QGIS Application - Bug report #20439
Layers not being properly added to canvas from spatialite

2018-11-10 10:58 AM - Dan Isaacs

Status:	Feedback		
Priority:	Normal		
Assignee:	Julien Cabieces		
Category:	DB Manager		
Affected QGIS version:	3.6.0	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	28259

Description

Basically, a table in an sqlite database (created by importing a layer from the exisitng project in QGIS), when added to the canvas, does not show any features on the map. The attribute table shows all the features as expected, even 'Zoom to Layer' zooms to exactly the extents of where the points should be, but there are none there.

It's not just a display issue because, whilst 'Select All' shows all the features selected in the attribute table, none of the geographical selection methods ('Select by Radius' for example) work when dragged across where the points should be.

Interestingly, if I create a view by selecting features from the table, the features show up fine. Also if I choose 'Update SQL Layer' and just update it without making any changes, the points all show up. The trouble is that views cannot be edited and I need a proper functioning editable spatialite layer.

The trouble is, the problem seems to be intermittent. I've tried creating a test project to upload with this issue, but in the test project it works fine.

My guess is that it's something to do with the way DB manager handles all the geometry columns tables that are added when a new geometry layer is added. I've looked at the indices on an ordinary SQL database viewer and there are indices leftover from tables that were removed long ago. I can't see exactly how it could happen, but it's the only thing I can think of that would be different between the actual project I'm working on (which doesn't work) and the test project where it works as expected.

History

#1 - 2018-11-10 11:06 AM - Dan Isaacs

I've just managed to pin down the problem a bit more. I was, by chance, loading the same layer into the project under a different table name and found it added to the canvas exactly as expected. After a bit more experimentation, the problem seem to be confined to importing a table to replace an existing one. Whenever 'Replace destination table (if exists)' is used (presuming a table with the same name actually does exist), the table will not display properly when added to the canvas. Obviously, DB manager is not properly updating some parameter when the table is updated.

It might be best to remove the option to replace an existing table in the dialog until it works as expected.

#2 - 2019-01-18 06:12 PM - Julien Cabieces

- Status changed from Open to Feedback

- Assignee set to Julien Cabieces

There was an issue corrected recently that looks like the one you describe.

Could you please check if you have the problem with current 3.4 nightly release ?

If there is still the issue, could you please describe a set of instructions so I can reproduce.

#3 - 2019-03-02 09:18 AM - Dan Isaacs

Julien Cabieces wrote:

> *There was an issue corrected recently that looks like the one you describe.*
>
> *Could you please check if you have the problem with current 3.4 nightly release ?*
>
> *If there is still the issue, could you please describe a set of instructions so I can reproduce.*

Just had a chance to test this with 3.6, and the problem persists exactly the same. What was the problem that was solved recently?

Steps to reproduce;

1. Create a layer in spatialite database, with more than one feature.
2. Delete that layer in DB manager.
3. Close Qgis and re-open to get DB manager to forget about the deleted table name (otherwise it will not allow the next stage)
4. Import a layer (again with multiple features) into the same database with the same name as the layer previously deleted. This layer should have different features to the first.
5. Adding this layer to Qgis will cause errors in display.
6. For comparison, import the same layer, but this time call it something new. Adding the layer to Qgis works perfectly.

A more simple way to look at the problem, download DB Browser for SQLITE. This allows you to open an Sqlite file and view all tables. Carry out steps 1 and 2. Load the Sqlite into DB Browser and look at the idx_ and views_geometry_ tables. All the deleted layers, every layer that's ever been deleted, are still there in the geometry tables, and what's worse when you create a new layer with the name of an old layer, the tables don't get updated with the new layer's data, they just get left with the old layer.

I don't know how DB manager is programmed to create triggers, but all that needs to be done is to get it to delete geometry table entries when a table is dropped and I think this would solve the problem.

#4 - 2019-03-02 09:45 AM - Dan Isaacs

Actually, I've just noticed that this is already a known issue from 3 years ago [[https://issues.qgis.org/issues/14239]]. You seem to suggest, in the issue I'm referring to, that we need to wait for spatialite to fix the problem, is that mainly the problem, that they are not very well supported anymore? It'd be good to know (if you have any experience on the matter) because I invest quite a bit of work in spatial data and if spatialite is not a good format, I might be better off choosing another.

#5 - 2019-03-02 12:19 PM - Giovanni Manghi

> *Steps to reproduce;*
>
> *1. Create a layer in spatialite database, with more than one feature.*
> *2. Delete that layer in DB manager.*
> *3. Close Qgis and re-open to get DB manager to forget about the deleted table name (otherwise it will not allow the next stage)*

This (point 3) is bug itself, it as already been reported?

> *4. Import a layer (again with multiple features) into the same database with the same name as the layer previously deleted. This layer should have different features to the first.*
> *5. Adding this layer to Qgis will cause errors in display.*
> *6. For comparison, import the same layer, but this time call it something new. Adding the layer to Qgis works perfectly.*

I can't replicate this, I'm using a recent (not latest) master release. Can you give a try to master?

#6 - 2019-03-02 01:08 PM - Giovanni Manghi

Dan Isaacs wrote:

> *Actually, I've just noticed that this is already a known issue from 3 years ago [[https://issues.qgis.org/issues/14239]]. You seem to suggest, in the issue I'm referring to, that we need to wait for spatialite to fix the problem, is that mainly the problem, that they are not very well supported anymore? It'd be good to know (if you have any experience on the matter) because I invest quite a bit of work in spatial data and if spatialite is not a good format, I might be better off choosing another.*

this seems slightly different (and is still valid on master)

Spatialite is a great format, but its releases, documentation, etc. are a mess (of course this is my opinion). For what I have understand in QGIS we use a version of SL declared as stable. It is old, in the meantime a lot of other versions have been releases, but none is declared as stable, so QGIS has not yet adopted them. I could be wrong anyway (as said before, to me understanding anything about how SL is managed is hard).

