QGIS Application - Bug report #17809
Rendering performance issues

2018-01-05 07:06 PM - Saber Razmjooei

Status:	Closed			
Priority:	High			
Assignee:				
Category:	Map Canvas			
Affected QGIS version:	master	**Regression?:**	Yes	
Operating System:		**Easy fix?:**	No	
Pull Request or Patch supplied:		**Resolution:**	not reproducable	
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	25705	

Description

I have done some tests rendering a single Shapefile using QGIS 2.18 and master. Rendering time in the master is significantly slower than in 2.18.

To reproduce the issue:

1- Load a relatively large file in QGIS master

2- Under Settings > Options > Rendering > Debugging, enable Map canvas referesh

3- Open the log message toolbar and under Rendering tab, check the time it takes to refresh the map

4- Repeat the same for QGIS 2.18 for the same map extent

For my test file with 14410 features, QGIS master (421ef88de0) takes 52% time longer than QGIS 2.18.5

Associated revisions

Revision 65827adc - 2018-02-11 11:56 PM - Nyall Dawson

Provide an optimised constructor for a null QgsRectangle

This is used a lot, yet the current constructor calls the
normalize() method which does a bunch of operations for no
result.

So instead provide a simple optimised null QgsRectangle
constructor and save a lot of cycles.

Refs #17809

Revision ff7bb9aa - 2018-02-11 11:56 PM - Nyall Dawson

Remove inefficient method for creating empty geometry of same
WKB type, and use optimized method instead

Refs #17809

Revision c7e257ef - 2018-02-12 12:04 AM - Nyall Dawson

More efficient simplification of linestring geometries during rendering

Refs #17809

Revision db12f006 - 2018-02-12 12:04 AM - Nyall Dawson

Avoid cloning in QgsGeometry::convertToMultiType where possible

Refs #17809

Revision b60489db - 2018-02-12 12:04 AM - Nyall Dawson

More efficiency improvements for map to pixel simplification

Refs #17809

Revision e4ab1a67 - 2018-02-12 12:04 AM - Nyall Dawson

Don't generate symbol expression context scope when it's not
being used

Refs #17809

History

#1 - 2018-01-05 09:07 PM - Nyall Dawson
- Status changed from Open to Feedback

Are you running a debug enabled build?

#2 - 2018-01-06 03:18 AM - Mathieu Pellerin - nIRV

I haven't done such test myself, but if it's not a debug-specific issue, it might have to do with QGIS 3.0 properly advertising and upgrading features to
Multi*.

#3 - 2018-01-06 07:42 AM - Nyall Dawson

I strongly suspect it's comparison between a debug enabled build (such as the osgeo4w ones) and a release build.

#4 - 2018-01-06 10:18 AM - Saber Razmjooei

I have both QGIS master and 2.18 compiled from source with pretty much the same settings (apart from qt and python).
Neither of them are built with debug mode enabled (CMAKE_BUILD_TYPE is left blank).

#5 - 2018-01-06 10:18 AM - Saber Razmjooei
- Status changed from Feedback to Open

#6 - 2018-01-15 02:55 PM - Andreas Neumann

Is this performance regression limited to Shapefiles or does it also surface with other data providers? Namely Geopackage or Postgis?

#7 - 2018-01-15 03:37 PM - Saber Razmjooei

Here is a zip file containing:

https://www.dropbox.com/s/496efk99vfspme1/17809.zip?dl=0

- 2 projects: one in QGIS 3 and another in QGIS 2
- Bot projects read the following files:
  - 4 layers from a geopackage file
  - A Shapefile
  - A geotiff file

For me, QGIS 3 project takes just over 15 seconds. QGIS 2 project takes 10 seconds. See above for my QGIS configurations and set up: both latest 2.18 and master compiled from source with no debugging and same configurations (apart from python and qt)

#8 - 2018-01-17 04:14 AM - Nyall Dawson

- Status changed from Open to Feedback

Just to check - do you have a hidpi display?

