

## Redmine - Feature #779

### Multiple SCM per project

2008-03-04 01:24 - Jonathan Dray

<b>Status:</b> Closed	<b>Start date:</b> 2008-03-04
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b> Jean-Philippe Lang	<b>% Done:</b> 80%
<b>Category:</b> SCM	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> 1.4.0	
<b>Resolution:</b>	
<b>Description</b>	
It would be nice to have multiple scm per project.	
A few strategies are available :	
- put all the code in the same repository.	
- manage each program in a seperate repository.	
For a project, users should have the choice how to organise and manage versionning.	
<b>Related issues:</b>	
Related to Redmine - Feature # 4052: Cross-project redmine links with alterna...	<b>New</b> 2009-10-19
Related to Redmine - Feature # 3687: Possibility to link all projects to a si...	<b>New</b> 2009-07-29
Related to Redmine - Feature # 7409: Cross project Redmine links	<b>Closed</b> 2011-01-22
Related to Redmine - Feature # 8826: support for multiply repositories repres...	<b>Closed</b> 2011-07-15
Related to Redmine - Defect # 3087: Revision referring to issues across all p...	<b>Closed</b> 2009-03-31
Related to Redmine - Feature # 3346: Support for cross-project revision links	<b>Closed</b> 2009-05-12
Related to Redmine - Feature # 9703: Define repositories independently from p...	<b>New</b>
Related to Redmine - Defect # 10026: Git: Mercurial: Branch dropdown broken o...	<b>Resolved</b>
Duplicated by Redmine - Feature # 1172: Multiple repository modules	<b>Closed</b> 2008-05-05
Duplicated by Redmine - Feature # 3169: Multiple repositories for projects	<b>Closed</b> 2009-04-14

#### Associated revisions

##### Revision 8648 - 2012-01-15 15:23 - Jean-Philippe Lang

Resourcified repositories for CRUD operations to prepare for multiple SCM per project (#779).

##### Revision 8650 - 2012-01-15 19:19 - Jean-Philippe Lang

Adds support for multiple repositories per project (#779).

##### Revision 8651 - 2012-01-15 19:30 - Jean-Philippe Lang

Adds repository\_id param for activity and search results (#779).

##### Revision 8653 - 2012-01-15 23:23 - Jean-Philippe Lang

Don't use a disabled SCM as a default repository SCM (#779).

**Revision 8654 - 2012-01-15 23:25 - Jean-Philippe Lang**

Display identifier for the default repository too (#779).

**Revision 8656 - 2012-01-16 00:09 - Jean-Philippe Lang**

Display repository identifier in breadcrumbs (#779).

**Revision 8659 - 2012-01-16 18:55 - Jean-Philippe Lang**

Changes the style of selected repository (#779).

**Revision 8661 - 2012-01-16 19:03 - Jean-Philippe Lang**

Search for changeset in the current repository (#779).

**Revision 8668 - 2012-01-17 19:40 - Jean-Philippe Lang**

Moved the "Main repository" checkbox before the identifier (#779).

**Revision 8669 - 2012-01-17 19:43 - Jean-Philippe Lang**

Sort repositories in settings (#779).

**Revision 8671 - 2012-01-20 02:39 - Toshi MARUYAMA**

generate i18n "field\_repository\_is\_default" key (#779)

**Revision 8694 - 2012-01-22 15:23 - Jean-Philippe Lang**

Adds support for multiple repositories to redmine links (#779).

**Revision 8881 - 2012-02-15 19:24 - Etienne Massip**

Prevents SCM browser to overlay over the new repository sidebar (#779).

**Revision 9255 - 2012-03-24 14:03 - Jean-Philippe Lang**

Adds the repository identifier to Changeset#text\_tag (#779).

**Revision 9256 - 2012-03-24 14:21 - Jean-Philippe Lang**

Makes Redmine.pm work with multiple repositories per project (#779).

## Revision 9257 - 2012-03-24 14:51 - Jean-Philippe Lang

Adds the repository identifier in the activity and search results (#779).

### History

---

#### #1 - 2008-03-05 15:52 - David Petersen

We would also like to be able to have multiple repositories per project. A "Project" doesn't always mean that your using one set of code.

#### #2 - 2008-03-12 06:15 - Aruo Miura

I add another example.

We use two repositories for one project.

One is for sources, another is for documents.

And documents repository is common for each projects.

So I hope Redmine can manage SCM independently from "Project".

I think setting relations between "Project" and "SCM" is better way.

#### #3 - 2008-03-17 16:20 - David Petersen

We also have the need to have multiple projects use the same SCM. Redmine can currently do this but it imports the commits multiple times each project. This is ok but the ability to "share" SCM across project would be nice.

#### #4 - 2008-03-18 07:24 - Richard Nichols

Just echoing the requirements above. Ideal would be to move repositories to be a top level object which could be linked to one or more projects independantly.

#### #5 - 2008-03-26 08:09 - Aruo Miura

- *File IssueRepositoriesGlobal.patch added*

I make a patch for ad-hoc solution. (for r1291)

#Maybe issue got illegal data on regular Redmine. (Currently maybe no effect.)

Maybe there are a lot of faults in this.

However, it show what want to do.

This can do only:

Another project's SCM changeset comment effect another project's issue.

"refs #\*\*\*", "fixes #\*\*\*" and like them.

usage:

- make SCM crawler projects. No issue management.
- make Management project.
- Commit with comment, and SCM project's repository updated, then you can get associated revisions on the issue.

#scheduled fetch\_changeset is better way.

**#6 - 2008-03-31 17:27 - Clyde Goffe**

Hello,

Any updates on this request? I'm also interested in hearing whether this will be incorporated into a future release/plugin.

Thanks,

Clyde

**#7 - 2008-03-31 17:31 - Thomas Lecavelier**

This kind of feature leads to very deep changes in redmine architecture and ergonomiy. Worst of all: data migration would be a real pain. The common usage of multi-SCM can't be replaced by sub-projects?

**#8 - 2008-04-02 04:50 - Aruo Miura**

Can sub-project SCM's comments effect parent's issue?

Can all SCM adapters handle same repositories in another project?(CVS doubt).

If above 2 issues become clear, I can accept.

