

openQA Project - action #68956

Restart the parent and child jobs of a test in a START_DIRECTLY_AFTER_TEST test chain

2020-07-14 12:43 - Julie_CAO

Status:	Resolved	Start date:	2020-07-14
Priority:	Normal	Due date:	
Assignee:	mktittler	% Done:	0%
Category:	Feature requests	Estimated time:	0.00 hour
Target version:	Ready		
Difficulty:			

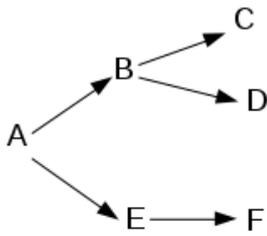
Description

Hi openQA experts,

I found START_DIRECTLY_AFTER_TEST has limitation in case of restarting tests, i.e. only parent jobs are run with openqa-client(including openqa-clone-job), and only child jobs are run by pressing restart button on openQA web UI. Can we have another button on the web UI to restart both the parent and child jobs for a test?

Our virtualization team has the plan to reconstruct our tests based on the test suite setting START_DIRECTLY_AFTER_TEST. We met problem based on current restarting logic.

fg. for this most common tree of chained tests(the tree is attached):



they will be triggered in daily build in this order, A->B->C->D->E->F. In the case B fails, C & D are canceled and A->E->F will continue to run. but how do I rerun the failing chain: A-> B -> C -> D? The restart button in web UI with existing logic is not capable to handle this.

What I can think of is to openqa-clone-job C & D, but this approach has two drawbacks: first, it is not so friendly as restart button in web UI; second, we need to know the test chain well, ie. to trigger the canceled C & D rather than the failing B directly, it becomes even impossible when the leaf nodes (like C & D) are plenty enough.

So could openQA provide a feature to restart jobs with its both parent and child jobs? We would not like to change the existing restarting logic as we need it as well. We'd like one more button as we need them both.

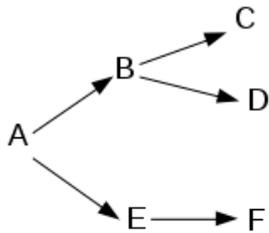
Related issues:

Related to openQA Project - action #69979: Advanced job restarting via the we...	New	2020-08-13
Related to openQA Project - action #70618: Automatically avoid restarting the...	New	2020-08-27

History

#1 - 2020-07-14 12:50 - Julie_CAO

- File chained_test.png added



attach the chained test tree.

#2 - 2020-07-14 13:32 - okurz

- Priority changed from Normal to Low
- Target version set to future

#3 - 2020-07-15 01:41 - xlai

[okurz](#) Why set the priority to low? **From customer needs, I would suggest "High"**, because now there are already 3 customer groups on this feature, kernel&network team, virtualization team and QAM. Besides, performance team is also a likely customer for it in nearby future. We all have the need to use `start_directly_after_test` on ipmi workers, and thus the "restart" behavior when a job in a sub-chain fails, we need this new feature for reasonable restart of "all its child jobs and parent jobs to the root".

[coolo](#) FYI.

#4 - 2020-07-15 05:32 - okurz

- Assignee set to mkittler
- Priority changed from Low to Normal
- Target version changed from future to Ready

xlai wrote:

[okurz](#) Why set the priority to low? **From customer needs, I would suggest "High"**, because now there are already 3 customer groups on this feature, kernel&network team, virtualization team and QAM. Besides, performance team is also a likely customer for it in nearby future. We all have the need to use `start_directly_after_test` on ipmi workers, and thus the "restart" behavior when a job in a sub-chain fails, we need this new feature for reasonable restart of "all its child jobs and parent jobs to the root".

[xlai](#) It is good to receive this feedback. And having multiple teams wanting this definitely is a good sign to handle this with higher priority. As I do not see a problem that needs to be handled within days I would avoid "High" due to the current team capacity and to ensure <https://progress.opensuse.org/projects/qa/wiki#SLOs-service-level-objectives>. There had been a post to the internal mailing list openqa@suse.de which I assume you have seen already? This explains to my understanding that the restart of the cluster should work in general. We should clarify what is the current expected behaviour, where is the diff in the request of this ticket and what is feasible to do.

