace

```
QTransform::type :
QRasterPaintEngine::drawImage :
QPainter::drawImage :
QgsHighlight::paint :
QGraphicsScene::dragMoveEvent :
QGraphicsScene::drawItems :
QGraphicsScene::drawItems :
QGraphicsView::paintEvent :
QWidget::event :
QFrame::event :
QGraphicsView::viewportEvent :
QCoreApplicationPrivate::sendThroughObjectEventFilters :
QApplicationPrivate::notify_helper :
QApplication::notify :
QgsApplication::notify :
QCoreApplication::notifyInternal2 :
QWidgetPrivate::drawWidget :
QWidgetPrivate::paintSiblingsRecursive :
QWidgetPrivate::drawWidget :
QWidgetPrivate::paintSiblingsRecursive :
QWidgetPrivate::drawWidget :
QWidgetPrivate::paintSiblingsRecursive :
QWidgetPrivate::drawWidget :
QWidgetPrivate::paintSiblingsRecursive :
QWidgetPrivate::paintSiblingsRecursive :
QWidgetPrivate::drawWidget :
QApplication::windowIcon :
QWidget::event :
QMainWindow::event :
QgisApp::event :
QApplicationPrivate::notify_helper :
QApplication::notify :
QgsApplication::notify :
QCoreApplication::notifyInternal2 :
QCoreApplicationPrivate::sendPostedEvents :
QGraphicsScene::~QGraphicsScene :
QObject::event :
QGraphicsScene::event :
QApplicationPrivate::notify_helper :
QApplication::notify :
```

QgsApplication::notify :
QCoreApplication::notifyInternal2 :
QCoreApplicationPrivate::sendPostedEvents :
qt_plugin_query_metadata :
QEventDispatcherWin32::processEvents :
DispatchMessageW :
NotifyWinEvent :
QEventDispatcherWin32::processEvents :
qt_plugin_query_metadata :
QEventLoop::exec :
QCoreApplication::exec :
main :
BaseThreadInitThunk :
RtlUserThreadStart :

QGIS Info
QGIS Version: 3.0.0-Girona
QGIS code revision: commit:001c80b0c3
Compiled against Qt: 5.9.2
Running against Qt: 5.9.2
Compiled against GDAL: 2.2.3
Running against GDAL: 2.2.3

System Info
CPU Type: x86_64
Kernel Type: winnt
Kernel Version: 6.3.9600

Associated revisions

Revision 45c400c2 - 2018-04-09 04:43 AM - Nyall Dawson

QgsSvgCache fetches remote SVG files in a background task

Previously QgsSvgCache would often try to fetch remote images
using a network request on the main thread, by calling
processEvents repeatedly until the request was complete.

This caused lots of bugs, since the main thread processEvents
would proceed with all kinds of stuff assuming that the
svg fetch operation was complete, leading to frequent crashes
and deadlocks and making remote svg use impossible (it's
likely that the SVG cache remote fetching code was written
in the pre-multi-threaded rendering era).

There's no way to fix this with async svg fetching - we
HAVE to remove the processEvents call, and a QEventLoop
won't help either (since the method may be called on the
main thread). Accordingly the only solution is to
fetch the requested svg in the background, and return
a temporary "downloading" svg for use in the meantime.
We use a QgsNetworkContentFetcherTask to do this, so it's
nicely integrated with task manager.

A request task is fired up when a remote svg is requested
for the first time, with the temporary downloading svg
returned for use by the caller asynchronously. QgsSvgCache

then emits the remoteSvgFetched signal when a previously
requested remote SVG has been successfully fetched,
triggering a map canvas redraw with the correct SVG
graphic.

Fixes #18504

Revision da0de649 - 2018-04-09 06:37 AM - Nyall Dawson

QgsSvgCache fetches remote SVG files in a background task

Previously QgsSvgCache would often try to fetch remote images
using a network request on the main thread, by calling
processEvents repeatedly until the request was complete.

This caused lots of bugs, since the main thread processEvents
would proceed with all kinds of stuff assuming that the
svg fetch operation was complete, leading to frequent crashes
and deadlocks and making remote svg use impossible (it's
likely that the SVG cache remote fetching code was written
in the pre-multi-threaded rendering era).

There's no way to fix this with async svg fetching - we
HAVE to remove the processEvents call, and a QEventLoop
won't help either (since the method may be called on the
main thread). Accordingly the only solution is to
fetch the requested svg in the background, and return
a temporary "downloading" svg for use in the meantime.
We use a QgsNetworkContentFetcherTask to do this, so it's
nicely integrated with task manager.

A request task is fired up when a remote svg is requested
for the first time, with the temporary downloading svg
returned for use by the caller asynchronously. QgsSvgCache
then emits the remoteSvgFetched signal when a previously
requested remote SVG has been successfully fetched,
triggering a map canvas redraw with the correct SVG
graphic.

Fixes #18504

(cherry-picked from 45c400c25)

History

#1 - 2018-03-21 06:35 PM - Gerardo Aliberti

Issue happens if the style/simbology icon of the layer points to an non local path (es. http://myurl/milcon.svg)

#2 - 2018-03-29 12:46 AM - Nyall Dawson

- Status changed from Open to In Progress

#3 - 2018-04-09 04:42 AM - Nyall Dawson

- % Done changed from 0 to 100

- Status changed from In Progress to Closed

Applied in changeset commit:qgis|45c400c25c128061923b0aded682557653ad1c6a.