Redmine - Defect #38995

Redmine query is very slow after upgrading from MySql 5.7 to 8

2023-08-29 13:42 - Kushal Singh

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Database	**Estimated time:**	0.00 hour
Target version:			
Resolution:		**Affected version:**	4.1.1

Description

Recently we upgraded the MySql version for Redmine from 5.7 to 8.0.32. After the upgrade the MyPage query is taking a very long time to execute (Around 1 minute). Previously in version 5.7 it used to execute within a few seconds. This is resulting in our CPU usage to reach 99% very quickly.

On initial research looks like MySql 8 uses derived query optimization concept due to which the logic that the query is interpreted in 5.7 and 8 is different.

Redmine Version: 4.1.1.stable (Tried this on latest redmine version 5.0.5 but faced the same issue. Also all the appropriate index have been done as well)

My Page Query:

```
SELECT
  issues.id AS t0_r0,
  issues.tracker_id AS t0_r1,
  issues.project_id AS t0_r2,
  issues.subject AS t0_r3,
  issues.description AS t0_r4,
  issues.due_date AS t0_r5,
  issues.category_id AS t0_r6,
  issues.status_id AS t0_r7,
  issues.assigned_to_id AS t0_r8,
  issues.priority_id AS t0_r9,
  issues.fixed_version_id AS t0_r10,
  issues.author_id AS t0_r11,
  issues.lock_version AS t0_r12,
  issues.created_on AS t0_r13,
  issues.updated_on AS t0_r14,
  issues.start_date AS t0_r15,
  issues.done_ratio AS t0_r16,
  issues.estimated_hours AS t0_r17,
  issues.parent_id AS t0_r18,
  issues.root_id AS t0_r19,
  issues.lft AS t0_r20,
  issues.rgt AS t0_r21,
  issues.is_private AS t0_r22,
  issues.position AS t0_r23,
  issues.remaining_hours AS t0_r24,
  issues.story_points AS t0_r25,
  issues.closed_on AS t0_r26,
  issue_statuses.id AS t1_r0,
  issue_statuses.name AS t1_r1,
  issue_statuses.is_closed AS t1_r2,
  issue_statuses.position AS t1_r3,
  issue_statuses.default_done_ratio AS t1_r4,
  projects.id AS t2_r0,
  projects.name AS t2_r1,
  projects.description AS t2_r2,
  projects.homepage AS t2_r3,
  projects.is_public AS t2_r4,
  projects.parent_id AS t2_r5,
  projects.created_on AS t2_r6,
  projects.updated_on AS t2_r7,
  projects.identifier AS t2_r8,
  projects.status AS t2_r9,
  projects.lft AS t2_r10,
  projects.rgt AS t2_r11,
  projects.inherit_members AS t2_r12,
  projects.default_version_id AS t2_r13,
  projects.default_assigned_to_id AS t2_r14
FROM
  issues
  INNER JOIN projects ON projects.id = issues.project_id
  INNER JOIN issue_statuses ON issue_statuses.id = issues.status_id
  LEFT OUTER JOIN enumerations ON enumerations.id = issues.priority_id
WHERE
  (projects.status <> 9
  AND EXISTS (
    SELECT 1
    FROM enabled_modules em
    WHERE em.project_id = projects.id AND em.name='issue_tracking'
  ))
  AND (
    issues.status_id IN (SELECT id FROM issue_statuses WHERE is_closed=FALSE)
    AND issues.assigned_to_id IN ('1051', '2643')
    AND projects.status IN ('1')
  )
ORDER BY
  enumerations.position DESC,
  issues.updated_on DESC,
  issues.id DESC
LIMIT 10;
```

Explain for MySQL 8.0.32

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	issue_statuses		ALL	PRIMARY				19	100	Using temporary; Using filesort
1	SIMPLE	issue_statuses		eq_ref	PRIMARY,index_issue_statuses_on_is_closed	PRIMARY	4	deermine.issue_statuses.id	1	89.47	Using where
1	SIMPLE	<subquery2>		ALL						100	Using where; Using join buffer (hash join)
1	SIMPLE	projects		eq_ref	PRIMARY	PRIMARY	4	<subquery2>.project_id	1	9	Using where
1	SIMPLE	issues		ref	issues_project_id,index_issues_on_status_id,index_issues_on_assi	issues_project_id	4	<subquery2>.project_id	1917	0.03	Using where

					gned_to_id						
1	SIMPLE	enumerations		eq_ref	PRIMARY,index_enumerations_on_id_and_type	PRIMARY	4	deermine.issues.priority_id	1	100	
2	MATERIALIZED	em		ALL	enabled_modules_project_id				3545	10	Using where

Explain for MySQL 5.7

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	PRIMARY	issues		range	issues_project_id,index_issues_on_status_id,index_issues_on_assigned_to_id	index_issues_on_assigned_to_id	5		2560	100	Using index condition; Using temporary; Using filesort
1	PRIMARY	issue_statuses		eq_ref	PRIMARY,index_issue_statuses_on_is_closed	PRIMARY	4	deermine.issues.status_id	1	89.47	Using where
1	PRIMARY	issue_statuses		eq_ref	PRIMARY	PRIMARY	4	deermine.issues.status_id	1	100	
1	PRIMARY	enumerations		eq_ref	PRIMARY,index_enumerations_on_id_and_type	PRIMARY	4	deermine.issues.priority_id	1	100	
1	PRIMARY	projects		eq_ref	PRIMARY	PRIMARY	4	deermine.issues.project_id	1	9	Using where
2	DEPENDENT SUBQUERY	em		ref	enabled_modules_project_id	enabled_modules_project_id	5	deermine.projects.id	5	10	Using where

As you can see, in 5.7 all the rows are being pulled in single query where as for 8.0.32 it is being pulled in multiple query.

History

#1 - 2023-08-31 11:41 - Kushal Singh

We fixed this issue by turning of the 'materialization' switch off in the optimizer_switch in MySQL 8. After turning off the materialization switch, SQL is not creating temp tables for where exists conditions which is leading to faster execution of the queries.

The execution time drop from 1+ minutes to less than a millisecond.

#2 - 2023-09-20 08:09 - Tobias Blixt

Did you only run set @@optimizer_switch="materialization=off"?

I'm having similar issues after upgrading from Redmine 4 > 5 and Mysql 5 > 8.0.34. Most of the pages seems to be fine, but for some reason, My Page is really chugging.