QGIS Application - Bug report #18064
QGIS3 QtSqlDatabase 'QSPATIALITE' driver crash

2018-02-08 06:54 PM - Jakub Kosik

Status:	Closed		
Priority:	High		
Assignee:			
Category:	Python plugins		
Affected QGIS version:	master	**Regression?:**	Yes
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	Yes	**Copied to github as #:**	25960

Description

I'm trying to port my plugin to python3/Qt5/QGIS3. I'm using spatialite database with QtSqlDatabase to manage plugin requests:
db = QSqlDatabase.addDatabase('QSPATIALITE')
When I execute any QSqlQuery on that db - QGIS crashes. Tested with driver 'QSQLITE', no crash (and no spatial functions).

History

#1 - 2018-02-22 07:43 PM - Giovanni Manghi

- Priority changed from Normal to High

#2 - 2018-05-03 07:52 PM - Johannes Liem

Hello, I have the very same problem porting my plugin.

Maybe the following helps somehow (not sure if is it appropriate to post here):

```
from qgis.PyQt.QtSql import QSqlDatabase, QSqlQuery

db = QSqlDatabase.addDatabase('QSPATIALITE')
db.setDatabaseName("Path to Spatialite File e.g. manually created with QGIS")
print(db.isValid(), QSqlDatabase.isDriverAvailable('QSPATIALITE')) #returns both True, if query below is commented out

if not db.open():
  print("DB not open")
else:
  print("DB open")
  q = QSqlQuery(db)
  q.exec_("SELECT sqlite_version(), spatialite_version()")
  q.first()
  print(str(q.value(0)), str(q.value(1)))
  db.close()
```

Running this code in the QGIS python console (3.0.2) results in a crash (it works in 2.18.19):

```
QSqlCachedResult::detachFromResultSet :
QSqlQuery::~QSqlQuery :
PyInit_sip :
Py_HashPointer :
Py_CheckFunctionResult :
```

PyEval_EvalFrameDefault :
PyErr_Occurred :
PyEval_EvalCode
... much more here ...

QGIS Info
QGIS Version: 3.0.2-Girona
QGIS code revision: 307d082e3d
Compiled against Qt: 5.9.2
Running against Qt: 5.9.2
Compiled against GDAL: 2.2.4
Running against GDAL: 2.2.4

System Info
CPU Type: x86_64
Kernel Type: winnt
Kernel Version: 10.0.16299

I am thinking about a work around using qgis.utils.spatialite_connect for all spatial queries (https://gis.stackexchange.com/questions/260527/how-to-create-a-spatialite-layer-in-qgis-3-with-python) and the 'QSQLITE' driver for non spatial queries to populate PyQt Views/Models/DataMappers (which in my case do not require spatial queries)

#3 - 2018-05-04 08:46 AM - Jakub Kosik

I've done with same workaround as you thinking - works for now.

```
db = QSqlDatabase.addDatabase('QSQLITE')
query = QSqlQuery(db)
request = query.exec_(exp)
if request:
  [...]
else:
  connection = spatialite_connect(dbfile)
  cursor = connection.cursor()
  [...]
```

#4 - 2018-08-21 06:35 PM - Johannes Liem

This problem appears to be fixed #19419
Just successfully tried the code I posted above with QGIS version: 3.2.2-Bonn, QGIS code revision: 26842169e9

#5 - 2018-08-21 07:30 PM - Giovanni Manghi

- Status changed from Open to Feedback

Johannes Liem wrote:

> *This problem appears to be fixed*

can the ticket issuer confirm? Thanks!

#6 - 2018-08-22 12:29 AM - Jakub Kosik

Yes, I can confirm, works now! Tested on nightly 3.3.0 rev 85d740026a

#7 - 2018-08-22 09:45 AM - Giovanni Manghi

- Status changed from Feedback to Closed

- Resolution set to fixed/implemented