Redmine - Defect #7218

Date range error on issue query

2011-01-03 12:11 - Felix Schäfer

Status:	Closed		**Start date:**	2011-01-03
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Issues		**Estimated time:**	0.00 hour
Target version:	1.2.0			
Resolution:	Fixed		**Affected version:**	

Description

When filtering issues for "this week" on issues the ranges are broken on core as well as on custom fields. The following examples were all tested against r4471 on MySQL with german locale (week starts on monday), the tests were realized on a sqlite installation too yielding similar but sometimes even "more broken" results.

When filtering issues for "this week" on core issue field, for example "Due date", the query returns results for 8 days instead of 7. This query executed today returns results from 03.01.2011 to 10.01.2011 (both dates included), whereas the current week is from 03.01.2011 to 09.01.2011.

When filtering issues for "this week" on a custom issue field with the type date, the query doesn't return results for the first day of the expected range (in this case monday) but does return them for one extra day at the end of the range. This query executed today returns results from 04.01.2011 to 10.01.2011 (both dates included), whereas the current week is from 03.01.2011 to 09.01.2011.

History

#1 - 2011-01-03 12:16 - Felix Schäfer

The culprit is source:/trunk/app/models/query.rb#L606, though I'm not sure where the problem is (I'd wager it has something to do with querying a date column with time values), but using the same "trick"/hack as in r2054, I was able to produce following diff:

```
diff --git a/app/models/query.rb b/app/models/query.rb
index 7a42de1..b1b919d 100644
--- a/app/models/query.rb
+++ b/app/models/query.rb
@@ -602,7 +602,7 @@ class Query < ActiveRecord::Base
       ((Date.today.cwday == 7) ? Time.now.at_beginning_of_day : Time.now.at_beginning_of_week - 1.day) :
         # week starts on monday (Rails default)
         Time.now.at_beginning_of_week
-      sql = "#{db_table}.#{db_field} BETWEEN '%s' AND '%s'" % [connection.quoted_date(from), connection.quote
d_date(from + 7.days)]
+      sql = "#{db_table}.#{db_field} BETWEEN '%s' AND '%s'" % [connection.quoted_date((from - 1.day).to_time.
end_of_day), connection.quoted_date((from + 6.days).to_time.end_of_day)]
     when "l"
       from = l(:general_first_day_of_week) == '7' ?
         # week starts on sunday
```

This seems to alleviate the problem for me on MySQL and as far as I tested it on SQLite too. I'd really like this operator tested though (currently the only test done is if it runs, not if the result is correct), but I haven't come up with a good way to do it yet.

#2 - 2011-01-03 12:18 - Felix Schäfer

(I missed a part of the diff in the reply above, I have edited it to include the missing part, sorry)

#3 - 2011-04-18 11:32 - Etienne Massip

See that on this site, but isn't it due to localization ? I'm on GMT+1, what timezone is Redmine website running in ?

#4 - 2011-04-29 13:30 - Jean-Philippe Lang

- Status changed from New to Closed

- Target version set to 1.2.0

- Resolution set to Fixed

Fixed in r5596 using #date_range_clause.