

Using the GUI: Wrong results are returned for FOUND_ROWS().

FOUND_ROWS() is an 'information function' in MySQL. The use of it is that you can use a LIMIT clause to reduce the result set and also retrieve a rowcount for the table from the same query like "SELECT SQL_CALC_FOUND_ROWS * FROM tbl_name LIMIT"

It is described in MySQL manual here:

<http://dev.mysql.com/doc/refman/5.1/en/information-functions.html>

Notice in particular this: "In the absence of the SQL_CALC_FOUND_ROWS option in the most recent successful SELECT statement, FOUND_ROWS() returns the number of rows in the result set returned by that statement."

That can be a problem with GUI tools because they may execute SELECT statements (mostly SELECT .. FROM INFORMATION_SCHEMA ..) in order provide user with information in the GUI. SQLyog will mostly use SHOW statements, but there are places where we SELECT .. FROM INFORMATION_SCHEMA .. because it is most efficient or because it is the only option.

SELECT .. FROM INFORMATION_SCHEMA .. is used

- 1) Sometimes when populating Object Browser details for 'stored programs'
- 2) For getting profiling information for the Query Profiler feature. It is not feasible unfortunately to use a SHOW statement instead of SELECT .. FROM INFORMATION_SCHEMA .. The SHOW PROFILES statement may return a much too large dataset to be handled efficiently (as there is no option to use a LIKE-clause or a WHERE-clause with SHOW PROFILES).

The problem with FOUND_ROWS() is that it is conflicting with the Query Profiler as Profiler will SELECT .. FROM INFORMATION_SCHEMA ... To use FOUND_ROWS() you will have to turn Query Profiler OFF. If you don't FOUND_ROWS() will return a rowcount for the `profiling` table in Information_Schema and not a rowcount for the table where user executed his query. We are considering how we could at least warn about this, of course.

Unique solution ID: #1178

Author: Peter Laursen

Last update: 2009-08-12 06:31