

openQA Project - action #56789

coordination # 58184 (Blocked): [saga][epic][use case] full version control awareness within openQA, e.g. user forks and branches, fully versioned test schedules and configuration settings

coordination # 94750 (New): [epic] Support needles from git-cloned or separate repo if casedir points to a git-repo

New needles from git repository not working with openqa-clone-custom-git-refspec

2019-09-11 08:14 - tjyrinki_suse

Status:	New	Start date:	2019-09-11
Priority:	Low	Due date:	
Assignee:		% Done:	0%
Category:	Feature requests	Estimated time:	0.00 hour
Target version:	future		
Difficulty:	hard		

Description

Use case:

To be able to test tests in production (openqa.opensuse.org or openqa.suse.de) before merging the needles and tests themselves, since results have proven to vary on those loaded machines compared to local openQA instance or VM.

It is currently working if there are no new needles involved, but this issue describes the problem with custom needles which would be theoretically supported by openqa-clone-custom-git-refspec but not working in practice.

Problem description:

First of all, documentation [1] says "Path to needles subdirectory to use, defaults to "needles" within PRODUCTDIR. Can be a git repository URL, comparable to CASEDIR", and CASEDIR states "for example git@github.com:os-autoinst/os-autoinst-distri-opensuse.git#feature/test".

[1] https://github.com/os-autoinst/os-autoinst/blob/master/doc/backend_vars.asciidoc

However,

```
openqa-clone-custom-git-refspec https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/8332 https://openqa.opensuse.org/tests/xxxxxxxx --apikey=XXX --apisecret=XXX NEEDLES_DIR=git@github.com:tjyrinki/os-autoinst-needles-opensuse.git#poo-12345
```

results in:

```
Could not create directory '/var/lib/empty/.ssh'.
Host key verification failed.
fatal: Could not read from remote repository.
```

Using https instead (NEEDLES_DIR=<https://github.com/tjyrinki/os-autoinst-needles-opensuse.git>) gets one further, but openQA complains about wrong needles location:

```
init needles from /var/lib/openqa/pool/15/os-autoinst-needles-opensuse
Needle /var/lib/openqa/pool/15/os-autoinst-needles-opensuse/ssh-login-ok-20160813.json is not under project directory /var/lib/openqa/cache/xxx at /usr/lib/os-autoinst/needle.pm line 63.
```

Oliver Kurz suggested playing around with the PRODUCTDIR to try to workaround the forbidding use of (correctly checked out) needles dir which I did but it didn't get any better with trying to set that to eg pool/... directories, which also changes on every run.

Related issues:

Related to openQA Infrastructure - action #57962: Needle load time increase	Resolved	2019-10-10
Related to openQA Project - action #58289: Huge amount of "Needle file .* not...	Resolved	2019-10-17
Related to openQA Project - action #54965: Cannot inspect the source code of ...	Resolved	2019-08-01
Related to openQA Tests - action #58574: [openqa] test fails in dashboard, de...	Resolved	2019-10-23
Related to openQA Project - action #59184: Research about testing with a cust...	Resolved	2019-11-07
Related to openQA Project - action #70333: providing a git branch for NEEDLES...	Resolved	2020-08-20

Related to openQA Project - action #91905: webui fails to display needles whe...
Copied to openQA Project - action #59330: "needledir not found: .*null/": ope...

New 2021-04-28
Resolved 2019-11-12

History

#1 - 2019-09-11 08:16 - coolo

- Category set to Feature requests
- Target version set to Ready

git@ urls are no option for sure. But https needle dirs should be supported, but we may need to know where to put it, so possibly we need yet another variable for that

#2 - 2019-09-11 08:22 - tjyrinki_suse

- Description updated
- Category deleted (Feature requests)
- Target version deleted (Ready)

#3 - 2019-09-11 08:23 - tjyrinki_suse

- Category set to Feature requests
- Target version set to Ready

#4 - 2019-09-11 18:29 - okurz

coolo wrote:

but we may need to know where to put [the needles], so possibly we need yet another variable for that

they are already cloned to the pool directory however loading from there is prohibited which is merely a security feature, not a real technical limitation

#5 - 2019-10-01 15:16 - mkittler

- Status changed from New to In Progress
- Assignee set to mkittler
- Target version changed from Ready to Current Sprint

I guess allowing \$pool_directory/os-autoinst-needles-opensuse in addition to \$bmwqemu::vars{PRJDIR} would to the trick then.

