Redmine - Defect #2755
Failing test after r2257 with MySQL 4.1.20: "test_sort_by_float_custom_field_asc(QueryTest)"

2009-02-15 15:00 - Mischa The Evil

Status:	Closed		**Start date:**	2009-02-15
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Database		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Wont fix		**Affected version:**	

Description

After re-starting my instance of CCRB I noticed the specified test failing after the check-in of r2257 on MySQL 4.1.20 with the following trace:

    1) Error:
    test_sort_by_float_custom_field_asc(QueryTest):
    ActiveRecord::StatementInvalid: Mysql::Error: You have an error in your SQL syntax; check the manual that corresponds to
    your MySQL server version for the right syntax to use near 'decimal(60,3)) FROM custom_values cv_sort WHERE
    cv_sort.customized_type='Issue' ' at line 1: 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, `users`.`id` AS t1_r0, `users`.`login` AS t1_r1, `users`.`hashed_password` AS t1_r2,
    `users`.`firstname` AS t1_r3, `users`.`lastname` AS t1_r4, `users`.`mail` AS t1_r5, `users`.`mail_notification` AS t1_r6,
    `users`.`admin` AS t1_r7, `users`.`status` AS t1_r8, `users`.`last_login_on` AS t1_r9, `users`.`language` AS t1_r10,
    `users`.`auth_source_id` AS t1_r11, `users`.`created_on` AS t1_r12, `users`.`updated_on` AS t1_r13, `users`.`type` AS
    t1_r14, `users`.`identity_url` AS t1_r15, `issue_statuses`.`id` AS t2_r0, `issue_statuses`.`name` AS t2_r1,
    `issue_statuses`.`is_closed` AS t2_r2, `issue_statuses`.`is_default` AS t2_r3, `issue_statuses`.`position` AS t2_r4,
    `trackers`.`id` AS t3_r0, `trackers`.`name` AS t3_r1, `trackers`.`is_in_chlog` AS t3_r2, `trackers`.`position` AS t3_r3,
    `trackers`.`is_in_roadmap` AS t3_r4, `projects`.`id` AS t4_r0, `projects`.`name` AS t4_r1, `projects`.`description` AS t4_r2,
    `projects`.`homepage` AS t4_r3, `projects`.`is_public` AS t4_r4, `projects`.`parent_id` AS t4_r5, `projects`.`created_on` AS
    t4_r6, `projects`.`updated_on` AS t4_r7, `projects`.`identifier` AS t4_r8, `projects`.`status` AS t4_r9, `projects`.`lft` AS t4_r10,
    `projects`.`rgt` AS t4_r11, `enumerations`.`id` AS t5_r0, `enumerations`.`opt` AS t5_r1, `enumerations`.`name` AS t5_r2,
    `enumerations`.`position` AS t5_r3, `enumerations`.`is_default` AS t5_r4 FROM `issues` LEFT OUTER JOIN `users` ON
    `users`.id = `issues`.assigned_to_id LEFT OUTER JOIN `issue_statuses` ON `issue_statuses`.id = `issues`.status_id LEFT
    OUTER JOIN `trackers` ON `trackers`.id = `issues`.tracker_id LEFT OUTER JOIN `projects` ON `projects`.id =
    `issues`.project_id LEFT OUTER JOIN `enumerations` ON `enumerations`.id = `issues`.priority_id   WHERE
    (((projects.status=1 AND EXISTS (SELECT em.id FROM enabled_modules em WHERE em.name='issue_tracking' AND
    em.project_id=projects.id)) AND (1=0 OR projects.is_public = 1))) ORDER BY (SELECT CAST(cv_sort.value AS
    decimal(60,3)) FROM custom_values cv_sort WHERE cv_sort.customized_type='Issue' AND cv_sort.customized_id=issues.id
    AND cv_sort.custom_field_id=6 AND cv_sort.value <> '' AND cv_sort.value IS NOT NULL LIMIT 1) ASC

    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/connection_adapters/abstract_adapter.rb
    _adapter.rb:147:in `log'

