

## Redmine - Defect #32046

### SQL to get counts when using created\_on in MySQL to group issues does not work correctly

2019-09-11 02:36 - Mizuki ISHIKAWA

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Go MAEDA	<b>% Done:</b>	0%
<b>Category:</b>	Issues filter	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Affected version:</b>	
<b>Resolution:</b>	Fixed		

**Description**

Reproduction procedure

- Use MySQL
- [Time zone data](#) is loaded (If it is not loaded, time zone conversion will fail.)
- Change user language from UTC to something else
- Group by created\_on at issues/index

[query.result\\_count\\_by\\_group](#) executed in QueriesHelper#grouped\_query\_results returns the following result.

Actual result:

```
{2006-07-18 14:00:00 +0000=>2,  
2019-08-13 14:00:00 +0000=>1,  
2019-08-18 14:00:00 +0000=>1,  
2019-08-23 14:00:00 +0000=>2,  
2019-08-25 14:00:00 +0000=>1,  
2019-08-28 14:00:00 +0000=>3}
```

Expected result:

```
{Wed, 19 Jul 2006=>2,  
Tue, 13 Aug 2019=>1,  
Sun, 18 Aug 2019=>1,  
Fri, 23 Aug 2019=>2,  
Sun, 25 Aug 2019=>1,  
Wed, 28 Aug 2019=>3}
```

The number of records cannot be retrieved properly because the keys format is different than expected.

This issue is related to [#13803](#).

**Related issues:**

Related to Redmine - Feature #13803: Implement grouping issues by date (start...	<b>Closed</b>
--	---------------

#### Associated revisions

##### Revision 18460 - 2019-09-13 00:55 - Go MAEDA

Fix that date grouping for MySQL is not working (#32046, #13803).

Patch by Mizuki ISHIKAWA.

#### History

##### #1 - 2019-09-11 02:37 - Mizuki ISHIKAWA

You have to make changes like the diff below.

```
diff --git a/lib/redmine/database.rb b/lib/redmine/database.rb  
index 979363329..0f0876157 100644  
--- a/lib/redmine/database.rb
```

```
+++ b/lib/redmine/database.rb
@@ -80,7 +80,7 @@ module Redmine
  if time_zone
    user_identifier = ActiveSupport::TimeZone.find_tzinfo(time_zone.name).identifier
    local_identifier = ActiveSupport::TimeZone.find_tzinfo(Time.zone.name).identifier
-   "CONVERT_TZ (DATE (#{column}), '#{local_identifier}', '#{user_identifier}') "
+   "DATE (CONVERT_TZ (#{column}, '#{local_identifier}', '#{user_identifier}')) "
  else
    "DATE (#{column}) "
  end
end
```

#### **#2 - 2019-09-11 03:41 - Go MAEDA**

- *Related to Feature #13803: Implement grouping issues by date (start, due, creation, update, closing dates) added*

#### **#3 - 2019-09-11 08:05 - Go MAEDA**

- *Assignee set to Marius BĂLTEANU*

- *Target version set to 4.1.0*

Marius, could you review this patch? It updates MySQL support for the date grouping filter option committed in [r17745](#).

#### **#4 - 2019-09-12 23:46 - Marius BĂLTEANU**

- *Assignee changed from Marius BĂLTEANU to Go MAEDA*

Go MAEDA wrote:

Marius, could you review this patch? It updates MySQL support for the date grouping filter option committed in [r17745](#).

The fix works well, thanks for catching and fixing this issue.

#### **#5 - 2019-09-13 00:56 - Go MAEDA**

- *Status changed from New to Closed*

- *Target version deleted (4.1.0)*

- *Resolution set to Fixed*

Committed the fix as a part of [#13803](#). Thank you all for working on fixing the issue.