

## openQA Project - action #72082

### Reduce test runtime, e.g. less reliance on test fixtures or test database instances

2020-09-29 14:14 - okurz

<b>Status:</b> Resolved	<b>Start date:</b> 2020-09-18
<b>Priority:</b> Low	<b>Due date:</b>
<b>Assignee:</b> okurz	<b>% Done:</b> 0%
<b>Category:</b> Feature requests	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> Ready	
<b>Difficulty:</b>	
<b>Description</b>	
<b>Motivation</b>	
Our tests can run very long which affects feedback cycles. We have many integration tests always initializing a test database with schema. Maybe we can improve by relying less on test fixtures or even completely on test database instances	
<b>Acceptance criteria</b>	
<ul style="list-style-type: none"><li>• <b>AC1:</b> All tests have been reviewed for their use of fixtures and the test database</li><li>• <b>AC2:</b> Test databases are only instantiated where explicitly needed</li><li>• <b>AC3:</b> Test databases are only initialized with data that is actually needed for specific tests</li></ul>	
<b>Suggestions</b>	
<ul style="list-style-type: none"><li>• Find all tests that instantiate a test database with fixtures and try instead to use initialization of only necessary data</li><li>• Find all tests that instantiate a test database and try instead to use instantiation of data objects directly instead of using a database</li><li>• Optional: Rework tests to not need a database at all anymore</li></ul>	
<b>Related issues:</b>	
Related to openQA Project - action #90371: Warnings "Subroutine JSON::PP::Boo..."	<b>Resolved</b> <b>2021-03-19</b>
Copied from openQA Project - action #71500: Potential optimization by skippin...	<b>Resolved</b> <b>2020-09-18</b>

## History

### #1 - 2020-09-29 14:14 - okurz

- Copied from action #71500: Potential optimization by skipping deployment checks in our tests (all except explicit deployment check tests) added

### #2 - 2020-09-29 14:17 - okurz

- <https://github.com/os-autoinst/openQA/pull/3414> for an extracted method (merged)
- <https://github.com/os-autoinst/openQA/pull/3421> t: Remove obsolete test job database entry in all initializations
- <https://github.com/os-autoinst/openQA/pull/3416> t: Skip fixtures by default to prevent excessive runtime

### #3 - 2020-10-19 10:45 - cdywan

okurz wrote:

- <https://github.com/os-autoinst/openQA/pull/3414> for an extracted method (merged)
- <https://github.com/os-autoinst/openQA/pull/3421> t: Remove obsolete test job database entry in all initializations
- <https://github.com/os-autoinst/openQA/pull/3416> t: Skip fixtures by default to prevent excessive runtime

We are down on the use of fixtures in most tests now, and the redundant initialization of the db in Selenium tests is also gone.

That last PR is still an open draft. Are you planning to pursue this further?

Arguably it's a cosmetic fix since it's not removing fixtures but merely adjusting the defaults. So leaving it as-is would make sense to me as well.

### #4 - 2020-10-20 21:54 - okurz

cdywan wrote:

That last PR is still an open draft. Are you planning to pursue this further?

yeah, you know how it is with "Low" tickets :) Eventually I would still like to have that in.

#### #5 - 2020-11-19 10:47 - okurz

<https://github.com/os-autoinst/openQA/pull/3421> is merged meanwhile. I did work on <https://github.com/os-autoinst/openQA/pull/3416> but then was annoyed by circleCI failing often (handled in other ticket). I looked into this the past days and continued with tiny steps.

#### #6 - 2021-03-05 11:05 - cdywan

Discussing this in the weekly we've identified 3 types of issues:

- Interdependencies of fixtures
- Fixtures from perl hashes are painfully slow - consider straight SQL (suggested by [kraith](#))
- Subroutine JSON::PP::Boolean::(++ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48. unclear what's causing this

#### #7 - 2021-03-05 12:33 - okurz

- Status changed from Feedback to In Progress

#### #8 - 2021-03-05 13:09 - okurz

After we discussed the topic today I managed to fix one little issue so added commit <https://github.com/os-autoinst/openQA/pull/3416/commits/f2c0ba668d442bf91e2fef4e7eab4f65336e8bf8> onto the PR. Based on test results and the fresh knowledge within the team now I am feeling motivated to bring other issues up to the team more quickly :)

#### #9 - 2021-03-05 18:28 - okurz

- Status changed from In Progress to Feedback

<https://github.com/os-autoinst/openQA/pull/3416> ready for review

#### #10 - 2021-03-08 10:02 - okurz

- Status changed from Feedback to Resolved

PR merged.