[mkittler](#) I know you already did a very good job in explaining by email. May I please ask you to look into this ticket, maybe add your email response as comment as well and make sure the description is up-to-date with the information so that we also have a workable ticket following our usual template? Thanks in advance.

[coolo](#) FYI.

coolo isn't PO for QA tools anymore. Sorry, I wanted to communicate that more openly but wasn't allowed to do so.

#5 - 2020-07-15 07:03 - xlai

okurz wrote:

xlai wrote:

[okurz](#) Why set the priority to low? **From customer needs, I would suggest "High"**, because now there are already 3 customer groups on this feature, kernel&network team, virtualization team and QAM. Besides, performance team is also a likely customer for it in nearby future. We all have the need to use `start_directly_after_test` on ipmi workers, and thus the "restart" behavior when a job in a sub-chain fails, we need this new feature for reasonable restart of "all its child jobs and parent jobs to the root".

[xlai](#) It is good to receive this feedback. And having multiple teams wanting this definitely is a good sign to handle this with higher priority. As I do not see a problem that needs to be handled within days I would avoid "High" due to the current team capacity and to ensure <https://progress.opensuse.org/projects/qa/wiki#SLOs-service-level-objectives>. There had been a post to the internal mailing list openqa@suse.de which I assume you have seen already? This explains to my understanding that the restart of the cluster should work in general.

We should clarify what is the current expected behaviour, where is the diff in the request of this ticket and what is feasible to do.

Sure, if current description from julie is not enough, I am sure she is glad to give clearer one for this part. Good thing is that [mkittler](#) seems to have understood what is wanted in this feature.

[mkittler](#) I know you already did a very good job in explaining by email. May I please ask you to look into this ticket, maybe add your email response as comment as well and make sure the description is up-to-date with the information so that we also have a workable ticket following our usual template? Thanks in advance.

Thanks from me, too.

[coolo](#) FYI.

coolo isn't PO for QA tools anymore. Sorry, I wanted to communicate that more openly but wasn't allowed to do so.

Oh, really? I am surprised about this. If it is fine to talk offline, Who is the new PO now? Is it you, [okurz](#)? Can you let me know in some allowed way?

#6 - 2020-07-15 07:05 - okurz

coolo isn't PO for QA tools anymore. Sorry, I wanted to communicate that more openly but wasn't allowed to do so.

Oh, really? I am surprised about this. If it is fine to talk offline, Who is the new PO now? Is it you, [okurz](#)? Can you let me know in some allowed way?

I will :)

#7 - 2020-07-15 09:04 - mkittler

Can we have another button on the web UI to restart both the parent and child jobs for a test?

Not starting the parent isn't actually very useful. (See the two problems I've mentioned in the mail.) So I would even suggest to change the current restart button to restart both - parent and child jobs. Of course this would only affect *directly* chained dependencies and that's also how it already works for parallel dependencies. Note that would restart *all* parents up to the top-level parent within the "direct subtree". (So a regularly chained parent would not be restarted.)

We would not like to change the existing restarting logic as we need it as well.

Really? But *not* restarting A and only B -> C -> D makes not generally sense as the direct chaining between A and B is not guaranteed anymore. What is your use-case? Note that if B is not actually required to be started directly after A you can simply use the regular START_AFTER_TEST dependency. It is possible to mix dependency types within the same dependency tree.

i.e. only parent jobs are run with openqa-client

As I said in the mail: You should really specify what you've did with the client because it is a quite generic tool. Generally you should be able to trigger the restart functionality with the client in the same way as with the web UI. (See the mail for the command.)

So when I change the restarting behavior as explained that will also affect restarts triggered via the client.

#8 - 2020-07-15 09:19 - mkittler

Just for the record - the relevant part of the mail:

...it is actually tricky anyways if one restarts a job in the middle of the "direct chain" (like B or E in your example) because of two reasons:

1. The worker has already been used to run a different job.
2. A completely different worker is assigned on the 2nd run as we currently not even ensure the same worker is used again.

