

Redmine - Defect #39803

Plugin modules no longer working

2023-12-10 14:58 - Guus Teley

Status:	Needs feedback	Start date:	
Priority:	Normal	Due date:	
Assignee:	Guus Teley	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:			
Description <p>As of Redmine 5.1.x (Rails 7.1.2) plugins that rely on modules in the plugin/lib directory are no longer working. Debugging lib/redmine/plugin_loader.rb shows that in add_autoload_paths in engine_cfg.all_eager_load_paths.each do dir the following directories are added:</p> <ul style="list-style-type: none">• app/controllers• app/helpers• app/models• lib <p>Note that the lib directory also has been added before with an explicit engine_cfg.paths.add 'lib', eager_load: true</p> <p>Trying to force load the required module with require File.expand_path('../lib/my_module', __FILE__) or require_relative 'lib/my_module' doesn't help.</p> <p>Before migrating to Rails 7, a simple require MyModule worked fine.</p>			
Related issues:			
Related to Redmine - Feature #36320: Migrate to Rails 7.2		Closed	
Related to Redmine - Defect #40856: random failing integration test for plugi...		New	

Associated revisions

Revision 22515 - 2023-12-12 17:26 - Marius BĂLTEANU

Reverts r22507, r22508 and r22510 (#36320, #39803).

Revision 22516 - 2023-12-12 17:45 - Marius BĂLTEANU

Update integration test for plugin routes after r22515 (#36320, #39803).

Revision 22517 - 2023-12-12 17:52 - Marius BĂLTEANU

Fix NoMethodError: undefined method `rake_run' for Rails::TestUnit::Runner::Class (NoMethodError) after updating to Rails 7.1 (#36320, #39803)

Patch by @tohosaku.

Revision 22524 - 2023-12-20 04:19 - Marius BĂLTEANU

Fixes that redmine:plugins:test should run plugins unit, functional, integration and system tests in one single test session instead of one session for each type of test. This behaviour was changed in Rails 7.1 because rake test and rails test are integrated (#39803, #36320).

Revision 22525 - 2023-12-20 04:30 - Marius BĂLTEANU

Run plugin tests only if test files exist (#39803, #36320).

History

#1 - 2023-12-10 16:23 - Marius BĂLTEANU

- Related to Feature #36320: Migrate to Rails 7.2 added

#2 - 2023-12-10 16:25 - Marius BĂLTEANU

Redmine 5.1 is using Rails 6.1, only the current trunk was updated to Rails 7.1.2 (see #36320) and will be available in the upcoming Redmine 6.0.0.

Can you check your environment?

#3 - 2023-12-10 16:25 - Marius BĂLTEANU

- Status changed from New to Needs feedback

#4 - 2023-12-10 17:57 - Guus Teley

Hello Marius,

You're correct. I seem to have the 5.1.1-devel version installed which pulled Rails 7.1.2 with it. Does this mean the support of plugins has fundamentally changed or should there come some updates so that plugins are working again as before?

#5 - 2023-12-11 13:34 - Marius BĂLTEANU

There were some changes, but it should work as before.

Can you try with the latest version of trunk ([r22514](#))? all the tests finally pass now. If still doesn't work, please provide the steps to reproduce the problem.

#6 - 2023-12-11 13:44 - Takashi Kato

Hello Guus.

From Redmine 5.0, plugin modules are loaded automatically by the Ruby on Rails autoloader. It's important for the autoloader to match both the Ruby file name and class name, as detailed in the [Rails Guide](#)

Potential reasons for a plugin not working might include:

- Mismatch between class names and file names
- RAILS_ENV being set to test (a recent commit changed the directory for plugin lookup based on RAILS_ENV)

I've developed a sample plugin that functions with the current Redmine trunk, and you can find it here.

https://github.com/tohosaku/redmine_simple_example/tree/master

To test it:

- Clone the repository into the plugins directory.
- Run the following command in the terminal: RAILS_ENV=development bin/rails s.
- Access localhost:3000/example in your browser.

Try accessing the pages with "hello" and "Hello fuga."

If you can view both pages successfully, it indicates that both the autoloader and the module loaded by require_relative are working as intended.

#7 - 2023-12-11 15:14 - Alexander Meindl

Marius BALTEANU wrote in [#note-5](#):

There were some changes, but it should work as before.

Hi,

our plugins to not work as before, see e.g. https://github.com/alphanodes/redmine_ref_issues/actions/runs/7167043514. All tests with master / Rails 7.1 are broken.

