Redmine - Defect #5822

Problem with Gantt and Calendar with SQL Server 2005

2010-07-05 19:58 - Victor Hugo Bueno Silva

Status:	Closed	**Start date:**	2010-07-05
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Database	**Estimated time:**	0.00 hour
Target version:			
Resolution:	Wont fix	**Affected version:**	0.9.3

Description

```
OS: Windows Server 2003
Database: SQL Server 2005
Ruby: ruby 1.8.7 (2010-01-10 patchlevel 249) [i386-mingw32]
Rails: 2.3.8
Redmine: 0.9.3
(gem) ruby-odbc (0.99991)
(gem) activerecord-sqlserver-adapter (2.3.8)
```

When acessing the calendar page for a project, the calendar appears empty. And when trying to acess the gantt page, the following error shows up on the log file. I guess it is a datetime format problem, but I could not find anything related around.

```
Rendering template within layouts/base
Rendering issues/gantt.rhtml

ActionView::TemplateError (comparison of Date with Time failed) on line #186 of app/views/issues/g
antt.rhtml:
183:     i_done_date = (i_done_date <= @gantt.date_from ? @gantt.date_from : i_done_date )
184:     i_done_date = (i_done_date >= @gantt.date_to ? @gantt.date_to : i_done_date )
185:
186:     i_late_date = [i_end_date, Date.today].min if i_start_date < Date.today
187:
188:     i_left = ((i_start_date - @gantt.date_from)*zoom).floor
189:     i_width = ((i_end_date - i_start_date + 1)*zoom).floor - 2                  # total width
 of the issue (- 2 for left and right borders)

    app/views/issues/gantt.rhtml:186:in `each'
    app/views/issues/gantt.rhtml:186:in `min'
    app/views/issues/gantt.rhtml:186:in `_run_rhtml_app47views47issues47gantt46rhtml'
    app/views/issues/gantt.rhtml:177:in `each'
    app/views/issues/gantt.rhtml:177:in `_run_rhtml_app47views47issues47gantt46rhtml'
    app/controllers/issues_controller.rb:384
    app/controllers/issues_controller.rb:383:in `gantt'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/connection.rb:76:in `pre
_process'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/connection.rb:74:in `cat
ch'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/connection.rb:74:in `pre
_process'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/connection.rb:57:in `pro
cess'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/connection.rb:42:in `rec
eive_data'
    eventmachine (0.12.10) lib/eventmachine.rb:256:in `run_machine'
    eventmachine (0.12.10) lib/eventmachine.rb:256:in `run'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/backends/base.rb:57:in `
start'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/server.rb:156:in `start'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/controllers/controller.r
b:80:in `start'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/runner.rb:177:in `send'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/runner.rb:177:in `run_co
```

```
mmand'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/lib/thin/runner.rb:143:in `run!'
    C:/server/ruby/lib/ruby/gems/1.8/gems/thin-1.2.7-x86-mingw32/bin/thin:6
    C:/server/ruby/bin/thin:19:in `load'
    C:/server/ruby/bin/thin:19

Rendering C:/server/localhost/redmine/public/500.html (500 Internal Server Error)
```

Related issues:

Related to Redmine - Feature #12713: Microsoft SQLServer support	**Closed**

History

#1 - 2010-07-05 20:08 - Felix Schäfer

Rails 2.3.8 is currently not supported, I'd rather suspect the problem there. Please try again with rails 2.3.5.

#2 - 2010-07-06 01:54 - Holger Just

Also SQL Server is not officially supported. Stuff might (and probably will) break.

#3 - 2010-07-25 12:38 - Jean-Philippe Lang

- Status changed from New to Closed

- Resolution set to Wont fix

Indeed, SQL server is not supported.

#4 - 2010-09-24 21:46 - William Wedin

- File Redmine_Datetime_Patch.patch added

- Status changed from Closed to Reopened

I've tracked the issue down to some oddities with datetimes values from certain database servers and date arithmetic. It appears to be a trend for commercial, "enterprise" database servers (so far, SQL Server and Oracle: http://www.redmine.org/boards/2/topics/7745) to return a full datetime, rather than just the date portion; Rails doesn't seem to make any effort to shield the application from these differences. Trying to add/subtract a date to/from a datetime causes a somewhat cryptic "Cannot convert from Date to Float" error. At least, that's the most that I can gather.

I've updated to the latest trunk and manually corrected the errors as they came up. It essentially consists of calling to_date on any date value that comes from the database. I realize that's not probably not the most elegant or maintainable solution, but I thought it would make a useful proof-of-concept. Would anyone be willing to test my patch to make sure it doesn't cause problems for other configurations?

Perhaps the ultimate solution would be to update the models to give consistent behavior across the board, or to track down the problems with the different Active Record adapters and submit a patch there. I'm willing to commit some time to help out if I can.

#5 - 2010-09-24 22:21 - Eric Davis

William Wedin wrote:

> track down the problems with the different Active Record adapters and submit a patch there. I'm willing to commit some time to help out if I can.

I think that will be the best option. Calling to_date everywhere in the code is very brittle and would create a lot of extra objects for the adapters that don't need it (e.g. MySQL). If the SQL Server (Oracle, etc) adapter called to_date, then Redmine and other applications wouldn't have to worry about it at all.

#6 - 2010-09-24 23:07 - Felix Schäfer

William Wedin wrote:

> track down the problems with the different Active Record adapters and submit a patch there. I'm willing to commit some time to help out if I can.

Looking through the AR AbstractAdapter: http://github.com/rails/rails/blob/1c11437a32a973fa9b521c32caa7256f9772acd7/activerecord/lib/active_record/connection_adapters/abstract/schema_definitions.rb#L51, a column defined as date in the migrations should return a Date, not a DateTime. This is clearly the job of the adapter.

#7 - 2010-09-27 20:49 - William Wedin

It looks like the adapter authors are aware of this problem, at least in the case of SQL Server. Check out http://github.com/rails-sqlserver/activerecord-sqlserver-adapter under the section **Date/Time Data Type Hinting**. It seems that, since SQL Server prior to 2008 didn't allow you to distinguish between Date and Time datatypes, the adapter needs a little help coercing the value to the correct type. I haven't tried out the suggested workaround yet.

#8 - 2010-09-28 20:21 - William Wedin

- File Redmine_SQL_Server_Dates_Patch.patch added

I tried out the method suggested by the author of activerecord-sqlserver-adapter. I had to call coerce_sqlserver_date on start_date and due_date in Issue, Project, and Version, and also on Version.effective_date. That fixed the calendar and the gantt chart html, png, and pdf output modes all at once. I'm not sure if it would be useful for users running on SQL Server to call this method on other columns; the seven I used were chosen because they caused errors with my particular data.

I've attached a patch. Someone with more knowledge might be able to point out some important date fields that also should be coerced.

I understand that supporting SQL Server is not a priority for the Redmine team, but perhaps the developers could consider making this small concession for us poor folks stuck using it. Otherwise, perhaps this patch could prove useful to someone else stuck in a similar situation.

#9 - 2013-01-12 23:52 - Daniel Felix

This refers to #12713.

#10 - 2013-01-13 12:30 - Jean-Philippe Lang

- Status changed from Reopened to Closed

I'm closing it. Inferring attributes definition from the database is a AR principle. I'm not in favour of adding this patch that kind of breaks this principle and adds adapter specific code to models. This works fine with SQL Server 2012 without any patch.

Files

Redmine_Datetime_Patch.patch	7.43 KB	2010-09-24	William Wedin
Redmine_SQL_Server_Dates_Patch.patch	1.6 KB	2010-09-28	William Wedin