#9 - 2020-07-15 09:24 - Julie_CAO

Hi Marius,

I thought START_AFTER_TEST and START_DIRECTLY_AFTER_TEST share the same restarting logic, so I suggest add an extra button. Yes, restarting child jobs only may be useful for START_AFTER_TEST but useless for START_DIRECTLY_AFTER_TEST. Of course it is best to change the current restart button to restart both - parent and child jobs for START_DIRECTLY_AFTER_TEST only if possible.

#10 - 2020-07-15 14:07 - mkittler

- Status changed from New to In Progress

No, they don't necessarily share the same restarting logic. Different dependency types already have different restarting logic. This change basically means making directly chained dependencies behave like parallel dependencies in this regard.

#11 - 2020-07-15 16:29 - mkittler

PR: <https://github.com/os-autoinst/openQA/pull/3262>

#12 - 2020-07-16 00:58 - Julie_CAO

Thank you for the quick fix, Marius. :-) Let me know if you'd like test or verification from my side.

#13 - 2020-07-23 16:29 - mkittler

- Status changed from In Progress to Feedback

The PR has been merged. It would be nice if you could give me feedback from your side.

#14 - 2020-07-24 13:05 - Julie_CAO

I have upgraded my local openQA to the latest version and got the fix. I'll do some tests and update to you. Thanks, Marius.

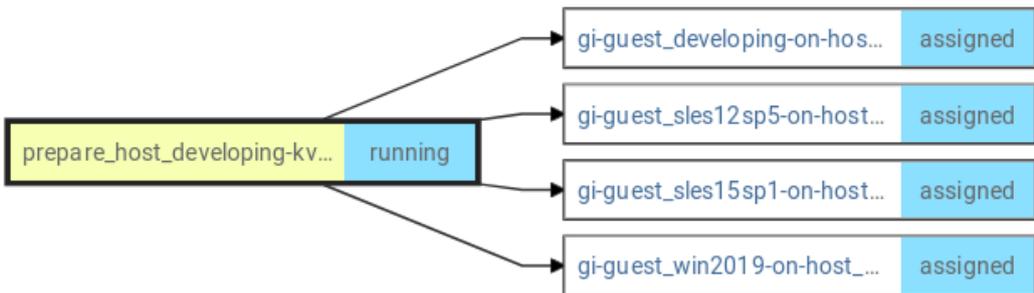
#15 - 2020-07-28 09:29 - Julie_CAO

- File Screenshot_20200728_171714.png added
- File Screenshot_20200728_172309.png added
- File Screenshot_20200728_172515.png added

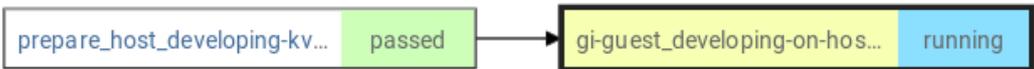
Hi Marius,

The test chain got restarted as expected(both parent jobs and child jobs are restarted), but there is a problem --- only one chain from the entire tree is allowed to be restarted. It can be reproduced by:

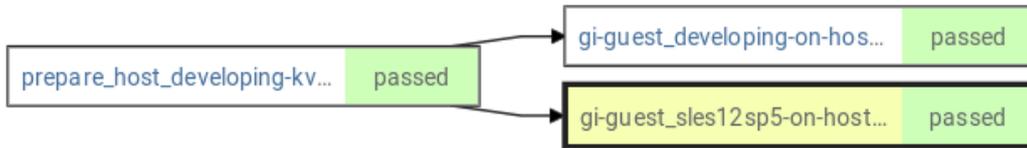
```
1.run 5 jobs by command line(/usr/share/openqa/script/client isos post --host http://10.67.133.97 DISTRI=sle ... _NOBSOLETEBUILD=1 TEST=prepare_host_developing-kvm-1,gi-guest_developing-on-host_developing-kvm,gi-guest_sles12sp5-on-host_developing-kvm,gi-guest_sles15sp1-on-host_developing-kvm,gi-guest_win2019-on-host_developing-kvm)
```



2.After all tests finish, restart the first child job, gi-guest_developing-on-host_developing-kvm, by clicking the 'restart' button on web UI. Its parent job, prepare_host_developing-kvm-1, is restarted as well.



