

Changing plugin loading order

2022/05/09 18:11 - Admin Redmine

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/24939	status_id:	1
category_id:	20	tracker_id:	3
version_id:	0	plus1:	0
issue_org_id:	24939	affected_version:	
author_id:	155112	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	7		
説明			
Hi,			
I have one plugin that depends on another, and I needed to specify the order plugins are loaded so the classes and migrations are loaded in the proper order, but the dependent plugin is loaded first because of alphabetical order of their names.			
I made this patch on lib/redmine/plugin.rb so I can determine plugins loading order by setting its priorities on a Hash in the file config/additional_environment.rb.			
This sorting relies that plugin directory name is equals to its id.			
Regards,			
journals			
I forgot to mention, I am using Redmine 3.3 version.			
The plugin load order is a headache, I fully agree.			
Instead of making a kind of ad-hoc fix, I believe the best solution involves:			
<ul style="list-style-type: none"> • an optional declarative statement "depends_on" that can be added to the init.rb of a plugin • a topological sort based on these dependency relationships 			
This has the benefit to scale and be dynamic... Your solution is overriding by giving a hard coded list... which I consider being a workaround and not a clean solution. Of course the clean solution means more efforts :-)			
I'll try to come with that solution...			
Relates to #23131			
Hi Jaap de Haan,			
I know that is not the ideal solution, but it is sufficient for most of cases, and the as you already said, the best solution would take a lot of additional effort.			
Thanks for considering.			
Best regards,			

We could try loading the plugin, have it check dependencies and, if they are not met, raise a specific exception to defer loading until the next plugin loading iteration. If at the end of a plugin loading iteration there isn't any progress, we know that the remaining plugins have unmet dependencies, like, missing plugins and/or circular dependencies.

Hi Jaap,

I've met another use case when the topological order defined by plugin dependencies won't work.

As Redmine allows you to override views, model class methods among other things in code, the order the plugins are loaded are crucial to define the application behavior.

Sometimes one use two or more independent plugins and you need to define the order they are loaded to meet your requirements and you can't just declare one as dependent on the other.

Regards,

Jaap de Haan wrote:

The plugin load order is a headache, I fully agree.

Instead of making a kind of ad-hoc fix, I believe the best solution involves:

- an optional declarative statement "depends_on" that can be added to the init.rb of a plugin
- a topological sort based on these dependency relationships

This has the benefit to scale and be dynamic... Your solution is overriding by giving a hard coded list... which I consider being a workaround and not a clean solution. Of course the clean solution means more efforts :-)

I'll try to come with that solution...

Relates to #23131

related_issues

relates,Reopened,23131,Plugin load order defined by inter-plugin dependencies

履歴

#1 - 2022/05/10 17:06 - Admin Redmine

- カテゴリをPlugin API_20 にセット