

ステータス:	Reopen	開始日:	2010/07/21
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/5932	status_id:	8
category_id:	20	tracker_id:	1
version_id:	0	plus1:	0
issue_org_id:	5932	affected_version:	
author_id:	443	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	4		

説明

I wrote a plugin to add new commands in repository views (upload files, create directory and delete file/directory) and I noticed that :view_repositories_show_contextual hook lacks of 2 context parameters:

- path
- revision

As a "workaround", I patched the views (show.rhtml, entry.rhtml, changes.rhtml and annotate.rhtml) to replace:

```
<%= call_hook(:view_repositories_show_contextual, { :repository => @repository, :project => @project }) %>
```

with

```
<%= call_hook(:view_repositories_show_contextual, { :repository => @repository, :project => @project, :path => @path, :revision => @rev }) %>
```

but it would be better if included in the next release of Redmine...

journals

As you can see in "one of my plugins":

http://dev.holgerjust.de/projects/redmine-checkout/repository/entry/trunk/lib/checkout/repository_hooks.rb#L15 (line 15), you can get these variables today without having to patch the hook calls.

Many thanks!!! I didn't know how to do that and I really like your solution: I will undo my patches ASAP!!!

I disagree... Many things are accessible through @context[:controller]@ but it's not the right way to go for me. Hooks should provide instance variables directly so plugin developers don't have to do hacky things like @instance_variable_get@. Else, with this argument, we could remove nearly any context variables passed to the hooks ;)

It makes me think of using @.send()@ to bypass object privacy: possible, but not so good in most cases.

Is there a particular reason why we should refuse adding those variables ? Maybe something I missed..

履歴

#1 - 2022/05/10 17:22 - Admin Redmine

- カテゴリをPlugin API_20 にセット