3.but the dependence graph of another child job changes to



4.restart the child job on web UI, "Unable to restart job: URL for new job not available" shows up.

We need the capability to restart any child jobs in case two or more of them fails. Could you fix it?

#16 - 2020-07-28 10:07 - Julie_CAO

Another problem, for those 5 jobs above, if I openqa-clone-job one child job, its parent job is created together, but they are running at the same time, not on the same SUT.

#17 - 2020-07-29 14:08 - mkittler

That the dependency graph of other child jobs change as well is unfortunately expected. The dependency graph only shows the last job of a "cloning/restarting chain". But of course this isn't nice. It would be better if the graph would show the actual parents and the behavior to show only the latest job would only apply to children. This might be a little bit tricky to implement, though.

About 4.: What happens when restarting with the CLI client as explained in the mail? It should behave identical but would likely provide a better error message. But I guess I'll have to reproduce this locally anyways.

About openqa-clone-job: I'll have to reproduce this locally as well to be sure but I suppose openqa-clone-job has a race condition because it doesn't create jobs atomically. That makes is unusable for cloning jobs with parallel or directly chained dependencies.

#18 - 2020-07-31 10:09 - Julie_CAO

- File Screenshot_20200731_180724.png added

Hi Marius, find another problem, not sure if they are the same issue. one child job is 'skipped' when its sibling job is incomplete.



#19 - 2020-08-05 07:58 - Julie_CAO

Hi Marius, are you able to reproduce locally? Let me know if I can help. We hope these issues can be fixed before SP3 testing starts at the beginning of September.

#20 - 2020-08-05 16:39 - mkittler

I can reproduce the problematic graph rendering but I have not encountered the error message "Unable to restart job: URL for new job not available" so far when restarting a job.

We need the capability to restart any child jobs in case two or more of them fails. Could you fix it?

That sounds useful, indeed. Restarting only failed children is is not covered so far. (When restarting the parent *all* children are restarted.) We could add an option to only restart children which have failed (or children which have failed).

#21 - 2020-08-05 16:57 - okurz

mkittler wrote:

[...]

That sounds useful, indeed. Restarting only failed children is is not covered so far. (When restarting the parent *all* children are restarted.) We could add an option to only restart children which have failed (or children which have failed).

I guess it should be possible to add that as an optional filter when triggering over API. If you think it's trivial to do then go ahead but as this ticket is open for longer already I would prefer new ticket(s) for any further work which is not a direct "fix" to the original issue.

[mkittler](#) can you comment what is your plan here for the next steps if any as we already have <https://github.com/os-autoinst/openQA/pull/3262> done?

#22 - 2020-08-06 06:48 - Julie_CAO

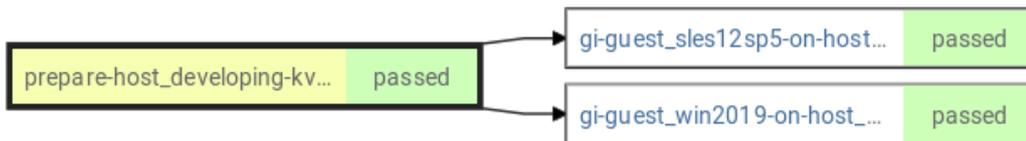
- File Screenshot_20200806_142047.png added
- File Screenshot_20200806_142736.png added
- File Screenshot_20200806_144712.png added

mkittler wrote:

I can reproduce the problematic graph rendering but I have not encountered the error message "Unable to restart job: URL for new job not available" so far when restarting a job.

I reproduce it locally:

step 1: run a test chain with 3 tests, 1 parent and two children. the test id is 869, 870, 871



step 2: After test finish, restart one child 871 on web UI, then its parent and itself got restarted as 872 & 873.

step 3: restart the other child from the original tree, 870. Got "Unable to restart job: URL for new job not available". refer to picture pasted as below.