E.g.:

```
Run bundle exec rake redmine:plugins:test NAME=redmine_ref_issues RUBYOPT="-W0"
rake aborted!
NoMethodError: undefined method `rake_run' for Rails::TestUnit::Runner:Class (NoMethodError)

      Rails::TestUnit::Runner.rake_run ["plugins/#{ENV['NAME']} || '*'}/test/unit/**/*.rb"]
                             ^^^^^^^^^
/home/runner/work/redmine_ref_issues/redmine_ref_issues/redmine/lib/tasks/redmine.rake:182:in `block (4 levels)
in <top (required)>'
/home/runner/work/redmine_ref_issues/redmine_ref_issues/redmine/lib/tasks/redmine.rake:172:in `block (3 levels)
in <top (required)>'
/opt/hostedtoolcache/Ruby/3.2.2/x64/bin/bundle:25:in `load'
/opt/hostedtoolcache/Ruby/3.2.2/x64/bin/bundle:25:in `<main>'
Tasks: TOP => redmine:plugins:test:units
(See full trace by running task with --trace)
Error: Process completed with exit code 1.
```

In <https://github.com/alphanodes/additionals/actions/runs/7166976930/job/19512117928> this error:

```
Run bundle exec rake generate_secret_token
```

```
rake aborted!
NameError: uninitialized constant Additional::Patches
Did you mean? Pathname
/home/runner/work/additionals/additionals/redmine/plugins/additionals/init.rb:52:in `block in <top (required)>'
/home/runner/work/additionals/additionals/redmine/plugins/additionals/init.rb:51:in `<top (required)>'
/home/runner/work/additionals/additionals/redmine/plugins/additionals/lib/additionals.rb:186:in `block in <class:Plugin>'
/home/runner/work/additionals/additionals/redmine/config/environment.rb:16:in `<top (required)>'
/opt/hostedtoolcache/Ruby/3.0.6/x64/bin/bundle:23:in `load'
/opt/hostedtoolcache/Ruby/3.0.6/x64/bin/bundle:23:in `<main>'
Tasks: TOP => db:create => db:load_config => environment
(See full trace by running task with --trace)
```

Looks like lib directory of a plugin is not found.

#8 - 2023-12-11 15:43 - Guus Teley

This works (I get an form showing "Hello" and "Hello fuga")

But trying to get my own plugin to work fails with the program being unable to find the methods in the library module.
I saw you named your Module 'class' in hoge.rb and 'module' plus 'class' in redmine_simple_example.rb.
It's a mystery to me why I now all of a sudden have to explicitly specify the module/class name with every call to my methods.
Is that something new with 7.1?

#9 - 2023-12-12 08:00 - Marius BĂLTEANU

Alexander Meindl wrote in [#note-7](#):

Marius BALTEANU wrote in [#note-5](#):

There were some changes, but it should work as before.

Hi,

our plugins to not work as before, see e.g. https://github.com/alphanodes/redmine_ref_issues/actions/runs/7167043514. All tests with master / Rails 7.1 are broken.

E.g.:

[...]

In <https://github.com/alphanodes/additionals/actions/runs/7166976930/job/19512117928> this error:

[...]

Looks like lib directory of a plugin is not found.

Method rake_run was removed in Rails 7, we will properly fix this issue.

Also, there is another issue introduced by [r22507](#) which changes the plugins directory for testing environment to another folder in order to test the autoloading and I'm not sure if we should revert this change or each plugin should set in their config environment the line `config.redmine_plugin_directory = 'plugins'`.

What do you think?

#10 - 2023-12-12 10:50 - Takashi Kato

Guus Teley wrote in [#note-8](#):

This works (I get an form showing "Hello" and "Hello fuga")

But trying to get my own plugin to work fails with the program being unable to find the methods in the library module.
I saw you named your Module 'class' in hoge.rb and 'module' plus 'class' in redmine_simple_example.rb.
It's a mystery to me why I now all of a sudden have to explicitly specify the module/class name with every call to my methods.
Is that something new with 7.1?

If all is well, can you upload your plugin?

If you can remove the parts you don't think are necessary for the survey and leave us with at least two files, init.rb and the unloaded module, we can investigate how it is working.

#11 - 2023-12-12 11:32 - Alexander Meindl

Marius BALTEANU wrote in [#note-9](#):

[...]

Method rake_run was removed in Rails 7, we will properly fix this issue.

Also, there is another issue introduced by [r22507](#) which changes the plugins directory for testing environment to another folder in order to test the autoloading and I'm not sure if we should revert this change or each plugin should set in their config environment the line `config.redmine_plugin_directory = 'plugins'`.

What do you think?

I think, it would be better we do not change redmine_plugin_directory with config/environments/test.rb for autoloading tests. For this tests we can set the plugin directory in the test file.

In general I think it is a good idea, to make the plugin directory configurable - as it is already committed.

#12 - 2023-12-12 13:55 - Takashi Kato

Alexander Meindl wrote in [#note-11](#):

[...]

I think, it would be better we do not change redmine_plugin_directory with config/environments/test.rb for autoloading tests. For this tests we can set the plugin directory in the test file.

In general I think it is a good idea, to make the plugin directory configurable - as it is already committed.

I have opened a new issue on plugin testing([#39834](#)).