(I don't know my usage is common or not :))

**#9 - 2008-06-09 05:25 - James Turnbull**

Aruo Miura wrote:

*Can sub-project SCM's comments effect parent's issue?  
Can all SCM adapters handle same repositories in another project?(CVS doubt).  
If above 2 issues become clear, I can accept.  
(I don't know my usage is common or not :))*

For me I don't see the need for multiple SCMs. But I would like support for branching in all of the SCM adapters. For me particularly the Git adapter. For example, ticket #1387 and this forum post - <http://www.redmine.org/boards/1/topics/show/999>. YMMV.

**#10 - 2008-06-14 17:42 - Jean-Philippe Lang**

- Subject changed from Multiple SCP per project to Multiple SCM per project

**#11 - 2008-06-18 17:16 - Jonathan Dray**

I understand Thomas Lecavelier when he says that it will lead to deep changes in redmine architecture.

Is it possible to open a new development branch in Redmine for this kind of feature ?

I'm afraid I don't have the necessary skills to develop such a feature and to dive into redmine architecture =(

Sub-projects are not a replacement for separate repository management.

Think of a project that has multiple (let's say more than 3) svn repositories with some of them shared with other projects.

Maintaining a hierarchical project organisation will quickly become a nightmare.

Also think about events and comments propagations between parents and children projects that will become a hassle to understand -> see Aruo Mirua's questions

Last point : Thomas, you say that data migration would be a real pain.

I don't understand why if you manage your repository separately with their backup strategy.

Could you please be more precise on that point?

Thank you.

#### #12 - 2008-06-18 18:12 - Eric Davis

Jonathan Dray wrote:

*I understand Thomas Lecavalier when he says that it will lead to deep changes in redmine architecture.  
Is it possible to open a new development branch in Redmine for this kind of feature ?  
I'm afraid I don't have the necessary skills to develop such a feature and to dive into redmine architecture =(*

A new development branch would be possible but it will need to be done by someone with a lot of time. The main branch progresses rapidly, so the Multiple SCM branch will be very busy keeping up on the mainline changes in addition to implementing their feature. Git could migrate some of the merging issues, since Subversion doesn't track branches.

*Last point : Thomas, you say that data migration would be a real pain.  
I don't understand why if you manage your repository separately with their backup strategy.*

In order to have multiple SCMs, several parts of the underlying database schema will need to be changed and data will need to be shuffled around. I assume you have a great backup strategy but I've seen many, **many** people just update software without a backup. If there is even the slightest error, their installation is broken and they blame the project.

I see this being a useful feature, especially in the use case you gave above. I don't have the time myself to work on it but if you create a branch and start on something, I'm sure others will help out. I see at least 5 people interested in this issue alone. JPL might be able to give you svn access to create a branch or you can fork my [GitHub repository](#).

Eric

#### #13 - 2008-06-26 11:05 - minoru fukuda

- File `cvs_alias.rb` added

I made an executable script for multiple CVS modules per project.

It wraps cvs command, and modifies the output from cvs command.

Usage:

- save the attached script `cvs_alias.rb` onto your server.
- register a virtual module name in Repository of the Project in Redmine.
- edit `cvs_alias.rb`, modify the constant `ALIASES` that associates virtual module name and real module names, and modify `CVSROOT` as your environment.
- edit "`lib/redmine/scm/adapters/cvs_adapter.rb`", then modify the constant `CVS_BIN` to `/path/to/cvs_alias.rb`.

**#14 - 2009-01-09 12:17 - Matthew W**

Another vote from me for this functionality. It would be very useful.

(Although #1387, support for branches in git repositories, might make up a bit for the lack of multiple repository support)

**#15 - 2009-02-03 17:06 - Will Silva**

Matthew W wrote:

*Another vote from me for this functionality. It would be very useful.*

*(Although #1387, support for branches in git repositories, might make up a bit for the lack of multiple repository support)*

More one vote here.

It's the only issue that is blocking my company to use redmine... =/

**#16 - 2009-02-13 16:40 - Ralf Petersilka**

Willen Silva wrote:

*Matthew W wrote:*

*Another vote from me for this functionality. It would be very useful.*

*(Although #1387, support for branches in git repositories, might make up a bit for the lack of multiple repository support)*

*More one vote here.*

*It's the only issue that is blocking my company to use redmine... =/*

Hi,

I think this feature would be a great step forward. We are using different (subversion) repositories to separate code from other project documentation.

Having several repositories would help to get a complete picture.

Thanks Ralf

**#17 - 2009-03-20 15:48 - Daniel Svensson**

*More one vote here.*

*It's the only issue that is blocking my company to use redmine... =/*

*I think this feature would be a great step forward.*

+1

Definatly something we miss in our redmine setup here at work.

**#18 - 2009-03-21 15:05 - Nicklas Holm**

+1

**#19 - 2009-03-23 19:50 - Christian Zagrodnick**

+1

**#20 - 2009-04-20 15:43 - Nicolas Maeder**

+1

**#21 - 2009-04-21 15:51 - Oskar Nordquist**

+1

**#22 - 2009-04-22 16:20 - Ollivier Robert**

+1

I have the case where the primary repo is in CVS on SF.net and my fork is on mercurial on another server. Being able to see both in redmine would be lovely.

**#23 - 2009-05-19 20:01 - Erick Pérez Castellanos**

I also think this is useful. I actually need for my projects, cause, all they use two or three repositories, even from different SCM.

And i was reading this, i think that global repos, cold wait for a more develop design, but (let's say this, DON'T KNOW ALMOST ANYTHING ABOUT REDMINE INTERNALS) i think that multiple repos for projects can't be extremely difficult or so for redmine's developer.

By the way, GREAT JOB on Redmine.

**#24 - 2009-05-19 23:29 - Daniel Moore**

+1, I think...

I *think* my problem fits into the scope of this request... we maintain several products in a single repo, but products share code & are branched too:

/trunk/productA

  /productB

  /sharedA

  /sharedB

  /sharedC

/releases/productA-1-0/productA

  /sharedA

/sharedB

/sharedC

/productB-2-0/productB

/sharedC

All products live together in /trunk & each uses a different subset of the shared pieces.

Products are branched into /releases along with the code they share.

There's no single place to point the base of the project repo at that makes sense - the best compromise is pointing at the root of the repo so that every project sees every change. This would flood each project with unrelated commits.

For our purposes, some kind of filtering system might suffice?

