

Foreman - Feature #1377

Foreman should support a "versioning" backend (for templates, params/vars, etc)

12/07/2011 09:59 AM - Marcello de Sousa

Status: Closed	
Priority: Normal	
Assignee:	
Category:	
Target version: 1.0	
Difficulty: medium	Fixed in Releases:
Triaged:	Found in Releases:
Bugzilla link:	Red Hat JIRA:
Pull request:	
Description Use case example : Say you use a template to provision a machine. Afterwards someone edits this template. This is saved directly in the DB and if you need to re-provision that machine (or a "clone") Foreman is not able to guarantee you are deploying the same machine as it's not aware of the version of the template you used. You are also not able to rollback to a previous working version. Possible other places where you might want versioning (Similar story where you can't roll back to a previous version.): - Smartvars / Parameters - Hostgroups - etc We had different discussions about this implementation and I guess it all boils down to 2 options: Option 1- Implement that on the DB creating your own version control system Option 2- Use an existing alternative backend like GIT to store that information Both options have advantages and disadvantages and we could probably discuss them here in this ticket.	
Related issues: Has duplicate Foreman - Feature #1555: Template should be audited and/or vers... Duplicate 03/27/2012	

Associated revisions

Revision 09ce1364 - 06/20/2012 03:43 PM - Amos Benari

audit as twitter and versions for template.

This also closes #1377 as it offer versions and diff view of the tempates.

it is not ideal, but it should handle the most common case for now,
where we should open other feature requests to support all of the ideas in #1377

History

#1 - 12/07/2011 10:04 AM - Marcello de Sousa

I have a certain bias towards option 2 as I get the feeling we are reinventing the wheel with option 1.
But I understand there might be some challenges or disadvantages there...

Please comment !

#2 - 12/07/2011 10:20 AM - Marcello de Sousa

Just wondering ...

Would it make sense to be able to regularly export Foreman "state" to the file system (every time that state is changed) ?

Then we could check that in our own version control system(VCS). In other words, the VCS would run in parallel to the DB and foreman would interface with it.

Foreman could be aware of that VCS and should allow us to import all or part of what was checked-in.

#3 - 12/09/2011 10:23 AM - Benjamin Papillon

Is it possible to replicate the puppet method ? Foreman update its database itself as soon as it detects the directory has been updated.

From this point, I see 2 routes :

- Foreman got its interface readonly and inform the user to use its VCS for files modifications
- Foreman still has r/w access to the data : How to commit? How to manage conflicts?

The first step would be cool for the moment. I see the conflicts management as a hard issue to overcome.

#4 - 12/09/2011 04:03 PM - Tim Speetjens

My 2 cents:

I always loved text representations for being able to do

- version control at your liking
- show differences textually. Database differences are way much harder to achieve.

Although I love git, I can imagine that some still prefer SVN. Forcing the use of git, is not the best approach imho.

It would be great to be able to do some kind of export to a directory, but without some sort of import mechanism, it's only half as useful.

Each db update could trigger the same kind of change in the filesystem representation, but it is a brittle method: Either everything should be exported over and over again, guaranteeing a consistent export, either we risk inconsistencies when someone decided to change file contents.

#5 - 03/21/2012 09:11 PM - Doug Weimer

Has anyone started working on implementing one of these methods? I'm interested in finding a way to get foreman information into a version control system and have some time to work on the problem. My initial thought is something similar to a post commit hook where a script is called when a change is made. That would give a lot of flexibility in implementing different versioning backends. The downside would be that it could be difficult to handle rollback and provide version history through the web UI.

To go the other direction (text file changes -> foreman), could the API be used to send in updates? If so, then all that would be needed is a standardized file layout for the configuration files and post commit hooks could be used to submit the contents for the changed files via the API.

#6 - 05/21/2012 08:57 AM - Ohad Levy

- *Difficulty set to medium*

#7 - 06/18/2012 07:50 AM - Ohad Levy

It is possible to use the API to get the template content, and its even possible to use the API to store new content, so a simple wrapper around an editor / VCS should not be too hard to do.

saying that, I feel that versions does make a lot of sense to foreman, e.g.

say you have appX which is now ready for production, that includes a list of puppet classes, templates, operating systems, and maybe even virtual hw description.

assuming some time passed, and now you want to have a new version of appX (maybe you want to run it on a newer OS), then I would assume that you don't want to break the older deployment (e.g. you still want to be able to deploy the old version, and the newer one).

imho, there are a few other good reasons you want to store it in foreman (for diff, rollback/undo, audit log etc), which makes my preference toward storing it in the db.

but in all cases, I don't see a good reason why 1 and 2 cant co exists, and I think it makes a lot of sense to allow custom hooks, in which every time something get saved / updated/ destroyed etc, some other action can be performed, to me, it makes sense to have a notification system which would allow people to be notified and act (via the [API](#)) for their own wishes.

to try and have a little bit of progress, I suggest to start working on 1, and making 2 a separate ticket, in which over time (not for v1.0), we'll try to make more pluggable.

if anyone has some free bandwidth to work on this, I'll be more than happy to assist in anyway I can.

#8 - 06/20/2012 04:54 PM - Amos Benari

- *Status changed from New to Closed*

- *% Done changed from 0 to 100*

Applied in changeset [09ce13649ce41444f392dde148d29eb3437be3c2](#).