#13 - 2023-12-12 14:35 - Guus Teley

@Takashi Kato: After installing your example in the development environment I'll noticed everything worked.

Then I noticed the same generated errors when switching back to the production environment.

The error was: "Exception `NameError` at /usr/local/lib/ruby/gems/3.1/gems/zeitwerk-2.6.12/lib/zeitwerk/loader/helpers.rb:135 - uninitialized constant Hoge"

So there's a difference in behaviour between the production and development environment which I'm not aware of.

After changing the config.eager_load value to 'false' for the production environment, everything seems to worked as expected.

It definitely looks like this is the culprit.

Hope this helps.

#14 - 2023-12-12 15:16 - Guus Teley

- File init.rb added

- File itgn_module.rb added

As an addition: My methods in my Module still don't work.

Error message: "NoMethodError (undefined method `access_level' for ItgnModule:Module"

I've attached the init.rb and library module.

#15 - 2023-12-12 17:54 - Marius BĂLTEANU

Alexander Meindl wrote in [#note-11](#):

Marius BALTEANU wrote in [#note-9](#):

[...]

Method rake_run was removed in Rails 7, we will properly fix this issue.

Also, there is another issue introduced by [r22507](#) which changes the plugins directory for testing environment to another folder in order to test the autoloading and I'm not sure if we should revert this change or each plugin should set in their config environment the line `config.redmine_plugin_directory = 'plugins'`.

What do you think?

I think, it would be better we do not change redmine_plugin_directory with config/environments/test.rb for autoloading tests. For this tests we can set the plugin directory in the test file.

In general I think it is a good idea, to make the plugin directory configurable - as it is already committed.

I've reverted those changes and I've committed the fix posted by tohosaku for @undefined method `rake_run`. Can you confirm that your plugins work as before?

#16 - 2023-12-12 20:53 - Alexander Meindl

Hi Marius,

it looks good. All tests with master branch runs without problems: <https://github.com/alphanodes/additional/actions/runs/7186533064>

This means both works: revert of plugin directory and rake_run fix.

Thanks for the fast work on it!

#17 - 2023-12-13 13:35 - Alexander Meindl

I found a side effect with change from Rails::TestUnit::Runner.rake_run to Rails::TestUnit::Runner.run_from_rake with revision [r22517](#)

If I run rake task redmine:plugins:test to run all tests of a plugin (unit, functional and integration tests), there starts there new sessions - after each other.

In general the same test helper is used for all tests. For coverage in the helper there can be

```
require 'simplecov'
SimpleCov.start :rails do
  add_filter 'init.rb'
  root File.expand_path "#{File.dirname __FILE__}/.."
end
```

as an example.

Now this code is executed 3 times - for unit tests, than before functional tests and the before integration tests.

See this output as an example:

```
bundle exec rails redmine:plugins:test RAILS_ENV=test NAME=myplugin
Emptying /Users/alex/dev/redmine_master/test/reports
```

```
Started with run options --seed 23283
```

```
Finished in 238.72515s
391 tests, 1601 assertions, 0 failures, 0 errors, 1 skips
Emptying /Users/alex/dev/redmine_master/test/reports
```

```
Started with run options --seed 62191
```

```
Finished in 157.84991s
407 tests, 2403 assertions, 0 failures, 0 errors, 0 skips
Emptying /Users/alex/dev/redmine_master/test/reports
```

```
Started with run options --seed 12353
```

```
Finished in 78.11091s
110 tests, 706 assertions, 0 failures, 0 errors, 0 skips
```

The reports cleaned after each test group, which is not correct - because test results are deleted of the test group before.

#18 - 2023-12-17 04:48 - Marius BĂLTEANU

- File 39803.patch added

Alexander, the problem is not caused by Redmine, but by the Rails changes. I've wrapped the test run execution in a condition that checks if any file exists in each test directory as an workaround, can you check, please?

#19 - 2023-12-20 04:31 - Marius BĂLTEANU

Marius BALTEANU wrote in [#note-18](#):

Alexander, the problem is not caused by Redmine, but by the Rails changes. I've wrapped the test run execution in a condition that checks if any file exists in each test directory as an workaround, can you check, please?

I've fixed the issue using a different approach, please see [r22524](#).

I'm leaving this issue open to get clarify the issue reported by [Guus Teley](#).

#20 - 2023-12-20 04:51 - Mischa The Evil

- Assignee set to Guus Teley

#21 - 2023-12-20 08:13 - Alexander Meindl

Hi Marius,

your new approach solved my problem. Everything with plugin testing works as before. Thanks for your time on that!

#22 - 2025-05-29 03:36 - Go MAEDA

- Related to Defect #40856: random failing integration test for plugin routes in 5.1.3 added

Files			
init.rb	2.71 KB	2023-12-12	Guus Teley
itgn_module.rb	1.21 KB	2023-12-12	Guus Teley
39803.patch	3.31 KB	2023-12-17	Marius BĂLTEANU