

# QGIS Application - Bug report #15082

## SpatialLite index issue

2016-06-19 11:21 AM - R. R.

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Category:</b> DB Manager	
<b>Affected QGIS version:</b> master	<b>Regression?:</b> No
<b>Operating System:</b>	<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b> No	<b>Resolution:</b> end of life
<b>Crashes QGIS or corrupts data:</b> No	<b>Copied to github as #:</b> 23024
<b>Description</b>	
Spatial indexes are not listed in 'Indexes' tab and the 'DB Manager' crashes when adding a index (see screencast).	

### History

**#1 - 2016-06-19 11:21 AM - R. R.**

- File 15082.mp4 added

**#2 - 2016-06-20 02:49 AM - Jukka Rahkonen**

The issue is real but how to improve the current situation needs some thinking because spatial index in SpatialLite/GeoPackage is not an index but a table with some well known triggers.

The list of true indexes from SQLite can be obtained with a PRAGMA command

```
PRAGMA index_list('table_name')
```

I suppose that the Indexes tab in Layer properties makes exactly that request.

From SpatialLite databases created with recent versions the existence of the spatial index is supposed to be checked from the geometry columns table which has a column "spatial\_index\_enabled".

```
SELECT ROWID, "f_table_name", "f_geometry_column", "type", "coord_dimension", "srid", "spatial_index_enabled"  
FROM "geometry_columns"  
ORDER BY ROWID
```

If user is pressing "Create spatial index" button and spatial index already exists then the application should perhaps make a question:

"Table xxx has already a spatial index. Do you want to recover it?" If user says "Yes" then RecoverSpatialIndex function would be performed.

For the GeoPackage case I suppose that if spatial index exists or not should be checked with the test that is included in the standard, on page 75:

@A.3.1.3 Spatial Indexes

A.3.1.3.1 Data

A.3.1.3.1.1 Spatial Indexes Implementation

...snip...

3) For each row table\_name, column\_name from step 1

a. SELECT sql FROM sqlite\_master WHERE tbl\_name = 'rtree\_' || result\_set\_table\_name || '\_' || result\_set\_column\_name

b. Not testable if result set is empty

- c. Fail if returned sql != 'CREATE VIRTUAL TABLE rtree\_' || result\_set\_table\_name || '\_' || result\_set\_column\_name || USING rtree(id, minx, maxx, miny, maxy)
  - d. SELECT sql FROM sqlite\_master WHERE type = 'trigger' AND tname = 'rtree\_' || result\_set\_table\_name || '\_' || result\_set\_column\_name || '\_insert'
  - e. Fail if returned sql != result of populating insert triggers template in Annex L using result\_set\_table\_name for <t> and result\_set\_column\_name for <c>
  - f. SELECT sql FROM sqlite\_master WHERE type = 'trigger' AND name LIKE 'rtree\_' || result\_set\_table\_name || '\_' || result\_set\_column\_name || '\_update%'
  - g. Fail if returned sql != result of populating 4 update triggers templates in Annex L using result\_set\_table\_name for <t> and result\_set\_column\_name for <c>
  - h. SELECT sql FROM sqlite\_master WHERE type='trigger' AND name = 'rtree\_' || result\_set\_table\_name || '\_' || result\_set\_column\_name || '\_delete'
  - i. Fail if returned sql != result of populating delete trigger template in Annex L using result\_set\_table\_name for <t> and result\_set\_column\_name for <c>
  - j. Log pass otherwise
- 4) Pass if logged pass and no fails@

**#3 - 2017-05-01 01:05 AM - Giovanni Manghi**

- Easy fix? set to No
- Regression? set to No

**#4 - 2019-03-09 04:23 PM - Giovanni Manghi**

- Resolution set to end of life
- Status changed from Open 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/>

**Files**

---

15082.mp4	1.34 MB	2016-06-19	R. R.
-----------	---------	------------	-------