#7 - 2019-03-02 07:04 PM - Dan Isaacs

Giovanni Manghi wrote:

> *I can't replicate this, I'm using a recent (not latest) master release. Can you give a try to master?*

I'm on 3.6. I've just carried it out with and got the same problematic results. I'll try to be more specific about my process.

1. Created a new blank project.
2. created a new spatialite layer using 'create - spalialite layer' in an otherwise blank database. Called 'test'
3. Added four polygons with some random data in four columns.
4. Saved changes.
5. Exported the new layer to a Geopackage.
6. Removed the layer from Qgis and deleted it from the database using DB manager.
7. Closed and the re-opened Qgis.
8. Edited the geopackage layer (added two more polygons and a couple more random columns), added this new layer to Qgis.
9. Imported this geopackage to the spatialite database and called it 'test' (the same as the old layer). Added this layer to Qgis.

Only the previous four polygons show up even though the layer with only four polygons doesn't even exist anymore. All six polygons show up in the attribute table.

#8 - 2019-03-02 07:11 PM - Dan Isaacs

Giovanni Manghi wrote:

> *this seems slightly different (and is still valid on master)*

Actually, I think it's the same. The reason is if you look at the index and geometry tables within a spatialite database created by Qgis where a table has been deleted, the geometry entries for the deleted table are still there. No matter what you do (delete, rename, replace) the geometry tables don't update and still refer to the very first layer to have that name.

I'm going try and test with the spatialie gui, to see if it's spatialite itself or the way DB manager handles saptialite.

> *Spatialite is a great format, but its releases, documentation, etc. are a mess (of course this is my opinion). For what I have understand in QGIS we use a version of SL declared as stable. It is old, in the meantime a lot of other versions have been releases, but none is declared as stable, so QGIS has not yet adopted them. I could be wrong anyway (as said before, to me understanding anything about how SL is managed is hard).*

Absolutely, It's very difficult, I've tried to raise a couple of issues with the developers, and they've basically said that the version I'm using is really old, but then it's the newest **stable** version... what can we do?

#9 - 2019-03-02 11:34 PM - Giovanni Manghi

Dan Isaacs wrote:

> *Giovanni Manghi wrote:*
>
> > *I can't replicate this, I'm using a recent (not latest) master release. Can you give a try to master?*
>
> *I'm on 3.6. I've just carried it out with and got the same problematic results. I'll try to be more specific about my process.*
>
> *1. Created a new blank project.*
> *2. created a new spatialite layer using 'create - spalialite layer' in an otherwise blank database. Called 'test'*
> *3. Added four polygons with some random data in four columns.*
> *4. Saved changes.*
> *5. Exported the new layer to a Geopackage.*
> *6. Removed the layer from Qgis and deleted it from the database using DB manager.*
> *7. Closed and the re-opened Qgis.*
> *8. Edited the geopackage layer (added two more polygons and a couple more random columns), added this new layer to Qgis.*
> *9. Imported this geopackage to the spatialite database and called it 'test' (the same as the old layer). Added this layer to Qgis.*
>
> *Only the previous four polygons show up even though the layer with only four polygons doesn't even exist anymore. All six polygons show up in the attribute table.*

followed this exact steps on 3.6 on windows and linux, cannot replicate.

#10 - 2019-03-02 11:36 PM - Giovanni Manghi

> *Actually, I think it's the same. The reason is if you look at the index and geometry tables within a spatialite database created by Qgis where a table has been deleted, the geometry entries for the deleted table are still there. No matter what you do (delete, rename, replace) the geometry tables don't update and still refer to the very first layer to have that name.*

I can't replicate this, but I can the other, so to me are different ;)

> *Absolutely, It's very difficult, I've tried to raise a couple of issues with the developers, and they've basically said that the version I'm using is really old, but then it's the newest **stable** version... what can we do?*

raise this issue (again) in the qgis developer mailing list.

#11 - 2019-03-02 11:36 PM - Giovanni Manghi

- Priority changed from High to Normal

- Affected QGIS version changed from 3.4.1 to 3.6.0

#12 - 2019-03-03 08:24 AM - Dan Isaacs

- File spatialite test.zip added

Giovanni Manghi wrote:

> *followed this exact steps on 3.6 on windows and linux, cannot replicate.*

Weird. I've uploaded a test project with a Spatialite layer (above) and the Geopackage it was imported from below. I'm seeing only four polygons in the Spatialite layer despite there being six in the attributes table. What do you see on your system?

#13 - 2019-03-03 08:49 AM - Dan Isaacs

I've tested the layer with Spatialite gui 1.7.1. It loads fine with all six features showing, but it does report an invalid spatial index. I repaired the spatial index and then re-opened the project in Qgis and all six polygons now show as expected.

So something in my exact workflow (but evidently not yours) caused the spatial index to not be formed properly? When you imported the Geopackage layer to the Spatialite database, did you tick 'create spatial index'? I did, but I forgot to specify that in my instructions, so I'm thinking possibly it's the problem?

#14 - 2019-03-03 06:38 PM - Giovanni Manghi

Dan Isaacs wrote:

> *I did, but I forgot to specify that in my instructions, so I'm thinking possibly it's the problem?*

yes. Have you added a spatial index to the spatialite layer created from scratch at the beginning of your procedure? if yes how (db manager? sl gui?)?

#15 - 2019-03-03 06:42 PM - Giovanni Manghi

> *So something in my exact workflow (but evidently not yours)*

attach a screencast how your whole process

Files

spatialite test.zip	427 KB	2019-03-03	Dan Isaacs