

Redmine - Defect #8508

Projet with Subtask won't open after migration from 1.1.3 to 1.2.0

2011-06-02 21:53 - Mario Morneau

Status:	Closed	Start date:	2011-06-02
Priority:	Urgent	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Resolution:	Invalid	Affected version:	1.2.0

Description

After migrating to version Redmine 1.2.0 all my project with subtask can not open see error below

If you create an issues ans then add un subtask same problem

the problem is order by clause is missing the identifying table

as follow

Select ORDER BY "lft"

should be depending of what is the scope of the lists to show

Select ORDER BY "issues"."lft" or

Select ORDER BY "projctcs"."lft" or

Select ORDER BY "projctcs"."lft", "issues"."lft"

Rendered issues/ action menu (4.3ms)

[[4;35;1]issue Load Including Associations (0.0ms) [0m [0mSQLite3::SQLException: ambiguous column name: lft: SELECT "issues"."id" AS t0_r0, "issues"."tracker_id" AS t0_r1, "issues"."project_id" AS t0_r2, "issues"."subject" AS t0_r3, "issues"."description" AS t0_r4, "issues"."due_date" AS t0_r5, "issues"."category_id" AS t0_r6, "issues"."status_id" AS t0_r7,

"issues"."assigned to id" AS t0_r8, "issues"."priority id" AS t0_r9, "issues"."fixed version id" AS t0_r10, "issues"."author id" AS t0_r11, "issues"."lock version" AS t0_r12, "issues"."created on" AS

```
t0_r13, "issues"."updated_on" AS t0_r14, "issues"."start_date" AS t0_r15, "issues"."done_ratio" AS t0_r16, "issues"."estimated_hours" AS t0_r17, "issues"."parent_id" AS t0_r18, "issues"."root_id" AS
```

t0_r19, "issues"."lft" AS t0_r20, "issues"."rgt" AS t0_r21, "issues"."is_private" AS t0_r22, "projects"."id" AS t1_r0, "projects"."name" AS t1_r1, "projects"."description" AS t1_r2, "projects"."homepage"

AS t1 r3. "projects"."is_public" AS t1 r4. "projects"."parent_id" AS t1 r5. "projects"."created_on" AS t1 r6. "projects"."updated_on" AS t1 r7. "projects"."identifier" AS t1 r8. "projects"."status" AS

```
t1_r9, "projects"."lft" AS t1_r10, "projects"."rgt" AS t1_r11 FROM "issues" LEFT OUTER JOIN "projects" ON "projects".id = "issues".project_id WHERE (((projects.status=1 AND projects.id IN (SELECT em.project_id FROM enabled_modules em WHERE em.name='issue_tracking')) AND (issues.id != 26)) AND ((issues."lft" <= 2 AND issues."rgt" >= 3) AND ("issues"."root_id" = 21))) ORDER BY "lft"
```

[[0m

ActionView::TemplateError (SQLite3::SQLException: ambiguous column name: lft: SELECT "issues"."id" AS t0_r0, "issues"."tracker_id" AS t0_r1, "issues"."project_id" AS t0_r2, "issues"."subject" AS t0_r3, "issues"."description" AS t0_r4, "issues"."due_date" AS t0_r5, "issues"."category_id" AS t0_r6, "issues"."status_id" AS t0_r7, "issues"."assigned_to_id" AS t0_r8, "issues"."priority_id" AS t0_r9, "issues"."fixed_version_id" AS t0_r10, "issues"."author_id" AS t0_r11, "issues"."lock_version" AS t0_r12, "issues"."created_on" AS t0_r13, "issues"."updated_on" AS t0_r14, "issues"."start_date" AS t0_r15, "issues"."done_ratio" AS t0_r16, "issues"."estimated_hours" AS t0_r17, "issues"."parent_id" AS t0_r18, "issues"."root_id" AS t0_r19, "issues"."lft" AS t0_r20, "issues"."rgt" AS t0_r21, "issues"."is_private" AS t0_r22, "projects"."id" AS t1_r0, "projects"."name" AS t1_r1, "projects"."description" AS t1_r2, "projects"."homepage" AS t1_r3, "projects"."is_public" AS t1_r4, "projects"."parent_id" AS t1_r5, "projects"."created_on" AS t1_r6, "projects"."updated_on" AS t1_r7, "projects"."identifier" AS t1_r8, "projects"."status" AS t1_r9, "projects"."lft" AS t1_r10, "projects"."rgt" AS t1_r11 FROM "issues" LEFT OUTER JOIN "projects" ON "projects".id = "issues".project_id WHERE (((projects.status=1 AND projects.id IN (SELECT em.project_id FROM enabled_modules em WHERE em.name='issue_tracking')) AND (issues.id != 26)) AND ((issues."lft" <= 2 AND issues."rgt" >= 3) AND ("issues"."root_id" = 21))) ORDER BY "lft") on line #9 of app/views/issues/show.rhtml:

6: <%= avatar(@issue.author, :size => "50") >^M

7: ^M

8: <div class="subject"> ^M

9: <= render_issue_subject_with_tree(@issue) >^M

10: </div>^M

11: <p class="author">^M

12: <= authoring @issue.created_on, @issue.author %>^M

```
app/helpers/issues_helper.rb:64:in `render_issue_subject_with_tree'
app/views/issues/show.rhtml:9:in `_run_rhtml_app47views47issues47show46rhtml'
app/controllers/issues_controller.rb:121:in `show'
app/controllers/issues_controller.rb:120:in `show'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/httpserver.rb:104:in `service'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/httpserver.rb:65:in `run'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:173:in `start_thread'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:162:in `start'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:162:in `start_thread'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:95:in `start'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:92:in `each'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:92:in `start'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:23:in `start'
/home/users/limdev/tools/ruby1.8.7/lib/ruby/1.8/webrick/server.rb:82:in `start'
```

Rendering /home/users/limdev/tools/redmine-1.2.0/public/500.html (500 Internal Server Error)

History

#1 - 2011-06-03 09:17 - Etienne Massip

Did you run migration ?

#2 - 2011-06-03 16:00 - Mario Morneau

Yes i ran the

rake db:migrate RAILS_ENV=production

and i empty the cache and the session like the upgrade document

see the result here

rake db:migrate RAILS_ENV=production

(in /home/users/xxxxxx/tools/redmine-1.2.0)

AddWorkflowsAssigneeAndAuthor: migrating =====

-- add_column(:workflows, :assignee, :boolean, {:default=>false, :null=>false})

> 0.0010s

-- add_column(:workflows, :author, :boolean, {:default=>false, :null=>false})

> 0.0005s

AddWorkflowsAssigneeAndAuthor: migrated (0.0298s) =====

AddUsersSalt: migrating =====

-- add_column(:users, :salt, :string, {:limit=>64})

> 0.0007s

AddUsersSalt: migrated (0.0008s) ===== SaltUserPasswords: migrating

=====

-- Salting user passwords, this may take some time...

```

-> 0.0317s
SaltUserPasswords: migrated (0.0318s) ===== AddRepositoriesPathEncoding: migrating
=====
-- add_column(:repositories, :path_encoding, :string, {:limit=>64, :default=>nil})
-> 0.0007s
AddRepositoriesPathEncoding: migrated (0.0008s) ===== ChangeRepositoriesPasswordLimit: migrating
=====
-- change_column(:repositories, :password, :string, {:limit=>nil, :default=>""})
-> 0.0076s
ChangeRepositoriesPasswordLimit: migrated (0.0077s) ===== ChangeAuthSourcesAccountPasswordLimit: migrating
=====
-- change_column(:auth_sources, :account_password, :string, {:limit=>nil, :default=>""})
-> 0.0089s
ChangeAuthSourcesAccountPasswordLimit: migrated (0.0090s) ===== ChangeJournalDetailsValuesToText: migrating
=====
-- change_column(:journal_details, :old_value, :text)
> 0.0524s
-- change_column(:journal_details, :value, :text)
-> 0.1477s
ChangeJournalDetailsValuesToText: migrated (0.2002s) ===== AddRepositoriesLogEncoding: migrating
=====
-- add_column(:repositories, :log_encoding, :string, {:limit=>64, :default=>nil})
-> 0.0010s
AddRepositoriesLogEncoding: migrated (0.0011s) ===== CopyRepositoriesLogEncoding: migrating
=====
CopyRepositoriesLogEncoding: migrated (0.0178s) ===== AddIndexToUsersType: migrating
=====
-- add_index(:users, :type)
-> 0.0010s
AddIndexToUsersType: migrated (0.0011s) ===== AddRolesIssuesVisibility: migrating
=====
-- add_column(:roles, :issues_visibility, :string, {:limit=>30, :null=>false, :default=>"default"})
-> 0.0008s
AddRolesIssuesVisibility: migrated (0.0009s) ===== AddIssuesIsPrivate: migrating
=====
-- add_column(:issues, :is_private, :boolean, {:null=>false, :default=>false})
-> 0.0010s
AddIssuesIsPrivate: migrated (0.0011s) ===== AddRepositoriesExtraInfo: migrating
=====
-- add_column(:repositories, :extra_info, :text)
-> 0.0012s
AddRepositoriesExtraInfo: migrated (0.0013s) =====