Unable to restart job: URL for new job not available

Results for sle-15-SP2-Full-x86_64-Build211.1-gi-guest_win2019-on-host_developing-kvm@64bit-ipmi

Result: **passed** finished 1.4 minutes ago (01:13 minutes)

Scheduled product: sle-15-SP2-Full-x86_64-211.1

Assigned worker: linux-oqtb:1

step 4: if I run 870 via openqa-clone-job, no error message prompt up and 2 jobs got run. BUT, the restarted 2 jobs behave like two independent jobs completely --- They begin at same time, in no order, and run on different SUT.

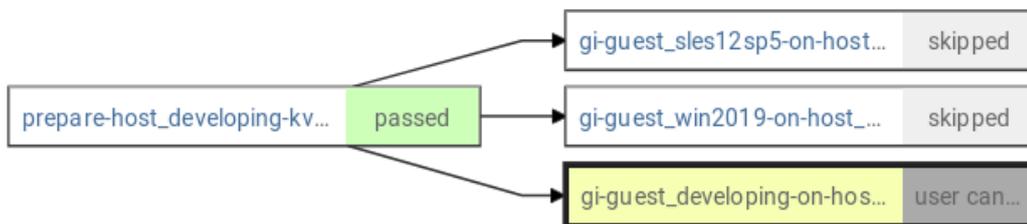
:-> openqa-clone-job --apikey xxx --apisecret xxx --within-instance <http://10.67.133.97/> 870

Cloning dependencies of sle-15-SP2-Full-x86_64-Build211.1-gi-guest_win2019-on-host_developing-kvm@64bit-ipmi

Created job #875: sle-15-SP2-Full-x86_64-Build211.1-prepare-host_developing-kvm-1@64bit-ipmi -> <http://10.67.133.97/t875>

Created job #876: sle-15-SP2-Full-x86_64-Build211.1-gi-guest_win2019-on-host_developing-kvm@64bit-ipmi -> <http://10.67.133.97/t876>

And regarding the problem I described in comment #18, the similar situation affect our tests. When we cancel one child job, the tree with its parent as root got canceled, including its siblings. But we expect only its children and itself got canceled, its siblings should continue. Could this be considered to be fixed as well?



#23 - 2020-08-06 06:59 - Julie_CAO

We need the capability to restart any child jobs in case two or more of them fails. Could you fix it?

That sounds useful, indeed. Restarting only failed children is is not covered so far. (When restarting the parent all children are restarted.) We could add an option to only restart children which have failed (or children which have failed).

We appreciate if this 'restarting multiple children' ability can be provided. It helps a lot as many tests will still under their parent, sharing worker resources.

#24 - 2020-08-06 10:09 - mkittler

Ah, you've restarted the other child. In this case I get the error message as well and there's actually no new job created. The problem is that one of the job it likes to duplicate (its parent) has already been cloned and it is not possible to clone a job twice. Unfortunately the error is not shown nicely within the web UI (which is yet another thing we could improve).

I suppose restarting multiple failed children one-by-one is not a good idea anyways (from the perspective of how dependency handling works internally when restarting a job but it also sounds like a tedious task for the user). So it would make more sense if you would restart *the parent* after all but there's an option to consider only children which have not passed or softfailed.

I can also think of a more advanced restart feature which we would incorporate into the dependency tree. It would allow you to select all the jobs you would like to restart from the dependency tree explicitly and to submit the restart at once as a single action. Note that implementing this idea would likely take a while so for now we better go with the simpler idea mentioned in the previous paragraph.

By the way, the restart API already allows to start multiple jobs at once. But it does not seem usable via the client and judging by the code it would run into the same problem as restarting the jobs on-by-one.

#25 - 2020-08-06 13:13 - mkittler

When we cancel one child job, the tree with its parent as root got canceled, including its siblings. But we expect only its children and itself got canceled, its siblings should continue. Could this be considered to be fixed as well?

I've also noticed that problem. I can try to fix it.

