Redmine - Patch #36416
Cleanup more dependent objects on project delete
2022-01-11 20:11 - Holger Just

Status: New
Priority: Normal
Assignee: Marius BALTEANU
Category: Database
Target version: 5.0.0

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

Description

On project deletion, there are currently some rows in the database which are not correctly destroyed and thus become orphaned.

With this patch, we will destroy overridden time entry activities (and all of their children) of a project on project destroy.

We now also destroy dependent objects of stored queries for a project on destroy. This will cleanup rows from the habtm join table queries_roles which were also orphaned before.

The attached patch is against current trunk but should also apply to older versions. When we apply this, we may also want to add a migration to cleanup previously orphaned rows in the database.

Associated revisions

Revision 21437 - 2022-02-24 20:44 - Marius BALTEANU
Destroy overridden time entry activities and rows from the habtm join table queries_roles on project deletion (#36416).
Patch by Holger Just.

Revision 21438 - 2022-02-24 20:55 - Marius BALTEANU
Add migration to delete orphaned time entry activities (#36416).
Patch by Holger Just.

Revision 21443 - 2022-02-27 15:06 - Marius BALTEANU
Fix rubocop offense (#36416).

Revision 21444 - 2022-02-28 22:16 - Marius BALTEANU
Add habtm relation between roles and queries_roles (#36416).

History

#1 - 2022-01-12 00:27 - Go MAEDA
- Target version set to Candidate for next major release

#2 - 2022-01-25 18:53 - Holger Just
- File 0001-Cleanup-more-dependent-objects-on-project-delete.patch added
Attached is an updated patch which included a migration to delete potentially orphaned database rows.

#3 - 2022-01-25 18:53 - Holger Just
- File deleted (0001-Cleanup-more-dependent-objects-on-project-delete.patch)

#4 - 2022-01-30 11:41 - Marius BALTEANU
- Target version changed from Candidate for next major release to 5.0.0

#5 - 2022-02-24 21:03 - Marius BALTEANU
- Assignee set to Marius BALTEANU

Holger, thanks for the patches. I've committed most of the changes (r21437 and r21438), but the migration for OrphanedQueriesRoles throws the below error on PostgreSQL after I changed from `DELETE #{queries_roles} FROM #{queries_roles}...` to `DELETE FROM #{queries_roles}...`.

Caused by:

```
PG::SyntaxError: ERROR: syntax error at or near "LEFT"
LINE 1: DELETE FROM queries_roles LEFT OUTER JOIN queries ON queries...
```

#6 - 2022-02-24 21:19 - Marius BALTEANU

What do you think about the following query migration?

```
quaries_roles = "#{Query.table_name_prefix}queries_roles#{Query.table_name_suffix}"
quaries = Query.table_name
ActiveRecord::Base.connection.execute "DELETE FROM #{queries_roles} WHERE query_id IN (SELECT #{queries_roles}.query_id FROM #{queries_roles} LEFT OUTER JOIN #{queries} ON #{queries_roles}.query_id = #{queries}.id WHERE #{queries}.id IS NULL)"
```

it will generate the following query:

```
DELETE FROM queries_roles WHERE query_id IN (
  SELECT queries_roles.query_id FROM queries_roles
  LEFT OUTER JOIN queries ON queries_roles.query_id = queries.id WHERE queries.id IS NULL
)
```

#7 - 2022-02-24 21:31 - Marius BALTEANU

Also, it seems like the habtm join table queries_roles is not clean up when a role is destroyed and we should fix this case as well.

#8 - 2022-02-25 18:30 - Holger Just

Marius BALTEANU wrote:

```
What do you think about the following query migration?

[...]
```
This fails on MySQL (or at least older MySQL versions):

> You can't specify target table 'queries_roles' for update in FROM clause

The version I have proposed works there.

> Also, it seems like the habtm join table queries_roles is not clean up when a role is destroyed and we should fix this case as well.

That is true. It might be enough to add a simple habtm relation to the Role model here. If I'm not mistaken, then Rails will automatically cleanup rows from the join table on destroy.

#9 - 2022-02-28 22:58 - Marius BALTEANU

Holger Just wrote:

> That is true. It might be enough to add a simple habtm relation to the Role model here. If I'm not mistaken, then Rails will automatically cleanup rows from the join table on destroy.

Yes, it works as you said, I've added this change in r21444.

Regarding the migrations, I tested the following queries on both MySQL and PostgreSQL and it works as expected:

Orphaned by query:

```
DELETE FROM queries_roles WHERE query_id NOT IN (SELECT DISTINCT(id) FROM queries);
```  

Orphaned by role:

```
DELETE FROM queries_roles WHERE role_id NOT IN (SELECT DISTINCT(id) FROM roles);
```  

Migration:

```ruby
class DeleteOrphanedQueriesRoles < ActiveRecord::Migration[6.1]
  def self.up
    queries_roles = "#{Query.table_name_prefix}queries_roles#{Query.table_name_suffix}"
    queries = Query.table_name
    roles = Role.table_name

    ActiveRecord::Base.connection.execute "DELETE FROM #{queries_roles} WHERE query_id NOT IN (SELECT DISTINCT(id) FROM #{queries})"
    ActiveRecord::Base.connection.execute "DELETE FROM #{queries_roles} WHERE role_id NOT IN (SELECT DISTINCT(id) FROM #{roles})"
  end

  def self.down
    # no-op
  end
```

2022-03-27
Do you see any issue with this approach?

#10 - 2022-03-18 19:02 - Marius BALTEANU

Holger, did you have a chance to check my last comment?

#11 - 2022-03-24 14:47 - Holger Just

Sorry for the late reply. Your approach is a nice alternative which should probably work fine. I don't see any obvious issues there.

Files

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001-Cleanup-more-dependent-objects-on-project-delete.patch</td>
<td>4.02 KB</td>
<td>2022-01-25</td>
<td>Holger Just</td>
</tr>
</tbody>
</table>