I'm not sure when exactly it happens (it's worked sometime ago), but things becomes broken for Principal with empty plugin.

How to reproduce:

$ cat vendor/plugins/test_plugin/init.rb
require 'redmine'

require 'dispatcher'
Dispatcher.to_prepare :test_plugin do
  require_dependency 'principal'
end

Redmine::Plugin.register :test_plugin do
end
(no other files in vendor/test_plugin)

$ rake test:units
<...>
1) Failure:
test_roles_updated(GroupTest) [/test/unit/group_test.rb:57]:
<[1, 2]> expected but was
2) Failure:
test_visible_scope_for_user(IssueTest) [/test/unit/issue_test.rb:94]:
<false> is not true.
478 tests, 2036 assertions, 2 failures, 0 errors

Related issues:
Related to Redmine - Defect # 5434: Redmine 0.9.3 - HTTP500 when trying to fi...  Closed  2010-05-04

History
#1 - 2009-12-07 12:47 - Alexey Palazhchenko

Any ideas? Please help, it's showstopper for me.
Jean-Philippe, any ideas?

Ok, second failure was bisect down to r3039. No idea about first yet...

Aleksey:

That's really odd. I have a few plugins that work with Users that are still working. Is this failing in just a test plugin or something you are working on?

Eric, I'm working on plugin for some Redmine models. When I found this 2 broken tests, I started to remove code from my plugin – and it's ended with this bare skeleton.

Had the same kind of problem recently with memberships. It seems Rails has some troubles reloading things correctly when some classes has been reloaded through engines. It may have something to do with STI (doesn't affect other classes from what I've seen).

For instance, with your test plugin, in test/unit/group_test.rb, line 57:

```ruby
assert_equal [1, 2], user.reload.roles_for_project(project).collect(&:id).sort
#=> KO
assert_equal [1, 2], User.find(9).roles_for_project(project).collect(&:id).sort
#=> OK
```

I tried to mark some classes as unloadable in the core, doesn't change anything...

I continue to think it's a rails bug, but it's solved by forcing memberships reload in User model:

```ruby
def reload(*args)
    @name = nil
    self.memberships.reload
    super(*args)
```

I continue to think it's a rails bug, but it's solved by forcing memberships reload in User model:
Any thought about integrating that into the core? (note it also works when monkey patching User class this way in the plugin...)

#8 - 2010-10-23 14:03 - Alexey Palazhchenko

Any thought about integrating that into the core?

I'm for it, with a big FIXME comment.

Is it a know Rails bug?

#9 - 2010-10-23 18:48 - Jean-Baptiste Barth

Alexey Palazhchenko wrote:

Is it a know Rails bug?

Didn't find it on lighthouse. It seems inherited scopes (in our case User#like) are also lost when reloading...

#10 - 2012-06-10 22:42 - Jean-Baptiste Barth

- Assignee deleted (Jean-Baptiste Barth)

#11 - 2013-01-13 21:04 - Jean-Philippe Lang

- Status changed from 7 to New

Assigned issue with no assignee back to New status.