# QGIS Application - Feature request #9163 Extend DBManager to Oracle

2013-12-04 06:33 AM - Regis Haubourg

Status:	Closed				
Priority:	Low				
Assignee:					
Category:	DB Manager				
Pull Request or Patch supplied:		<b>Resolution:</b>	Resolution:		
Easy fix?:	No	Copied to github as	Copied to github as #: 17796		
Description					
Hi all,					
I miss the ability to cr	eate a datasource from a custom SQL quer	in Oracle.			
Extending DBmanage	er (or have a dedicated plugin like FastSQLI	ayers) working with Oracl	e would be har	ndy.	
I do not use MSSQL,	but I guess that is interesting someone				
No hurry, I'm filing the	e feature request for future uses.				
Régis					
Related issues:					
Related to QGIS Application - Feature request # 14527: DB Manager - Add suppo			Open	2016-03-18	
Duplicated by QGIS Application - Feature request # 11652: Add support for Ora			Closed	2014-11-15	
Associated revision	ns				

### Revision 644bdf04 - 2015-07-26 04:39 AM - Jürgen Fischer

Merge pull request #2222 from medspx/dbmanager-oracle

DBManager Oracle Spatial support (fixes #9163)

### History

### #1 - 2013-12-04 07:15 AM - Regis Haubourg

In fact,

I tested typing a sql query directly in the qgs file, and it fails to load project, when doing the same with postgis works. Does the provider supports this? Maybe we encounter table / schema double quote problemes (Oracle table names are upper case)

### #2 - 2014-08-08 04:33 AM - Médéric RIBREUX

It seems that the Oracle data provider from QG is 2.4 is able to handle custom queries. I've done lots of tests and the performances are quite good (same level than ArcGIS 10.2).

In order to load a custom SQL query from Oracle you have to modify the qgs file and change the table definition (table= variable). Just add parenthesis and use escaping to deal with " characters. You also have to add a key definition in the layer definition (add key=field\_name before table statement), otherwise QG is is not able to find a primary key for the layer and returns an error.

Here is an example of a layer definition in the qgs file:

```
<datasource>dbname='geobase' port=1521 user='USER' password='password' estimatedmetadata=true srid=-1 type=POLYGON key=GID
table="(SELECT a.GID, a.FIELD1, a.FIELD2, a.FIELD3, a.FIELD4, a.FIELD5, a.GEOM FROM \\"SCHEMA\\".\\"LAYER1\\" a, \\"SCHEMA\\".\\"LAYER2\\"
b WHERE SDO_ANYINTERACT(a.GEOM, b.GEOM) = 'TRUE' AND b.FIELD=1234561)" (GEOM) sql=</datasource>
```

Now that we have a quite reliable Oracle provider to handle dynamic queries, I agree that we need to have DBManager support for Oracle...

#### #3 - 2014-08-08 06:02 AM - Jürgen Fischer

Médéric RIBREUX wrote:

In order to load a custom SQL query from Oracle you have to modify the qgs file and change the table definition (table= variable). Just add parenthesis and use escaping to deal with " characters. You also have to add a key definition in the layer definition (add key=field\_name before table statement), otherwise QG is is not able to find a primary key for the layer and returns an error.

No need to modify the project file. You can also use QgsDataSourceURI:

uri = QgsDataSourceURI() uri.setConnection(",'1521',"geobase",'USER','password') uri.setSrid("-1") # Edit: has to be a string uri.setUseEstimatedMetadata(True) uri.setWkbType(QGis.WKBPolygon) uri.setDataSource("", '(SELECT a.GID, a.FIELD1, a.FIELD2, a.FIELD3, a.FIELD4, a.FIELD5, a.GEOM FROM "SCHEMA"."LAYER1" a,"SCHEMA"."LAYER2" b WHERE SDO\_ANYINTERACT(a.GEOM, b.GEOM) = \\'TRUE\\' AND b.FIELD=1234561)', 'GEOM', ", 'GID')

to produce the appropriate uri and use it with iface.addVectorLayer().