#9 - 2018-01-20 02:58 PM - Saber Razmjooei

- Operating System set to Kubuntu 17.10
- Status changed from Feedback to Open

Correct...it is hidpi indeed.

#10 - 2018-01-20 08:43 PM - Nyall Dawson

- Status changed from Open to Feedback

This probably relates. Since qt5 respects hi dpi. Qgis is now drawing a LOT more pixels in the canvas (maybe 4 times as many, or even more). A good test would be to manually set your resolution to something lowdpi and compare 2.18 and 3.

#11 - 2018-01-25 09:55 AM - Saber Razmjooei

- Status changed from Feedback to Open

I have tested it with lower no hidpi resolution (on an external monitor) with the same results....i.e. still slower to render/load in QGIS 3 than in QGIS 2.

#12 - 2018-02-09 08:20 AM - Nyall Dawson

Can you test https://github.com/qgis/QGIS/pull/6300 ? I'm not hopeful, but that's all I was able to find by profiling. Everything else looks correct - so it's

possible this is related to a Qt5 speed regression.

#13 - 2018-02-12 01:15 AM - Nyall Dawson

- Status changed from Open to Feedback

Saber - can you please run a test of this code to determine whether the speed regression you see is related to the rendering or layer feature iteration?

```
layer = iface.activeLayer()
   for f in layer.getFeatures():
      pass
```

Run it over some large layers in both 2.18 and 3.0, and see if there's any significant difference in the time it takes...

#14 - 2018-02-28 09:09 PM - mr twister

I have the same issue with large project files (many layers, many features per layer). I tested 3.0.0 and master on Windows. Not only map rendering is slow, but the whole GUI is slow - or better - laggy. Nyall, with your test I didn't get any differences.

#15 - 2018-03-02 11:53 AM - mr twister

- File rendering_debugging.txt added

I did some tests (turning labels off; removing WMS layers; disable line anti aliasing) but ditn't get any improvements.

I also turned on debugging for Map canvas refresh and got some warnings (see attached file).

Also QGIS sometimes keeps beeing listed as process in task manager after closing.

#16 - 2018-03-02 05:07 PM - Saber Razmjooei

Could you also check you are using the same settings for multi-threaded rendering?

#17 - 2018-03-03 06:31 PM - Giovanni Manghi

- Regression? changed from No to Yes

#18 - 2018-03-05 12:24 PM - mr twister

Saber Razmjooei wrote:

> *Could you also check you are using the same settings for multi-threaded rendering?*

It makes no difference whether multi-threaded rendering is activated or not.

#19 - 2018-03-06 11:43 AM - Giovanni Manghi

- Status changed from Feedback to Open

#20 - 2018-03-14 06:32 PM - Justin Gamble

Same here, running QGIS 3.0 on Windows 7. Have built 2 large projects (10+ layers) and even with only simple polyline features rendering is very slow

#21 - 2018-03-14 09:48 PM - Nyall Dawson

There's really nothing we can do without more troubleshooting here -- there must be something specific to the setups or data where this issue is encountered, and that's what we need to know.

#22 - 2018-03-15 01:53 AM - Nyall Dawson

- Status changed from Open to Feedback

#23 - 2018-03-15 12:07 PM - Giovanni Manghi

maybe is related to the slowness many are seeing in rendering of postgis layers?

#24 - 2018-03-15 12:14 PM - Saber Razmjooei

- Operating System deleted (Kubuntu 17.10)

Please see the link to the example file I have posted earlier, which is based on flat files (gpkg):

https://www.dropbox.com/s/496efk99vfspme1/17809.zip?dl=0

It will be great if someone else could test this project and report the rendering time in QGIS 2.18 and 3.0

#25 - 2018-03-15 10:21 PM - Nyall Dawson

Saber - I tested that project, no issue here.