Together with [#69274](#) a lot of rough edges have piled up when it comes to directly chain dependencies so it will take a while to remove all problems we're currently seeing.

#26 - 2020-08-07 13:03 - mkittler

Here's a draft PR with an explaining comment: <https://github.com/os-autoinst/openQA/pull/3300#issuecomment-670504578>

@Julie_CAO You could give me some feedback whether I'm on the right track.

#27 - 2020-08-11 13:56 - mkittler

The PR <https://github.com/os-autoinst/openQA/pull/3300> has been merged. See <https://github.com/os-autoinst/openQA/pull/3300#issuecomment-670504578> for more details.

Further points which came up:

1. Add a UI for the advanced restarting features: I suppose this deserves its own ticket and this feature request should not only be thought of in the context of bare metal testing.
2. Better error handling for the restart API: I've already created another ticket ([#69871](#)) because this is independent from bare metal testing.
3. Selecting explicitly which jobs to restart: I've mentioned it in a [previous comment](#). I mention it here again because of the PR comment "In the case if one of the test is expected to be fail due to a bug, how can we skip it to restart other faied tests? especially it is a long-run test, other tests have to queue until it finishes.". That feature would certainly help with that use-case. However, it is still questionable whether implementing it is worth the effort.

#28 - 2020-08-12 06:12 - Julie_CAO

mkittler wrote:

The PR <https://github.com/os-autoinst/openQA/pull/3300> has been merged. See <https://github.com/os-autoinst/openQA/pull/3300#issuecomment-670504578> for more details.

Julie: Thanks Marius for the good idea and effort.

Further points which came up:

1. Add a UI for the advanced restarting features: I suppose this deserves its own ticket and this feature request should not only be thought of in the context of bare metal testing.

Julie: With the restart logic in the PR, for our directly chained tests, in the case of one child test fails, we will use the 'restart' button on web UI to rerun it, then its parent job and itself will get rerun as a result. In the case of more than one children fail, we will use REST API with `skip_ok_result_children=1` to restart all the failing ones, then a new sub-chain is created, we take the same approach to the sub-chain. then sub-tree creates again ... recursively.

1. Better error handling for the restart API: I've already created another ticket ([#69871](#)) because this is independent from bare metal testing.
2. Selecting explicitly which jobs to restart: I've mentioned it in a [previous comment](#). I mention it here again because of the PR comment "In the case if one of the test is expected to be fail due to a bug, how can we skip it to restart other faied tests? especially it is a long-run test,

other tests have to queue until it finishes.". That feature would certainly help with that use-case. However, it is still questionable whether implementing it is worth the effort.

Julie: It would be much helpful if we can specify which tests would be restarted, especially in web UI.

#29 - 2020-08-14 13:42 - mkittler

- Another discussion came up here: <https://github.com/os-autoinst/openQA/pull/3262#issuecomment-673382461>
- I've also created a PR to improve the error handling: <https://github.com/os-autoinst/openQA/pull/3312>

#30 - 2020-08-18 15:36 - mkittler

A summary of the mentioned discussion:

1. There is a use-case where restarting directly chained parents interferes with the workflow: Apparently users use directly chained dependencies in a very constraint setup. In this setup the user knows that the bare metal worker is still in the right state and that running the directly chained parent is not required to re-establish that state. The user also ensures that the same worker is used when restarting the job manually by having only one worker (with that particular worker class).
 1. Solutions to this use-case within the current state of openQA:
 1. Use the API to restart only the child job, e.g. `script/openqa-cli api -X POST jobs/$child_job_id/restart skip_parents=1`
 2. Use regularly chained dependencies which should work in the same way as directly chained dependencies within the constrained setup.
 2. Ideas to improve the usability of 1.1.1:
 1. Allow advanced job restarting via the web UI ([#69979](#))
 1. Possibly even allow to select the particular jobs to be restarted within the dependency tree.
 2. Make defaults for restarting configurable per openQA instance
 3. Idea to generally improve the use-case: Make openQA smart enough to discover the situation of this use-case automatically and avoid restarting the parent unnecessarily automatically.
2. It is generally not nice that the dependency graph of cloned jobs is not shown anymore. I've created a ticket for this: [#69976](#)