**#25 - 2009-05-20 00:13 - Adam Piotr Żochowski**

I think Bob Roberts has provided a patch in issue : #3087

This allows to run multiple SCMs against one issue location

- root project has issues module enabled, repository module disabled

- subprojects have issues module disabled, but repository module enabled

Subprojects receive repository updates, and they can refer/close parent project issues.

Kind regards

**#26 - 2009-06-24 14:39 - Seth Dziengeleski**

+1

Using subprojects with repo but no issues seems a bit messy, but might be workable... Is there any work being done on this feature at present? Or plans to include #3087 in a future release?

**#27 - 2009-06-30 01:29 - Robert Prince**

+1

**#28 - 2009-07-28 05:28 - Jiongliang Zhang**

+1

**#29 - 2009-08-10 15:27 - Yelve Y.**

+1



**#30 - 2009-09-09 00:03 - Roman Musin**

+1

**#31 - 2009-12-28 13:33 - pasquale [:dedalus]**

David Petersen wrote:

| *We would also like to be able to have multiple repositories per project. A "Project" doesn't always mean that you're using one set of code.*

+1.

I have configured my subprojects as "categories" (example, "core", "selector" etc). Then I have multiple repositories for "one" project... this feature is a blocker for me.

Any change to have in next 0.9 release?

**#32 - 2009-12-28 13:43 - Jean-Philippe Lang**

dedalus - wrote:

| *Any change to have in next 0.9 release?*

0.9 features are frozen.

**#33 - 2009-12-28 14:49 - pasquale [:dedalus]**

Jean-Philippe Lang wrote:

| *0.9 features are frozen.*

Then I hope for next 1.0 release... meanwhile I continue to use readmine :-D

**#34 - 2010-03-02 14:27 - Daniel Beland**

+1

We use multiple SVN repositories (URLs) per project.

ProjectA:

<http://host/common/trunk>

<http://host/projectA/trunk>

ProjectB:

<http://host/common/trunk>

<http://host/projectB/trunk>

**#35 - 2010-03-11 22:05 - Pieter Smith**

Can this not be adequately solved with sub-projects?

**#36 - 2010-03-17 01:30 - Pascal Schoenhardt**

+1 for version 1.0

We have many projects, but they all use the same CVS repository and its HUGE with thousands of branches and commits spanning many years. Importing the whole thing for every project is a big problem.

**#37 - 2010-05-04 15:27 - armin walland**

+1

**#38 - 2010-06-03 15:21 - Tim Henigan**

+1 Adding this feature would eliminate another roadblock in my corporate environment.

**#39 - 2010-06-24 12:48 - fnord 999**

+1 for version 1.0

We have one repository for each development project, since we otherwise would have lots of overlaps. Redmine projects, however, span multiple such development projects... This would really be a cool feature for 1.0.

**#40 - 2010-06-25 15:27 - Uli Köhler**

+1

**#41 - 2010-07-01 17:19 - Thomas L. Kjeldsen**

+1

**#42 - 2010-07-29 04:21 - yusuke kokubo**

+1

Trac 0.12 has this feature...

<http://trac.edgewall.org/wiki/TracRepositoryAdmin>

**#43 - 2010-10-08 16:13 - Richard Hurt**

+1

We are also looking to track multiple repositories for a project. I'm not saying that it will keep us from using Redmine, but it sure would be a great feature point compared to other products.

**#44 - 2010-10-21 21:15 - Fardeen GHULAM**

+1

We are very interested also in this feature. It would avoid to have only one huge repository containing everything...

**#45 - 2010-10-28 04:49 - yusuke kokubo**

- File 0001-.patch added

This is a very very adhoc patch to able to have Mutliple SCM per project.  
(still incomplete)

Please try it and review it.

**#46 - 2010-10-28 05:58 - Eric Thomas**

yusuke kokubo wrote:

*This is a very very adhoc patch to able to have Mutliple SCM per project.  
(still incomplete)*

*Please try it and review it.*

Tests...?

**#47 - 2010-10-28 07:06 - yusuke kokubo**

Eric Thomas wrote:

*Tests...?*

no yet...

**#48 - 2010-11-29 01:42 - Billy T**

+1

It also fix our problem.

**#49 - 2010-11-30 11:46 - Giovanni Vizzotto**

+1

We are very interested also in this feature.

We use REDMINE for electronics development, so we have multiple repositories, one for each development area:

```
/ProjectA
  /DocumentsA
  /Hardware
    /HardwareDocumentation
    /Board1 (repo1)
    /Board2 (repo2)
    ...
  /SoftwareA
    /SWBoard1 (repo3)
    /SWBoard2 (repo4)
    ...
  /FpgaA
    /Fpga1 (repo5) - Board Especific
    /Library1 (repo6)..
    /Library2 (repo7)..
    ....

/MechanicalA
/SystemIntegrationAndTests
```

Each product can have more than one PCB, each PCB usually have a processor and/or FPGA.

We do a great reuse of the code and boards between diferent projects. So we need to mantain all the pieces "glued" together.

Currently, we use REDMINE sub-versions, one for each development area (Documents,Hardware, Software, FPGA, Mechanical, System).

Is not practical to open one sub-project for each repository.

So we try to use the SVN:Externals property to link the repositories, but doesn't work ( <http://www.redmine.org/boards/1/topics/19639> ).

But the solution of multiple repositories per project is more generic.

#### **#50 - 2010-12-07 20:43 - Luis Garcia**

*- File 0055-multiple-SCM-fix-navigation.patch added*

Hi. I am also very interested in this feature.

I am responsible for LliureX (an Ubuntu derivative distribution for education). Currently we are thinking about moving our svn+trac to bazaar, so support for multiple bazaar branches per project (like this patch or #2799) are features we absolutely need.

Your patch works great for us! I had noticed just a small problem when clicking 'root' and directory links in the repository title links. The links revert to 'default' repository (index 0) due to absence of 'rid' parameter.

I think that the (very simple) attached patch fixes it, and can be added to your code.

Sorry about the code (and for my english too!), but I have absolutely no familiarity with ruby on rails, so feel free to modify it as you like.

You can count with me for testing.

yusuke kokubo wrote:

*This is a very very adhoc patch to able to have Mutliple SCM per project.  
(still incomplete)*

*Please try it and review it.*

**#51 - 2010-12-08 15:42 - Damien Couderc**

+1

**#52 - 2010-12-15 14:39 - Takumi TSUKINAGA**

+1

Nice patch.thanks.

