Redmine - Defect #22646

PostgreSQL: Exception in issue report view.

2016-04-27 17:44 - Lars Kanis

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Issues		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Invalid		**Affected version:**	

Description

The following exception is raised, when viewing /projects/projektmanagement/issues/report on PostgreSQL:

```
ActiveRecord::StatementInvalid (PG::AmbiguousColumn: ERROR:  column reference "status_id" is ambig
uous
LINE 1: SELECT COUNT(*) AS count_all, status_id AS status_id, is_clo...
                                      ^
: SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, tracker_id AS trac
ker_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "
issue_statuses" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "trackers" ON "trackers
"."id" = "issues"."tracker_id" WHERE ((projects.id = 204) AND (projects.status <> 9 AND projects.i
d IN (SELECT em.project_id FROM enabled_modules em WHERE em.name='issue_tracking'))) GROUP BY "iss
ues"."status_id", "is_closed", "issues"."tracker_id"):
  app/models/issue.rb:1348:in `count_and_group_by'
  app/models/issue.rb:1305:in `by_tracker'
  app/controllers/reports_controller.rb:31:in `issue_report'
  lib/redmine/sudo_mode.rb:63:in `sudo_mode'
```

The attached patch fixes this.

History

#1 - 2016-04-27 18:16 - Toshi MARUYAMA

- Description updated

#2 - 2016-04-27 18:47 - Toshi MARUYAMA

- Status changed from New to Needs feedback

I cannot reproduce on my vanilla Redmine 3.2.1 CentOS 6.

```
$ rpm -q postgresql
postgresql-8.4.20-5.el6_7.x86_64
$ bundle show pg
/home/xxxxx/.rvm/gems/ruby-1.9.3-p551/gems/pg-0.18.4

Started GET "/projects/ecookbook/issues/report" for 192.168.11.10 at 2016-04-28 01:40:47 +0900
Processing by ReportsController#issue_report as HTML
  Parameters: {"id"=>"ecookbook"}
   (0.4ms)  SELECT MAX("settings"."updated_on") FROM "settings"
  User Load (0.3ms)  SELECT  "users".* FROM "users" WHERE "users"."type" IN ('User', 'AnonymousUser') AND "use
rs"."status" = $1 AND "users"."id" = $2 LIMIT 1  [["status", 1], ["id", 1]]
  Current user: admin (id=1)
  Setting Load (0.1ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "force_default_language_for_loggedin"]]
  Project Load (0.2ms)  SELECT  "projects".* FROM "projects" WHERE "projects"."identifier" = $1 LIMIT 1  [["id
entifier", "ecookbook"]]
   (0.1ms)  SELECT "enabled_modules"."name" FROM "enabled_modules" WHERE "enabled_modules"."project_id" = $1
[["project_id", 1]]
  IssueStatus Load (0.3ms)  SELECT "issue_statuses".* FROM "issue_statuses"  ORDER BY "issue_statuses"."positi
on" ASC
  Version Load (0.9ms)  SELECT "versions".* FROM "versions" INNER JOIN "projects" ON "projects"."id" = "versio
ns"."project_id" WHERE (projects.id = 1 OR (projects.status <> 9 AND ( versions.sharing = 'system' OR (project
```

```
s.lft >= 1 AND projects.rgt <= 10 AND versions.sharing = 'tree') OR (projects.lft < 1 AND projects.rgt > 10 AN
D versions.sharing IN ('hierarchy', 'descendants')) OR (projects.lft > 1 AND projects.rgt < 10 AND versions.sh
aring = 'hierarchy'))))
  Project Load (0.3ms)  SELECT "projects".* FROM "projects" WHERE "projects"."id" IN (1, 2, 3, 5)
  IssuePriority Load (0.4ms)  SELECT "enumerations".* FROM "enumerations" WHERE "enumerations"."type" IN ('Iss
uePriority')  ORDER BY "enumerations"."position" ASC
  Setting Load (0.1ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "issue_group_assignment"]]
  User Load (0.2ms)  SELECT DISTINCT "users".* FROM "users" INNER JOIN "members" ON "members"."user_id" = "use
rs"."id" WHERE "users"."type" IN ('User', 'AnonymousUser') AND "users"."status" = $1 AND (members.project_id =
 1)  [["status", 1]]
  Setting Load (0.1ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "user_format"]]
   (1.4ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, tracker_id AS tracke
r_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_statuses
" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "trackers" ON "trackers"."id" = "issues"."tracker
_id" WHERE ((projects.id = 1) AND (projects.status <> 9 AND projects.id IN (SELECT em.project_id FROM enabled_
modules em WHERE em.name='issue_tracking'))) GROUP BY "issues"."status_id", "is_closed", "issues"."tracker_id"

   (1.0ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, fixed_version_id AS
fixed_version_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "is
sue_statuses" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "versions" ON "versions"."id" = "issu
es"."fixed_version_id" WHERE ((projects.id = 1) AND (projects.status <> 9 AND projects.id IN (SELECT em.projec
t_id FROM enabled_modules em WHERE em.name='issue_tracking'))) GROUP BY "issues"."status_id", "is_closed", "is
sues"."fixed_version_id"
   (1.0ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, priority_id AS prior
ity_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_status
es" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "enumerations" ON "enumerations"."id" = "issues
"."priority_id" AND "enumerations"."type" IN ('IssuePriority') WHERE ((projects.id = 1) AND (projects.status <