### #4 - 2014-08-20 03:15 AM - Regis Haubourg

Thanks for the hacking hints. My request is to have a GUI for that.. Anyway, I couldn't have python do the trick: 1- uri.setSrid(-1)returns an error:

Traceback (most recent call last): File "<input>", line 1, in <module> TypeError: QgsDataSourceURI.setSrid(QString): argument 1 has unexpected type 'int'

2-When I can load attribute table from provider, I can't load the same with python, vector layer is invalid.

here is what I tried:

uri = QgsDataSourceURI() uri.setConnection('vslin189','1521',"EXP",'visu','visu') uri.setUseEstimatedMetadata(True) uri.setDataSource("", '(SELECT ROWNUM, comm\_comm\_insee, comm\_dept\_num, lac\_code from "GOC"."GEN\_COMMUNE\_LAC"', ", 'ROWNUM') vlayer = QgsVectorLayer(uri.uri(), "test\_oracle\_layer", "oracle") vlayer.isValid() >>False

Am I doing it wrong with not defining geometry type and letting geom column name empty? I could'nt find in the docs how to define a uri for an attribute table. Cheers Régis

#### #5 - 2014-08-20 04:51 AM - Jürgen Fischer

Regis Haubourg wrote:

Thanks for the hacking hints. My request is to have a GUI for that..

Might still be a starting point for volunteers.

Anyway, I couldn't have python do the trick: 1- uri.setSrid(-1)returns an error: [...]

Right, actually it's uri.setSrid("-1")

2-When I can load attribute table from provider, I can't load the same with python, vector layer is invalid.

uri = QgsDataSourceURI()
uri.setConnection('somehost', '1521', "someservicename", 'someuser', 'somepassword')
uri.setSrid("-1")
uri.setUseEstimatedMetadata(True)
uri.setWkbType(QGis.WKBNoGeometry)
uri.setDataSource("", '(SELECT somecolumn,someothercolumn FROM sometable WHERE somecolumn<somevalue)', None, ", 'somecolumn')
v = iface.addVectorLayer( uri.uri(), "sometable", "oracle" )
v.isValid()</pre>

works for me.

### #6 - 2014-08-20 05:28 AM - Regis Haubourg

Got it !

I missed WKBNoGeometry, None value for geom field and parenthesis for SQL definition!

Thanks a lot Jürgen, If I find some time, I could give life back to FastSQLLayer and adapt it for Oracle.. or fund dbmanager extension..

Cheers,

Régis

#### #7 - 2014-08-20 06:18 AM - Regis Haubourg

One last question, locator is not activated here (and will not), do you think it is possible to fake geometry by manually computing point WKT? I tried this, with no success:

```
uri.setDataSource("", '(select id , \\'SRID=2154;POINT(\\'||y||\\',\\'||x|| \\')\\' geomwkt from goc.gen_localis where rownum <10)', 'geomwkt', ", 'id')
```

That would be sufficient for some uncommon usecases here (we have ETL jobs to load data in postgis, but sometimes, having a direct read access for geometries is cool)

#### #8 - 2014-08-20 06:58 AM - Regis Haubourg

Regis Haubourg wrote:

One last question, locator is not activated here (and will not), do you think it is possible to fake geometry by manually computing point WKT? I tried this, with no success:

[...]

That would be sufficient for some uncommon usecases here (we have ETL jobs to load data in postgis, but sometimes, having a direct read access for geometries is cool)

I have my answer in error log of qgis, this is not possible since QGIS asks SDO functions to get srid of geom field..

### #9 - 2014-08-28 07:08 AM - Médéric RIBREUX

Hello,

I've written an implementation of DBManager for Oracle. It needs cx\_Oracle to work which is quite easy to install under MS-Windows and seems to provide more work to install it under GNU/Linux.

Hope it will help and it will be implemented in the official QGis source code (it is the only objective of this work)...

best regards

### #10 - 2014-08-28 11:26 AM - Jürgen Fischer

Médéric RIBREUX wrote:

I've written an implementation of DBManager for Oracle. It needs cx\_Oracle to work which is quite easy to install under MS-Windows and seems to provide more work to install it under GNU/Linux.

Could you also use QtSql and qocispatial that is shipped with QGIS?

#### #11 - 2014-11-16 01:42 AM - Paolo Cavallini

See also #11652 (extend also to SQL server, possibly other DBs).

#### #12 - 2015-06-06 02:43 PM - Jürgen Fischer

Jürgen Fischer wrote:

Médéric RIBREUX wrote:

I've written <u>an implementation of DBManager for Oracle</u>. It needs cx\_Oracle to work which is quite easy to install under MS-Windows and seems to provide more work to install it under GNU/Linux.

Could you also use QtSql and qocispatial that is shipped with QGIS?

See also https://github.com/medspx/dbmanager-oracle/pull/3

## #13 - 2015-07-25 07:39 PM - Jürgen Fischer

- Status changed from Open to Closed

Fixed in changeset commit:"644bdf046fc305ddd068b7d31b08080baaba859a".