

Redmine - Feature #3667 granular post-commit hook

2009-07-24 14:44 - Felix Schäfer

| | | | |
|---|--------|------------------------|------------|
| Status: | New | Start date: | 2009-07-24 |
| Priority: | Normal | Due date: | |
| Assignee: | | % Done: | 0% |
| Category: | SCM | Estimated time: | 0.00 hour |
| Target version: | | | |
| Resolution: | | | |
| Description | | | |
| <p>I have written a commit-hook that will fetch changes only for the project with the same name as the svn repository, see http://www.redmine.org/boards/2/topics/7633 for the full discussion, code copied here:</p> <pre>\$ cat /path/to/your/redmine/directory/extra/svn/trigger_Redmine_fetch_changeset #!/usr/bin/env /path/to/your/redmine/directory/script/runner Project.find_by_identifier(File.basename(ARGV[0])).repository.fetch_changesets \$ cat /path/to/your/svn/directory/identifier/hooks/post-commit #!/bin/sh REPOS="\$1" RAILS_ENV="production" /path/to/your/redmine/directory/extra/svn/trigger_Redmine_fetch_changeset \$1</pre> <p>This will obviously only work if the svn repo has the same name as the identifier of your project, but that is given if created with reposman.rb. I haven't integrated my git with redmine yet, but I think the script could be made to work with a git project by just adding something like <code>ARGV[0].chomp!("/") .chomp!(".git")</code> (think the space away, I'd get an image markup from the !s without it...) before the call to Project and passing GIT_DIR to the script.</p> <p>Anyway, I figured I'm not the only one who would like to get a post-commit triggered refresh without having to wait for ages because redmine sorts through all repositories, maybe this should go in extra/svn/ or maybe in the wiki?</p> | | | |

History

#1 - 2009-07-28 10:16 - Felix Schäfer

Felix Schäfer wrote:

```
This will obviously only work if the svn repo has the same name as the identifier of your project, but that is given if created with reposman.rb. I haven't integrated my git with redmine yet, but I think the script could be made to work with a git project by just adding something like ARGV[0].chomp!("/") .chomp!(".git") (think the space away, I'd get an image markup from the !s without it...) before the call to Project and passing GIT_DIR to the script.
```

I can confirm this works with a git repository, just change `REPOS="$1"` to `REPOS=$(pwd)` (`$GIT_DIR` will give you a relative path, which in this case resolves to `.` and is worthless...) and put in not in the post-commit, but in the post-receive hook.

#2 - 2009-10-05 15:44 - Micha Kersloot

If you then commit the following change to reposman.rb all is automatically fine:

```
--- reposman.rb.orig 2009-10-05 15:35:28.000000000 +0200
+++ reposman.rb 2009-10-05 15:32:03.000000000 +0200
@@ -105,6 +105,7 @@
 module Subversion
   def self.create(path)
     system_or_raise "svnadmin create #{path}"
+   system_or_raise "ln -s /usr/local/bin/post-commit #{path}/hooks/post-commit"
   end
 end
```

#3 - 2010-07-14 10:04 - Mikhail Tcishewsky

Guys, does anyone plans to take this feature in work?

I have lot of repositories and when someone push few changes Redmine start fetching process for each repository. This process take lot of time and resources and this is sad.

#4 - 2010-07-14 11:23 - Felix Schäfer

Mikhail Tcishewsky wrote:

```
| Guys, does anyone plans to take this feature in work?
| I have lot of repositories and when someone push few changes Redmine start fetching process for each repository. This process take lot of time
| and resources and this is sad.
```

This already works, but I don't think it will make it into core because there now are other means to achieve this, see [\[\[HowTo_setup_automatic_refresh_of_repositories_in_Redmine_on_commit\]\]](#).