

## Redmine - Defect #39443

### Invalid statement query error on MSSQL when role filter is used in issues query

2023-10-30 11:01 - Marius BĂLTEANU

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Marius BĂLTEANU	<b>% Done:</b>	0%
<b>Category:</b>	Database	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	5.1.1	<b>Affected version:</b>	
<b>Resolution:</b>	Fixed		

#### Description

The following tests are failing on MS SQL. Unfortunately, I don't have a MS SQL local environment (M1 Mac) to take a quick look to it.

At the same time, maybe it's better to drop support for SQL on Redmine 6.0 and stick only with MySQL and PostgreSQL.

Error:

```
QueryTest#test_assigned_to_role_filter_should_search_assigned_to_for_users_without_the_Role:
Query::StatementInvalid: TinyTds::Error: An expression of non-boolean type specified in a context
where a condition is expected, near ', '.
  app/models/issue_query.rb:424:in `rescue in issues'
  app/models/issue_query.rb:380:in `issues'
  test/unit/query_test.rb:161:in `block in assert_query_result'
  test/unit/query_test.rb:160:in `assert_query_result'
  test/unit/query_test.rb:2898:in `block in <class:QueryTest>'
rails test test/unit/query_test.rb:2894
.....E
```

Error:

```
QueryTest#test_assigned_to_role_filter_should_search_assigned_to_for_users_with_the_Role_on_the_is
ue_project:
Query::StatementInvalid: TinyTds::Error: An expression of non-boolean type specified in a context
where a condition is expected, near ', '.
  app/models/issue_query.rb:424:in `rescue in issues'
  app/models/issue_query.rb:380:in `issues'
  test/unit/query_test.rb:161:in `block in assert_query_result'
  test/unit/query_test.rb:160:in `assert_query_result'
  test/unit/query_test.rb:2883:in `block in <class:QueryTest>'
rails test test/unit/query_test.rb:2877
.....E
```

Error:

```
QueryTest#test_assigned_to_role_filter_should_return_an_empty_set_with_empty_role:
Query::StatementInvalid: TinyTds::Error: An expression of non-boolean type specified in a context
where a condition is expected, near ', '.
  app/models/issue_query.rb:424:in `rescue in issues'
  app/models/issue_query.rb:380:in `issues'
  test/unit/query_test.rb:161:in `block in assert_query_result'
  test/unit/query_test.rb:160:in `assert_query_result'
  test/unit/query_test.rb:2891:in `block in <class:QueryTest>'
rails test test/unit/query_test.rb:2886
.....E
```

Error:

```
QueryTest#test_assigned_to_role_filter_should_search_assigned_to_for_users_with_the_Role:
Query::StatementInvalid: TinyTds::Error: An expression of non-boolean type specified in a context
where a condition is expected, near ', '.
  app/models/issue_query.rb:424:in `rescue in issues'
  app/models/issue_query.rb:380:in `issues'
  test/unit/query_test.rb:161:in `block in assert_query_result'
  test/unit/query_test.rb:160:in `assert_query_result'
  test/unit/query_test.rb:2874:in `block in <class:QueryTest>'
rails test test/unit/query_test.rb:2870
.....E
```

Error:

```
QueryTest#test_assigned_to_role_filter_should_return_issues_with_non_empty_role:
Query::StatementInvalid: TinyTds::Error: An expression of non-boolean type specified in a context
where a condition is expected, near ', '.
  app/models/issue_query.rb:424:in `rescue in issues'
  app/models/issue_query.rb:380:in `issues'
  test/unit/query_test.rb:161:in `block in assert_query_result'
  test/unit/query_test.rb:160:in `assert_query_result'
  test/unit/query_test.rb:2920:in `block in <class:QueryTest>'
rails test test/unit/query_test.rb:2915
```

#### Related issues:

Related to Redmine - Patch #38319: Optimize IssueQuery#sql\_for\_assigned\_to\_ro...

Closed

#### Associated revisions

##### Revision 22417 - 2023-11-11 07:00 - Marius BĂLTEANU

Fix an expression of non-boolean type specified in a context where a condition is expected when MSSQL is used (#39443).

Patch by Pavel Rosický.

##### Revision 22420 - 2023-11-11 08:12 - Marius BĂLTEANU

Merged r22417 from trunk to 5.1-stable (#39443).

#### History

##### #1 - 2023-11-06 21:49 - Pavel Rosický

I'm not a mssql user, but it should be simple to fix... I'll investigate...