My thoughts so far:

- Idea 1.3 sounds great to have but would be very hard to implement. Of course it would be possible to detect the situation of 1. where restarting the parent is unnecessary. If the user can do that the system can do it as well. However, we must not only detect that situation but also ensure it is retained until the very moment the jobs would be assigned to the worker. We would also need to take care that the same worker is used as during the previous run. Maybe this could be done by assigning the jobs to the worker directly from the restart code and thus bypassing the scheduler. This would mean messing around with how openQA usually works a lot. At this point I do not think implementing 1.3 is worth the development effort and added complexity but it would be conceivable to implement.
 - By the way, the "situation" is basically: The job to be restarted has a directly chained parent and the worker which has been executing the directly chained parent has only been executing directly chained children of that parent so far.
 - There are actually more tweaks conceivable: For instance, when a directly chained child fails we could lock the worker so its state is preserved until it is manually unlocked or a timeout is exceeded. The locked worker would be exclusively considered when restarting the directly chained children.
- Since 1.2 is actually a viable solution in this case, why not just go for it?
- The ideas 1.2.1 and 2 sound useful also outside of the scope of restarting directly chained dependencies. So implementing them would likely be worth the effort.
- The idea 1.2.2 is not a nice solution. I would consider it a workaround for a special setup and I'm not sure whether we want that workaround considering that there are also 1.2 and 1.2.1.

#31 - 2020-08-19 10:17 - mkittler

- Status changed from *Feedback* to *Resolved*

I've created tickets for the additional ideas which make most sense to me as not only jobs with `START_DIRECTLY_AFTER_TEST` dependencies would benefit from them (see referenced tickets in the previous comment). Even these tickets have been set to the future target version by @okurz. Hence I suppose that going for the even more advanced ideas which would be even specific to `START_DIRECTLY_AFTER_TEST` dependencies is *not* something we want to focus on in the next time as well. (And if we wanted that I would create a separate ticket.)

So I would actually mark this ticket as resolved for now because otherwise I'd be blocked as the further ideas are only for the future milestone. Marking the ticket as resolved does of course not mean you can not continue the discussion here or in the other "future" tickets.

#32 - 2020-08-19 13:54 - okurz

Agreed. Changing the default was already questionable from the beginning. I think we provided a good base for further extensions for others to follow. For the tools team I see the priority rather on the side of the overall design work which we have in our backlog.

#33 - 2020-08-25 14:56 - mkittler

- Status changed from *Resolved* to *In Progress*

Unfortunately the scheduler has problems with jobs restarted via the feature implemented in <https://github.com/os-autoinst/openQA/pull/3300>. I'll have to fix the scheduling half-restarted directly chained dependencies.

#34 - 2020-08-25 16:09 - mkittler

Draft PR for fixing the problem mentioned in the previous comment: <https://github.com/os-autoinst/openQA/pull/3332>

#35 - 2020-08-27 08:53 - mkittler

- Status changed from In Progress to Resolved

The PR has been merged and works on OSD (hot-patched).

#36 - 2020-09-11 08:51 - okurz

- Related to action #69979: Advanced job restarting via the web UI added

#37 - 2020-09-11 09:02 - okurz

- Related to action #70618: Automatically avoid restarting the directly chained parent if possible to save time added

Files

chained_test.png	3.82 KB	2020-07-14	Julie_CAO
Screenshot_20200728_171714.png	20 KB	2020-07-28	Julie_CAO
Screenshot_20200728_172309.png	5.94 KB	2020-07-28	Julie_CAO
Screenshot_20200728_172515.png	10.2 KB	2020-07-28	Julie_CAO
Screenshot_20200731_180724.png	16.2 KB	2020-07-31	Julie_CAO
Screenshot_20200806_142047.png	10.5 KB	2020-08-06	Julie_CAO
Screenshot_20200806_142736.png	22.4 KB	2020-08-06	Julie_CAO
Screenshot_20200806_144712.png	14.7 KB	2020-08-06	Julie_CAO