Redmine - Defect #11609

Override redmine routes.rb in a plugin in redmine 2.x

2012-08-09 04:46 - Bob Bottle

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Jean-Baptiste Barth		**% Done:**	0%
Category:	Plugin API		**Estimated time:**	0.00 hour
Target version:				
Resolution:	No feedback		**Affected version:**	2.3.0

Description

Currently, it doesn't seem possible to override redmine application routes in a plugin.

For instance, I want to override the home link to go to projects. I can hack redmine/config/routes.rb to have

```
root :to => 'projects#index', :as => 'home'
```

and that works.

However, when I make a plugin and add this to routes.rb

```
RedmineApp::Application.routes.draw do
  root :to => 'projects#index', :as => 'home'
end
```

It has no effect. I can see the route is added by seeing them in rake routes however.

It seems this was possible in redmine 1.x ([#3867](#))

Related issues:

Related to Redmine - Defect #12590: Plugin routes.rb files are not getting in...	**Closed**

History

#1 - 2012-08-12 13:42 - Roger Mbiama Assogo

- File ChangeLog added

#2 - 2012-08-12 13:48 - Roger Mbiama Assogo

- File Makefile.apxs added

#3 - 2012-08-13 21:42 - Jean-Philippe Lang

- File deleted (ChangeLog)

#4 - 2012-08-13 21:43 - Jean-Philippe Lang

- File deleted (Makefile.apxs)

#5 - 2012-08-16 14:48 - Sergey Gnuskov

Same here, can't redefine root path in plugin to my controller on redmine 2.x. On 1.x it works.

#6 - 2012-09-05 13:50 - cam lafit

Hi

I confirm this problem, with redmine 2.x It's impossible to change route on plugi.
Actually, i find a solution via patch behaviour (http://www.redmine.org/projects/redmine/wiki/Plugin_Internals)

Not very simply

#7 - 2012-09-20 10:44 - Leo L.

Bob Bottle wrote:

For instance, I want to override the home link to go to projects. I can hack redmine/config/routes.rb to have
root :to => 'projects#index', :as => 'home'
and that works.

I did found that hack too, but if I hack routes.rb this way, when I try to access a project from the projects menu, I got an "500 internal server error". Don't you ?

--
Leo.

#8 - 2012-09-20 12:51 - Leo L.

Concerning my last comment, when I hack the routes.rb file the way Bob Bottle did, when I click on an evolution id/name or incident id/name, I got :

ActionController::RoutingError (No route matches {:controller=>"welcome"}):
app/views/issues/_edit.html.erb:34:in `block in *app_views_issues*_edit_html_erb___1214659690394619739_17271917100'
app/helpers/application_helper.rb:952:in `labelled_form_for'
app/views/issues/_edit.html.erb:1:in `_app_views_issues__edit_html_erb___1214659690394619739_17271917100'
app/views/issues/show.html.erb:129:in `_app_views_issues_show_html_erb__1382867738471145656_17340890220'
app/controllers/issues_controller.rb:118:in `block (2 levels) in show'
app/controllers/issues_controller.rb:115:in `show'

I don't understand why... Is there something I missed ?

--
Leo.

#9 - 2012-09-20 15:08 - Leo L.

Leo L wrote:

> Concerning my last comment, when I hack the routes.rb file the way Bob Bottle did, when I click on an evolution id/name or incident id/name, I got :
> [...]
> I don't understand why... Is there something I missed ?

Resolved.
Modifying the home page implies adding a new line below the root's home :

```
  root :to => 'projects#index', :as => 'home'
 #add this new line :
  match '/', :to => 'welcome#index', :as => 'home'
```

--
Leo.

#10 - 2013-02-27 11:27 - Ilya S

Have same problem. Redmine 2.2.3.dev and 2.1.6 stable. This hack

```
root :to => 'projects#index', :as => 'home'
 #add this new line :
  match '/', :to => 'welcome#index', :as => 'home'
```

doesn't works too. How i can fix it?

