

## QGIS Application - Bug report #16705

### Handle missing PostGIS with useful warning/error messages

2017-06-12 04:14 PM - Arjan Mossel

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Category:</b> DB Manager	
<b>Affected QGIS version:</b> 2.18.9	<b>Regression?:</b> No
<b>Operating System:</b>	<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b>	<b>Resolution:</b> end of life
<b>Crashes QGIS or corrupts data:</b>	<b>Copied to github as #:</b> 24604

#### Description

If a PostgreSQL database doesn't have PostGIS installed or configured properly, the database manager errors out without specific indications what's wrong (at least to a novice postgres user as myself). In my case I had installed the postgis extension (in a separate schema), and configured the search\_path to include it. I could connect to the database in "Add PostGIS layers" and in the Browser Panel, and see the schemas I had access to, but in the database manager I only got some errors (see below). It turns out I had followed a mistaken snippet for configuring the search\_path (assigned a string instead of a list of schemas), so PostGIS was actually not available on the search\_path.

I think from a user perspective, two things would be helpful:

- 1) Add a check for PostGIS, and a warning if the extension is missing, and display this in the "Add PostGIS Table(s)" dialog, the Browser Panel, and the Database Manager.
- 2) Make the Database Manager behave like the "Add PostGIS Table" and Browser Panel, so show the schemas and tables you have access to even if PostGIS is missing (plus the warning).

The errors (stacktrace):

@An error has occurred while executing Python code:

```
TypeError: 'NoneType' object has no attribute '__getitem__'
```

Traceback (most recent call last):

```
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_model.py", line 438, in rowCount
    self._refreshIndex(parent, True)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_model.py", line 489, in refreshIndex
    if item.populate():
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_model.py", line 164, in populate
    if not connection.connect():
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/plugin.py", line 96, in connect
    return self.connectToUri(uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/plugin.py", line 106, in connectToUri
    self.db = self.databasesFactory(self, uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/plugin.py", line 66, in databasesFactory
    return PGDatabase(connection, uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/plugin.py", line 104, in init
    Database._init__(self, connection, uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/plugin.py", line 215, in init
    self.connector = self.connectorsFactory(uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/plugin.py", line 107, in connectorsFactory
    return PostGisDBConnector(uri)
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/connector.py", line 134, in init
    self._checkGeometryColumnsTable()
```

```
File "C:/OSGEO4~1/apps/qgis/.python/plugins/db_manager/db_plugins/postgis/connector.py", line 167, in
checkGeometryColumnsTable
    self.has_geometry_columns_access = priv
TypeError: 'NoneType' object has no attribute '_getitem__'@
```

## History

---

### #1 - 2017-10-17 02:18 AM - Jürgen Fischer

- Description updated

### #2 - 2019-03-09 03:10 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/>

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 3x 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.