#26 - 2018-04-12 10:55 AM - mr twister

I did some research with Saber's and my projects: Both run smoothly when stored on a local path. However when loaded from a network drive, QGIS 3 is getting slow. With QGIS 2.18 I have no such problems - it makes no difference whether the files are stored on a network drive or locally.

#27 - 2018-06-29 03:39 PM - mr twister

Still present in 3.2. Could this be related with the following QT-issue?

https://bugreports.qt.io/browse/QTBUG-13182?focusedCommentId=201836&page=com.atlassian.jira.plugin.system.issuetabpanels%3Acomment-tbpanel#comment-201836

#28 - 2018-07-17 10:48 PM - Spencer Gardner

I've experienced *very* slow rendering times as well that I think must be related. In my case it's with PostGIS layers pulled over our network. The slowdown seems to be related to network latency. When I use layers on a PostGIS server in my local network things aren't terribly slow. When I reference a layer on a server across the country it's painfully slow. This was never an issue in the 2.x series.

I'd love to help troubleshoot further but I'm not sure what info can be useful at this point.

#29 - 2018-07-18 01:47 AM - Nyall Dawson

Spencer are you on 3.2?

#30 - 2018-08-04 02:19 AM - Willem Buitendyk

QGIS 3.2.1 on the Mac has far too many network issues. WFS 1.0, 1.1 or 2.0 won't work at all - network timeouts. Postgis will load the data but the attribute data will show 'error' in every field. On windows WFS 1.0 and 1.1 will work but the data comes in one big lump instead of incrementally like QGIS 2.18 or possibly even QGIS 3.1? I was about to try under linux but might leave that for another day.

#31 - 2018-08-04 10:34 AM - Giovanni Manghi

Willem Buitendyk wrote:

> *QGIS 3.2.1 on the Mac has far too many network issues. WFS 1.0, 1.1 or 2.0 won't work at all - network timeouts.*

so the same exact layers/connections work without issues on 2.18? Can you share the url?

> *Postgis will load the data but the attribute data will show 'error' in every field.*

the same datasource is ok on 2.18?

#32 - 2018-11-13 08:50 AM - mr twister
- File 3.5.LOG added

Since I still have this issue, I tested the 3.5.0 nightly build. With DebugView I get two different and constantly repeating warnings (see full log attached). With 2.18.25 nightly there are no such warnings.

```
00000215   31.26636505   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[78ms] draw request outside view extent.
00000216   31.26671982   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[0ms] draw request outside view extent.
00000217   31.26707649   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[0ms] draw request outside view extent.
00000218   34.49539948   [7316] ..\..\..\src\providers\ogr\qgsogrprovider.cpp(3456) : (QgsOgrProvider::crs) [3229ms] no spatial reference found

00000219   34.50426865   [7316] ..\..\..\src\providers\ogr\qgsogrprovider.cpp(3456) : (QgsOgrProvider::crs) [16ms] no spatial reference found
00000220   34.52859497   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[31ms] draw request outside view extent.
00000221   34.52896881   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[0ms] draw request outside view extent.
00000222   34.57506180   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
[47ms] draw request outside view extent.
00000223   34.57543182   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)
```

[0ms] draw request outside view extent.

00000224   34.57579803   [7316] ..\..\..\src\core\raster\qgsrasterlayerrenderer.cpp(140) : (QgsRasterLayerRenderer::QgsRasterLayerRenderer)

[0ms] draw request outside view extent.

#33 - 2019-01-14 11:58 PM - Nyall Dawson

- Resolution set to not reproducable

- Status changed from Feedback to Closed

Closing this report -- it's become a mix of conflicting possibly-related issues. Let's open specific issues if there's outstanding issues still in 3.4.3

#34 - 2019-04-25 02:43 PM - mr twister

For me this issue is solved by replacing our old network storage (Netgear ReadyNAS Duo) with a new device. I think the cause of this issue was a high network latency of the old NAS in combination with Qt 5.x.

Files

rendering_debugging.txt	41 KB	2018-03-02	mr twister
3.5.LOG	39.9 KB	2018-11-13	mr twister