#6 - 2019-10-07 11:56 - mkittler

- Status changed from In Progress to Feedback

PR has been merged: <https://github.com/os-autoinst/os-autoinst/pull/1217>

Let's see whether it works in production.

#7 - 2019-10-09 08:18 - tjyrinki_suse

Looking initially good on my own updated instance!

Fail without NEEDLES_DIR: <http://d432.gam.suse.de/tests/125>
Pass with NEEDLES_DIR: <http://d432.gam.suse.de/tests/124>

I will re-test on openqa.suse.de once it's deployed.

#8 - 2019-10-09 15:22 - okurz

tjyrinki_suse wrote:

I will re-test on openqa.suse.de once it's deployed.

already the case

#9 - 2019-10-10 16:56 - coolo

- Status changed from Feedback to New

We reverted this in master - and will shortly also do it on osd due to issue 57962

#10 - 2019-10-10 18:31 - okurz

- Related to action #57962: Needle load time increase added

#11 - 2019-10-11 15:05 - mkittler

- Status changed from New to In Progress

Another attempt: <https://github.com/os-autoinst/os-autoinst/pull/1225>

I've checked the performance this time:

before:

```
HEAD ist jetzt bei e2642e31 Merge pull request #1224 from os-autoinst/revert-1217-consider-needles-in-pool-directory
[martchus@linux-9lzf os-autoinst]$ time perl -I. -e 'use bmqemu; use needle; needle::init($bmqemu::vars{NEEDLES_DIR} = $bmqemu::vars{PRJDIR} = "$ENV{OPENQA_BASEDIR}/openqa/share/tests/opensuse/products/opensuse/needles");'
[2019-10-11T16:51:06.861 CEST] [debug] init needles from /hdd/openqa-devel/openqa/share/tests/opensuse/products/opensuse/needles
[2019-10-11T16:51:07.404 CEST] [debug] loaded 5818 needles

real    0m0,823s
user    0m0,698s
sys     0m0,125s
```

after:

```
Ihr Branch ist auf demselben Stand wie 'Martchus/needle-dir-3'.
[martchus@linux-9lzf os-autoinst]$ time perl -I. -e 'use bmqemu; use needle; $bmqemu::vars{PRODUCTDIR} = "$ENV{OPENQA_BASEDIR}/openqa/share/tests/opensuse/products/opensuse"; needle::init;'
[2019-10-11T16:40:15.445 CEST] [debug] init needles from /hdd/openqa-devel/openqa/share/tests/opensuse/products/opensuse/needles
[2019-10-11T16:40:16.014 CEST] [debug] loaded 5818 needles

real    0m0,851s
user    0m0,738s
sys     0m0,112s
```

#12 - 2019-10-15 09:21 - tjyrinki_suse

PR 1226 is the actual new needle loading fix: <https://github.com/os-autoinst/os-autoinst/pull/1226>

#13 - 2019-10-15 10:24 - okurz

- Parent task set to #58184

#14 - 2019-10-17 09:02 - okurz

- Related to action #58289: Huge amount of "Needle file .* not found where expected. Check /var/lib/openqa for distri symlinks" on o3 in /var/log/openqa added

#15 - 2019-10-17 11:05 - mkittler

- Status changed from In Progress to Feedback

We needed to revert the change again: <https://github.com/os-autoinst/os-autoinst/commit/5bc47a5ca12feb1a890735e7fe6daea2e125ff96>

The problem is that os-autoinst now no longer references needles relative to the project directory. Instead, needles are referenced relative to the needle directory which is not what openQA expects.

This reveals a further difficulty of implementing this feature: The openQA web UI doesn't know the custom needles but it relies on knowing the involved needles to display candidates. I could now revert the revert and amend it to emit paths relative to the project directory. However, it is not clear how the case of custom needles should be handled.

I still like the general refactoring of the needle directory handling of my change which is now lost. Maybe I'll create a separate PR for this refactoring which basically simplifies the use of \$needle::needlesdir and \$bmqemu::vars{NEEDLES_DIR} (\$bmqemu::vars{NEEDLES_DIR} has previously been used without validation by the needle downloader).

