

Redmine - Defect #1336

After 0.6.3 -- 0.7.1 upgrade, project views fail with SQL error about "tracker_id" column

2008-05-31 07:57 - Anonymous

Status:	Closed	Start date:	2008-05-31
Priority:	High	Due date:	
Assignee:		% Done:	0%
Category:	Projects	Estimated time:	0.00 hour
Target version:		Affected version:	0.7.1
Resolution:	Cant reproduce		

Description

When installing Redmine, I initially checked out SVN tag 0.6.3, because the docs said that was the latest stable revision. Everything worked fine and I set up a few projects, issues, news items and even a forum. Then I viewed the Redmine changelog and found that the latest stable release is actually 0.7.1. So I did an "svn switch" to the corresponding tag, and followed the upgrade instructions, including the three 'rake' commands.

Now there's one big problem: all of the project overview pages (/projects/show/xxxx) now fail to render due to a Ruby exception:

```
ActiveRecord::StatementInvalid (SQLite3::SQLException: no such column: tracker_id: SELECT COUNT(*)
  AS count_all, tracker_id AS tracker_id FROM (SELECT DISTINCT issues.id FROM issues  LEFT OUTER JO
IN projects ON projects.id = issues.project_id  LEFT OUTER JOIN issue_statuses ON issue_statuses.i
d = issues.status_id  LEFT OUTER JOIN trackers ON trackers.id = issues.tracker_id  WHERE (project
s.status=1) AND (((projects.id = 1 OR projects.parent_id = 1)) AND issue_statuses.is_closed='f')
GROUP BY tracker_id )):
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract_
adapter.rb:150:in `log'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_ad
apter.rb:132:in `execute'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_ad
apter.rb:345:in `catch_schema_changes'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_ad
apter.rb:132:in `execute'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_ad
apter.rb:256:in `select'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/
database_statements.rb:7:in `select_all_without_query_cache'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/
query_cache.rb:53:in `select_all'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/
query_cache.rb:74:in `cache_sql'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/
query_cache.rb:53:in `select_all'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:220:in `execu
te_grouped_calculation'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:121:in `calcu
late'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:119:in `catch
'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:119:in `calcu
late'
  /usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:46:in `count'
  /app/controllers/projects_controller.rb:96:in `show'
  ....
```

That query does appear to be invalid, since the nested SELECT doesn't return a tracker_id column. But I don't know nearly enough about the innards of Rails to guess what's causing this. I looked at the schema of my sqlite database, and the correct tables have 'tracker_id' columns (matching the schema.rb file). Everything else seems to be working fine, but not being able to view a project's main page is kind of a showstopper...

History

#1 - 2008-05-31 07:59 - Anonymous

FYI, the page that led me astray by saying 0.6.3 is the latest release is:

<http://www.redmine.org/wiki/redmine/CheckingoutRedmine>

I don't have privileges to edit the page, but someone ought to fix that to say 0.7.1.

#2 - 2008-05-31 20:14 - Anonymous

I tried starting over from scratch: I killed the app process, moved aside the sqlite3 database, ran the rake commands to clear the cache and sessions, and then followed the installation instructions starting from the "rake migrate" command. As expected, I got a virgin installation. I set up my account, created a project, and then went to the project's overview page. Unfortunately I got exactly the same exception:

```
ActiveRecord::StatementInvalid (SQLite3::SQLException: no such column: tracker_id: SELECT COUNT AS count_all, tracker_id AS tracker_id FROM (SELECT DISTINCT issues.id FROM issues LEFT OUTER JOIN projects ON projects.id = issues.project_id LEFT OUTER JOIN issue_statuses ON issue_statuses.id = issues.status_id LEFT OUTER JOIN trackers ON trackers.id = issues.tracker_id WHERE (projects.status=1) AND (((projects.id = 1 OR projects.parent_id = 1)) AND issue_statuses.is_closed='f') GROUP BY tracker_id )):
```

It must be something to do with my configuration?

- Dreamhost (regular shared hosting plan)
- Using Passenger (aka mod_rails)
- Redmine installed at the root of a domain (projects.mooseyard.com)
- Rails 2.0.2
- Ruby 1.8.5
- sqlite3 3.2.1

Help! I'm not sure what to do now, other than start backing up to each previous release until I find one that works again like 0.6.3 did.