    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/connection_adapters/mysql_adapter.rb:30
    dapter.rb:302:in `execute'

    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/connection_adapters/mysql_adapter.rb:53
    dapter.rb:537:in `select'

/opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/connection_adapters/abstract/database_s
/database_statements.rb:7:in `select_all_without_query_cache'

/opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/connection_adapters/abstract/query_cach
/query_cache.rb:61:in `select_all'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/associations.rb:1429:in
`select_all_rows'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/associations.rb:1260:in
`find_with_associations'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/associations.rb:1258:in
`catch'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/associations.rb:1258:in
`find_with_associations'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/base.rb:1343:in
`find_every'
    /opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activerecord-2.1.2/lib/active_record/base.rb:540:in `find'
    ./test/unit/query_test.rb:231:in `test_sort_by_float_custom_field_asc'

/opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activesupport-2.1.2/lib/active_support/testing/setup_and_teardown.rb:67:in `_
n.rb:67:in `__send__'

/opt/ruby-enterprise-1.8.6-20080810/lib/ruby/gems/1.8/gems/activesupport-2.1.2/lib/active_support/testing/setup_and_teardown.rb:67:in `ru
n.rb:67:in `run'

Related issues:		
Related to Redmine - Defect # 5778: Defect in the filters (custom field) from...	**Closed**	**2010-06-28**
Related to Redmine - Defect # 5726: Cannot sort issues on custom fields	**Closed**	**2010-06-22**

History

#1 - 2009-02-15 15:29 - Jean-Philippe Lang

I've just had a look at the Mysql 4 doc, the CAST function is supported but casting as decimal is not:
http://dev.mysql.com/doc/refman/4.1/en/cast-functions.html

I run the tests with Mysql 5, maybe I should change the requirements.

#2 - 2009-02-15 15:55 - Mischa The Evil

Jean-Philippe Lang wrote:

> *I run the tests with Mysql 5, maybe I should change the requirements.*

With that you mean dropping support for MySQL 4.x for Redmine 0.9?

I think that is something to consider in the eye of the extended support EOL of MySQL 4.1.x untill 2009-12-31, which means that OS-vendors can still
maintain an OS which includes MySQL 4.1.x, supported untill 2009-12-31 (meaning they can work on a new stable release of their product [including a
migrating to MySQL 5.x] ut-most the rest of this year)...

PS: this painfully reminds me of my issue with my private server's OS-vendor currently still not providing a supported update to MySQL 5.x... :evil:

#3 - 2009-02-16 23:43 - Eric Davis

I run MySQL 5 myself (Debian stable) but keeping MySQL 4 support might be needed if there are a lot of users still on 4. I've got a private CruiseControl.rb system running Redmine's tests for every commit against a sqlite3 database. I'm thinking about setting up other instances to test against MySQL and Postgres, I could add a MySQL 4 server in there if it's helpful.

#4 - 2009-02-21 14:45 - Jean-Philippe Lang

I don't really want to drop MySQL 4 support. The problem is I can't find a SQL syntax that works with MySQL 4 & 5, Postgresql 8 and SQLite 3. And I would prefer not to add database specific statements. I don't even know if the database version can be retrieved from within the app.

#5 - 2009-03-29 18:05 - Mischa The Evil

After testing this physically on Redmine trunk at r2641 I can confirm that this *logically* leads to a 500-error in the browser, when sorting an issuelist on a custom field of the type float, which persists the *entire lenght of the session*...

#6 - 2009-11-23 02:31 - Mischa The Evil
- Category set to Database

#7 - 2010-07-11 19:24 - Jean-Baptiste Barth

JP, Eric: maybe you should confirm what have been said in #5726, i.e. Redmine no longer support Mysql 4.1 (extended support from the community has ended since 12/2009). And close this issue if it's the case.

#8 - 2010-12-17 06:21 - Mischa The Evil

Any updates on this?

#9 - 2011-02-21 12:08 - Etienne Massip
- Status changed from New to Closed
- Resolution set to Wont fix

Closed as MySQL 4 is not supported anymore in [[RedmineInstall#Database]]