The actual issue is harder to solve. We could just flag those needles in the result-*.json to prevent the web UI from displaying them. Or we somehow

embed custom needle data within the results.

#16 - 2019-10-17 11:09 - mkittler

By the way, we have unit tests (01-test_needle.t) which claim in comments that they cover what has been broken by the change. So these tests don't seem to be correct.

When checking the result-*.json files manually the problem is actually easy to spot. Before:

```
{
  "dents" : 0,
  "details" : [
    {
      "area" : [
        {
          "h" : 300,
          "result" : "ok",
          "similarity" : 100,
          "w" : 618,
          "x" : 0,
          "y" : 0
        }
      ],
      "error" : 0,
      "frametime" : [
        "0.54",
        "0.58"
      ],
      "json" : "tests/opensuse/products/opensuse/needles/bootmenu-TW-20190425.json",
      "needle" : "bootmenu-TW-20190425",
    }
  ]
}
```

After:

```
{
  "dents" : 0,
  "details" : [
    {
      "area" : [
        {
          "h" : 300,
          "result" : "ok",
          "similarity" : 100,
          "w" : 618,
          "x" : 0,
          "y" : 0
        }
      ],
      "error" : 0,
      "frametime" : [
        "0.58",
        "0.62"
      ],
      "json" : "bootmenu-TW-20190425.json",
      "needle" : "bootmenu-TW-20190425",
    }
  ]
}
```

#17 - 2019-10-17 14:22 - mkittler

Note that the whole issue should only occur when caching is enabled (like it is on all of our production instances). Otherwise the pool directory is actually within the project directory. See Worker.pm:

```
my $self = $class->SUPER::new(
    instance_number    => $instance_number,
    no_cleanup         => $cli_options->{'no-cleanup'},
    pool_directory     => "$OpenQA::Utils::prjdir/pool/$instance_number",
```

#18 - 2019-10-17 14:52 - okurz

We propose to get rid of PRJDIR at least in the scope of os-autoinst where we should not need it.

openQA changes:

```
diff --git a/lib/OpenQA/Schema/Result/Needles.pm b/lib/OpenQA/Schema/Result/Needles.pm
index 5a0ee0193..255c730c1 100644
--- a/lib/OpenQA/Schema/Result/Needles.pm
+++ b/lib/OpenQA/Schema/Result/Needles.pm
```

```

@@ -113,7 +113,7 @@ sub update_needle {
    my $guard = $schema->txn_scope_guard;

    if (!-f $filename) {
-       $filename = catdir($OpenQA::Utils::sharedir, $filename);
+       $filename = catdir($module->job->needle_dir(), $filename);
        if (!-f $filename) {
            log_error(
                "Needle file $filename not found where expected. Check $OpenQA::Utils::prjdir for distri syml
inks");
diff --git a/lib/OpenQA/Worker/Engines/isotovideo.pm b/lib/OpenQA/Worker/Engines/isotovideo.pm
index 9a447b11..276ddfb77 100644
--- a/lib/OpenQA/Worker/Engines/isotovideo.pm
+++ b/lib/OpenQA/Worker/Engines/isotovideo.pm
@@ -204,7 +204,6 @@ sub engine_workit {
    OPENQA_URL      => $openqa_url,
    WORKER_INSTANCE => $instance,
    WORKER_ID       => $workerid,
-   PRJDIR          => $OpenQA::Utils::sharedir,
    %$job_settings
    );
    log_debug('Job settings:');
@@ -231,8 +230,6 @@ sub engine_workit {
    );
    my $rsync_request_description = "Rsync cache request from '$rsync_source' to '$shared_cache'";

-   $vars{PRJDIR} = $shared_cache;
-
    # enqueue rsync task; retry in some error cases
    for (my $remaining_tries = 3; $remaining_tries > 0; --$remaining_tries) {
        return {error => "Failed to send $rsync_request_description"} unless $rsync_request->enqueue;

