

openQA Project - coordination #55364

coordination # 80142 (Blocked): [saga][epic] Scale out: Redundant/load-balancing deployments of openQA, easy containers, containers on kubernetes

[epic] Let's make codecov reports reliable

2019-08-12 06:15 - okurz

Status:	Resolved	Start date:	2020-09-24
Priority:	Normal	Due date:	
Assignee:	okurz	% Done:	100%
Category:	Organisational	Estimated time:	0.00 hour
Target version:	Ready		
Difficulty:	hard		
Description			
Observation			
codecov reports often report about coverage changes which are obviously not related to the actual changes of a PR, e.g. when documentation is changed or as in https://github.com/os-autoinst/openQA/pull/2253#issuecomment-520042072 . It seems some of our tests – maybe the full stack test – introduced a flakyness in coverage.			
Suggestions			
We should try to make the reports more reliable and hence less annoying with big red color when it is not helpful. Let's try to collect our observations here:			
<ul style="list-style-type: none">• https://github.com/os-autoinst/openQA/pull/2253#issuecomment-520042072<ul style="list-style-type: none">◦ lib/OpenQA/Worker/Settings.pm 84.9% (-13.21%) arrow_down◦ lib/OpenQA/Worker/WebUIConnection.pm 84.42% (+2.51%) arrow_up -> https://github.com/os-autoinst/openQA/pull/3718• https://github.com/os-autoinst/openQA/pull/2135#issuecomment-506641541<ul style="list-style-type: none">◦ lib/OpenQA/WebSockets.pm 88.4% (-5.8%) arrow_down◦ lib/OpenQA/Scheduler/Model/Jobs.pm 86.11% (-2.78%) arrow_down◦ lib/OpenQA/Worker/Commands.pm 81.17% (-2.36%) arrow_down◦ lib/OpenQA/Utils.pm 95.81% (-0.2%) arrow_down• https://github.com/os-autoinst/openQA/pull/2161#issuecomment-513477665<ul style="list-style-type: none">◦ lib/OpenQA/Setup.pm 90.83% (∅) arrow_up◦ lib/OpenQA/Worker/Settings.pm 84.31% (-13.73%) arrow_down◦ lib/OpenQA/Worker/Engines/isotovideo.pm 94.66% (+0.48%) arrow_up• https://github.com/os-autoinst/openQA/pull/2161#issuecomment-513477665<ul style="list-style-type: none">◦ lib/OpenQA/Setup.pm 90.83% (∅) arrow_up◦ lib/OpenQA/Worker/Settings.pm 84.31% (-13.73%) arrow_down◦ lib/OpenQA/Worker/Engines/isotovideo.pm 94.66% (+0.48%) arrow_up• https://codecov.io/gh/os-autoinst/openQA/pull/2258/changes• https://github.com/os-autoinst/openQA/pull/2920#issuecomment-611509627<ul style="list-style-type: none">◦ OpenQA::Schema::Result::Jobs::done is only invoked in some testruns with \$args{newbuild}• https://codecov.io/gh/os-autoinst/openQA/pull/3276/changes from https://github.com/os-autoinst/openQA/pull/3276#issuecomment-694707108<ul style="list-style-type: none">◦ t/lib/OpenQA/SeleniumTest.pm:371◦ t/34-developer_mode-unit.t:100-102			
Other ideas for improving			
<ul style="list-style-type: none">• Mark individual sections as required to have 100% code coverage (how to do that?)• Aim for 100% statement coverage at least• Include coverage for javascript• Check coverage for perl template code			
Subtasks:			
action # 71857: flaky/unstable/sporadic test coverage from t/34-developer_mode-unit.t			Resolved
action # 78043: unstable/flaky/inconsistent statement coverage in t/lib/OpenQA/Selenium...			Resolved
action # 80268: Fix flaky coverage - t/05-scheduler-full.t			Resolved

action # 80274: Fix flaky coverage - t/lib/OpenQA/Test/Utils.pm size:M	Resolved
action # 80298: Fix flaky coverage - lib/OpenQA/WebSockets.pm	Resolved
action # 88594: codecov reports for individual files yield "forbidden" and files can no...	Resolved
action # 89899: Fix flaky coverage - t/ui/27-plugin_obs_rsync_status_details.t	Resolved
Related issues:	
Related to openQA Project - action #57050: Turn off travis	Resolved 2019-09-30
Copied to openQA Project - action #55643: Prevent duplicate codecov reports	Resolved 2019-08-12

History

#1 - 2019-08-12 10:12 - okurz

- Description updated

#2 - 2019-08-13 16:05 - okurz

- Description updated

#3 - 2019-08-13 20:16 - okurz

It also seems like we are getting duplicate codecov reports which I understand can happen if both an old bot are activated along with the github app? As the bot runs under the account "coolo" we can look into this with him.

