openQA Project - action #56591

Improve feedback for jobs in "Uploading"

2019-09-08 20:46 - okurz

Status:	Resolved		**Start date:**	2019-09-08
Priority:	Normal		**Due date:**	
Assignee:	mkittler		**% Done:**	60%
Category:	Feature requests		**Estimated time:**	0.00 hour
Target version:	Ready			
Difficulty:				

Description

Motivation

As ideas based on our investigation work in [#54902#note-10](#) . Jobs can spend quite some time in "Uploading" and it's not really clear to users what is going on or an open browser tab is handling it inefficiently

Acceptance criteria

- **AC1:** The job details page preserves a user selection, e.g. the currently selected tab, when a job changes from "running" over "uploading" to the final state **DONE**
- **AC2:** During longer uploads the current status can be read from live logs in the webUI
- **AC3:** Jobs are only shown as Done when all log files are completely uploaded **DONE**

Suggestions

- Prevent the repeating auto-reloads of the webpage
- Provide the current status from logs over the webUI
- Ensure the job to be considered complete and upload worker-log.txt only after all other files are uploaded

Related issues:

Related to openQA Project - action #55904: /status updates are too heavy with...	**New**	**2019-08-23**

History

#1 - 2019-10-02 13:41 - okurz

- Related to action #55904: /status updates are too heavy with external results added

#2 - 2019-12-10 11:22 - okurz

- Description updated

- Priority changed from Low to Normal

- Target version deleted (future)

https://openqa.suse.de/tests/3684739 shows another interesting problem: According to the worker log as well as autoinst log tests ran from 10:01 to 10:46 but the openQA webUI summary says that the test ran for 1:26h. Only checking the systemd journal on the worker instance malbec:4 one can see the problem:

```
Dec 10 10:46:21 malbec worker[105342]: [debug] [pid:22963] Uploading artefact t16_teaming_lacp-173.txt
Dec 10 10:46:22 malbec worker[105342]: [debug] [pid:105342] REST-API call: POST http://openqa.suse.de/api/v1/j
obs/3684739/status
Dec 10 10:46:22 malbec worker[105342]: [info] [pid:105342] +++ worker notes +++
Dec 10 10:46:22 malbec worker[105342]: [info] [pid:105342] End time: 2019-12-10 09:46:22
Dec 10 10:46:22 malbec worker[105342]: [info] [pid:105342] Result: done
Dec 10 10:46:22 malbec worker[105342]: [info] [pid:25032] Uploading before_test_post.tar.gz
…
Dec 10 10:46:31 malbec worker[105342]: [info] [pid:25032] Uploading t13_teaming_activebackup_pre.tar.gz
Dec 10 10:46:32 malbec worker[105342]: [info] [pid:105342] Test schedule has changed, reloading test_order.jso
n
Dec 10 10:46:32 malbec worker[105342]: [debug] [pid:105342] REST-API call: POST http://openqa.suse.de/api/v1/j
obs/3684739/status
Dec 10 10:46:33 malbec worker[105342]: [info] [pid:25032] Uploading t14_teaming_broadcast_post.tar.gz
Dec 10 10:46:33 malbec worker[105342]: [info] [pid:25032] Uploading t14_teaming_broadcast_pre.tar.gz
```

…
```
Dec 10 10:46:54 malbec worker[105342]: [info] [pid:25032] Uploading worker-log.txt
Dec 10 10:46:54 malbec worker[105342]: [debug] [pid:25061] Uploading artefact before_test-48.txt
Dec 10 10:46:54 malbec worker[105342]: [debug] [pid:105342] Setting job 3684739 to done
Dec 10 10:46:54 malbec worker[105342]: [info] [pid:105342] Test schedule has changed, reloading test_order.json
Dec 10 10:46:55 malbec worker[105342]: [debug] [pid:105342] REST-API call: POST http://openqa.suse.de/api/v1/jobs/3684739/status
Dec 10 10:46:55 malbec worker[105342]: [debug] [pid:25061] Uploading artefact before_test-49.txt
Dec 10 10:46:55 malbec worker[105342]: [debug] [pid:25061] Uploading artefact before_test-50.txt
```
…

so the job is set to done and the worker log is uploaded before all other log files are finished uploading.

#3 - 2019-12-10 11:31 - okurz

- Description updated

#4 - 2019-12-11 12:56 - mkittler

+1 to that ticket

Note that we also reload the page in the states scheduled and assigned (and possibly setup, not sure about it right now). It would make most sense to come up with an approach which takes all of these non-final states into account. Likely we can extend the AJAX-polling code for the running state to cover the other non-final job states as well.

Prevent the repeating auto-reloads of the webpage

This is a really important point but it is likely acceptable to reload the page when the job state changes. We just shouldn't reload it all the time when the job state doesn't even change.

#5 - 2019-12-11 13:17 - coolo

That nonsense reloading sounds like a bug to me - and should be tracked independent of the rest.

#6 - 2019-12-13 15:10 - okurz

- Status changed from New to Workable

yes, could be tracked separately but I would not know what to provide in more details there. Anyone picking up on or the other "part" of this ticket can also split it out if desired. However I consider the ticket in the current state "Workable".

#7 - 2020-05-13 14:40 - mkittler

The test details page has been generally enhanced as part of [#65402](#) so AC1 has already been solved.

Not sure about AC3. Jobs have never been shown as done while the upload was still ongoing.

