

Updating custom fields does not trigger update to "updated_on" field in the customized object

2022/05/09 14:40 - Admin Redmine

ステータス:	Closed	開始日:	2010/04/20
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Custom fields_14	予定工数:	0.00時間
対象バージョン:	4.2.0_152	作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/5354	status_id:	5
category_id:	14	tracker_id:	1
version_id:	152	plus1:	0
issue_org_id:	5354	affected_version:	
author_id:	12633	closed_on:	
assigned_to_id:	332	affected_version_id:	17
comments:	20		
説明			
When updating a custom field (which has been defined within the Redmine settings), the updated_on field in the Projects table is not updated to reflect the date of modification to the project's settings.			
journals			
Patch which corrects the behavior is attached.			
Sorry I don't see the point : I can confirm updated_on field is not updated when you just change a custom field, but why do you need it to be updated ? I don't see any place where we display or use this information, so it doesn't seem to be a problem to me. Can you explain your use-case please ?			
The field is used by plugins and reports at our site to determine the last date of activity on the project's general information. It could potentially be used by other plugins or reporting tools as well.			
Not updating the field is not the expected behavior to end users, and would be considered a bug if a user is expecting the updated_on value to actually reflect the last date of update for the project's general information (whether native or a custom field).			
• Stu			
Jean-Baptiste Barth wrote:			
Sorry I don't see the point : I can confirm updated_on field is not updated when you just change a custom field, but why do you need it to be updated ? I don't see any place where we display or use this information, so it doesn't seem to be a problem to me. Can you explain your use-case please ?			
New patch revision against Redmine 1.0.1			
Please don't assign issues to people without them telling you to do so explicitly.			
Also, your patch has some issues.			
• You save projects two times and don't check the second time. • Patches always have to be against trunk to be accepted.			
* You are missing some tests.			

Bullet points 1 and 2 fixed:

```
# Force a save of the project record with the correct updated_on time
# to cover corner cases of custom fields, etc. being updated by user
# which do not natively trigger the base project save method
@project.updated_on = Time.now
if validate_parent_id && @project.save
  @project.set_allowed_parent!(params[:project]['parent_id']) if params[:project].has_key?('parent_id')
  flash[:notice] = I(:notice_successful_update)
```

Before submitting a new patch against Trunk, what specific tests need to be wrapped around setting project.updated_on in order to get this accepted upstream? ----- Thanks for the patch. You shouldn't set @updated_on@ yourself though. Rails provides a @:touch@ option that will automatically update @updated_on@ when any custom field changes. <http://github.com/rails/rails/commit/abb899c54e8777428b7a607774370ba29a5573bd> Also, your proposed patch would work for editing a project through the web only. If a project is updated any other way, it wouldn't be updated.

----- New patch which handles the update within the custom_value model is attached. Note that I attempted to use the :touch property within the belongs_to in custom_value, but ran into what may be a bug or undefined behavior for polymorphic fields in ActiveRecord (though did not have time to try and debug activerecord itself). I found a way around it by using a callback and then checking if the parent relation was a project. If there is a better methodology to address this within Redmine's model, please do suggest so that it can be implemented and submitted for inclusion.

----- Quick question: Does the development team prefer that a patch issue be opened up w/ the diff for the refactored patch, or just keep the fix attached to this defect?

----- Bump - I have this problem as well. Environment: Redmine version 4.0.2.stable Ruby version 2.4.5-p335 (2018-10-18) [x86_64-linux] Rails version 5.2.2 Environment production Database adapter Mysql2 Mailer queue ActiveSupport::QueueAdapters::AsyncAdapter Mailer delivery sendmail SCM: Subversion 1.7.14 Git 2.19.1 Filesystem Redmine plugins: redmine_agile 1.4.11 redmine_checklists 3.1.16 redmine_default_custom_query 1.4.0 redmine_image_clipboard_paste 3.3.0 redmineup_tags 2.0.4 ----- Update @updated_on@ of the class (e.g. Project) that uses 'acts_as_customizable' with the following patch.

```
diff --git a/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb b/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
index 96fdbf8b6..3703a5e65 100644
--- a/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
+++ b/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
@@ -145,7 +145,9 @@ module Redmine
   end
   end
   self.custom_values = target_custom_values
+  custom_values_changed = custom_values.any?(&:changed?)
+  custom_values.each(&:save)
+  touch if persisted? && custom_values_changed
   @custom_field_values_changed = false
   true
 end
```

----- Setting the target version to 4.2.0. -----

Committed the patch. Thank you. ----- The patch breaks an existing test.

Failure:

IssueTest#test_closed_on_should_be_set_when_closing_an_open_issue [/var/lib/jenkins/workspace/trunk/DATABASE_ADAPTER/postgresql/RUBY_VER/ruby-2.6/test/unit/issue_test.rb:3071]:

No visible difference in the ActiveSupport::TimeWithZone#inspect output.

You should look at the implementation of #== on ActiveSupport::TimeWithZone or its members.

Tue, 11 Feb 2020 04:40:14 UTC +00:00

bin/rails test test/unit/issue_test.rb:3064

Go MAEDA wrote:

The patch breaks an existing test.

[...]

The test passes with the following patch. The @saved_changes?@ method is available in Rails 5.1.0 and later.

https://github.com/rails/rails/blob/v5.1.0/activerecord/lib/active_record/attribute_methods/dirty.rb#L170

<https://github.com/rails/rails/pull/25337>

```
diff --git a/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb b/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
index 3703a5e65..73b2ae847 100644
--- a/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
+++ b/lib/plugins/acts_as_customizable/lib/acts_as_customizable.rb
@@ -145,9 +145,8 @@ module Redmine
   end
   end
   self.custom_values = target_custom_values
-  custom_values_changed = custom_values.any?(&:changed?)
-  custom_values.each(&:save)
-  touch if persisted? && custom_values_changed
+  touch if !saved_changes? && custom_values.any?(&:saved_changes?)
+  @custom_field_values_changed = false
   true
 end
```

Committed the new patch. Thank you.

履歴

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

- カテゴリ を Custom fields_14 にセット

- 対象バージョン を 4.2.0_152 にセット