> 9 AND projects.id IN (SELECT em.project_id FROM enabled_modules em WHERE em.name='issue_tracking'))) GROUP B
Y "issues"."status_id", "is_closed", "issues"."priority_id"
   (1.0ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, category_id AS categ
ory_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_status
es" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "issue_categories" ON "issue_categories"."id" =
 "issues"."category_id" WHERE ((projects.id = 1) AND (projects.status <> 9 AND projects.id IN (SELECT em.proje
ct_id FROM enabled_modules em WHERE em.name='issue_tracking'))) GROUP BY "issues"."status_id", "is_closed", "i
ssues"."category_id"
   (1.6ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, assigned_to_id AS as
signed_to_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_
statuses" ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "users" ON "users"."id" = "issues"."assig
ned_to_id" WHERE ((projects.id = 1) AND (projects.status <> 9 AND projects.id IN (SELECT em.project_id FROM en
abled_modules em WHERE em.name='issue_tracking'))) GROUP BY "issues"."status_id", "is_closed", "issues"."assig
ned_to_id"
   (1.0ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, author_id AS author_
id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_statuses"
ON "issue_statuses"."id" = "issues"."status_id" INNER JOIN "users" ON "users"."id" = "issues"."author_id" AND
"users"."type" IN ('User', 'AnonymousUser') WHERE ((projects.id = 1) AND (projects.status <> 9 AND projects.id
 IN (SELECT em.project_id FROM enabled_modules em WHERE em.name='issue_tracking'))) GROUP BY "issues"."status_
id", "is_closed", "issues"."author_id"
   (0.9ms)  SELECT COUNT(*) AS count_all, status_id AS status_id, is_closed AS is_closed, project_id AS projec
t_id FROM "issues" INNER JOIN "projects" ON "projects"."id" = "issues"."project_id" INNER JOIN "issue_statuses
" ON "issue_statuses"."id" = "issues"."status_id" WHERE (((projects.id = 1 OR (projects.lft > 1 AND projects.r
gt < 10))) AND (projects.status <> 9 AND projects.id IN (SELECT em.project_id FROM enabled_modules em WHERE em
.name='issue_tracking'))) GROUP BY "issues"."status_id", "is_closed", "issues"."project_id"
  Setting Load (0.2ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "ui_theme"]]
  Tracker Exists (0.1ms)  SELECT  1 AS one FROM "trackers" INNER JOIN "projects_trackers" ON "trackers"."id" =
 "projects_trackers"."tracker_id" WHERE "projects_trackers"."project_id" = $1 LIMIT 1  [["project_id", 1]]
  Tracker Load (0.1ms)  SELECT "trackers".* FROM "trackers" INNER JOIN "projects_trackers" ON "trackers"."id"
= "projects_trackers"."tracker_id" WHERE "projects_trackers"."project_id" = $1  ORDER BY "trackers"."position"
 ASC  [["project_id", 1]]
  Rendered reports/_simple.html.erb (86.1ms)
  Rendered reports/_simple.html.erb (3.8ms)
  Rendered reports/_simple.html.erb (2.0ms)
  Rendered reports/_simple.html.erb (1.6ms)
  Rendered reports/_simple.html.erb (4.0ms)
   (0.2ms)  SELECT COUNT(*) FROM "projects" WHERE "projects"."parent_id" = $1  [["parent_id", 1]]
   (0.3ms)  SELECT COUNT(*) FROM "projects" WHERE (projects.lft > 1 AND projects.rgt < 10) AND (projects.statu
s <> 9)
  Project Load (0.2ms)  SELECT "projects".* FROM "projects" WHERE (projects.lft > 1 AND projects.rgt < 10) AND
 (projects.status <> 9)  ORDER BY "projects"."lft" ASC
  Rendered reports/_simple.html.erb (5.3ms)
  IssueCategory Exists (0.1ms)  SELECT  1 AS one FROM "issue_categories" WHERE "issue_categories"."project_id"
 = $1 LIMIT 1  [["project_id", 1]]
  IssueCategory Load (0.1ms)  SELECT "issue_categories".* FROM "issue_categories" WHERE "issue_categories"."pr
```

```
oject_id" = $1  ORDER BY issue_categories.name  [["project_id", 1]]
  Rendered reports/_simple.html.erb (11.7ms)
  Rendered reports/issue_report.html.erb within layouts/base (488.6ms)
  Setting Load (0.4ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "app_title"]]
  UserPreference Load (0.1ms)  SELECT  "user_preferences".* FROM "user_preferences" WHERE "user_preferences"."
user_id" = $1 LIMIT 1  [["user_id", 1]]
  Setting Load (0.2ms)  SELECT  "settings".* FROM "settings" WHERE "settings"."name" = $1  ORDER BY "settings"
."id" DESC LIMIT 1  [["name", "gravatar_enabled"]]
  Project Load (0.2ms)  SELECT "projects"."id", "projects"."name", "projects"."identifier", "projects"."lft",
"projects"."rgt" FROM "projects" INNER JOIN "members" ON "projects"."id" = "members"."project_id" WHERE "membe
rs"."user_id" = $1 AND (projects.status<>9) AND "projects"."status" = $2  [["user_id", 1], ["status", 1]]
  Wiki Load (0.1ms)  SELECT  "wikis".* FROM "wikis" WHERE "wikis"."project_id" = $1 LIMIT 1  [["project_id", 1
]]
  Board Exists (0.1ms)  SELECT  1 AS one FROM "boards" WHERE "boards"."project_id" = $1 LIMIT 1  [["project_id
", 1]]
  Repository Load (0.1ms)  SELECT  "repositories".* FROM "repositories" WHERE "repositories"."project_id" = $1
 AND (is_default = 't') LIMIT 1  [["project_id", 1]]
Completed 200 OK in 1460ms (Views: 697.6ms | ActiveRecord: 33.8ms)
```

