

QGIS Application - Bug report #7458

QGIS crashes when starting with a project while loading print composers

2013-03-27 04:14 AM - Andreas Neumann

Status: Closed	
Priority: Severe/Regression	
Assignee: Larry Shaffer	
Category: Project Loading/Saving	
Affected QGIS version: master	Regression?: No
Operating System:	Easy fix?: No
Pull Request or Patch supplied: No	Resolution: fixed
Crashes QGIS or corrupts data: Yes	Copied to github as #: 16413

Description

When starting QGIS with the --project switch with a project containing print composer definitions it crashes on loading print composers.

It crashes in loadComposersFromProject.

If one manually loads the same project after QGIS has started without a project, QGIS does not crash and the project loads fine.

History

#1 - 2013-03-27 11:11 AM - Larry Shaffer

Hi Andreas,

Can not confirm on Mac.

Does this happen consistently for the projects that do crash?

Try this: in Options->General set the 'Open project on launch' to 'Most recent' and apply settings; then, open the problem project manually. On next launch (just launching app, not from file) QGIS should auto-open the most recent project. Please let me know if it crashes then.

You can also just set the 'Open project on launch' to 'Specific' and directly specify the project's path.

If it continues to crash (in either setup), subsequent launches should show a red (critical) message bar indicating that the auto-opening project could not be previously opened, and the Options->General set the 'Open project on launch' will be reset to 'New'.

This will help to determine if the problem stems from the code I introduced there, or possibly something wrong in the new Composer GUI updates.

Also, do you already have 'Open project on launch' set to something other than New, then loading a project from the command line?

#2 - 2013-03-27 11:22 AM - Matthias Kuhn

Larry, the code crashing is not being compiled on Mac.

lines 4790 to 4793 in qgisapp.cpp

```
#ifndef Q_OS_MACX
    composer->setWindowState( Qt::WindowMinimized );
    composer->show();
#endif
```

#3 - 2013-03-27 11:28 AM - Larry Shaffer

Matthias Kuhn wrote:

Larry, the code crashing is not being compiled on Mac.

lines 4790 to 4793 in qgisapp.cpp

```
#ifndef Q_OS_MACX
composer->setWindowState( Qt::WindowMinimized );
composer->show();
#endif
```

Ok, I updated that code and tested on Ubuntu and Win 7, but maybe my test composers weren't complicated enough (or something like that). Could you try "reverting that change in the commit":commit:160a5c47273a8052c7155e451961be379f63264e#src/app/qgisapp.cpp, and see if that helps?

#4 - 2013-03-27 11:32 AM - Matthias Kuhn

No. Still crashes...

#5 - 2013-03-27 11:33 AM - Andreas Neumann

- *Crashes QGIS or corrupts data changed from No to Yes*

Will have to test tomorrow at work - I am at home now and do not have the projects at hand. I wasn't aware about this new setting "Open project on launch" - so I did not use that (knowingly).

We had a similar problem some time ago, which was fixed by you (I did not report a bug - just mentioned it on the qgis-dev list) - then the problem was gone. Now there seems to be a regression.

See e-mail thread at <http://lists.osgeo.org/pipermail/qgis-developer/2013-February/024469.html>

and your fix at

commit:cc0c2b1853d1608a43845b05dd9677b6eb8f03db

Maybe this is related?

Will do more tests tomorrow.

#6 - 2013-03-27 11:36 AM - Larry Shaffer

Matthias Kuhn wrote:

No. Still crashes...

And when you comment those lines out?

#7 - 2013-03-27 01:11 PM - Marco Hugentobler

No crash for me here on KDE / Kubuntu 12.04

#8 - 2013-03-27 01:18 PM - Matthias Kuhn

The problem is related to svg symbol loading (commit:34f3d50099c7e90e614fc68a5ebe37c9bd04cae0) details have been discussed on #qgis and Larry is currently working on a patch

No need for further investigation. Sorry for not updating this topic.

#9 - 2013-03-27 07:57 PM - Larry Shaffer

Ok, I think this may work (I can't get anything to crash on launch with project from command line, so...):

https://github.com/dakcarto/Quantum-GIS/tree/fix_7458_2

Please test. It's not ideal. The repetitive loading of the same SVG previews for every composer picture is what is slowing things down (and the need for the busy indicator), especially on Mac. The SVG preview list widget should be moved to model-view setup or a shared cache. (Don't have time to do that this week, myself.)

#10 - 2013-03-28 12:14 AM - Matthias Kuhn

This fix works here (just gave it a short test and no crash).

#11 - 2013-04-18 06:42 AM - Matthias Kuhn

- Assignee set to Larry Shaffer

Larry, did you merge this branch?