```

and then the plugging update
rake db:migrate:upgrade_plugin_migrations RAILS_ENV=production
(in /home/users/xxx/tools/redmine-1.2.0)
Cannot find old migration table - assuming nothing needs to be done

and then

```

~/tools/redmine-1.2.0$ rake db:migrate_plugins RAILS_ENV=production
(in /home/users/xxx/tools/redmine-1.2.0)
Migrating engines...
Migrating acts_as_activity_provider...
Migrating acts_as_attachable...
Migrating acts_as_customizable...
Migrating acts_as_event...
Migrating acts_as_list...
Migrating acts_as_searchable...
Migrating acts_as_tree...
Migrating acts_as_versioned...
Migrating acts_as_watchable...
Migrating awesome_nested_set...
Migrating classic_pagination...
Migrating gravatar...
Migrating open_id_authentication...
Migrating prepend_engine_views...
Migrating redmine_time_tracker...
Migrating rfpdf...
Migrating ruby-net-ldap-0.0.4...

```

#3 - 2011-06-06 13:52 - Mario Morneau

Etienne Massip wrote:

Did you run migration ?

Yes Sorry for the late answer
I put the info in an update of the current problem

#4 - 2011-06-09 14:37 - Mario Morneau

- Status changed from New to Resolved

I found my problem (my installation is on linux)

i did for Step 3 Perform the upgrade (option 1)
step 5 Copy the folders of your installed plugins into new installation directory

```
cp -pr $HOME/redmine-x.x.x/vendor/plugins $HOME/redmine-y.y.y/vendor/.
```

This is wrong i had to copy only the sub-directory of the plugins that i have installed

so i did the following

```
diff $HOME/redmine-x.x.x/vendor/plugins $HOME/redmine-y.y.y/vendor/plugins  
only in $HOME/redmine-x.x.x/vendor/plugins/redmine_time_tracker
```

and then i did

```
cp -pr $HOME/redmine-x.x.x/vendor/plugins/redmine_time_tracker $HOME/redmine-y.y.y/vendor/plugins/.
```

Like someone said in another issue those simple upgrade should be put in a script for unix and BAT or CMD for windows
The script would have as input old_redmine_root_directory and new_redmine_root_directory.

the script would do the following

- 1) copy the database.yml and email.yml from old installation to the new one
- 2) read the settings.yml from the old directory and update the settings in the new one
- 3) if new properties in the settings.yml need the user to change a value asks for those values and update the file
- 4) copy the files subdirectory from old directory to the new directory
- 5) find all plugins from old directory that need to be copy over in the new directory
- 6) generate sessions store if needed
- 7) find all themes from old directory that need to be copy over in the new directory
- 8) copy all the db needed from old directory to the new directory
- 9) do the migration of database if necessary
- 10) do the migration of the plugins if necessary
- 11) clear the cache and the sessions

this script would be use after the unzip and then the service would be started

Note the script could be a java application

Thank's and sorry for the urgent status id did not read carefully the step 5 (not clear enough for a Québécois)

#5 - 2011-06-09 17:19 - Etienne Massip

- Status changed from Resolved to Closed

- Resolution set to Invalid

Thanks for your feedback !