

QGIS Application - Feature request #8689

Oracle provider - speed of listing

2013-09-26 08:09 AM - Jonathan Moules

Status:	Open	Resolution: Copied to github as #: 17410
Priority:	Normal	
Assignee:		
Category:	Data Provider/Oracle	
Pull Request or Patch supplied:	No	
Easy fix?:	No	
Description		
<p>Not sure if this qualifies as a bug or feature request.</p> <p>My Oracle is listing several hundred tables (500?), the problem is that the scanning for the metadata for these tables can take an unreasonably long time. Take the following, all done during the same QGIS-session:</p> <p>First time.</p> <ul style="list-style-type: none">- 117seconds to list everything on our live database <p>"Only look in meta data table"</p> <p>"use estimated table metadata"</p> <p>"only existing geometry types"</p> <p>were all selected.</p> <p>Second time (without having closed QGIS remember!)</p> <ul style="list-style-type: none">- 63 seconds. <p>The third time (instantly after the second)</p> <ul style="list-style-type: none">- 3 seconds. <p>Fourth time</p> <ul style="list-style-type: none">- 11 seconds <p>There are two things here:</p> <p>a) If a full scan of everything must be made, can the results be stored locally/in memory for the session? If QGIS feels paranoid, when a user has selected a layer to add it, QGIS can re-scan that table to make sure stuff hasn't changed.</p> <p>b) The better option IMHO - don't scan the metadata until the user is actually adding it to QGIS. Most users probably don't care what the SRID, Geometry column, or datatype is when they're adding their data so won't miss it on that screen. You could add a checkbox for it to show it (or not show it, depends on the default).</p> <p>I think both of these would provide considerable increases to the speed of listing tables in the QGIS Oracle provider where there are lots of tables.</p>		
Related issues:		
Related to QGIS Application - Bug report # 13838: Oracle Connection takes too...		Closed 2015-11-16

History

#1 - 2014-04-09 12:52 AM - Jürgen Fischer

- Assignee changed from Jürgen Fischer to Martin Dobias

#2 - 2014-04-09 01:03 AM - Martin Dobias

- Assignee deleted (Martin Dobias)

The option a) has been implemented in commit:5b72da

For the option b) there is still area for improvement - similarly to Postgres provider, detect the layer metadata in a second step (in background thread)

#3 - 2016-04-11 04:12 AM - Ken Karlsson

Hi!

We have the same problem, but we have to wait for hours before all tables are listed when trying to add an Oracle Spatial-layer.

We would like to be able to first choose the table we want to add, and then only scan that table's metadata when added if necessary.

This issue seems to be open, but is anyone working on this?

We are using QGIS 2.14.1.

Thanks in advance!

Best regards, Ken Karlsson

#4 - 2017-05-01 12:48 AM - Giovanni Manghi

- Easy fix? set to No

#5 - 2017-08-02 12:20 AM - Russell Foster

Hi All,

At work we have recently been trialling QGIS as an alternative to some proprietary GIS's. The problem outlined above is our main issue with the software. It takes so long to scan the database tables, gathering all the info it wants. We have connections to databases that have hundreds of spatial objects available, and directories with hundreds of files. It would make sense to use Jonathan's option 'b' to us (mentioned above). We know what we're looking for in the databases, we know it's going to be in the projection we made it, all we need are a list of names so we can click on one to add. A simple query on a spatial db could simply cross check with say ALL_SDO_GEOM_METADATA (for Oracle) just to make sure something is spatial, and that would be enough. The details only become important when trying to add the layer. I hope this is something that gets fixed in later versions, **because so much about QGIS is positive, and a great piece of software**. I'd hate to see it not implemented fully at work because of this one issue, but we don't have time to sit around waiting for it to finally list the one layer we're interested in adding.