#3 - 2008-06-01 08:13 - Anonymous

I've regressed it further: the problem does not occur in 0.6.4, but does occur in 0.7.0, so it must have been introduced in the latter.

It looks like there were a huge number of changes made in 0.7.0, so that may not narrow it down by much. If you can't reproduce this, and want me to try specific SVN revision numbers, just let me know. I've written some simple shellscripts to check out a particular revision and rebuild a virgin database, so it's pretty easy for me to switch back and forth now.

#4 - 2008-06-02 19:34 - Jean-Philippe Lang

I'm not able to reproduce with 0.7.1 release and Rails 2.0.2 and a sqlite3 database.

The generated query is valid and different from yours:

```
SELECT count(DISTINCT issues.id) AS count_all, tracker_id AS tracker_id FROM issues LEFT OUTER JOIN projects ON projects.id = issues.project_id LEFT OUTER JOIN issue_statuses ON issue_statuses.id = issues.status_id LEFT OUTER JOIN trackers ON trackers.id = issues.tracker_id WHERE (projects.status=1) AND (((projects.id = 1 OR projects.parent_id = 1)) AND issue_statuses.is_closed='f') GROUP BY tracker_id
```

Since this query is generated by Rails, you may use a different version of Rails.

Can you try to freeze Redmine with the Rails 2.0.2 official release:

1. rake rails:freeze:edge TAG=rel_2-0-2 (from your redmine directory)
2. restart the app

#5 - 2008-06-02 22:35 - Anonymous

- File *production.sqlite3* added

- File *schema.rb* added

It didn't change anything — I updated to trunk, froze to 2.0.2 and did the installation again. I got exactly the same exception viewing the project's overview.

I've attached the sqlite database and the schema file, in case there's anything wrong with them.

#6 - 2008-06-03 01:21 - Anonymous

Even in 0.6.4, which had been working fine, I've now found a case where a very similar exception occurs: when viewing details of a version (i.e. "/versions/show/4"). I found this as soon as I created versions and added an issue to one of them -- versions with no issues assigned render OK.

```
ActionView::TemplateError (SQLite3::SQLException: no such column: category_id: SELECT COUNT(*) AS count_all, category_id AS category_id FROM (SELECT DISTINCT issues.id FROM issues LEFT OUTER JOIN issue_statuses ON issue_statuses.id = issues.status_id WHERE (issues.fixed_version_id = 4 AND issue_statuses.is_closed = 'f') GROUP BY category_id )) on line #8 of versions/show.rhtml:
5: <h2><%= h(@version.name) %></h2>
6:
```

```

7: <div id="status_by">
8: <%= render_issue_status_by(@version, params[:status_by]) if @version.fixed_issues.count > 0 %>
9: </div>
10:
11: <%= render :partial => 'versions/overview', :locals => {:version => @version} %>

/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract_adapter.rb:1
50:in `log'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_adapter.rb:132
:in `execute'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_adapter.rb:345
:in `catch_schema_changes'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_adapter.rb:132
:in `execute'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/sqlite_adapter.rb:256
:in `select'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/database_statements.rb:7:in `select_all_without_query_cache'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/query_cache.rb:53:in `select_all'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/query_cache.rb:74:in `cache_sql'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/connection_adapters/abstract/query_cache.rb:53:in `select_all'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:220:in `execute_grouped_calculation'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:121:in `calculate'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:119:in `catch'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:119:in `calculate'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.0.2/lib/active_record/calculations.rb:46:in `count'
app/helpers/versions_helper.rb:32:in `render_issue_status_by'
app/views/versions/show.rhtml:8:in `_run_erb_47app47views47versions47show46rhtml'
/usr/lib/ruby/gems/1.8/gems/actionpack-2.0.2/lib/action_view/base.rb:637:in `send'
/usr/lib/ruby/gems/1.8/gems/actionpack-2.0.2/lib/action_view/base.rb:637:in `compile_and_render_template'
...