#4 - 2019-08-16 12:46 - okurz

- Copied to action #55643: Prevent duplicate codecov reports added

#5 - 2019-08-29 14:22 - andriinikitin

I have strong impression that this happens when PR is not rebased to latest master. I.e. codecov compares coverage stats in PR with current coverage on master, which will look incorrect if base of PR is behind several commits

#6 - 2019-08-30 08:14 - okurz

yes, I thought the same but it still seems as if the "usual suspects" crop up, e.g. lib/OpenQA/Worker/Settings.pm even though we hardly ever touch it.

#7 - 2019-11-05 10:57 - okurz

- Related to action #57050: Turn off travis added

#8 - 2019-11-05 10:58 - okurz

[andriinikitin](#) I assume this is closely related to your work on circle CI setup. Can you comment on what are your recent experiences regarding codecov reliability?

#9 - 2019-11-25 09:20 - andriinikitin

okurz wrote:

[andriinikitin](#) I assume this is closely related to your work on circle CI setup. Can you comment on what are your recent experiences regarding codecov reliability?

I believe few lines are unstable, but overall I don't see it as a big problem.

#10 - 2020-02-25 11:44 - okurz

- Status changed from New to Resolved

- Assignee set to andriinikitin

[andriinikitin](#) thanks for your assessment. I personally trust coverage reports for openQA as well as os-autoinst lately so I guess we can call this done, probably mainly thanks to andriinikitin, hence assigning to him.

#11 - 2020-02-27 17:29 - okurz

- Status changed from Resolved to Workable

- Assignee deleted (andriinikitin)

Actually I found out that recent pull requests seem to report often the same flaky, changing coverage diff, e.g. <https://github.com/os-autoinst/openQA/pull/2782#issuecomment-591963100> :

- lib/OpenQA/Worker/Settings.pm 84.9% (-13.21%)
- lib/OpenQA/Utils.pm 96.84% (-0.23%)

-> <https://codecov.io/gh/os-autoinst/openQA/pull/2782/changes>

<https://github.com/os-autoinst/openQA/pull/2783#issuecomment-592008196> :

- lib/OpenQA/Worker/Settings.pm 84.9% (-13.21%)
- lib/OpenQA/Scheduler/Model/Jobs.pm 91.49% (+2.83%)

-> <https://codecov.io/gh/os-autoinst/openQA/pull/2783/changes>

that is

<https://github.com/os-autoinst/openQA/blob/master/lib/OpenQA/Worker/Settings.pm#L100>

which should have been "uncoverable".

and <https://github.com/os-autoinst/openQA/blob/master/lib/OpenQA/Scheduler/Model/Jobs.pm#L305> and following.

and <https://github.com/os-autoinst/openQA/blob/master/lib/OpenQA/Utils.pm#L807>

I guess for the latest at least it depends on the full stack test if that part is actually executed based on timing behaviour. Probably a more level or unit test to cover the lines explicitly can help.

#12 - 2020-03-04 12:49 - okurz

- Status changed from Workable to In Progress

- Assignee set to okurz

<https://github.com/os-autoinst/openQA/pull/2813> to cover <https://codecov.io/gh/os-autoinst/openQA/pull/2765/changes> about lib/OpenQA/Worker/Job.pm:315 which wasn't even found by me earlier.

#13 - 2020-03-11 22:24 - okurz

<https://github.com/os-autoinst/openQA/pull/2826> to cover uncovered parts of `_pick_siblings_of_running` in lib/OpenQA/Scheduler/Model/Jobs.pm

#14 - 2020-03-12 21:54 - okurz

- Status changed from In Progress to Feedback

PR merged, next stop: <https://codecov.io/gh/os-autoinst/openQA/pull/2811/changes> with lib/OpenQA/Worker/WebUIConnection.pm and lib/OpenQA/WebAPI/Controller/API/V1/Job.pm

But I could also wait for more results from next pull requests.

#15 - 2020-03-13 02:07 - andriinikitin

Not sure if it is obvious, but will mention anyway: my understanding is that there is some race condition on end of some of test(s), when Worker shutdown doesn't always happen in time, so codecov stats don't always reported before test ends.