```

os-autoinst changes:

```

diff --git a/autotest.pm b/autotest.pm
index fc06f3de..2b7f83fe 100644
--- a/autotest.pm
+++ b/autotest.pm
@@ -54,7 +54,8 @@ loadtest is called.
sub find_script {
    my ($script) = @_;
    my $casedir = $bmwgemu::vars{CASEDIR};
-   my $script_override_path = join('/', $bmwgemu::vars{PRJDIR} // '', 'factory/other', $script);
+   # TODO use proper way not relying on PRJDIR
+   my $script_override_path = join('/', 'factory/other', $script);
    if (-f $script_override_path) {
        bmwgemu::diag("Found override test module for $script: $script_override_path");
        return File::Spec->abs2rel($script_override_path, $casedir);
diff --git a/bmwqemu.pm b/bmwqemu.pm
index a8dd756d..2a3302f5 100755
--- a/bmwqemu.pm
+++ b/bmwqemu.pm
@@ -48,7 +48,6 @@ $| = 1;

our $default_timeout = 30; # assert timeout, 0 is a valid timeout
-our $openqa_default_share = '/var/lib/openqa/share';

my @ocorrect;

@@ -162,15 +161,6 @@ sub ensure_valid_vars {
    die "CASEDIR variable not set, unknown test case directory" if !defined $vars{CASEDIR};
    die "No scripts in $vars{CASEDIR}" if !-e "$vars{CASEDIR}";
    unless ($vars{PRJDIR}) {
-       if (index($vars{CASEDIR}, $openqa_default_share) != 0) {
-           diag("PRJDIR not specified and CASEDIR ($vars{CASEDIR}) does not appear to be a subdir of default
($openqa_default_share), using CASEDIR instead");
-           $vars{PRJDIR} = $vars{CASEDIR};
-       }
-       else {
-           $vars{PRJDIR} = $openqa_default_share;
-       }
    }
    save_vars();

```

```

}

diff --git a/needle.pm b/needle.pm
index 2e22ad84..525d4a05 100644
--- a/needle.pm
+++ b/needle.pm
@@ -53,20 +53,20 @@ sub new {
    }

    my $self = {};
    - if (index($jsonfile, $bmwqemu::vars{PRJDIR}) == 0) {
    -     $self->{file} = substr($jsonfile, length($bmwqemu::vars{PRJDIR}) + 1);
    + if (index($jsonfile, $bmwqemu::vars{CASEDIR}) == 0) {
    +     $self->{file} = substr($jsonfile, length($bmwqemu::vars{CASEDIR}) + 1);
    }
    - elsif (-f File::Spec->catfile($bmwqemu::vars{PRJDIR}, $jsonfile)) {
    + elsif (-f File::Spec->catfile($bmwqemu::vars{CASEDIR}, $jsonfile)) {
        # json file path already relative
        $self->{file} = $jsonfile;
    -     $jsonfile = File::Spec->catfile($bmwqemu::vars{PRJDIR}, $jsonfile);
    +     $jsonfile = File::Spec->catfile($bmwqemu::vars{CASEDIR}, $jsonfile);
    }
    else {
    -     die "Needle $jsonfile is not under project directory $bmwqemu::vars{PRJDIR}";
    +     die "Needle $jsonfile is not under CASEDIR $bmwqemu::vars{CASEDIR}";
    }

    - # $json->{file} contains path relative to $bmwqemu::vars{PRJDIR}
    - # $jsonfile contains absolute path within $bmwqemu::vars{PRJDIR}
    + # $json->{file} contains path relative to $bmwqemu::vars{CASEDIR}
    + # $jsonfile contains absolute path within $bmwqemu::vars{CASEDIR}

    if (!$json) {
        try {

```

#19 - 2019-10-17 14:56 - mkittler

And my previous comment is incorrect. The variable `prjdir` is used inconsistently. On the worker side (the diff from my last comment) it is something like `/var/lib/openqa` and within `isotovideo` `/var/lib/openqa/share`. So the pool directory is *not* within the `prjdir` from `isotovideo`'s point of view.

