

Redmine - Defect #12803

Cross-year date range in time tracking report misses some hours

2013-01-10 19:43 - Murray Melvin

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Time tracking	Estimated time:	0.00 hour
Target version:		Affected version:	2.2.0
Resolution:	Duplicate		
Description			
How to reproduce:			
<ol style="list-style-type: none">1. Have some tickets that have time tracked in both 2012 and 2013.2. Select "last 2 weeks" date preset (or manually enter 2012-12-24 to 2013-01-06)3. Select the Report tab4. Add Details Week- Project5. Add Details Week- Issue6. Note that the total at the top of the report does not equal the total in the bottom right corner of the report. Times from 2013 have been omitted, although the projects and issues are included.7. Also note that the weeks shown in the column headers are 2012-52 and 2012-1			
This has also been reported to be a problem in 1.4.4.stable.9879.			
Related issues:			
Is duplicate of Redmine - Defect #5329: Time entries report broken on first w...		Closed	2010-04-15

History

#1 - 2013-01-16 21:31 - Uwe Koloska

Here is a patch against 1.4.4.stable.9879 (the lines that have to be changed are the same in trunk)

```
--- lib/redmine/helpers/time_report.rb (Revision 9879)
+++ lib/redmine/helpers/time_report.rb (Arbeitskopie)
@@ -67,7 +67,7 @@
     when 'month'
       row['month'] = "#{row['tyear']}-#{row['tmonth']}"
     when 'week'
-       row['week'] = "#{row['tyear']}-#{row['tweek']}"
+       row['week'] = Date.parse("#{row['spent_on']}").cwyar.to_s + "-#{row['tweek']}"
     when 'day'
       row['day'] = "#{row['spent_on']}"
     end
@@ -98,7 +98,7 @@
     @periods << "#{date_from.year}-#{date_from.month}"
     date_from = (date_from + 1.month).at_beginning_of_month
     when 'week'
-     @periods << "#{date_from.year}-#{date_from.to_date.cweek}"
+     @periods << "#{date_from.to_date.cwyar}-#{date_from.to_date.cweek}"
     date_from = (date_from + 7.day).at_beginning_of_week
     when 'day'
       @periods << "#{date_from.to_date}"
```

Reason is the difference between the year for the week and the year from the date. The first ISO week of this year looks like this in the time_entries table:

year	tmonth	tweek	spent_on
2012	12	1	2012-12-31
2013	1	1	2013-01-01
2013	1	1	2013-01-02
2013	1	1	2013-01-03

Maybe someone could help me with the following questions, as I'm only a noob with redmine and rails development:

- is there a more elegant way to get the year for the ISO week than parsing `spent_on`?
- How can I create a test for this bug?
- How can I use the logger in this function? `logger.debug` gives undefined local variable or method ``logger'`

#2 - 2013-03-03 23:40 - Filou Centrinov

Duplicate of [#5329](#)

#3 - 2013-03-04 00:02 - Jean-Philippe Lang

- *Status changed from New to Closed*

- *Resolution set to Duplicate*