**#53 - 2010-12-18 13:21 - Ivan Cenov**

Hi,

This feature would be very useful. Our projects use multiple svn repositories:

```
Project
sw      -- software
el      -- electronics, electrotechnics
mech    -- mechanical costruction
doc     -- text documents (tech specifications, test result protocols, bills of materials ...)
ud      -- User documentations (manuals)
```

By now, we create 'Project' and several subprojects with names 'Project-sw', 'Project-el' and so on. This way works but the number of the projects grows very fast and the system becomes not so easy to handle with.

On the other hand, multiple repositories for one project would make Redmine very complicated. It would be a big development effort... and I'm afraid that Redmine may become too heavy to develop and maintain.

Ivan

**#54 - 2011-01-06 00:15 - Robert Gruendler**

is this patch working for for anyone on redmine v 1.1.0 ?

i'm getting these errors with git:

```
git apply --check 0001-.patch
error: patch failed: app/controllers/repositories_controller.rb:199
error: app/controllers/repositories_controller.rb: patch does not apply
error: patch failed: app/helpers/application_helper.rb:105
error: app/helpers/application_helper.rb: patch does not apply
error: patch failed: app/helpers/repositories_helper.rb:87
```

error: app/helpers/repositories\_helper.rb: patch does not apply  
error: patch failed: app/models/changeset.rb:161  
error: app/models/changeset.rb: patch does not apply  
error: patch failed: app/views/repositories/\_dir\_list\_content.rhtml:12  
error: app/views/repositories/\_dir\_list\_content.rhtml: patch does not apply

**#55 - 2011-01-06 09:08 - yusuke kokubo**

- File 0001-Multiple-SCM-per-Project.patch added

updated the patch for trunk ( r4642 )

1. still adhoc...

| Luis Garcia

Thanks your try and review.  
this new patch including your code.

**#56 - 2011-01-06 09:11 - yusuke kokubo**

| updated the patch for trunk ( r4642 )

sorry.  
latest update of trunk is r4636

**#57 - 2011-01-06 12:32 - Robert Gruendler**

- File multi\_repo.patch added

Sorry, i did not notice your replies and in the meantime i've fixed the first patch and merged the second one into it. Also, i've added some features/fixes:

- Added a name field to the repository model - repos can be named now so they don't show the full path to the repo
- Fixed sys\_controller > fetch\_changesets so people who fetch the changesets this way can use the patch too

I've created a [multi\\_repo branch](#) on my github account, where i will maintain those changes as long as they are not merged into trunk.

I'd like to note that i'm no ruby/rails developer either, and i've only tested my changes on git repositories, but they should work on all other supported scms too.

Maybe we can work on this patch together on github/sourceforge/etc, so we don't have to manually merge fixes/additions of each other manually.

**#58 - 2011-01-06 16:26 - Robert Gruendler**

i've updated the generalized unit tests in my [github fork](#), but i couldn't fix the repository-specific tests, as i think they need certain local repositories to test against. Anyone knows if these tests are documented somewhere?



thanks. I have managed to create all scm test repositories, install the required scms and run all the tests. Some of the repository specific tests have errors, but i'm not sure if they're related to the patch, because all of the tests create the repository object under test like this:

```
def setup
  @project = Project.find(1)
  assert @repository = Repository::Bazaar.create(:project => @project, :url => "file:///#{REPOSITORY_PATH}")
end
```

Also, the cvs tests fails with this error:

```
Loaded suite test/unit/repository_cvs_test
Started
.Unknown command: `rls`
```

Not sure if i can fixe these ones.

#### **#64 - 2011-01-07 11:25 - Toshi MARUYAMA**

cvs version requires version 1.12.

darcs test passes in version 1.0.9.

You can download at <http://eric.kow.free.fr/download/darcs-1.0.9-i386-linux.gz> .

#### **#65 - 2011-01-07 18:16 - Andy Bolstridge**

I'd like to add a little inout here : this is a great addition to the project. Where I work, we have a single repository but multiple paths. ie. each version of our product is branched, but this is not in a single root that's meaningful to associate with a redmine project.

What I'd love to see is the ability to set a repository for each version within a project. I can add versions to a redmine project, but they are just text. Adding a link to the subversion repo would solve all our problems and I think would not complicate the basic design of redmine at all.

There's already a link to a wiki page for each version, so why not add a link to a repo (including path).

#### **#66 - 2011-01-08 17:53 - Robert Gruendler**

Couldn't this be solved by adding the link to the repo/path in the wiki page of eaach version? Maybe i don't understand you correctly.

#### **#67 - 2011-01-08 20:49 - Andy Bolstridge**

It could be a link in a wiki page - but that's just tucking it away where no-one will see it.

Imagine the usual subversion organisation of a repository with the trunk/tags/branches directory structure. I'd like to be able to set the trunk folder path in the project main repository, but then if I had 1 redmine version mapped to a subversion tag folder, then I'd like to put a link (that's a lot more obvious and 'in-your-face' than a wiki link) that would give you the full redmine repository page when you clicked on it. Of course, this link would only be accessible per version - like the wiki page is for each version.

so:



trunk -> set in the project main repository settings tab.  
tags/ReleaseA -> set as a link in a redmine version 'Release A'.  
and so on for release B etc.

then my users can get at the correct repository path for a version that they want, without having to open the main repository page and browsing through to get the correct folder.

(ok, in my subversion repo, this is more complicated as we have tag folders that are not sub-folders to a project - ie we have trunk/tags/branches at the top level and multiple projects underneath them. But the above solution would still work for me).

**#68 - 2011-01-09 11:53 - Robert Gruendler**

@Andy: i understand your point now, makes sense.

Before i start installing up the required scms to fully test the patches, i'd like to ask the redmine developers how realistic they think it is to merge this patch into redmine trunk. As far as i am concerned, it works for us and we use it on our production redmine instance. But as i've already mentioned i'm no ruby/rails developer and i can't really say if - especially my contributions - can satisfy the redmine coding standards.

**#69 - 2011-01-09 13:17 - Damien Couderc**

As this request exists since 3 years and that a lot of people is waiting for it, it would really be a shame if this would not go into trunk soon. Even if the diff does not satisfy coding standards it should not take long for a commiter to rework it as it is functional.