#20 - 2019-10-18 11:50 - mkittler

- Status changed from Feedback to In Progress

Draft PRs for removing `PRJDIR` in `os-autoinst` context and adapting `openQA` to it: <https://github.com/os-autoinst/os-autoinst/pull/1234>, <https://github.com/os-autoinst/openQA/pull/2410>

With that change needles are referenced relatively to the needle directory and not the `prjdir/sharedir`. This is a first step to make `os-autoinst/openQA` more flexible. Of course the `openQA` web UI will still not be able to display candidates from custom needle checkouts. (It will display the version from its version of the needle repo or won't display the candidate at all if it doesn't exist in its needle repo.)

#21 - 2019-10-24 11:54 - mkittler

The draft PRs are actually no longer drafts and one has already been merged. These PRs solve the problem of using these needles on the `os-autoinst` side.

However, the web UI would still be unaware. I previously suggested to embed the candidates into the results. After discussing the issue with the team it seems just referencing a Git hash makes more sense.

So I would implement it in the following way:

1. `os-autoinst` side

1. The Git `revspec/hash` is determined and assigned to a test variable (e.g. `NEEDLES_GIT_REVSPEC`) when checking out the custom repo. (The repo URL itself might just be something like `https://github.com/tjyrinki/os-autoinst-needles-opensuse.git` and therefore does not already contain the concrete `revspec` itself.)
2. We need a way to make up a URL like `https://raw.githubusercontent.com/...` on the web UI side for the job. So we need to assign all relevant details to test variables (e.g. `NEEDLES_GIT_HOST=github` and `NEEDLES_GIT_REPO_PATH=tjyrinki/os-autoinst-needles-opensuse`).

2. web UI side

1. The image preview and needle editor are made aware of the mentioned variables. If they exist and `NEEDLES_GIT_HOST` refers to a known Git host the usual links to candidate needles are replaced by links pointing to that the needle on the host (e.g. `https://raw.githubusercontent.com/tjyrinki/os-autoinst-needles-opensuse/f456756cdca46b607415df43332961bded6b3620/42-bootmenu-xen-kernel-20150918.png`).
2. The needle update code must be aware as well.

1. In particular, it must not reject a needle if it is not locally available if NEEDLES_GIT_* variables are present.
2. Not sure whether those "Git needles" should be tracked within the database. I'd start with not adding them to the database and see how well that goes. Likely the database entries for needles are only required for cleanup so tracking these needles within the openQA database wouldn't be necessary.
3. So if no special test variables are set everything works as before. The Git host specific code is only an optional addition. It shouldn't be hard to implement URL generators for different Git hosts if that's required (likely it is since we're also using GitLab for needles).

The entire approach could roughly apply to <https://progress.opensuse.org/issues/54965> as well. Just replace "needle" with "test code". However, in this case we would likely not show our own source code viewer anymore and simply redirect to the Git host web page.

#22 - 2019-10-24 12:33 - mkittler

- Related to action #54965: Cannot inspect the source code of the tests from my fork added

#23 - 2019-10-24 16:04 - okurz

I missed the monitoring alerts since this morning. Reverted openQA package updates on o3 with sudo zypper in --oldpackage /var/cache/zypp/packages/devel_openQA/noarch/openQA{-common,-client}-4.6.1571759996.a3da178cd-lp151.1938.1.noarch.rpm

Now workers fail, e.g. openqa-worker1:1 :

```
Oct 24 18:06:48 openqaworker1 worker[3203]: [info] Registered and connected via websockets with openQA host http://openqa-opensuse and worker ID 51
Oct 24 18:06:49 openqaworker1 worker[3203]: Use of uninitialized value in numeric ne (!=) at /usr/share/openqa/script/./lib/OpenQA/Worker/Job.pm line 694.
Oct 24 18:06:49 openqaworker1 worker[3203]: [info] Test schedule has changed, reloading test_order.json
Oct 24 18:06:49 openqaworker1 worker[3203]: [warn] Can't open /var/lib/openqa/pool/1/testresults/test_order.js on for result upload - likely isotovideo could not be started or failed early. Error message: No such file or dire>
Oct 24 18:06:49 openqaworker1 worker[3203]: Mojo::Reactor::Poll: Timer failed: Can't use an undefined value as an ARRAY reference at /usr/share/openqa/script/./lib/OpenQA/Worker/Job.pm line 1132.
Oct 24 18:06:49 openqaworker1 worker[3203]: [error] Stopping because a critical error occurred.
Oct 24 18:06:49 openqaworker1 worker[3203]: [error] Another error occurred when trying to stop gracefully due to an error. Trying to kill ourself forcefully now.
```