This leaves AC2. Considering how the page is structured now I suppose it is not hard to implement anymore. We just need to adjust the JavaScript code so the "Live" tab remains present during the upload. The screen and developer mode controls should likely still be hidden.

Currently [MDoucha](#) is working on the test details page (see https://github.com/os-autoinst/openQA/pull/3080) so I'd rather wait until these changes have been merged before starting to work on this ticket.

#8 - 2020-05-19 13:38 - mkittler

- Status changed from Workable to Blocked

- Assignee set to mkittler

The mentioned PR looks quite good now so I'd like to work on this feature when it has been merged.

#9 - 2020-05-25 12:44 - mkittler

- Status changed from Blocked to In Progress

#10 - 2020-05-25 15:38 - mkittler

PR: https://github.com/os-autoinst/openQA/pull/3116

#11 - 2020-05-26 11:20 - mkittler

This is trickier than expected so the PR is not working yet. It is easy to keep the 'Live' tab open while uploading in the frontend. The problem is that new live log contents are transferred from the worker to the web UI at the beginning of each "result upload sequence" within the worker. This happens all the time while the job is running and one last time while stopping the job. Uploading assets such as HDD images is done during the "stopping the job" phase so the "result upload sequence" isn't happening anymore alongside. So it will take some (possibly dangerous) restructuring within the worker to implement this.

#12 - 2020-05-27 09:35 - mkittler

I've fixed the PR which is now ready for review. The refactoring is likely not as dangerous as expected. It seems that continuing the "result upload sequence" while stopping the job (at least until logs and assets are uploaded) doesn't cause much trouble. I could even get rid of passing $is_final_upload around all the time.

#13 - 2020-06-04 09:59 - mkittler

When checking whether it works in production I've noticed that it doesn't work when job is already uploading at the point the test details page is opened. I'll come up with another PR to fix that.

#14 - 2020-06-04 11:12 - mkittler

PR which should at least partially fix the problem: https://github.com/os-autoinst/openQA/pull/3160

Not sure why it didn't work at all on the o3 jobs I've checked out because it works locally. Maybe the test really just didn't produce any further output at this point. Currently there are not tests uploading in production so I can not test it further.

#15 - 2020-06-05 11:45 - mkittler

> Maybe the test really just didn't produce any further output at this point.

That can not be the problem because not even the replay works. However, it seems the JavaScript-side is behaving as expected but the request for the live log keeps pending forever. So something must be wrong on the server-side. I couldn't spot any problem within the code and I can not reproduce it locally. Judging by the code it shouldn't make any difference whether the worker is on a remote host or local. That makes it hard to continue at this point. Maybe I'll have an idea later.

#16 - 2020-07-03 12:56 - cdywan

- Description updated

- Status changed from In Progress to Feedback

- % Done changed from 0 to 60

#17 - 2020-07-28 11:27 - okurz

- Target version set to Ready

#18 - 2020-08-04 16:07 - mkittler

While watching https://openqa.opensuse.org/tests/1352817 during the upload I could actually see messages like Uploading worker-log.txt within the live log. However, the upload looked nevertheless stuck when it uploaded images. The worker's journal also didn't contain status messages. I only observed the optipng calls in htop.

#19 - 2020-08-19 11:02 - mkittler

I've also tested a job with an asset upload on an o3 worker and it actually seems to work as well. The [2020-08-19T10:44:39.0800 UTC] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5502.qcow2: Processing chunk 1/5454, avg. speed ~976.562 KiB/s log messages appear within the live log. Not sure why it has not been working before.

Not sure whether it makes sense to add further log calls for the image/result upload. These uploads are usually already done while the job is being executed and should not take that long to finish during the uploading phase. When adding additional log calls we should make sure they're only produced during the uploading phase and not while isotovideo is still running because that would make the log just more cluttered.

#20 - 2020-08-20 14:04 - okurz

So you would regard the ticket as resolved?

#21 - 2020-08-27 10:06 - mkittler

- Status changed from Feedback to Resolved

I've tested it again and yes, I consider it resolved. It works when the live log is still open from watching the job while it was still running and it also works when opening the live log while the upload is already ongoing.

```
[2020-08-27T11:59:13.0362 CEST] [info] +++ worker notes +++
```

```
[2020-08-27T11:59:13.0362 CEST] [info] End time: 2020-08-27 09:59:13
[2020-08-27T11:59:13.0362 CEST] [info] Result: done
[2020-08-27T11:59:13.0373 CEST] [info] Uploading opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2
[2020-08-27T11:59:45.0453 CEST] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2: Processing chunk 10
/5422, avg. speed ~976.562 KiB/s
[2020-08-27T11:59:46.0462 CEST] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2: Processing chunk 20
/5422, avg. speed ~976.562 KiB/s
[2020-08-27T11:59:47.0466 CEST] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2: Processing chunk 30
/5422, avg. speed ~976.562 KiB/s
[2020-08-27T11:59:48.0489 CEST] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2: Processing chunk 40
/5422, avg. speed ~976.562 KiB/s
[2020-08-27T11:59:49.0484 CEST] [info] opensuse-Tumbleweed-x86_64@64bit-2G-:TW.5566.qcow2: Processing chunk 50
/5422, avg. speed ~976.562 KiB/s
…
```

I have no idea why it didn't work in my first test.

At this point I wouldn't care too much about the image upload because it is mostly done while the job is still running anyways and shouldn't take too long during the upload state.