

QGIS Application - Bug report #17785

Virtual layers (nested), all layers disappear from project file (.qgs) when the project is saved

2018-01-04 01:44 PM - Christian Nielsen

Status:	Closed	
Priority:	Normal	
Assignee:		
Category:	Virtual Layers	
Affected QGIS version:	2.18.2	Regression?: No
Operating System:		Easy fix?: No
Pull Request or Patch supplied:	No	Resolution: end of life
Crashes QGIS or corrupts data:	No	Copied to github as #: 25681
Description		
When using nested virtual all layers disappear from project file (.qgs) when the project is saved, if not the individual virtual layers have been manually updated. Example: <pre>layer_1: SELECT * FROM point_file WHERE point_id = 1 layer_2: SELECT st_buffer(geometry,100000) FROM layer_1</pre> If I change the point_id referenced in the WHERE clause for layer_1, I need to edit the layer_2 expression again (without any changes) for it to update. And if I save and close the project without doing that manual update, when i re-open later the project file will be corrupted and all layers will be missing from the project (including regular vectors/rasters). /Christian		

History

#1 - 2018-01-05 10:57 AM - Jakob Lanstorp

I can confirm this issue in QGIS 2.18.15. There should be an automatically or cascading update on virtual layers depending on each other. In the mean time you can condense you query to one virtual layer with a common table expression (with clause).

with pointid1 as

```
(
  select geometry from point_file where id = 1
)
select st_buffer(geometry, 10) from pointid1;
```

or just add a point_id where clause to layer_2

#2 - 2018-01-08 08:51 PM - Christian Nielsen

Thanks, I'll look into your suggested workaround in the meantime.

/Christian

Jakob Lanstorp wrote:

I can confirm this issue in QGIS 2.18.15. There should be an automatically or cascading update on virtual layers depending on each other. In the

mean time you can condense you query to one virtual layer with a common table expression (with clause).

```
with pointid1 as
(
select geometry from point_file where id = 1
)
select st_buffer(geometry, 10) from pointid1;

or just add a point_id where clause to layer_2
```

#3 - 2018-02-22 07:43 PM - Giovanni Manghi

- Priority changed from Normal to High

#4 - 2018-10-23 10:53 AM - Regis Haubourg

- Status changed from Open to Feedback

- Priority changed from High to Normal

Trying to replicate here.

If I change the point_id referenced in the WHERE clause for layer_1, I need to edit the layer_2 expression again (without any changes) for it to update

Can you explain more precisely how you change the point_id reference ? Manually in the Qggs?

Saving cascaded virtual layers in qgs file works here, in 2.18.21 and latest master 3.3. I just don't get how you try to change the definition of layer_1.

And I agree that embedding sub-queries directly is more elegant and less risky.

#5 - 2019-01-21 12:34 AM - Jürgen Fischer

Please test with QGIS 3.4 - QGIS 2.18 reached its end of life.

#6 - 2019-03-09 03:10 PM - Giovanni Manghi

- Resolution set to end of life

- Status changed from Feedback to Closed

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 3.x 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.