Redmine - Patch #33431
Better performance for Time entries without issue and activity filters

2020-05-11 13:47 - Alexander Meindl

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Performance	**Estimated time:**	0.00 hour
Target version:	Candidate for next major release		

Description

At the moment all time entry queries join issue and activities. If there is a large number of time entries and issues, this is a big performance issue.

Some of our customers have more than 500.000 time entries and even more issues. A query without a time filter requires minutes. With this attached patch this improves it to some seconds.

Issue and activity joins are only add, if they are required.

It would be great, to get this in Redmine.

History

#1 - 2020-05-11 16:09 - Go MAEDA

- Category changed from Time tracking to Performance

- Target version set to Candidate for next major release

#2 - 2020-05-12 02:33 - Go MAEDA

Looks good to me.

#3 - 2020-05-12 06:08 - Alexander Meindl

My patch has problems with column sort order.

But I found the real problem now. It is a missing condition to project_id for Enumeration. I am working on a new patch to fix this problem. Please do not commit the current patch.

#4 - 2020-05-12 07:57 - Marius BALTEANU

Alexander Meindl wrote:

> *My patch has problems with column sort order.*
>
> *But I found the real problem now. It is a missing condition to project_id for Enumeration. I am working on a new patch to fix this problem. Please do not commit the current patch.*

Nice feature, but we should have tests for this change.

#5 - 2020-05-12 14:55 - Alexander Meindl

- File time_query_performace_v2.patch added

Here is my new fixed version.

Adding

```
where("#{Enumeration.table_name}.project_id = #{TimeEntry.table_name}.project_id")
```

fixed the problem. But I moved activity join from base_scope to results_scope, because as I understood it, activity join is not required for totals and count.

I am not sure, how I can test this change with an additional test. Existing test should work, of course.

#6 - 2020-05-12 16:24 - Alexander Meindl

- File time_query_performace_v3.patch added

And again, now all tests are running.

With my test data, /time_entries without filters

- without patch: ActiveRecord: 345038.1ms
- with patch ActiveRecord: 6038.4ms

This is not perfect, but way better than before. At the moment I have no idea how it can be more improved. The problem is table layout of enumerations table.

#7 - 2022-11-24 12:26 - Alexander Meindl

- File time_query_performace_v4.patch added

2 years later, a new patch introduce skipping joins for count time entries, which are not required. Test added.

All tests run successful with this changes.

#8 - 2022-11-24 12:36 - Alexander Meindl

- File time_query_performace_v5.patch added

New version of patch has some optimization for count with activities.

#9 - 2022-11-26 07:22 - Omega Code

on the ther hand (in regard to #37962) this patch helped a lot under Redmine 4.2.8 (mysql 8):
time for time entries queries was reduced approx by 40% for me.
thank you!

Files

time_query_performace.patch	1.64 KB	2020-05-11	Alexander Meindl
time_query_performace_v2.patch	790 Bytes	2020-05-12	Alexander Meindl
time_query_performace_v3.patch	591 Bytes	2020-05-12	Alexander Meindl
time_query_performace_v4.patch	3.97 KB	2022-11-24	Alexander Meindl
time_query_performace_v5.patch	4.03 KB	2022-11-24	Alexander Meindl