##### #2 - 2023-11-06 22:19 - Marius BĂLTEANU

Pavel Rosický wrote in [#note-1](#):

I'm not a mssql user, but it should be simple to fix... I'll investigate...

Thanks Pavel!

##### #3 - 2023-11-07 00:07 - Pavel Rosický

[Marius BĂLTEANU](#) the issue is caused by this optimization [#38319](#)

I couldn't confirm the performance issue on MySQL 8 (all 3 variants behave exactly the same) and I don't have a large postgres db.

in order to fix the issue we can either

1/ revert the original change

2/ or here's an alternative approach

```
SELECT SQL_NO_CACHE
  issues.id
FROM
  issues
WHERE
  EXISTS (
    SELECT
      1
    FROM
      members
      INNER JOIN member_roles ON members.id = member_roles.member_id
    WHERE
      member_roles.role_id = 3
      AND issues.project_id = members.project_id
      AND members.user_id = issues.assigned_to_id)
```

```
diff --git a/app/models/issue_query.rb b/app/models/issue_query.rb
index 235bf2fbe..08240c117 100644
--- a/app/models/issue_query.rb
+++ b/app/models/issue_query.rb
@@ -567,10 +567,10 @@ class IssueQuery < Query
   sw = operator == "!" * ? 'NOT' : ''
   nl = operator == "!" * ? "#{Issue.table_name}.assigned_to_id IS NULL OR" : ''
   subquery =
-   "SELECT DISTINCT #{Member.table_name}.user_id" +
```

```

+     "SELECT 1" +
+     " FROM #{Member.table_name}" +
-     " WHERE #{Member.table_name}.project_id = #{Issue.table_name}.project_id"
-     " (#{nl} #{Issue.table_name}.assigned_to_id #{sw} IN (#{subquery}))"
+     " WHERE #{Issue.table_name}.project_id = #{Member.table_name}.project_id AND #{Member.table_name}.use
r_id = #{Issue.table_name}.assigned_to_id"
+     " (#{nl} #{sw} EXISTS (#{subquery}))"
+     when "=", "!"
+     role_cond =
+     if value.any?
@@ -581,10 +581,10 @@ class IssueQuery < Query
sw = operator == "!" ? 'NOT' : ''
nl = operator == "!" ? "#{Issue.table_name}.assigned_to_id IS NULL OR" : ''
subquery =
-     "SELECT DISTINCT #{Member.table_name}.user_id, #{Member.table_name}.project_id" +
-     " FROM #{Member.table_name}, #{MemberRole.table_name}" +
-     " WHERE #{Member.table_name}.id = #{MemberRole.table_name}.member_id AND #{role_cond}"
-     " (#{nl} (#{Issue.table_name}.assigned_to_id, #{Issue.table_name}.project_id) #{sw} IN (#{subquery}))"
+     "SELECT 1" +
+     " FROM #{Member.table_name} inner join #{MemberRole.table_name} on members.id = member_roles.member_i
d" +
+     " WHERE #{Issue.table_name}.project_id = #{Member.table_name}.project_id AND #{Member.table_name}.use
r_id = #{Issue.table_name}.assigned_to_id AND #{role_cond}"
+     " (#{nl} #{sw} EXISTS (#{subquery}))"
+     end
end

```

I'm not sure if it makes any difference on postgres. Perhaps the original reporter could retest it.

#### #4 - 2023-11-07 08:45 - Marius BĂLTEANU

- Target version set to 5.1.1

#### #5 - 2023-11-07 08:46 - Marius BĂLTEANU

- Related to Patch #38319: Optimize IssueQuery#sql\_for\_assigned\_to\_role\_field for PostgreSQL performance added

#### #6 - 2023-11-07 08:47 - Marius BĂLTEANU

I can confirm that the tests pass now on my custom CI.

Vincent Robert, can you check this fix against your PostgreSQL data?

#### #7 - 2023-11-07 15:53 - Vincent Robert

I have just tested this change with Postgres and a large dataset. Everything ran fine, and performance is good. This simpler code seems to work perfectly.

#### #8 - 2023-11-11 07:02 - Marius BĂLTEANU

- Subject changed from Tests failing on MS SQL to Invalid statement query error on MSSQL

- Category set to Database

- Status changed from New to Resolved

- Assignee set to Marius BĂLTEANU

- Resolution set to Fixed

#### #9 - 2023-11-11 08:12 - Marius BĂLTEANU

- Status changed from Resolved to Closed

#### #10 - 2023-11-11 08:14 - Marius BĂLTEANU

- Subject changed from Invalid statement query error on MSSQL to Invalid statement query error on MSSQL when role filter is used in issues query