#12 - 2013-04-18 07:12 AM - Larry Shaffer

Well, I could but it is not a good solution, more a band-aid, loading all SVG previews for all composer picture items (can be very slow if there are a lot) *before* the composer is shown.

Recently I pulled the [SVG selector for symbology out into its own class](#) to use in the label background settings, which, if edited further will offer a much better and basically permanent fix.

Here is what needs done to the class:

- Allow passing in already constructed QgsSvgSelectorListModel and QgsSvgSelectorGroupsModel. This will allow a base SVG and groups model that can be constructed in QgsApp or QgsComposer, independent of the views.
- Add a get/setter for retrieving/adding SVG search paths. This way, any instance of the views can offer the ability to edit the search paths.
- Add a parameter for construction that builds the widget in code instead (currently a .ui file), offering the option for horizontal (side-by-side) or vertical (groups above SVG listing) generation of the layout. This way the widget can be dropped in and easily configured for horizontal- or

vertical-aspect-prominent dialogs.

Shouldn't take too long, and could possibly be done under the concept of fixing this bad bug. Unfortunately, I don't have the time prior to 2.0.

Then, here's what can be done in the Print Composer:

- Add the widget to composer's picture item panel, setting its orientation to vertical (groups above SVG listing).
- On construction of the composer, the SVG selector widget for each composer picture item added to the QGraphicsScene does not need to load all of the SVG previews, just reference the existing model. This should be a much better model-view approach and not take long to load.
- Composer uses a set of special SVG paths in addition to the base search paths. This could be continued by utilizing the get/setter for the search paths. However, I think it is odd that there are search paths only available in the composer, and that the functionality be dropped in favor of only the paths that can be defined in the Options dialog, or added to the base search paths, if it is not.

Additionally, the SVG selector widget could have simple + and - buttons (or just one that says 'search paths', etc.) that pops up the SVG paths editor. All of this could be tied into remote repositories for SVGs as well, etc. like gradients in symbology.

Also, the SVG selector widget needs 'backported' to the symbology code, which I did not edit, so changes to the selector are available there as well. And, if the models are part of QgisApp, then the new label backgrounds SVG selector can be quickly adapted to use them as well.

#13 - 2013-05-15 01:21 AM - Andreas Neumann

Here is a dump-file from a crash when loading one of my projects by double-clicking the file in Windows explorer.

#14 - 2013-05-15 01:35 AM - Larry Shaffer

Hi Andreas,

In the labeling gui redesign branch I'm about to finish (today or tomorrow), I have reworked QgsSvgSelectorWidget to be a standalone SVG selector dialog. This aided in getting a data defined SVG setup for label backgrounds.

While this doesn't implement the better 'base SVG and groups model' idea noted above, it does offer a solid solution for this issue for 2.0. However, it does mean that the SVG previews will not be shown as they are now. There will be a line edit for SVG file path (already there), then a button to pop up the SVG selector. This essentially bypasses the issue, since the SVG previews are only ever loaded on user's interactive choice.

The SVG selector dialog does have signals available, so it is still possible to have a live update of the picture on the Composer canvas upon just selecting an SVG preview in the detached dialog. It does introduce an initial extra click to open the dialog and another to close it.

#15 - 2013-05-15 01:54 AM - Andreas Neumann

Hi Larry,

This sounds fine. I think it is not problem that there is an extra step for selecting the SVG symbol. Selecting an SVG for a symbol definition or composer is not a step that is done very often.

Would this also mean that the project loading would be a bit faster? I see that a lot of time is spent going through the SVG symbol library after the project loaded.

Thank you a lot for your work! Looking forward to see this fixed.

Andreas

#16 - 2013-05-27 06:06 AM - Andreas Neumann

I tried with the current master (which should have your revision you mention above, Larry?) and QGIS still crashes on program load.

#17 - 2013-05-27 01:33 PM - Larry Shaffer

Hi Andreas,

The fix is not pushed to master yet. I should be able to do that sometime this week. Sorry for the delay.

#18 - 2013-06-01 04:00 PM - Larry Shaffer

- Status changed from Open to Feedback

Hi Andreas and Matthias,

Should work now, as an OK fix, with commit commit:e2bd04f. Please test.

#19 - 2013-06-03 12:28 AM - Andreas Neumann

Hi Larry,

This looks very good.

I tested around 10 projects and do not get crashes anymore. The projects also load faster as the SVG preview generation is now postponed until the symbols are really needed.

And the annoying blinking in the SVG preview panel of the print composer is now gone.

Very nice - thank you for your work!

From my point of view we can close this bug.

Andreas

#20 - 2013-06-03 12:31 AM - Giovanni Manghi

- Status changed from Feedback to Closed

- Resolution set to fixed