Happy new year.

**#70 - 2011-01-12 14:22 - Andy Bolstridge**

FYI, this is how Trac handles multiple repos per project:  
<http://trac.edgewall.org/wiki/0.12/TracRepositoryAdmin>  
see the example for a 'project' and 'lib' repository links within the same trac project.

Still, if Robert's patch works, I'd be very happy to see it incorporated into the main Redmine codebase, I might even have a go trying it out later with our svn repos.

**#71 - 2011-01-12 14:25 - Robert Gruendler**

just for the record: the patch is from yusuke, i just added some minor stuff and am maintaining it on github.

**#72 - 2011-01-13 12:43 - Andy Bolstridge**

Well, I just tried it on my new Redmine 1.1 system, and it failed.

It failed to patch app/views/issues/\_changesets.rhtml, and that results in an 'internal error' when trying to view a project's settings.

I patched by going to the redmine directory and running 'patch -p1 < ../multi-repo.patch'. Any suggestions? Is this not ready for RM 1.1?

**#73 - 2011-01-13 13:59 - Robert Gruendler**

- File 002\_multi\_repo.patch added

i'd recommend to use the [github fork](#) until this feature is implemented, if possible. this fork implements the patch and i maintain it and merge all changes from the official github repository in there. however, i've attached an updated patch, which should work for 1.1.

#### #74 - 2011-01-13 16:02 - Andy Bolstridge

Robert Gruendler wrote:

```
| i'd recommend to use the github fork until this feature is implemented, if possible. this fork implements the patch and i maintain it and merge all  
| changes from the official github repository in there. however, i've attached an updated patch, which should work for 1.1.
```

Sorry, it still doesn't work. I tried with the patch and with the branch from your git repo (it's a right pain to download just one branch from git! go back to svn :) ) and I still get the error. FYI the patch still has the \_changesets.rhtml error in it.

So, what else do I have to do besides overwrite my files with the patched ones and restart (httpd via passenger in my case).

Here's the error from the production.log:

ActionView::TemplateError (undefined method `name' for #<Repository::Subversion:0xb64b5da0

```
| ) on line #11 of app/views/projects/settings/_repository.rhtml:
```

8:

9: <div class="box tabular">

10: <p><%= label\_tag('repository\_scm', l(:label\_scm)) ><= scm\_select\_tag(r, rid) ></p>

11: <= repository\_field\_tags(f, r) if r %>

12: </div>

13:

14: <div class="contextual">

lib/tabular\_form\_builder.rb:35:in `text\_field'

...

I am using the 1.1.0 version of RM, checked ut of SVN using the tags/1.1.0 branch (rev 4707). I assume this is the same one that built the 1.1.0 release (should be, but sometimes you never know!)

#### #75 - 2011-01-13 17:37 - Andy Bolstridge

Of course, I am an idiot for not migrating the DB too.

However, after restarting everything, the repository links on the project settings take an impossible amount of time to return (it hasn't yet, and right now I'm unconvinced it will)

I'll leave it spinning overnight, is there any way I can see what it's trying to do?

#### #76 - 2011-01-13 23:08 - Robert Gruendler

you could try to start redmine via the webrick server:

```
ruby script/server webrick -e production
```

and browse to <http://yourredmine.instance.com:3000>. You'll see the rails log entries in the console when using redmine.

#### #77 - 2011-01-14 17:31 - Andy Bolstridge

Thankyou. I think I have determined the error.

I created a new repository link in the project's settings - that works nicely.

then I browsed the project's repository page and all worked. I had 2 links at the top of the page to select which to use.

However, as I had only created a new one, the old link was the original, so it didn't have a name corresponding to it. I went to the settings, deleted that repository, and created a new one with the old path in it so I'd have 2 links at the top of the repo page with descriptive names.

This is when it hangs.

Webrick shows the following (mangled hostnames, etc):

```
Processing RepositoriesController#edit (for <ip> at 2011-01-14 16:06:06) [POST]
  Parameters: {"commit"=>"Create", "rid"=>"1", "repository"=>{"name"=>"version 1.4", "url"=>"http://<hostname>/svn/<path>",
"password"=>"[FILTERED]", "login"=>"admin"}, "action"=>"edit", "authenticity_token"=>"<blah>", "id"=>"uklv", "controller"=>"repositories",
"repository_scm"=>"Subversion"}
Completed in 223ms (View: 148, DB: 8) | 200 OK [http://<webhost>/projects/uklv/repository/edit?rid=1]
svn: Unable to find repository location for 'http://<hostname>/svn/<path>' in revision 200
svn: Unable to find repository location for '<hostname>/svn/<path>' in revision 400
```

and so on, up to 'Unable to find repository location .. in revision 3800. I don't know why it stops there instead of continuing to the end of the revision history, or even to revision 311743 where it was created from a branch.

I have 320,000 revisions in this repo. The path to the repo is correct (I cut and pasted from my tortoise properties).

I should say that webrick has completely hung at this point - stopping the browser and trying different links has no effect, so I guess somethings died internally after retrying and logging those 20 times.

I also couldn't stop webrick using ctrl+c, I had to open a new terminal and kill it (it reported 'svn: Write error: Broken pipe')

so.. is this a new bug in the patch code, or an assumption about the repository size that's failing when I try to use my big one, or some particularly inefficient code (I see it creating many svn processes one after the other), or is there some niggle when the old repo link is deleted? I have cleared the cache (rake tmp:cache:clear) but that didn't work.

Note: I tried again, this time I just deleted the original repo link. Same error.

I also reset the entire repo settings, started again - same error :(

#### #78 - 2011-01-18 13:06 - Andy Bolstridge

- Assignee set to Jean-Philippe Lang

- % Done changed from 0 to 80

running it in debug shows a lot more information.

It appears this patch works well, assuming you set the 'autofetch\_commits' setting to off, it won't trawl your repo looking for changesets, 200 at a time. I think the crashing problem was simply that the 'svn log' command would return nothing (ie fail) occasionally as redmine tried too many svn log commands too quickly. In the debugger, this results in an exception, in webrick this just hangs.