git log of openQA:

a3da178cd..da2de85d9

```
fd17bff6f (okurz/fix/package_tests) spec: Fix package tests already failing on database setup
adae4ab06 Fix VNC port info for ports > 99 in devel mode
8a682cc98 Prevent logging needle problems twice
e65f62948 Provide default filename for needles from the name
0d4152d19 Move needle_info() from Utils.pm to Step.pm
71f1bdf67 (okurz/fix/qemu_kvm) Fix missing 'qemu-kvm' in non-x86_64 package tests
36aa97443 Reduce use of project dir to be more flexible regarding custom test runs
9bf26a71b Calculate coverage in Circleci in all branches
```

and os-autoinst:

e3b331a2..5391548d

```
7d482890 (perlpunk/new-status) Increase version numbers
87276ac7 Add new status file that worker can read from
c52303a0 Consider tests with `tools/tidy --only-changed`
ab0554c2 (okurz/feature/container) spec: Fix missing, additional runtime requirements
a45e17fa Allow tidy to run only over local changes
37d31fc7 Improve 'check_ssh_serial'
55db2e0d Make start_serial_grab blocking
5266e820 Fix svirt backend's 100 % CPU usage
```

I put all git commits between yesterday (last good) and today in <https://progress.opensuse.org/issues/56789#note-23>. I suspect either 0d4152d19 Move needle_info() from Utils.pm to Step.pm or 36aa97443 Reduce use of project dir to be more flexible regarding custom test runs from openQA to be problematic or 87276ac7 Add new status file that worker can read from from os-autoinst. What's disturbing that the old web ui does not seem to work correctly even with a rolled back, old worker anymore. as if the settings in the vars.json files is now troubling. I will upgrade the webui again to the latest version. If we are lucky in the new version just the custom git clone jobs are problematic. seems the newer webui is better: <https://openqa.opensuse.org/tests/1065110#> is running ok, regardless if worker version is from today or yesterday. As expected again we have a lot of errors referencing needles again in the log file. ggardet suspected that only the custom-git jobs are affected but from the frequency of the message appearing I suspect it is affecting all. To check yourself use sudo tail -f /var/log/openqa | grep 'not found within ' on o3.

#24 - 2019-10-24 16:22 - cdywan

mkittler wrote:

So I would implement it in the following way:

1. os-autoinst side

1. The Git revspec/hash is determined and assigned to a test variable (e.g. NEEDLES_GIT_REVSPEC) when checking out the custom repo. (The repo URL itself might just be something like <https://github.com/tjyrinki/os-autoinst-needles-opensuse.git> and therefore does not already contain the concrete revspec itself.)
2. We need a way to make up a URL like <https://raw.githubusercontent.com/...> on the web UI side for the job. So we need to assign all relevant details to test variables (e.g. NEEDLES_GIT_HOST=github and NEEDLES_GIT_REPO_PATH=tjyrinki/os-autoinst-needles-opensuse).

How about using a template instead? Something like NEEDLES_GIT_RAW=[https://gitlab.com/\\$REPO/raw/\\$SHA1/\\$FILENAME](https://gitlab.com/$REPO/raw/$SHA1/$FILENAME) or NEEDLES_GIT_RAW=[https://raw.githubusercontent.com/\\$REPO/\\$SHA1/\\$FILENAME](https://raw.githubusercontent.com/$REPO/$SHA1/$FILENAME) depending on the host. IT wouldn't require any further knowledge except filling in the variables.

#25 - 2019-10-24 16:40 - mkittler

So every test which makes use of this needs these variables provided from the outside. I don't think that makes it easier for our users. And we still needed to be clever to deduce \$REPO (or require it to be provided, too).

#26 - 2019-10-24 18:11 - okurz

- Priority changed from Normal to Urgent

See notes in the above comment [#56789#note-23](#) . Please monitor the situation on o3 and crosscheck "normal" jobs as well as custom-git jobs in production as well.

