Redmine - Patch #33431
Better performance for Time entries without issue and activity filters
2020-05-11 13:47 - Alexander Meindl

Status: New
Priority: Normal
Assignee: Alexander Meindl
Category: Performance
Target version: 6.0.0

Start date: 
Due date: 
% Done: 0%
Estimated time: 0.00 hour

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 BÂLTEANU
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_performance_v2.patch added

Here is my new fixed version.

Adding

```ruby
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
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_performance_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_performance_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.
	hank you!

#10 - 2023-01-14 00:12 - John Ramsden

We have hundreds of thousands of time entries and redmine became almost unusable recently, this patch made a huge difference to us. It would be great if it can be pushed into an upcoming release.

#11 - 2023-01-16 00:34 - Go MAEDA

- Target version changed from Candidate for next major release to 5.1.0

#12 - 2023-05-04 11:59 - sly sly

I have a problem with the v5 patch.

If you move an issue to another project, the project_id no longer matches the request and is ignored when displaying the numbers.

An example:

Issue 2982 belonged to project with ID = 8 when the time entry 34446 was created and then was moved in another project ID = 17

MariaDB [redmine_import]> select te.*
, projects.id, enumerations.project_id, enumerations.type, users.id, users.type, projects.id FROM time_entries AS te
  INNER JOIN `projects` ON `projects`.id = te.project_id
  INNER JOIN enumerations ON enumerations.id = te.activity_id AND enumerations.type = 'TimeEntryActivity'
  INNER JOIN users ON users.id = te.user_id AND users.type IN ('User', 'AnonymousUser')
WHERE te.id = 34446;
+-------------------+-------------------+----+---+-----------+---------------------+---------------------+---------+------+----+--
| id | project_id | author_id | user_id | issue_id | hours | comments             | activity_id |
|----|------------|-----------|---------|----------|-------|---------------------|-------------|--------|----|--
| 34446 | 17 | 12 | 12 | 2982 | 4 | Fix alma container issue for devel | 172 | 2021-10-04 | 2021 | 10 | 40 | 2021-10-05 07:59:26 | 2021-10-05 07:59:26 | NULL | 17 | 172 | 8 | TimeEntryActivity | 12 | User | 17 |
So the DB queries was KO and doesn't display the good numbers, for an hotfix we just removed

2024-06-07
def base_result_scope
    base_scope.joins(:activity, :user)
        .includes(:activity)
        .left_join_issue
    # XXX - remove to let time entry report working fine
    # .where(activity_scope_sql)
end

#13 - 2023-10-21 03:02 - Go MAEDA
- Target version changed from 5.1.0 to 6.0.0

#14 - 2024-05-27 10:14 - Kaustubh Patil
Hello There,
We are facing the issue while loading global spent time. Can the above mentioned patches help us to mitgate our issue. Following are the details of our Redmine Environment:

Environment:
Redmine version                5.1.2.stable
Ruby version                   3.2.4-p170 (2024-04-23) [x86_64-linux]
Rails version                  6.1.7.7
Environment                    production
Database adapter               Mysql2
Mailer queue                   ActiveJob::QueueAdapters::SidekiqAdapter
Mailer delivery                smtp
Redmine settings:
Redmine theme                  Zenmine-510 (includes JavaScript)
SCM:
Subversion                     1.14.2
Mercurial                      6.3.2
Bazaar                         3.3.2
Git                            2.39.2
Filesystem
Redmine plugins:
redmine_agile                  1.6.9
redmine_budgets                1.0.5
redmine_checklists             3.1.23
redmine_contacts               4.4.1
redmine_contacts_helpdesk      4.2.2
redmine_contacts_invoices      4.2.9
redmine_reporter               2.0.4
We have tried installing the time_query_performance_v5.patch but things don't seem to work. Pls guide us on the same to resolve this issue.

#15 - 2024-05-29 07:09 - Kaustubh Patil
Hello there, could anyone pls provide any insights on this.

Files

time_query_performance.patch  1.64 KB  2020-05-11  Alexander Meindl
time_query_performance_v2.patch  790 Bytes  2020-05-12  Alexander Meindl
time_query_performance_v3.patch  591 Bytes  2020-05-12  Alexander Meindl
time_query_performance_v4.patch  3.97 KB  2022-11-24  Alexander Meindl
time_query_performance_v5.patch  4.03 KB  2022-11-24  Alexander Meindl

2024-06-07 3/3