Redmine - Patch #13927
Reduce coupling between plugins and the "plugins/" directory
2013-05-02 12:54 - Jean-Baptiste Barth

Status: New
Priority: Normal
Assignee: Jean-Baptiste Barth
Category: Plugin API
Target version: Unplanned backlogs

Start date:
Due date:
% Done: 0%
Estimated time: 0.00 hour

Description

Today plugins are just directories in `<redmine root>/plugins/` directory. There are a few implicit assumptions in Redmine code that this is the case, which doesn't add much benefit I think. This reference is needed in the plugin discovery mechanism, but I can see obvious reasons why it may appear anywhere else.

It has at least one disadvantage: it prevents from easily distributing plugins as gems or via an other mechanism (which I think has a long list of advantages that I could enumerate another day).

An other motivation is that removing this would lead to clearer code I think, we could remove some `Dir.glob` and `File.xxx` calls in favor of new, clear methods in `Redmine::Plugin`.

I'll try to work on that and link commit to this issue.

Related issues:
- Related to Redmine - Patch # 14402: Plugin migration directory should use plu... Closed
- Related to Redmine - Patch # 24007: Change plugins directory through the conf... Reopened
- Related to Redmine - Patch # 26139: Follow Gemfile's plugin loader as #24007 Reopened

Associated revisions

Revision 11766 - 2013-05-02 19:59 - Jean-Baptiste Barth

Allow plugins to define their base directory (#13927).

History

#1 - 2013-05-04 16:37 - Toshi MARUYAMA

The reason of moving "plugins" to "lib/plugins" is Rails3 prints deprecated warning.
See: r9533.

#2 - 2013-05-04 16:48 - Toshi MARUYAMA

Toshi MARUYAMA wrote:

The reason of moving "plugins" to "lib/plugins" is Rails3 prints deprecated warning.
See: r9533.

Sorry, I posted unrelated matters.

I think Redmine plugin mechanism problem is how to deal assets.
Current Redmine disables asset pipeline.

2020-05-05
Actually that's one problem in Redmine plugins, but I have a few dozen others I could talk about. Re-enabling the asset pipeline can be done today in a standard Redmine plugin but it's a bit hacky and rough around the edges (see this plugin for a proof-of-concept). I'll write about plugin concerns and some proposals in a blog post soon.

Hi,

I added another issue a while back with a patch that extends this with the (at least in my opinion) missing support for migrations. -> #14402

- Related to Patch #14402: Plugin migration directory should use plugin directory added

Jean-Baptiste Barth wrote:

Actually that's one problem in Redmine plugins, ...

FYI, OpenProject (GPL3, Redmine fork) uses plugin gem.
https://github.com/opf/openproject/blob/v4.2.2/doc/DEVELOP_PLUGINS.md

- Related to Patch #24007: Change plugins directory through the configuration.yml file added

- Related to Patch #26139: Follow Gemfile's plugin loader as #24007 added