#27 - 2019-10-25 11:12 - mkittler

Taking into account that we need to support custom GitLab instances with URLs like <https://gitlab.suse.de/openqa/os-autoinst-needles-sles.git> being clever with auto-assigning NEEDLES_GIT_HOST and NEEDLES_GIT_REPO_PATH like I suggested is hard. There could also be Git hosts like <https://martchus.no-ip.biz/gitea/Martchus/synctingtray.git> where NEEDLES_GIT_HOST is only part of the path. That means I will go for the approach [cdywan](#) suggested. However, the \$REPO part should better be provided from the outside as well because I'm not sure whether we can deduce it in the general case from the repo URL.

By the way, here's the fix for the problem mentioned by [okurz](#): <https://github.com/os-autoinst/openQA/pull/2432>

#28 - 2019-10-25 13:00 - okurz

- Related to action #58574: [openqa] test fails in dashboard, desktop runner shows up a little bit late added

#29 - 2019-10-25 13:09 - okurz

Btw, I haven't reviewed your recent suggestions for "features" yet as I am still in "urgent regression fixing mode" ;) I would appreciate you give me some time to review your ideas. I can of course also review them in the form of PRs but this would have involved more upfront work by you then.

<https://github.com/os-autoinst/openQA/pull/2432> merged and deployed on openqaworker1.o.o already. <https://openqa.opensuse.org/tests/1065586> is a "custom-git-clone" job running on the worker and it looks better than the previous incomplete <https://openqa.opensuse.org/tests/1065537/file/autoinst-log.txt>

#30 - 2019-10-25 13:29 - mkittler

- Priority changed from Urgent to Normal

The fix works so I'm lowering the priority. Of course candidates for jobs which ran with the broken version can not be restored.

I set the ticket to feedback to give you some time to review my idea.

#31 - 2019-10-25 13:29 - mkittler

- Status changed from In Progress to Feedback

#32 - 2019-10-25 14:11 - mkittler

I also have another idea: The full link (e.g. <https://raw.githubusercontent.com/tjyrinki/os-autoinst-needles-opensuse/f456756cdca46b607415df43332961bde6b3620/42-bootmenu-xen-kernel-20150918.png>) could already be generated on the os-autoinst side when writing the results. Then openQA really only needs to check whether the candidate reference has an URL and would simply use that URL. The advantage would be that the code dealing with Git would be spread less across the repositories. Additionally, the template substitution wouldn't run each and every time the test result is displayed but only once when it is generated. The performance impact of this feature on the job module upload would also be less because the web UI doesn't need to check the settings of the related job.

#33 - 2019-10-28 13:50 - mkittler

I've continued with the implementation to see myself whether the idea works out. I think there actually speaks nothing against being able to set URL templates for some "well known hosts" automatically (e.g. GitHub) and relying on the user to provide them for other hosts.

But there's a more annoying problem: The server-side controller code of the image preview and the needle editor relies on the local needle files to read the area information. However, the area info is actually just passed to the client. So in theory it would be possible to just pass the URL to the client like for the PNG (e.g. <https://raw.githubusercontent.com/tjyrinki/os-autoinst-needles-opensuse/f456756cdca46b607415df43332961bded6b3620/42-bootmenu-xen-kernel-20150918.json>) and fetch the info via AJAX. The problem with that is that cross origin requests like that are not allowed (e.g. try `$.ajax({url: "https://raw.githubusercontent.com/tjyrinki/os-autoinst-needles-opensuse/f456756cdca46b607415df43332961bded6b3620/42-bootmenu-xen-kernel-20150918.json"});` in the JavaScript console).

The alternative would be loading the JSON from the Git host in our server's controller code but that's not really nice because it adds additional load to the server and disables the possibility for lazy loading. We could also provide a special "proxy" routes on our server. That way we could at least only lazy-load the candidates the user selects.

And the other alternative is of course to go back to the original plan of embedding the candidates within the test results. We could also only embed the JSON/areas and still load the PNG directly from the Git host. Combining the two approaches feels a bit inconsistent, though.

