

ステータス:	New	開始日:	2022/05/09
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Plugin API_20	予定工数:	0.00時間
対象バージョン:		作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/35415	status_id:	1
category_id:	20	tracker_id:	2
version_id:	0	plus1:	0
issue_org_id:	35415	affected_version:	
author_id:	523899	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	0		

説明

Many plugins use different ways to include own lib-code. A common topic I came across is applying patches and consuming hooks. Sometimes the files are @require@d having something like

```
unless FoobarController.included_modules.include? MyPlugin::Patches::FooControllerPatch
  FooController.send(:include, MyPlugin::Patches::FooControllerPatch)
end
```

in the bottom.

Sometimes the @require@ is wrapped in Reloader's @to_prepare@-callback (which seems quite useless to me), other plugins have some kind of @apply_patch@-method implemented, which does the same as the code above in a more generic manner (IMO the most elegant way to go).

While all these techniques basically work, it often invites to mess up with the auto reloader. When doing some tweaks in the codebase (or even develop my own plugin), I either have to disable foreign plugins or painfully restart the dev-server with every change, when autoloading is not properly implemented. Otherwise I face the often reported problem of "Cannot autoload Constant ..." -Error.

My suggestion is to provide a more unified API for applying patches and consuming hooks.

I could imagine a simple approach e.g. of an @apply_patch@ function, which forwards the params to an array, which then is processed internally by the @ActiveSupport::Reloader.to_prepare@ callback.

Another (more opinionated) approach is to grab all patches from plugins/.../lib/patches/*.rb, after a plugin was registered, which also was applicable to the hooks.

Anyways, I think there should be some mechanism to provide plugins safely, which do not mess up with the autoloader. This could also catch compatibility-stuff I have also often seen like @Rails.version < '5.1' ? ActiveSupport::Callbacks : ActiveSupport::Reloader@.to_prepare.

履歴

#1 - 2022/05/10 16:59 - Admin Redmine

- カテゴリ を Plugin API_20 にセット