What I did is find all tests that instantiate a test database with fixtures and try instead to use initialization of only necessary data. Partially also instead of a test database we instead use instantiation of data objects directly instead of using a database. Well, what we did not proceed much with though is to rework more tests to not need a database at all anymore but that's ok.

#### #11 - 2021-03-18 20:58 - tinita

cdywan wrote:

- Subroutine JSON::PP::Boolean::(++ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48. unclear what's causing this

I debugged a little bit.

Mentioned already in <https://progress.opensuse.org/issues/38264>, also related <https://progress.opensuse.org/issues/29048>

<https://github.com/rurban/Cpanel-JSON-XS/issues/65> (closed but unfixed)

```
perl -wE 'use Cpanel::JSON::XS (); use JSON::PP ()'  
Subroutine JSON::PP::Boolean::(++ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(+ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(-- redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.
```

Happens only with Devel::Cover and JSON format.

Same with JSON::MaybeXS, which Devel::Cover uses

```
perl -wE 'use JSON::MaybeXS (); use JSON::PP ()'  
Subroutine JSON::PP::Boolean::(+ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(-- redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(++ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.
```

Can be avoided by loading JSON::PP before everything else:

```
export PERL5OPT="-MJSON::PP -MDevel::Cover...
```

Unclear why we only see it in the fullstack tests. Maybe JSON::PP is indirectly used in openQA somewhere before Cpanel::JSON::XS, while in os-autoinst code Cpanel::JSON::XS is used first. But that would also explain why -MJSON::PP "fixes" it, but a use JSON::PP at the top of the t/full-stack.t does not (because the problem lies in os-autoinst code).

Oh, and just for the record, if you're wondering where that warning comes from, a tiny reproducer:

```
perl -wE'  
package Foo;  
use overload "+" => sub { say 23 };  
use overload "+" => sub { say 23 };  
'  
Subroutine Foo::(+ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.
```

#### #12 - 2021-03-18 21:09 - okurz

ok, cool. Thanks for the context and the reminder about the other tickets where we have seen this. So in case you think there is anything related to JSON::PP we can do to improve or prevent further future confusing issues then please open a ticket or create a PR or something

#### #13 - 2021-03-19 16:18 - cdywan

tinita wrote:

Same with JSON::MaybeXS, which Devel::Cover uses

```
perl -wE 'use JSON::MaybeXS (); use JSON::PP ()'  
Subroutine JSON::PP::Boolean::(0+ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(-- redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.  
Subroutine JSON::PP::Boolean::(++ redefined at /usr/lib/perl5/5.26.1/overload.pm line 48.
```

Can be avoided by loading JSON::PP before everything else:

```
export PERL5OPT="-MJSON::PP -MDevel::Cover...
```

Unclear why we only see it in the fullstack tests. Maybe JSON::PP is indirectly used in openQA somewhere before Cpanel::JSON::XS, while in os-autoinst code Cpanel::JSON::XS is used first. But that would also explain why -MJSON::PP "fixes" it, but a use JSON::PP at the top of the t/full-stack.t does not (because the problem lies in os-autoinst code).

What I read there is *something* uses both JSON::MaybeXS and JSON::PP but if JSON::PP comes first, it's fine.

I would suggest a new ticket about this rather than this one which is about test runtime.

#### #14 - 2021-03-19 16:39 - tinita

- Related to action #90371: Warnings "Subroutine JSON::PP::Boolean::(0+ redefined" added

#### #15 - 2021-03-19 16:45 - tinita

I created <https://progress.opensuse.org/issues/90371>

I only mentioned it here because the warning was mentioned.