

## ChiliProject - Feature # 268: Show branches changset belongs to on issue page

<b>Status:</b>	Open	<b>Priority:</b>	Normal
<b>Author:</b>	Ivan Evtuhovich	<b>Category:</b>	
<b>Created:</b>	2011-03-09	<b>Assignee:</b>	
<b>Updated:</b>	2011-10-13	<b>Due date:</b>	
<b>Remote issue URL:</b>			
<b>Affected version:</b>			
<b>Description:</b> This patch (see pull request from evtuhovich on github) will help me a lot to understand, what commits belongs to what branches.  Now it works only for git, but it is quite easy to make it work for others DCVS (bazaar, hg and others)			

### History

**2011-03-09 09:22 am - Ivan Evtuhovich**

<https://github.com/chiliproject/chiliproject/pull/18>

**2011-03-09 10:50 am - Toshi MARUYAMA**

Git branch is not stable. Git branch is pointer to specific revision. So, Git branch cannot be stored in database.

Mercurial \*named branch\* is stable.

**2011-03-09 10:51 am - Toshi MARUYAMA**

Please see [http://mercurial.selenic.com/wiki/GitConcepts#Branch\\_model](http://mercurial.selenic.com/wiki/GitConcepts#Branch_model) .

**2011-03-09 12:20 pm - Ivan Evtuhovich**

This is right, but when i see some issue, i want to know which branches changesets belongs to.

And if user do not make force update of branches (and this is strictly not recommended to git users), changeset do not change the branches, it belongs to.

This is very useful, when you have branches for every version and master branch (v38, v39, master, for example) and cherry-pick a lot. And sometimes this is difficult to determine, does some commit exists in some branch. This patch solve this problem.

**2011-03-09 12:57 pm - Ivan Evtuhovich**

But there can be other problems - when you add new branch to git, you should regenerate all entries in chili-database :-)

I'll think about other implementation

**2011-03-10 08:59 am - Ivan Evtuhovich**

I've just added repository/git.refresh\_branches, it refresh branches in database.

I try to determine branches of commit "on the fly", but only git log takes 0.25s for one branch-commit pair (for our not very big repo). For our repository with 5 branches and 5 commits in issue, issue page load time is too big.

But with refresh\_branches, which periodically updates branches, this is a quite good solution.

If you think, this is not good enough, i'll try to make this feature as chili-plugin, because it is very useful for our team. I think, for other teams this can be helpful too.

**2011-03-10 10:37 am - Toshi MARUYAMA**

Chili master is behind my redmine trunk commits.

Your patch conflicts with redmine trunk.

**2011-03-10 10:41 am - Toshi MARUYAMA**

Redmine git adapter has serious problems.

\* <http://www.redmine.org/issues/5357>

Git: SCM revisions ordered by date/time (should be reverse commit order)

\* <http://www.redmine.org/issues/7146>

Git adapter lost commits before 7 days

\* <http://www.redmine.org/issues/7047>

Git adapter very slow when a commit modifies a lot of files

I think Redmine git adapter should be refactored such as Mercurial overhaul

<http://www.redmine.org/issues/4455>

**2011-03-10 10:44 am - Toshi MARUYAMA**

If you have a plan to refactor git adapter, please refer <http://www.redmine.org/issues/4773#note-13>.

**2011-03-10 09:16 pm - Ivan Evtuhovich**

> Chili master is behind my redmine trunk commits.

> Your patch conflicts with redmine trunk.

I know, i have another patch for redmine, that used in our team's redmine. But i think Chili is the "separate" project.

Thanx you for information about problems with git, but ordering issue and 7 days issue couldn't be resolved without DB changes, and i do not know, is it possible to change DB just for git?

Also i do not know HG, so mercurial overhaul patch is not helpful for me.

I think about using grit library (ruby git wrapper by github), but is it exists any test or criteria that some implementation of redmine git adapter is good enough?

**2011-03-11 08:24 am - Ivan Evtuhovich**

I've just sent new pull request, with git-only implementation. It is easy and quite fast.

There exists git branch --contains <commit-id> command, which do what i need. And it takes less then 0.1s for each commit

**2011-03-14 02:27 pm - Ivan Evtuhovich**

<https://github.com/chiliproject/chiliproject/pull/22> - what should i do you accept this request?

**2011-03-14 09:11 pm - Eric Davis**

Ivan Evtuhovich wrote:

> <https://github.com/chiliproject/chiliproject/pull/22> - what should i do you accept this request?

I'd like to see some tests for it. From what I see, it's also missing a migration and the view. If you meant for pull request 18 and 22 to be used together, can you merge them in your branch and send a pull request with the full code?

**2011-03-15 08:00 am - Ivan Evtuhovich**

Ok, i'll write a test for it.

And only 22 request is actual. I get branches from git "on the fly", so there no migrations.

**2011-03-16 10:01 am - Ivan Evtuhovich**

Eric Davis wrote:

> Ivan Evtuhovich wrote:

> > <https://github.com/chiliproject/chiliproject/pull/22> - what should i do you accept this request?

>

> I'd like to see some tests for it. From what I see, it's also missing a migration and the view. If you meant for pull request 18 and 22 to be used together, can you merge them in your branch and send a pull request with the full code?

I add another commit with tests

**2011-10-13 04:20 pm - Colin Mollenhour**

Just an FYI, before I knew this ticket existed I created the exact same feature for Redmine 1.2. See "Redmine #5386":<http://www.redmine.org/issues/5386> (patch included with view update and migration)

**Files**

---

chili.png	39.6 kB	2011-03-09	Ivan Evtuhovich
-----------	---------	------------	-----------------