#11 - 2013-05-02 01:53 - Jean-Baptiste Barth

- Status changed from New to Closed

- Assignee set to Jean-Baptiste Barth

- Affected version (unused) changed from 2.0.3 to 2.3.0

- Resolution set to Fixed

- Affected version changed from 2.0.3 to 2.3.0

Actually there's a prepend method in ActionDispatch::Routing::RouteSet (found it thanks to this stackoverflow question).

It doesn't work in a plugin's config/routes.rb because it is loaded after the initialization process, when first routes are already defined. But if you leave it in your init.rb file, it's evaluated soon enough, and it works. Example:

```
RedmineApp::Application.routes.prepend do
```

```
  root :to => 'my#page', :as => 'home'
end
```

Maybe you'll have to also redefine welcome#index action so that url_for() calls in Redmine core work, but I don't know if this is necessary..

I close this issue as I think it answers the initial question, feel free to reopen if not.

#12 - 2013-05-16 12:17 - Ilya S

Jean-Baptiste Barth wrote:

> Actually there's a prepend method in ActionDispatch::Routing::RouteSet (found it thanks to this stackoverflow question).
>
> It doesn't work in a plugin's config/routes.rb because it is loaded after the initialization process, when first routes are already defined. But if you leave it in your init.rb file, it's evaluated soon enough, and it works. Example: [...]
>
> Maybe you'll have to also redefine welcome#index action so that url_for() calls in Redmine core work, but I don't know if this is necessary..
>
> I close this issue as I think it answers the initial question, feel free to reopen if not.

i've tried solution you suggested, but it doesn't solve problem. Root path still leads to welcome controller. 'rake routes' says that's all fine - my custom route now on top, but problem still there. Any ideas?

#13 - 2013-05-16 12:35 - Toshi MARUYAMA

- Status changed from Closed to Reopened

#14 - 2013-05-16 15:49 - Jean-Baptiste Barth

No idea, it worked for me. As you imagine I wasn't sure it would work so I put this code in a plugin and rebooted my redmine instance, then it worked. Maybe it depends on the version of Rails, I tried with Redmine 2.3.1 (Rails 3.2.13), older versions of Redmine may behave differently.

#15 - 2013-05-17 15:40 - Ilya S

Jean-Baptiste Barth wrote:

> No idea, it worked for me. As you imagine I wasn't sure it would work so I put this code in a plugin and rebooted my redmine instance, then it worked. Maybe it depends on the version of Rails, I tried with Redmine 2.3.1 (Rails 3.2.13), older versions of Redmine may behave differently.

it's really strange, i use same conf: ruby 1.9.3, redmine 2.3.1 stable, rails 3.2.13

#16 - 2013-05-17 16:58 - Jean-Baptiste Barth

- Resolution deleted (Fixed)

Ok, I'll try to reproduce and see if I did everything correctly. Unfortunately I won't be at home until next tuesday, so I won't be able to run these tests before.

#17 - 2013-06-12 14:18 - Logan Raarup

I found that this is fixed by changing the following in routes.rb:

From:

```
Dir.glob File.expand_path("plugins/*", Rails.root) do |plugin_dir|
    file = File.join(plugin_dir, "config/routes.rb")
```

To:

```
Dir.glob File.join(File.expand_path("plugins", Rails.root), "*") do |plugin_dir|
    file = File.join(plugin_dir, "config", "routes.rb")
```

#18 - 2014-08-15 23:17 - Jean-Baptiste Barth

- Status changed from Reopened to Needs feedback

Logan Mazzoleni: could you please give us informations about your OS and ruby version ? I didn't manage to reproduce it (same as #12590), but your change would be ok I think if the problem still occurs on Redmine 2.5.2 with a supported ruby version. If not I guess it's better not to change anything. Sorry for the delay since last update :/

#19 - 2015-08-23 17:07 - Jan Niggemann (redmine.org team member)

- Status changed from Needs feedback to Closed

- Resolution set to No feedback