---

There are some outstanding niggles:

- The links at the top of the repositories page should be bigger and bolder, maybe highlighted as buttons rather than just links.
- The repository links could be matched with versions, as I think half the people using this need to show a single project with multiple releases, where each release is a different repository path (ie linked to a /tags directory - for example redmine.org has version 1.1.0 listed, that could have a link to /tags/1.1.0)

So anyway, after some testing, it does appear that this patch is a good 'un. How do we get it into the next release, as then some plugins can be upgrades (code review plugin doesn't work well with this, for example)

**#79 - 2011-01-18 15:39 - Toshi MARUYAMA**

- File *fix-broken-rev-link.83fae10224d087.diff* added

This is a patch to fix broken revision link.

Target git revision is <https://github.com/pulse00/redmine/commit/83fae10224d087b80dde0d82d13e68410a6eb3>.

**#80 - 2011-01-18 15:47 - Toshi MARUYAMA**

Github fork lost #5328 feature.

**#81 - 2011-01-18 15:54 - Toshi MARUYAMA**

- File *svn-shema-check.png* added

Original Redmine checks Subversion schema (ex. file://, svn://).

Github fork lost this feature.

**#82 - 2011-01-18 16:04 - Toshi MARUYAMA**

- File *deleted (svn-shema-check.png)*

**#83 - 2011-01-18 16:05 - Toshi MARUYAMA**

- File *svn-schema-check.png* added

Sorry, attachment file in note 81 has typo.

I fixed and re-attach.

**#84 - 2011-01-18 16:06 - Robert Gruendler**

current trunk from edavis has been merged right now into the github fork. patch will follow when i have more time.

**#85 - 2011-02-02 18:29 - Andy Bolstridge**

I've been looking through the patch (partly to learn ruby, so forgive any comments where I'm just plain wrong - but do tell me!) and I have a few comments.

Each repository has an id associated with it, and the repository\_controller selects which repo to use depending on a parameter that is passed in on the url. This should always set the @repository instance variable, but - some links (especially those that render the page rather than link to a new one) tend to reset the repository id.

so I'm beginning to question the validity of this approach. Should rid be set as another instance variable or symbol and set by the calling function? For example, the initial link from 'repositories' menu does not specify which one to use (so it defaults to 0) but that means the rid param is not set so items like the repo name is not shown unless you click the link to one (at the top).

Similarly, the 'view all revisions' link shows all revs for the first repo, not the desired one as the rid param can't be passed in properly.

I've started updating the code to use an instance variable that is set in the find\_repository method and use that throughout the rest of the class. I'll let you know how it goes.

**#86 - 2011-04-05 13:20 - Larry Cai**

Tim Henigan wrote:

| *+1 Adding this feature would eliminate another roadblock in my corporate environment.*

+1, any plan to coming release

**#87 - 2011-04-09 22:16 - William Baum**

Robert, would it be possible for you to merge the 1.1-stable branch into a branch of your fork, perhaps at the 1.1.1 or 1.1.2 release points? And/or perhaps a couple comments about how to best go about doing that?

I did take a shot at it, but I'm new to git, and my merge attempt made rather a mess.

**#88 - 2011-05-06 15:52 - Louis Simons**

+1, in the same bin a Tim Henigan and Larry Cai

**#89 - 2011-07-13 17:18 - Joan Ginard**

+1

**#90 - 2011-07-15 19:37 - Mike Dubman**

+1

**#91 - 2011-07-22 10:22 - Artur Wasilewski**

+1

**#92 - 2011-07-22 10:27 - Artur Wasilewski**

Which patch works fine with current stable redmine release (1.2.11) ? Does any?

**#93 - 2011-08-23 22:58 - W Lao**

When can this feature be included in the official release?

The roadmap doesn't include this information.

**#94 - 2011-09-02 09:13 - Gilles Cornu**

+1

**#95 - 2011-09-03 09:29 - Don Doffe**

Would love to have multiple SCM support. It is the only reason we are going to keep Trac in place for a while (web based access to support ppl).

**#96 - 2011-10-04 11:46 - Carsten Nielsen**

+1

**#97 - 2011-11-04 04:46 - Jun Huang**

+1

**#98 - 2011-11-05 04:20 - Guillaume Chéramy**

- Assignee deleted (*Jean-Philippe Lang*)

hello,

+1 for this features

Where to get the latest patch ? The pluse00 git repos seems not to work

Sincerely

**#99 - 2011-11-14 20:18 - Kaspars Sprogis**

+1

Really interested in this feature too. Any working patches, anyone?

Thanks

**#100 - 2011-11-23 08:32 - Anatol Bollinger**

+1

One team, one project, two scm archives.

Cheers,

Anatol

**#101 - 2011-12-20 21:23 - Jean-Philippe Lang**

- Assignee set to Jean-Philippe Lang

- Target version set to 1.4.0

- % Done changed from 80 to 0

**#102 - 2011-12-21 17:04 - Colin Mollenhour**

While this ticket previously had my vote, I now prefer the solution proposed in #9703 since it allows for even more flexibility. When you have projects and sub-projects, one problem is that commits in a repo that reference a ticket in a sibling project don't get related. The solution from #9703 would solve this by allowing you to associate each repository with each project, or more likely each repository with the parent (assuming all children inherit relations in commits from the parent).

**#103 - 2011-12-21 19:12 - Jean-Philippe Lang**

#779 is needed by many people and it doesn't mean that some kind of repository sharing across projects won't be added later. I think this is the first step in this direction. Even if we don't have independent repositories, we could have a repository shared with all the subprojects and allow issue referencing between all of them.

**#104 - 2011-12-22 08:29 - Pedro Gutierrez**

--1

I'm more in favour of the solution proposed in #9703

I think that it is more flexible and would solve two problems and not only one:

- multiple repositories per project
- multiple projects per repository.

**#105 - 2012-01-15 22:41 - Ivan Cenov**

I've just tested the new features (r8648, r8649, r8650, r8651, r8652). Great addition to Redmine.

I would like to propose these visual enhancements:

- The currently selected repository to be somehow visually marked (bold, italic, or a check mark near the name)
- The label "Main repository" to be replaced with the identifier of the repository and the fact that it is the main repository to be visually marked.

Also, when adding new repository, all the SCM are selectable in the drop-down list. It was not the case before. The SCMs that had been installed were in the drop-down list only.

