

## Redmine - Defect #11073

### UserCustomField order\_statement returns wrong output

2012-06-01 15:43 - Devadatta Sahoo

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Etienne Massip	<b>% Done:</b>	0%
<b>Category:</b>	Custom fields	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	2.1.0	<b>Affected version:</b>	
<b>Resolution:</b>	Fixed		
<b>Description</b>			
The UserCustomField order_statement returns wrong output:			
Eg:			
For a user custom field named "Employee Code" (type string) the order statement is returned as follows:			
<pre>&gt;&gt; UserCustomField.find_by_name("Employee Code").order_statement =&gt; "COALESCE((SELECT cv_sort.value FROM custom_values cv_sort WHERE cv_sort.customized_type='User' AND cv_sort.customized_id=users.id AND cv_sort.custom_field_id=1 LIMIT 1), "")"</pre>			
Herein the value of customized_type is <b>User</b> , whereas in the actual database table, i.e. <b>custom_values</b> table, the value of customized_type for UserCustomFields is <b>Principal</b> (which is a superclass of User class).			
For this reason, the `UserCustomField.find_by_name("Employee Code").order_statement` gives a statement which always returns empty output when used in a query.			
The correct value of `UserCustomField.find_by_name("Employee Code").order_statement` should be:			
<pre>&gt;&gt; UserCustomField.find_by_name("Employee Code").order_statement =&gt; "COALESCE((SELECT cv_sort.value FROM custom_values cv_sort WHERE cv_sort.customized_type='Principal' AND cv_sort.customized_id=users.id AND cv_sort.custom_field_id=1 LIMIT 1), "")"</pre>			
<b>Related issues:</b>			
Related to Redmine - Feature #1139: Being able to sort the issue list by cust...		<b>Closed</b>	<b>2008-04-29</b>

#### Associated revisions

##### Revision 9880 - 2012-06-23 17:47 - Etienne Massip

Use base class name as customized type to fix UserCustomField#order\_statement.

##### Revision 9881 - 2012-06-23 18:51 - Etienne Massip

Removed test line committed accidentally (#11073).

#### History

##### #1 - 2012-06-01 15:46 - Devadatta Sahoo

- Assignee set to Jean-Philippe Lang

##### #2 - 2012-06-01 15:51 - Etienne Massip

- Assignee deleted (Jean-Philippe Lang)

- Priority changed from High to Normal

##### #3 - 2012-06-01 20:06 - Devadatta Sahoo

- File custom\_field.rb.patch added

A suggested patch has been attached:

File to be patched:

app/models/custom\_field.rb

**#4 - 2012-06-18 10:55 - Etienne Massip**

- Status changed from New to Confirmed
- Target version set to Candidate for next minor release

Rails stores the base class type (Principal) instead of the actual class type (User).

These RoR issues might be relevant:

- <https://rails.lighthouseapp.com/projects/8994/tickets/5617>
- <https://github.com/rails/rails/issues/724> (continuation of lh 5617)
- <https://github.com/rails/rails/issues/617>

**#5 - 2012-06-18 11:05 - Etienne Massip**

Maybe you can try to replace `self.class.customized_class.name` with `self.class.customized_class.base_class.name`?

**#6 - 2012-06-23 17:48 - Etienne Massip**

- Status changed from Confirmed to Resolved
- Assignee set to Etienne Massip
- Target version changed from Candidate for next minor release to 2.1.0
- Resolution set to Fixed

Should be fixed with [r9880](#), thanks for the report.

I don't know if it's useful to write a test since this statement is not used in Redmine code to query User model nor where I should write it?

**#7 - 2012-07-26 18:57 - Jean-Philippe Lang**

- Status changed from Resolved to Closed

**Files**

---

custom_field.rb.patch	1.64 KB	2012-06-01	Devadatta Sahoo
-----------------------	---------	------------	-----------------