Redmine - Patch #8294

Include repos in scheduled fetch

2011-05-05 03:36 - Christina Louise Warne

Status:	New		**Start date:**	2011-05-05
Priority:	Normal		**Due date:**	
Assignee:	Toshi MARUYAMA		**% Done:**	0%
Category:	SCM		**Estimated time:**	0.00 hour
Target version:				

Description

I had a requirement to link projects on my Redmine installation to a local Subversion repository and a remote one. The local one sits on the same machine and is scanned every 5 minutes by way of a cron job. Scanning the remote repository every 5 minutes is not really an option, so scanning the repository only when it's accessed is fine.

To allow for this configuration, this patch adds a flag to the project repository settings that allow you to specify whether or not it should be included in a scheduled fetch. To differentiate between the existing fetch_changesets features, two new methods have been added to the repository model. The modified methods are:-

- do_fetch_changesets - Performs the actual fetch. This is essentially a slightly modified version of the existing fetch_changesets method.
- fetch_changesets - Calls do_fetch_changesets and functions exactly as it does now, fetching the repositories for all projects that have them
- scheduled_fetch_changesets - Calls do_fetch_changesets, but only scans those repositories which have the 'include in scheduled fetch' flag set

To accompany this change, their is a minor db migration which adds the column to the repositories table and a task for rake.

The scheduled fetch can be invoked using this example:-

```
RAILS_ENV=production rake -f <PATHTOREDMINEROOT>/Rakefile redmine:scheduled_fetch_changesets
```

This change also allows you to have automatic scans running on two different schedules. Calling fetch_changesets will scan all repositories on one schedule and then scheduled_fetch_changesets can scan the flagged ones on another.

All repository unit tests seemed to pass, as did the majority of the other tests (I have only tested with Subversion). The patch is built against revision r5642. The feature itself has been tested on a Redmine 1.1.3 installation and it all appears to work correctly.

I have a bit of a query regarding one aspect of this as I'm sure a more experienced Ruby/Rails/Redmine developer will be able to come up with an answer. I had to explicitly read the value for the include_in_scheduled_fetch checkbox using this code:-

```
@repository.include_in_scheduled_fetch = !params[:include_in_scheduled_fetch].nil?
```

I was expecting this to be handled by Rails like the other parameters from the form, but that appeared not to be the case. As a consequence, I'm just a bit concerned I may have missed something.

History

#1 - 2011-05-05 04:51 - Toshi MARUYAMA

- Assignee set to Toshi MARUYAMA

#2 - 2011-05-05 09:44 - Christina Louise Warne

Had a thought about this overnight. A more complete way of achieving this might be to have the concept of a repository group. Each repos is assigned to a group and then the scheduled_fetch_changesets gets called with a group identifier. That would in my opinion round out this change. Unfortunately, that's way beyond my capabilities at the moment.

Christina

Files

20110504-IncludeReposInScheduledFetch.patch	6.2 KB	2011-05-05	Christina Louise Warne
test.log	8.17 KB	2011-05-05	Christina Louise Warne