openQA Tests - action #44582

[qam][functional][u] Test module "mariadb" seems to be not used anywhere

2018-11-30 13:05 - okurz

<table>
<thead>
<tr>
<th>Status</th>
<th>Rejected</th>
<th>Start date:</th>
<th>2018-11-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>High</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee</td>
<td>dheidler</td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Category</td>
<td>Bugs in existing tests</td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version</td>
<td>Milestone 30</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

**Observation**

```bash
$ git grep mariadb
lib/repo_tools.pm:    assert_screen 'smt-mariadb-password', 60;
lib/repo_tools.pm:    # install RMT and mariadb
lib/repo_tools.pm:    zypper_call 'in mariadb';
tests/console/mariadb.pm:# Summary: client part of mariadb ssl connection test
tests/console/mariadb.pm:    zypper_call('in mariadb-client');
tests/console/mariadb.pm:    mutex_lock('mariadb');
tests/console/mariadb.pm:    mutex_unlock('mariadb');
tests/console/mariadb.pm:    assert_screen 'mariadb-monitor-password';
tests/console/mariadb.pm:    assert_screen 'mariadb-monitor-opened', 60;
tests/console/mysql_odbc.pm:    assert_script_run "echo [mariadbodbc_mysql_dsn] > /etc/unixODBC/odbc.ini";
tests/console/mysql_odbc.pm:    assert_script_run "echo Driver=mariadbodbc_mysql >> /etc/unixODBC/odbc.ini";
tests/console/mysql_odbc.pm:    assert_script_run "echo [mariadbodbc_mysql] > /etc/unixODBC/odbc.ini";
tests/console/mysql_odbc.pm:    my $odbc = (!$is_sle('<15') && !$is_leap('<15.0')) ? 'mariadb-connector-odbc' : 'MyODBC-unixODBC';
tests/console/mysql_odbc.pm:    zypper_call 'in mysql mariadb-client sudo ' . $odbc;
tests/console/mysql_odbc.pm:    assert_script_run 'isql mariadbodbc_mysql_dsn root x -b < query.sql';
tests/support_server/setup.pm:sub setup_mariadb_server { tests/support_server/setup.pm:    zypper_call('in mariadb');
tests/support_server/setup.pm:    assert_screen 'mariadb-monitor-opened';
tests/support_server/setup.pm:    assert_screen 'mariadb-user-host';
tests/support_server/setup.pm:    assert_screen 'mariadb-grant-ok';
tests/support_server/setup.pm:    if (exists $server_roles{mariadb}) {
tests/support_server/setup.pm:        setup_mariadb_server;
tests/support_server/setup.pm:    }
tests/support_server/setup.pm:    push @mutexes, 'mariadb';
tests/x11/smt_disconnect_internal.pm:    assert_screen("smt-mariadb-password-required");
$ openqa-db_query_last_use_of_module --openqa-host openqa.suse.de --module mariadb (nothing)
```

**Acceptance criteria**

- AC1: "mariadb" is scheduled and working

**Related issues:**

 Related to openQA Tests - action #23348: [sle][security][fips] multi-machine tests for stunnel and mariadb added

**History**

#1 - 2018-11-30 13:09 - okurz

- Related to action #23348: [sle][security][fips] multi-machine tests for stunnel and mariadb added

#2 - 2018-11-30 13:09 - okurz

Seems my suspicion in [https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/3441#pullrequestreview-57732181](https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/3441#pullrequestreview-57732181) was correct …

2021-05-23
Then let's rethink the unused modules static test, because seems to be not working as expected...

sure, if you have a good idea. The problem about "mariadb" is that other test modules install mariadb so the term "mariadb" is mentioned by some test modules. What tools/detect_unused_modules does is looking if the according test module name is mentioned anywhere else. Enough to detect if "yast2_nfs_server" or "first_boot" is mentioned in main.pm, not enough to check if "mariadb" is actually scheduled as a test module ;)

@santi let's check how to proceed here together =)

to be rediscussed...

needs to be discussed offline

Related to action #63871: [functional][u] test fails in php7_mysql - no answer from test_mysql_connector.php added

See commit https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/9208

Assignee changed from dheidler to szarate

Assignee changed from dheidler to szarate
#17 - 2020-03-06 10:11 - szarate
- Related to deleted (action #63871: [functional][u] test fails in php7_mysql - no answer from test_mysql_connector.php)

#18 - 2020-03-06 10:14 - szarate
- Status changed from Feedback to Resolved

#19 - 2020-03-06 14:48 - okurz
- Status changed from Resolved to Feedback

this is certainly an interesting interpretation of "Resolved" :D

I consider it a bit dangerous to call the ticket resolved with a link to the PR. People that are actively looking for mariadb due to the recent test regression might assume that "test is there and works fine" but your resolution was actually to delete the complete test.

Just to make sure we have a common understanding. The acceptance criterion mentions "mariadb" is scheduled and working" which isn't fulfilled by having the test module removed.

#21 - 2020-03-11 10:42 - szarate
- Status changed from Feedback to Rejected

rejecting in favor of 64051

#22 - 2020-03-11 10:46 - szarate
rejecting in favor of 64051

okurz wrote:

this is certainly an interesting interpretation of "Resolved" :D

I can't remember why I didn't set it to rejected, in any case rejecting in favor of 64051

I consider it a bit dangerous to call the ticket resolved with a link to the PR. People that are actively looking for mariadb due to the recent test regression might assume that "test is there and works fine" but your resolution was actually to delete the complete test.

Well, since there's no plan, nobody actually looked for mariadb tests, the test module wasn't scheduled, and it was deleted... the ticket bureaucracy is too much, and tickets as old as this one, don't help

Just to make sure we have a common understanding. The acceptance criterion mentions "mariadb" is scheduled and working" which isn't fulfilled by having the test module removed.

See acceptance criteria of the aforementioned ticket

#23 - 2020-03-11 11:04 - okurz

szarate wrote:

[...]

I consider it a bit dangerous to call the ticket resolved with a link to the PR. People that are actively looking for mariadb due to the recent test regression might assume that "test is there and works fine" but your resolution was actually to delete the complete test.

Well, since there's no plan, nobody actually looked for mariadb tests, the test module wasn't scheduled, and it was deleted... the ticket bureaucracy is too much, and tickets as old as this one, don't help

I was looking for mariadb tests, the ticket was just not followed up with for a long time. And there are multiple incarnations of a test plan including main.pm from where mariadb probably vanished by mistake. Mistakes can happen but I doubt explicitly deleting the test module and accepting that in a code review is done "by mistake". I don't think mariadb mentioned anywhere else -- which I am sure is as part of "what should be tested" would have stopped you from deleting the test module. This is why I consider this a dangerous approach.