

Redmine - Defect #40157

Problem with Version types project CFs group titles in filters drop-down boxes

2024-01-31 04:48 - Stoyan Zhekov

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Custom fields	Estimated time:	0.00 hour
Target version:		Affected version:	5.1.0
Resolution:	Duplicate		

Description

Problem

Redmine-5.1.x does not handle properly **group titles** in filters drop-down boxes for association type custom fields (**custom field names, containing dot** , like **Version** type - example cf_1.status). It is trying to implement **localization** (l() method) on **CF name**, which lead to *Translation missing* error. It **should display directly the name** , not trying to localize it. See the attached screenshots for RM-5.0.3 and RM-5.1.0 filters drop-down box in *Projects* screen.

How to reproduce the problem

- 1.) Create Version-type project custom field and enable it to be used as a filter - see attached screenshot.
- 2.) Go to Projects screen and open Filters drop-down box
- 3.) There will be *Translation missing* group title

Possible fix for the problem

I think the problem originates in changing the orders of matchers in app/helpers/queries_helper.rb file, filters_options_for_select(query) method.

Up to RM-5.0 the /^cf_\d+\./.match?(field) matcher was taking place **before** field =~ /^(.+)\./ one. In RM-5.1 this matcher is on the top, so it is the first executed on **CF names, containing dot** (for example cf_1.status - version-type CF).

As a result group name cf_1 becomes :field_cf_1, which later, in localized_grouped = grouped.map {[k, v] [k.is_a?(Symbol) ? l(k) : k.to_s, v]} line is **never resolved** (because it is **unknown symbol, should be just the CF name**), leading to translation missing error for the group name.

The debug flow for filters_options_for_select(query) execution in RM-5.0.x vs RM-5.1.x for version-type CF follows:

RM-5.0.3 - proper group name handling

```
field: "cf_6"
field opts: name=Sprint type=list_optional
group: nil
grouped: {}
ungrouped: [{"", "id"}, [{"", "status"}, ["Sprint", "cf_6"]]
```

```
field: "cf_6.due_date"
field opts: name=Sprint type=date
cf_: Sprint <- ---[ !!! matcher: /^cf_\d+\./.match?(field) , cf_1 ]---
group: "Sprint"
grouped: {"Sprint"=>[["Sprint", "cf_6.due_date"]]}
ungrouped: [{"", "id"}, [{"", "status"}, ["Sprint", "cf_6"]]
```

```
field: "cf_6.status"
field opts: name=Sprint type=list
cf_: Sprint
group: "Sprint"
grouped: {"Sprint"=>[["Sprint", "cf_6.due_date"], ["Sprint", "cf_6.status"]]}
ungrouped: [{"", "id"}, [{"", "status"}, ["Sprint", "cf_6"]]
```

```
localized: ["Sprint", [["Sprint", "cf_6.due_date"], ["Sprint", "cf_6.status"]]]
```

RM-5.1.0 - translation missing error for group name

```
field: "cf_3"
field opts: name=Sprint type=list_optional
group: nil
grouped: {:label_string=>[], :label_date=>[], :label_time_tracking=>[], :label_attachment=>[]}
ungrouped: [{"", "id"}, [{"", "status"}, {"Sprint", "cf_3"}]

field: "cf_3.due_date"
field opts: name=Sprint type=date
association: cf_3 <- ---[ !!! matcher: field =~ /^(.+)\./ , cf_3 field_cf_3 ]---
group: :field_cf_3
grouped: {:label_string=>[], :label_date=>[], :label_time_tracking=>[], :label_attachment=>[], :field_cf_3=>[{"Sprint", "cf_3.due_date"}]}
ungrouped: [{"", "id"}, [{"", "status"}, {"Sprint", "cf_3"}]

field: "cf_3.status"
field opts: name=Sprint type=list
association: cf_3
group: :field_cf_3
grouped: {:label_string=>[], :label_date=>[], :label_time_tracking=>[], :label_attachment=>[], :field_cf_3=>[{"Sprint", "cf_3.due_date"}, {"Sprint", "cf_3.status"}]}
ungrouped: [{"", "id"}, [{"", "status"}, {"Sprint", "cf_3"}]

localized: [{"Translation missing: ja.field_cf_3"}, [{"Sprint", "cf_3.due_date"}, {"Sprint", "cf_3.status"}]]
```

To fix the problem, I think, a simple restore of the old matchers order (`/^cf_d+\.\/` matcher **before** `^(.+)\./` one) will be enough:

```
--- app/helpers/queries_helper.rb.orig 2024-01-31 12:37:14.000000000 +0900
+++ app/helpers/queries_helper.rb      2024-01-31 12:37:47.000000000 +0900
@@ -26,15 +26,15 @@
   ungrouped = []
   grouped = {label_string: [], label_date: [], label_time_tracking: [], label_attachment: []}
   query.available_filters.map do |field, field_options|
-     if field =~ /^(.+)\./
-       # association filters
-       group = "field_#{field}".to_sym
+     if field_options[:type] == :relation
+       group = :label_relations
+     elsif field_options[:type] == :tree
+       group = query.is_a?(IssueQuery) ? :label_relations : nil
+     elsif /^cf_d+\.\/.match?(field)
+       group = (field_options[:through] || field_options[:field]).try(:name)
+     elsif field =~ /^(.+)\./
+       # association filters
+       group = "field_#{field}".to_sym
+     elsif %w(member_of_group assigned_to_role).include?(field)
+       group = :field_assigned_to
+     elsif field_options[:type] == :date_past || field_options[:type] == :date
```

Related issues:

Related to Redmine - Defect #39714: Query grouping filter not working for cus...

Closed

History

#1 - 2024-01-31 04:51 - Marius BĂLTEANU

- Related to Defect #39714: Query grouping filter not working for custom field relations added

#2 - 2024-01-31 04:53 - Marius BĂLTEANU

We did some changes in [#39714](#).

#3 - 2024-01-31 06:57 - Go MAEDA

- File clipboard-202401311454-vipi5.png added

- Status changed from New to Closed

- Resolution set to Duplicate

I cannot reproduce the issue with the current trunk [r22665](#). I think the issue was fixed in Redmine 5.1.1 ([#39714](#)). Please check if the issue is still reproducible with Redmine 5.1.1.

clipboard-202401311454-vlpi5.png

#4 - 2024-01-31 09:30 - Go MAEDA

- Affected version changed from 5.1.1 to 5.1.0

Files

filters-5.0.png	25.4 KB	2024-01-31	Stoyan Zhekov
filters-5.1.png	26.7 KB	2024-01-31	Stoyan Zhekov
version_cf.png	17 KB	2024-01-31	Stoyan Zhekov
clipboard-202401311454-vlpi5.png	137 KB	2024-01-31	Go MAEDA