**#106 - 2012-01-16 23:57 - Jean-Philippe Lang**

- % Done changed from 0 to 80

I have made a few enhancements based on your feedback, thanks. The selected repository is highlighted in the sidebar and the "Main repository" label is replaced with the repository identifier (it is always at the top of the list so I don't know if a visual mark is required). I've also fixed the SCM drop-down list.

Magic links to revisions/files of other repositories other than the main one need to be implemented. Here is the proposed syntax:

- Changeset: identifier:c6f4d0fd
- Source: identifier:source:some/file

#### #107 - 2012-01-17 18:03 - Ivan Cenov

Jean-Philippe Lang wrote:

```
| Magic links to revisions/files of other repositories other than the main one need to be implemented. Here is the proposed syntax:
```

- *Changeset: identifier:c6f4d0fd*
- *Source: identifier:source:some/file*

I have experience with Subversion only, and can propose the following:

```
Source: identifier:source:some/file[?r=nn][?p=nn]
```

examples:

```
myrepo:source:some/file.c      -- some/file.c from HEAD
myrepo:source:some/file.c?r=45  -- some/file.c from revision 45
myrepo:source:some/file.c?r=7?p=20 -- some/file.c from operative revision 7 and peg revision 20
myrepo:source:some/file.c?p=20  -- some/file.c from operative and peg revision 20
```

#### #108 - 2012-01-17 18:49 - Etienne Massip

Jean-Philippe Lang wrote:

```
| Magic links to revisions/files of other repositories other than the main one need to be implemented. Here is the proposed syntax:
```

- *Changeset: identifier:c6f4d0fd*
- *Source: identifier:source:some/file*

What would you think of: - replacing Identifier property by a Name property (allowing spaces characters and such) because there are actually 2 Identifier columns (one for repo and one for the fetch account) and make it optional (or make it mandatory for the main repo too with a default value of "Main repo")

Explanation: in the first place I didn't understand what the Identifier field was useful to, then I understood this was to display in the list, so why not a more friendly Name instead?

- moving "Main repo" checkbox before the Name/Identifier field (no real reason)
- using syntax like
  - Changeset: identifier:c6f4d0fd (same as proposed by JPL)
  - Source: source:identifier:some/filewhere identifier is the id of the repository?



## #109 - 2012-01-17 19:35 - Jean-Philippe Lang

The repository identifier is used in urls (just like the project identifier):

```
/projects/foo/repository    => main repo (identifier is not required in the url to preserve existing urls)
/projects/foo/repository/other => other repo
```

We could add a Name attribute for displaying in the sidebar.

The problem with myrepo:source:some/file.c is that it can be ambiguous with existing cross-project links like myproject:source:some/file.c if a repository has the same identifier as a project.

I'd prefer a syntax that would prevent ambiguousness like the following (just an example, the slash may not be the best option):

```
myrepo/source:some/file.c
myproject:myrepo/source:some/file.c

myrepo/r123
myproject:myrepo/r123

myrepo/commit:c6f4d0fd
myproject:myrepo/commit:c6f4d0fd
```

## #110 - 2012-01-17 19:41 - Jean-Philippe Lang

Etienne Massip wrote:

```
| - moving "Main repo" checkbox before the Name/Identifier field (no real reason)
```

Indeed, changed in r8668.

## #111 - 2012-01-17 19:53 - Etienne Massip

Jean-Philippe Lang wrote:

```
| I'd prefer a syntax that would prevent ambiguousness like the following (just an example, the slash may not be the best option):
| [...]
|
```

IMO, the ideal could have been to have the type of reference as first argument, e.g.:

```
source:myrepo:some/file.c
source:myproject:myrepo:some/file.c

myrepo:r123
myproject:myrepo:r123
```

(or weirder:  
rmyrepo:123  
rmyproject:myrepo:123)

commit:myrepo:c6f4d0fd  
commit:myproject:myrepo:c6f4d0fd

I guess it's too late.

**#112 - 2012-01-17 21:09 - Nikunj Undhad**

- Assignee changed from Jean-Philippe Lang to Felix Schäfer

**#113 - 2012-01-17 22:16 - Ivan Cenov**

When pressing 'View all revisions' link, the side bar disappears. It was not expected for me when I saw it for first time but now I am not sure if this is good or bad. Just mention it now...

**#114 - 2012-01-18 06:26 - Toshi MARUYAMA**

- Assignee deleted (Felix Schäfer)

**#115 - 2012-01-18 19:05 - Terence Mill**

Will it be possible to grant different access rights role based to each repo?

**#116 - 2012-01-18 21:54 - John Kubiawicz**

Just a curiosity question about repo identifiers. Any reason that they are not globally unique (and unique with respect to project IDs as well)? Then, (1) the identifier of the primary repo of a project could default to the project identifier, and (2) the set of all repos have unique names so that they can appear in a single, flat namespace (if desired).

I am just thinking about plugins such as redmine\_git\_hosting that manages repositories now (in <= 1.3 Redmine) in a way that can rely on the uniqueness of the project IDs to name the actual repositories. Constraining the identifiers to be unique across projects (and not the same as project-ids) would make such management simple.

Is the flexibility to have identifiers that are not-unique across projects a feature?

**#117 - 2012-01-19 01:47 - Andy Bolstridge**

One important aspect is the need to have the same repo specified in multiple projects. Think of several projects to all use the same 3rd party repo for example. each repo would be unique, both globally and with respect to a project, but a repo cannot then be tied to a single project, not unless we had non-unique repo ids (so you could specify the same repo many times, once for each project)

**#118 - 2012-01-19 02:57 - John Kubiawicz**

Ah. So, it doesn't make sense for the same id to be used in different repos with different URLs? I guess your comment is a different take on what I said: every physical repository has a unique ID, possibly repeated in different projects? That seems like a use-case that might have a different "new

repository" screen than currently offered: **either** pick an existing repo by name (in which case the URL is taken from an earlier definition) or define a new repo...

In this use-case, do you assume that the same physical repo would have different access control depending on which project it was accessed through? That seems a bit weird (and really messes up something like gitolite). That being said, however, gitolite supports per-branch permissions (for git). I guess it might make sense to have different branches of a physical repo be attached to different projects. Oy!

Also, what is the story with the primary repo (/without an identifier)? Is it considered project-specific and unique (not cross project)?