#16 - 2020-03-13 20:45 - okurz

yes, good observation. What you mentioned is one option but I think the full stack tests that currently shows the coverage problems is doing the right thing on top level and actually pretty stable. On lower level we have different code paths that are there to provide robustness for different timing and network behaviour so I think we are better off covering these code branches with explicit low-level tests. As an alternative we could also run the full stack tests "often enough", e.g. with make variables `STABILITY_TEST=1 RETRY=3` in the hope we at least once cover all lines but that still seems risky.

<https://github.com/os-autoinst/openQA/pull/2815> is another PR, merged.

<https://github.com/os-autoinst/openQA/pull/2835> was another PR for more coverage that is now merged. Let's see what the next PRs will tell.

#17 - 2020-04-09 14:30 - mkittler

- Description updated

#18 - 2020-05-12 07:38 - cdywan

- <https://github.com/os-autoinst/openQA/pull/2920#issuecomment-611509627>
 - OpenQA::Schema::Result::Jobs::done is only invoked in some testruns with `$args{newbuild}`

I gather I'm currently improving coverage in that area.

#19 - 2020-07-20 17:01 - okurz

- Status changed from Feedback to Workable
- Assignee deleted (okurz)
- Target version set to Ready

The unreliability has decreased but there are still some areas that seem to report different results from time to time, e.g. see <https://github.com/os-autoinst/openQA/pull/3142#issuecomment-656651494> mentioning lib/OpenQA/WebAPI/Plugin/Helpers.pm despite the PR having clearly no related changes. <https://github.com/os-autoinst/openQA/pull/2897> is still the pending change planned to increase coverage.

#20 - 2020-09-15 08:45 - okurz

- Description updated

#21 - 2020-09-18 07:38 - okurz

- Description updated

#22 - 2020-09-25 12:06 - mkittler

Seems like <https://github.com/os-autoinst/openQA/pull/3399> didn't make this easier. Now we also see changes in code coverage within the test code itself, e.g. <https://codecov.io/gh/os-autoinst/openQA/pull/3423/changes>.

Loops like

```
for (0 .. 5) {
    $ret = $driver->find_element($element)->get_text();
    return $ret if &$test_break($ret);
    sleep .1;
}
```

within our test code are hard to make stable. The sleep may or may not be executed depending on the timing.

#23 - 2020-09-28 07:23 - okurz

I do not think the inclusion of test code is a problem but actually helps. We should be able to specify individual lines with "uncoverable statements" and introduce reliable coverage for all other places. And unreliable test code can likely be linked to unreliable production code coverage.

#24 - 2020-10-09 09:57 - okurz

- Subject changed from Let's make codecov reports reliable to [epic] Let's make codecov reports reliable
- Status changed from Workable to Blocked
- Assignee set to okurz
- Difficulty set to hard

Forced cdywan to discuss this with me ;) We agreed that it's actually not "Workable" for him and I should rework this, e.g. as an epic. There is already a subtask. That is not really the only task but we are not progressing anyway so I will block this epic based on the subtask.

#25 - 2020-10-12 13:33 - szarate

- Tracker changed from action to coordination
- Status changed from Blocked to New
- Difficulty deleted (hard)

#26 - 2020-10-12 13:44 - szarate

See for the reason of tracker change: <http://mailman.suse.de/mailman/private/qa-sle/2020-October/002722.html>

#27 - 2020-10-13 08:49 - okurz

- Status changed from New to Blocked
- Difficulty set to hard

#28 - 2020-11-24 10:37 - okurz

I have looked into recent codecov reports on <https://app.codecov.io/gh/os-autoinst/openQA/pulls?page=1&state=open&order=-pullid> and often I found two files with unreliable coverage results: t/05-scheduler-full.t and t/lib/OpenQA/Test/Utils.pm, created two new subtasks.

There is also <https://app.codecov.io/gh/os-autoinst/openQA/compare/3555/changes> showing uncovered lines in t/lib/OpenQA/SeleniumTest.pm still which I addressed in [#78043](#)

<https://codecov.io/gh/os-autoinst/openQA/pull/3467/changes> shows in lib/OpenQA/WebSockets.pm

```
97     = "Unable to assign job to worker $worker_id: the worker has not established a ws connection";  
98     return $result;
```

-> [#80274](#)

#29 - 2020-11-25 12:04 - okurz

- *Estimated time set to 39719.00 h*

#30 - 2020-11-25 12:10 - okurz

- *Estimated time deleted (39719.00 h)*

#31 - 2021-02-15 09:15 - okurz

- *Description updated*

#32 - 2021-05-05 18:08 - okurz

- *Parent task set to #80142*

#33 - 2021-09-02 21:09 - okurz

- *Status changed from Blocked to Resolved*

I think we finally managed – at least for now – to have consistent, trustworthy codecov reports \o/