

SCM Creator (+Github) - Feature #2075

Make an option to fully automate repository creation without editing the URL field

26 Jun 2012 13:59 - Humberto Anjos

|   |               |                        |             |
|---|---------------|------------------------|-------------|
| <b>Status:</b>  | Open          | <b>Start date:</b>     | 26 Jun 2012 |
| <b>Priority:</b>  | Normal        | <b>Due date:</b>       |             |
| <b>Assignee:</b>  | Andriy Lesyuk | <b>% Done:</b>         | 0%          |
| <b>Category:</b>  |               | <b>Estimated time:</b> | 0.00 hour   |
| <b>Target version:</b>  |               |                        |             |
| <b>External issue:</b>  |               |                        |             |
| <b>Description</b><br><br>Here's my situation: I need a SourceForge-like solution for internal use in my company, and Redmine seems like the best option. I need to create repositories only through Redmine, so this plugin fell like a glove. Thing is, the URL field is fully editable, so users can create repos anywhere on the server.<br><br>So my current options now are: <ul style="list-style-type: none"><li>• stick to Redmine 1.3 and SCM Creator 0.3; or</li><li>• migrate to Redmine 2, SCM Creator 0.4, use the <a href="#">Force repository creation for all projects</a> configuration and limit repos to one per project (max_repos: 1), so the user never gets the option of editing the URL.</li></ul><br>The best solution (for me) would be for the URL field to be non-editable, and for the URL to be fully determined by the project name, the path in scm.yml and the repository identifier. So, for a path /var/lib/svn, a project test and an repository main, the repo's full URL would be file:///var/lib/svn/test/main.<br><br>Of course, Redmine-based authentication should still work (it seems like a simple matter, but I don't know if there's any hidden surprises...). |               |                        |             |
| <b>Related issues:</b>  |               |                        |             |
| Related to SCM Creator (+Github) - Feature #2078: Show users a friendly URL i...  |               | Closed                 | 29 Jun 2012 |
| Follows SCM Creator (+Github) - Feature #2053: Set repository path according ...  |               | Open                   | 15 Jun 2012 |

History

#1 - 26 Jun 2012 14:53 - Andriy Lesyuk

- Status changed from New to Open

... the URL field is fully editable, so users can create repos anywhere on the server

No, they can't! If users change the path (and this path differs from the one specified in scm.yml) they will get an error.

The best solution (for me) would be for the URL field to be non-editable, and for the URL to be fully determined by the project name, the path in scm.yml and the repository identifier. So, for a path /var/lib/svn, a project test and an repository main, the repo's full URL would be file:///var/lib/svn/test/main.

Of course, Redmine-based authentication should still work (it seems like a simple matter, but I don't know if there's any hidden surprises...).

Not sure if file:///var/lib/svn/test/main will work with Redmine-based authentication... Seems it should be file:///var/lib/svn/test.main instead...

Anyway why do you need that main if there should be only one repo per project?.. I would recommend to go with [Force repository creation for all projects](#) in this case. Or I did not get something...

## #2 - 26 Jun 2012 19:41 - Humberto Anjos

My writing definitely needs some work...

No, they can't! If users change the path (and this path differs from the one specified in scm.yml) they will get an error.

I may have overstated the problem. The issue I actually encountered is that I can name my repos anything I want (inside scm.yml's path), and that name may collide with another project's.

For example, given a path `/var/lib/svn`: if I create a project `test`, I can create a repo at `/var/lib/svn/test`. But I can create another repo for `test` at `/var/lib/svn/test2`. If later on, another user creates a project named `test2`, he can't use `/var/lib/svn/test2` because that's already taken. From the second user's perspective, that seems like a bug.

Enabling fully automatic repo creation would mean that name collisions like this just can't happen. And it would be easier for the users, too; just pick a name for your new repo and you're good to go. This is how it is with Gitorious, for example.

My ultimate goal is something like Google Code/Github/Gitorious, where the user shouldn't have to bother at all with *where* on the server his repos should go. They all will be created for him, readily accessible with nice URLs, and all using Redmine for authentication. That may be a little outside this plugin's original calling, but it's so close...

Anyway why do you need that `main` if there should be only one repo per project?.. I would recommend to go with Force repository creation for all projects in this case. Or I did not get something...

The one repo per project and forced repo creation is my current workaround for the issue. Since the user never gets to muck around with the URL, there's no problem. I intend to enable multiple repos, except that letting my users edit the URL (almost) at will is asking for trouble...

Not sure if `file:///var/lib/svn/test/main` will work with Redmine-based authentication... Seems it should be `file:///var/lib/svn/test.main` instead...

Sure, why not

As long as Redmine-based auth works, the repos are created automatically and there's no name collisions, the actual path is not a problem

**Update:** fixed the quoting.

**#3 - 26 Jun 2012 19:57 - Andriy Lesyuk**

Ok, I got it.