And since the cross origin restrictions don't apply to the PNG: <https://www.peter-eigenschink.at/projects/steganographyjs> (not a serious suggestion)

#34 - 2019-10-28 16:09 - mkittler

- *Difficulty set to hard*

Or we really have the Git checkout on the web UI host, too. So when the job is added not only the worker/isotovideo clones the repository but also the web UI would add the remote and fetch the required branch. This could happen in a Minion task while the job is scheduled. The file contents for a specific version could be read via `git show $SHA:$NEEDLE_PATH`. This approach has the advantage that it is Git host independent. This approach adds a lot of complexity on the web UI side (e.g. we need to extend the needle cleanup) but it seems conceivable.

#35 - 2019-10-28 17:10 - mkittler

That's roughly how the idea described in the last comment would work:

1. Someone schedules a job with `NEEDLES_DIR=https://github.com/tjyrinki/os-autoinst-needles-opensuse.git#a635f114218f59baada8c4cb2bdbdc6b6e769a84`. We require to add the branch name or checksum here from the beginning.
2. When the job is executed isotovideo clones the repo as it is possible already. When <https://github.com/os-autoinst/os-autoinst/pull/1234> is merged os-autoinst will be able to actually use that checkout.
 - isotovideo should better preserve the `NEEDLES_DIR` variable (so far it overrides it with the local path to the cloned repository)
3. The web UI adds the remote to its needle checkout when the job is scheduled.
 - e.g. the following commands would do it:

```
git -C $needle_checkout remote add tjyrinki https://github.com/tjyrinki/os-autoinst-needles-opensuse.git # ignoring "fatal: remote tjyrinki already exists"
git -C $needle_checkout fetch tjyrinki a635f114218f59baada8c4cb2bdbdc6b6e769a84
```
 - The commands are quite fast but it might still make sense to run them as Minion job.
 - To avoid interference with the needle editor we could use a separate checkout here.
4. The web UI can read a needle files via e.g. `git show a635f114218f59baada8c4cb2bdbdc6b6e769a84:42-bootmenu-xen-kernel-20150918.png`.
 - [Mojo::LOLoop::ReadWriteFork](#) might help to implement this

Things which still need to be considered:

- Cleanup/tracking
- Possibly updating "last seen/matched" info
- Error handling when the Git repo isn't ready by the time the candidate needs to be displayed
- Deducing the jobs version info from the job settings variable `NEEDLES_DIR` all the time it is needed is likely inefficient. Maybe add a separate database column?

#36 - 2019-10-29 10:01 - mkittler

- *Status changed from Feedback to New*

- *Assignee deleted (mkittler)*

- *Target version changed from Current Sprint to Ready*

The Git checkout on the isotovideo side could be made more efficient by having a master needles repo permanently around and adding remotes as needed (like described for the web UI). We can then clone from that local repo via `git clone --shared` into the pool directory.

We agreed to improve the overall epic first before continuing with this. So I'm unassigning for now and remove it from the current sprint. I also set the ticket back to "New" as "Feedback" is now preserved for *user* feedback and not feedback from other openQA developers.

#37 - 2019-11-07 13:19 - tinita

- *Related to action #59184: Research about testing with a custom git ref added*

#38 - 2019-11-12 09:14 - okurz

- Copied to action #59330: "needledir not found: .*null/": openqa-clone-custom-git-refspec fails setting NEEDLES_DIR=null with new os-autoinst or openQA added

#39 - 2019-12-08 12:57 - okurz

I did a smaller change in <https://github.com/os-autoinst/os-autoinst/pull/1286> to allow specifying PRODUCTDIR and NEEDLESDIR relative to the CASEDIR top.

#40 - 2020-04-23 18:47 - okurz

- Priority changed from Normal to Low

The last mentioned change should help or maybe even resolves the problem, but certainly it's less severe.

#41 - 2020-06-03 07:24 - okurz

- Target version changed from Ready to future

we should focus on other issues in the parent [#58184](#) first before we look into any potential special handling of needle repos which are not shared with test repos. General idea I have: Use git submodules for needles if not the same repo as for tests.

#42 - 2020-08-24 08:06 - mkittler

- Related to action #70333: providing a git branch for NEEDLES_DIR does not appear to work on o3 added

#43 - 2021-04-28 14:33 - mkittler

- Related to action #91905: webui fails to display needles when using a git repository as CASEDIR added

#44 - 2021-06-25 19:10 - okurz

- Parent task changed from #58184 to #94750