```

#7 - 2008-06-03 23:52 - Jean-Philippe Lang

Can you post the content of your config/database.yml ?

#8 - 2008-06-04 06:36 - Anonymous

Sure, here it is. This is just copied over from a brand new Rails app I'd created the other day (with Rails 2.0.2).

```

development:
  adapter: sqlite3
  database: db/development.sqlite3
  timeout: 5000

# Warning: The database defined as 'test' will be erased and
# re-generated from your development database when you run 'rake'.
# Do not set this db to the same as development or production.
test:
  adapter: sqlite3
  database: db/test.sqlite3
  timeout: 5000

production:
  adapter: sqlite3
  database: db/production.sqlite3
  timeout: 5000

```

This is all very mysterious, and my suspicion is that it's something to do with the old versions of Ruby and sqlite that Dreamhost is using. Unfortunately they do this because that's what comes with the 2006-era release of Debian they're using on all their servers; they said they'll be upgrading to a newer Debian, but I don't know how long that will take.

#9 - 2008-10-14 22:40 - John Jetmore

Jens, did you narrow this down any? I am also on Dreamhost and experiencing the same problem as a new user with 0.7.3. Redmine looks like an excellent package and I would like very much to use it.

#10 - 2008-10-15 19:51 - Anonymous

No; I haven't been actively using Redmine for the past few months.

Worse, someone pointed out to me recently that my existing deployment of the old version of Redmine is now totally broken -- viewers just see a

Passenger "fatal error launching Rails app" message. I haven't had a chance to look at what happened; my guess is that Dreamhost upgraded something and the new configuration causes Redmine to fail completely.

#11 - 2009-06-06 00:32 - Mike Henderson

This is an issue with the latest version in trunk (2778). I get the following as soon as I add a Subversion repository and try to access the project. If I remove the repository, I continue to get the error. By the way, this is on dreamhost, too, using Passenger. If I use MySQL for the database, I do not have the error.

```
ActiveRecord::StatementInvalid (SQLite3::SQLException: no such column: tracker_id: SELECT COUNT AS count_all, tracker_id AS tracker_id FROM (SELECT DISTINCT "issues".id FROM "issues" LEFT OUTER JOIN "projects" ON "projects".id = "issues".project_id LEFT OUTER JOIN "issue_statuses" ON "issue_statuses".id = "issues".status_id LEFT OUTER JOIN "trackers" ON "trackers".id = "issues".tracker_id WHERE (((projects.id = 1 OR (projects.lft > 1 AND projects.rgt < 2))) AND issue_statuses.is_closed='f') AND (projects.status=1 AND EXISTS (SELECT em.id FROM enabled_modules em WHERE em.name='issue_tracking' AND em.project_id=projects.id)) GROUP BY tracker_id ) count_all_subquery):
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/abstract_adapter.rb:188:in `log'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/sqlite_adapter.rb:132:in `execute'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/sqlite_adapter.rb:372:in `catch_schema_changes'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/sqlite_adapter.rb:132:in `execute'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/sqlite_adapter.rb:275:in `select'
/usr/lib/ruby/gems/1.8/gems/activerecord-2.2.2/lib/active_record/connection_adapters/abstract/database_statements.rb:7:in
`select_all_without_query_cache'
```

```
$ ruby --version
ruby 1.8.5 (2006-08-25) [i386-linux]
$ rails --version
Rails 2.2.2
$ sqlite3 -version
3.2.1

]$ svn info
Path: .
URL: http://redmine.rubyforge.org/svn/trunk
Repository Root: http://redmine.rubyforge.org/svn
Repository UUID: e93f8b46-1217-0410-a6f0-8f06a7374b81
Revision: 2778
Node Kind: directory
Schedule: normal
Last Changed Author: jplang
Last Changed Rev: 2778
Last Changed Date: 2009-06-02 10:24:50 -0700 (Tue, 02 Jun 2009)
```

#12 - 2010-02-11 20:39 - Jean-Philippe Lang

- Status changed from New to Closed
- Resolution set to Cant reproduce

Files

production.sqlite3	98 KB	2008-06-02	Anonymous
schema.rb	19 KB	2008-06-02	Anonymous