

## Redmine - Patch #29378

### More call\_hooks for projects, issue statuses, time entries, etc.

2018-08-13 14:33 - Jonas Ineveld, van

<b>Status:</b>	Needs feedback	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Hook requests	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Description</b>			
<p>Currently there are only functional call_hooks for issues, and with that, i mean the 'controller_issues_edit_after_save' etc. Because i'm working on a seperate app, connected with redmine on the backend, i need more hooks to hook my 'update API' to.</p> <p>For now i've created the call_hooks for creation of a project, and for the deletion of a issue.</p> <p>Would it be something if these changes would go back into core? I can imagine more uses for these type of hooks.</p> <p>Svn patch file is included</p>			

### History

#### #1 - 2018-08-13 14:45 - Go MAEDA

- Tracker changed from Defect to Patch

#### #2 - 2018-08-13 14:49 - Holger Just

Do you strictly need these controller hooks to distinguish updates done by the controller from other updates?

Most of the time, people are interested in e.g. any update of wiki pages or projects. Here, it is already possible to use Active Record callbacks on the models which don't require callbacks in the controller in all places that could touch these models. See [https://guides.rubyonrails.org/active\\_record\\_callbacks.html](https://guides.rubyonrails.org/active_record_callbacks.html) for details. You could e.g. register a new hook for after\_save (or after\_update) or the Project model or an after\_destroy hook of the Issue model.

Since these hooks are on a model level, you won't have access to the controller params though.

#### #3 - 2018-08-23 13:22 - Holger Just

- Status changed from New to Needs feedback

### Files

call_hooks_patch_project_issue_destroy.patch	1.15 KB	2018-08-13	Jonas Ineveld, van
--	---------	------------	--------------------