My ultimate goal is something like Google Code/Github/Gitorious, where the user shouldn't have to bother at all with where on the server his repos should go. They all will be created for him, readily accessible with nice URLs, and all using Redmine for authentication. That may be a little outside this plugin's original calling, but it's so close...

With auto\_create=true you get this, don't you?

**#4 - 26 Jun 2012 22:34 - Humberto Anjos**

My ultimate goal is something like Google Code/Github/Gitorious, where the user shouldn't have to bother at all with where on the server his repos should go. They all will be created for him, readily accessible with nice URLs, and all using Redmine for authentication. That may be a little outside this plugin's original calling, but it's so close...

With auto\_create=true you get this, don't you?

I did try it, but I can't recall the results (gotta get an external HD for my brain ), and forced creation and one repo per project worked well enough.

I'm at home, so I can't verify this right now, but this issue will pop up whenever the user gets an editable URL field.

**#5 - 26 Jun 2012 23:02 - Andriy Lesyuk**

Yes, I understand.

Feature request accepted! Will do this.

**#6 - 26 Jun 2012 23:17 - Andriy Lesyuk**

- Target version set to 0.4.1

**#7 - 28 Jun 2012 22:20 - Andriy Lesyuk**

- Assignee set to Andriy Lesyuk

- Target version deleted (0.4.1)

I guess this issue should be resolved along with [#2053](#)... Otherwise how would it be possible to add more repositories? By accepting `.<number>`?

Will perhaps name the appropriate option `deny_url_edit`...

#### #8 - 29 Jun 2012 02:29 - Humberto Anjos

Andriy Lesyuk wrote:

I guess this issue should be resolved along with [#2053](#)... Otherwise how would it be possible to add more repositories? By accepting `.<number>`?

I didn't quite get [#2053](#)... With it implemented, wouldn't renaming the identifier move the repo around in the server's filesystem? And isn't the URL immutable once created? If so, for my situation at least, providing an identifier should be mandatory and immutable once given, just like a project identifier, shouldn't it? Or am I missing something?

#### #9 - 29 Jun 2012 09:19 - Andriy Lesyuk

I didn't quite get [#2053](#)... With it implemented, wouldn't renaming the identifier move the repo around in the server's filesystem? And isn't the URL immutable once created? If so, for my situation at least, providing an identifier should be mandatory and immutable once given, just like a project identifier, shouldn't it? Or am I missing something?

Saying you have right now (when you create a repository): repository identifier set to "" (*empty*) and URL set to `file:///var/lib/svn/test-project.2`. URL is immutable (*that is a user can't modify it*). When you change repository identifier to additional the URL is automatically changed to `file:///var/lib/svn/test-project.additional` (*it's still immutable for users, but not for JavaScript*). As soon as you click "Create new repository" both – Identifier and URL – become immutable.

Without this URL for additional repos will be set to something like `file:///var/lib/svn/test-project.2...` And there will be no way to change.

Hope this makes sense? Your thoughts?

#### #10 - 29 Jun 2012 10:43 - Humberto Anjos

Andriy Lesyuk wrote:

Saying you have right now (when you create a repository): repository identifier set to "" (*empty*) and URL set to `file:///var/lib/svn/test-project.2`. URL is immutable (*that is a user can't modify it*). When you change repository identifier to additional the URL is automatically changed to `file:///var/lib/svn/test-project.additional` (*it's still immutable for users, but not for JavaScript*). As soon as you click "Create new repository" both – Identifier and URL – become immutable.

Without this URL for additional repos will be set to something like file:///var/lib/svn/test-project.2... And there will be no way to change.

Hope this makes sense? Your thoughts?

OK, I got it now. This solution works almost perfectly for me! It would be perfect along [#2078](#)

**Update:** mentioning [#2078](#) as well.

#### **#11 - 27 Sep 2012 23:59 - Lluís -**

Andriy Lesyuk wrote:

When you change repository identifier to additional the URL is automatically changed to file:///var/lib/svn/test-project.additional *(it's still immutable for users, but not for JavaScript)*. As soon as you click "Create new repository" both – Identifier and URL – become immutable.

Just as a side note, URL editability should also be enforced in the server code (not just JavaScript), or malicious users would otherwise be able to create repositories at arbitrary locations in the system by forging the appropriate requests.

#### **#12 - 04 Jul 2014 08:07 - Ludovic Schoepps**

Hi,

Any news on this ? I would really appreciate this option for the users of the University I'm working on. It would be even better if the url field doesn't appear and we only ask the user for the identifier. This way, the user is not bothered with an url including "/var/lib/svn" since he only needs an url like "http://redmine.example.com/svn/<generated\_id>"

Best regards,

Ludovic Schoepps.