#### #119 - 2012-01-19 14:12 - Andy Bolstridge

I just wanted to make sure the most common use-case was catered for :)

I'm not sure about access-control. From my PoV, access control should be via the repository itself. I'm not sure if specifying different users even makes sense - you have a 'redmine' user and that's it. When it comes to reading the revision history, you want RM to have read access to the repo no matter what.

I guess some people will want access control for users viewing different repositories inside RM, but that is a project-based access control that has nothing to do with the repo link, you either allow a RM user to read the repo log comments that have been copied to the RM db, or you don't - RM itself still has to read the revision history or it just doesn't work.

Either way, I'm looking forward to this update, it's about time RM got some of the ancient feature requests resolved. I'm glad JPL is back on the case.

#### #120 - 2012-01-19 18:07 - Ivan Cenov

Andy Bolstridge wrote:

*One important aspect is the need to have the same repo specified in multiple projects. Think of several projects to all use the same 3rd party repo for example. each repo would be unique, both globally and with respect to a project, but a repo cannot then be tied to a single project, not unless we had non-unique repo ids (so you could specify the same repo many times, once for each project)*

There is another use case: many projects and every project has repositories 'software', 'electronics', 'mechanics', 'clients'. These repositories are different ones. Par example, project1.software and project2.software are not the same project.

#### #121 - 2012-01-22 13:33 - Jean-Philippe Lang

- Assignee set to Jean-Philippe Lang

Repository identifiers do not need to be globally unique because each of them is tied to a single project. Please understand that this feature does not cover #9703 that would allow multiple projects to use the same repositories.

#### #122 - 2012-01-22 20:06 - Etienne Massip

Jean-Philippe Lang wrote:

*Repository identifiers do not need to be globally unique because each of them is tied to a single project. Please understand that this feature does not cover #9703 that would allow multiple projects to use the same repositories.*

New dumb question: why not let administrator manage repositories instead of project manager, that would make more sense since this is some kind of host machine level communication config, don't you think so?

**#123 - 2012-01-22 22:17 - William Baum**

Jean-Philippe Lang wrote:

*Repository identifiers do not need to be globally unique because each of them is tied to a single project. Please understand that this feature does not cover #9703 that would allow multiple projects to use the same repositories.*

While I can see the benefit of common names for repositories (code, db, requirements, tests, etc.), that would **require** using both the project and repository identifiers to access the repository. If a unique name was required, the existing url schemes, etc., could be used to access the repository, changesets, etc., just with the repository ID rather than the project ID. If non-unique names are used, this will break source/export links as well as some third party plugins and integrations (hudson, jenkins come to mind).

Also, it seems like non-unique names gets us farther from shared repositories (#9703) rather than closer to it.

**#124 - 2012-02-02 00:35 - John Kubiawicz**

I would like to add another plugin that would be "broken" by non-unique IDs: redmine\_git\_hosting. (Perhaps "broken" is too strong a word, instead "complicated" is a better word.)

As it is, the uniqueness of the repository IDs is inherited from the uniqueness of the project IDs, and this plugin takes advantage of this property. The reason I first asked about uniqueness was that I am thinking of what would be involved in making this plugin work well with the new feature. One of my options was to "uniqueify" repository names by prepending the project-ID, but this has its own complications.

**#125 - 2012-02-16 10:07 - Brice Beamesnil**

If it's possible to add a name to each Repository and show this name on Activity's page (near sub-project name for example).

Thanks.

**#126 - 2012-03-24 15:02 - Jean-Philippe Lang**

*- Status changed from New to Closed*

Brice Beamesnil wrote:

*If it's possible to add a name to each Repository and show this name on Activity's page (near sub-project name for example).*

r9257 adds the repository identifier to the activity view.

**#127 - 2012-11-28 17:57 - Daniel Dehennin**

Hello,

I'm looking for a documentation to clone my sub git repository

Looking at r9256 I thought it would be `http://<base redmine url>/git/<project>.<subrepository name>` but it does not work.

Any hints?

Regards.

**#128 - 2012-11-28 18:44 - Ivan Cenov**

Daniel Dehennin wrote:

*Hello,*

*I'm looking for a documentation to clone my sub git repository*

*Looking at r9256 I thought it would be `http://<base redmine url>/git/<project>.<subrepository name>` but it does not work.*

*Any hints?*

*Regards.*

No, this is not the way and it does not work.

Create a bare repository in the file system that is visible by Redmine server. Then when configuring the repository in Redmine, set the path (the physical path in the file system) to this repository. This is different from what is done when configuring SVN repository -- SVN repos may be referred by URL. After you have done this, you should be able to see the repository in Redmine.

**#129 - 2012-11-29 09:26 - Daniel Dehennin**

Ivan Cenov wrote:

*No, this is not the way and it does not work.*

*Create a bare repository in the file system that is visible by Redmine server. Then when configuring the repository in Redmine, set the path (the physical path in the file system) to this repository. This is different from what is done when configuring SVN repository -- SVN repos may be referred by URL. After you have done this, you should be able to see the repository in Redmine.*

I'm not sure that `Apache::Authn::Redmine::access_handler` works with multiple repository, the repository should be "clonable" anonymously but authentication is asked and fail :-/

**#130 - 2015-04-13 07:45 - Olivier Mehani**

Daniel Dehennin wrote:

*I'm not sure that `Apache::Authn::Redmine::access_handler` works with multiple repository, the repository should be "clonable" anonymously but authentication is asked and fail :-/*

Yup, I just ran into this issue myself. Does this mean that it is not currently possible to authenticate to a repo which doesn't have exactly the name of

the project?

## Files

---

IssueRepositoriesGlobal.patch	2.33 KB	2008-03-26	Aruo Miura
cvs_alias.rb	1.91 KB	2008-06-26	minoru fukuda
0001-.patch	15.4 KB	2010-10-28	yusuke kokubo
0055-multiple-SCM-fix-navigation.patch	1.23 KB	2010-12-07	Luis Garcia
0001-Multiple-SCM-per-Project.patch	16.3 KB	2011-01-06	yusuke kokubo
multi_repo.patch	23.3 KB	2011-01-06	Robert Gruendler
002_multi_repo.patch	34.9 KB	2011-01-13	Robert Gruendler
fix-broken-rev-link.83fae10224d087.diff	906 Bytes	2011-01-18	Toshi MARUYAMA
svn-schema-check.png	37.3 KB	2011-01-18	Toshi MARUYAMA