#3 - 2016-04-27 18:54 - Toshi MARUYAMA

Tests cover.
source:tags/3.2.1/test/functional/reports_controller_test.rb

#4 - 2016-04-28 13:07 - Lars Kanis

- File test-pg-9.5.log added

This is interesting. Putting any of the above "SELECT COUNT(*) AS count_all..." statements into psql, results in the "column reference "status_id" is ambiguous" error for me. This is because both projects and issues have this column, but it is table-prefixed in the GROUP BY part only, not in the SELECT part. I wonder why it works on Centos-6.

I get this error on PostgreSQL versions 8.4.22, 9.3.11 and 9.5.2.
System is Ubuntu-16.04 with ruby-2.3.0.

Attached is the log of a full redmine test suite run on PostgreSQL-9.5.2. It results in 11 errors, because of this issue. With the above patch applied, all tests run green.

#5 - 2016-04-28 13:29 - Toshi MARUYAMA

Tests of trunk on Ruby 2.3 pass.
http://www.redmine.org/builds/index.html

#6 - 2016-04-28 13:49 - Toshi MARUYAMA

Redmine 3.2 does not support Ruby 2.3.
But test/functional/reports_controller_test.rb passes.

```
$ ruby test/functional/reports_controller_test.rb
/home/xxxxx/.rvm/gems/ruby-2.3.0/gems/htmlentities-4.3.1/lib/htmlentities/mappings/expanded.rb:465: warning: k
ey "inodot" is duplicated and overwritten on line 466
Run options: --seed 36978

# Running:

.....

Finished in 1.585129s, 3.1543 runs/s, 39.1135 assertions/s.

5 runs, 62 assertions, 0 failures, 0 errors, 0 skips
$ ruby --version
ruby 2.3.0p0 (2015-12-25 revision 53290) [x86_64-linux]
$ hg parent --template='{svnpath}\n{svnrev}\n'
/branches/3.2-stable
15358
```

#7 - 2016-04-28 14:04 - Toshi MARUYAMA

- Tracker changed from Patch to Defect

- Status changed from Needs feedback to Closed

- Resolution set to Invalid

Lars Kanis wrote:

> This is interesting. Putting any of the above "SELECT COUNT(*) AS count_all..." statements into psql, results in the "column reference "status_id" is ambiguous" error for me. This is because both projects and issues have this column,

"Projects" table does not have "status_id".

```
redmine_test=# \d projects
                            Table "public.projects"
      Column        |            Type             |                    Modifiers
--------------------+-----------------------------+--------------------------------------------------
 id                 | integer                     | not null default nextval('projects_id_seq'::regclass)
 name               | character varying(255)      | not null default ''::character varying
 description        | text                        |
 homepage           | character varying(255)      | default ''::character varying
 is_public          | boolean                     | not null default true
 parent_id          | integer                     |
 created_on         | timestamp without time zone |
 updated_on         | timestamp without time zone |
 identifier         | character varying(255)      |
 status             | integer                     | not null default 1
 lft                | integer                     |
 rgt                | integer                     |
 inherit_members    | boolean                     | not null default false
 default_version_id | integer                     |
Indexes:
    "projects_pkey" PRIMARY KEY, btree (id)
    "index_projects_on_lft" btree (lft)
    "index_projects_on_rgt" btree (rgt)
```

#8 - 2016-04-28 15:18 - Lars Kanis

Sorry for the noise! I did the database setup per "rake db:setup" instead of "rake db:migrate", because I thought, that db/schema.rb comes from the VCS. Setup per "rake db:migrate" fixed this issue. I'm not sure where this additional column in my db/schema.rb came from.

Thank you very much!

Files

0001-PostgreSQL-Fix-Exception-in-issue-report-view.patch	1.61 KB	2016-04-27	Lars Kanis
test-pg-9.5.log	19.2 KB	2016-04-28	Lars Kanis