

Redmine - Defect #15709

TimeEntry custom\_values are not deleted from the database when destroying the associated project

2013-12-16 05:36 - Mischa The Evil

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Jean-Philippe Lang	% Done:	0%
Category:	Custom fields	Estimated time:	0.00 hour
Target version:	2.5.0	Affected version:	
Resolution:	Fixed		

Description

This defect emerged while I was writing some tests (with new fixtures) for my patch for issue [#4138](#). When I ran these new tests they passed without issues, but when I ran the whole test suite to make sure I didn't break any existing functionality I encountered a failing unit test. The failing test is ProjectTest#test\_destroying\_root\_projects\_should\_clear\_data.

The assertion that triggers the test failure is

```
assert_equal 0, CustomValue.where(:customized_type => ['Project', 'Issue', 'TimeEntry', 'Version'])
).count
```

This assertion tests if all custom\_values of the designated customized\_type are deleted from the database when the associated project has been destroyed. This functionality works for custom\_values of the type Project, Issue and Version (where the first two are explicitly tested because they have fixtures, in contrast to the Version type that is not-yet tested because it is missing fixtures), but it does not work for custom\_values of the type TimeEntry.

The defect (and the error-free behavior of the Version type custom\_values) can be made visible by adding some custom\_value fixtures like shown below, followed by running ProjectTest#test\_destroying\_root\_projects\_should\_clear\_data:

```
Index: test/fixtures/custom_fields.yml
=====
--- test/fixtures/custom_fields.yml      (revision 12406)
+++ test/fixtures/custom_fields.yml      (working copy)
@@ -145,3 +145,35 @@
     SGXDqWzDp2prc2Tigqw2NTTDuQ==
     - Other value
     field_format: list
+custom_fields_012:
+  name: Searchable time entry field
+  regexp: ""
+  is_for_all: false
+  is_filter: false
+  type: TimeEntryCustomField
+  possible_values: ""
+  id: 12
+  is_required: false
+  field_format: string
+  searchable: true
+  default_value: ""
+  editable: true
+  position: 2
+custom_fields_013:
+  name: vState
+  regexp: ""
+  is_for_all: false
+  is_filter: false
+  type: VersionCustomField
+  possible_values:
+    - Stable
+    - Beta
+    - Alpha
```

```

+ - Planning
+ id: 13
+ is_required: false
+ field_format: list
+ searchable: false
+ default_value: ""
+ editable: true
+ position: 1
Index: test/fixtures/custom_values.yml
=====
--- test/fixtures/custom_values.yml      (revision 12406)
+++ test/fixtures/custom_values.yml      (working copy)
@@ -101,3 +101,15 @@
   customized_id: 1
   id: 17
   value: '2009-12-01'
+custom_values_018:
+  customized_type: TimeEntry
+  custom_field_id: 12
+  customized_id: 2
+  id: 18
+  value: "stringfortimeentry custom field search"
+custom_values_019:
+  customized_type: Version
+  custom_field_id: 13
+  customized_id: 2
+  id: 19
+  value: Stable

```

The assertion fails because the TimeEntry custom\_value is not deleted from the db when the associated project is destroyed (as opposed to the yet-untested Version custom\_value which is indeed properly deleted [as like for the already-tested Issue and Project custom\_values]).

This issue seems pretty much related to [#7385](#) and [#7904](#).

Environment information:

- Redmine Trunk at [r12406](#)
- Ruby 2.0.0-p353 (2013-11-22) [i686-linux]
- Rails 3.2.16

Please let me know if more information is needed...

## Associated revisions

### Revision 12421 - 2013-12-18 19:39 - Jean-Philippe Lang

Fixed that time entries custom values are not deleted when deleting a project or an issue (#15709).

## History

### #1 - 2013-12-17 20:40 - Jean-Philippe Lang

- Target version set to 2.5.0

### #2 - 2013-12-18 19:39 - Jean-Philippe Lang

- Status changed from New to Closed

- Assignee set to Jean-Philippe Lang

- Resolution set to Fixed

Fixed in [r12421](#), a migration takes care of removing orphan custom values that may be present.  
Thanks for reporting this.