

Redmine - Defect #32487

Copy issue and its subtasks

2019-11-19 11:06 - Dmitry Makurin

Status: Closed	Start date:
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category: Issues	Estimated time: 0.00 hour
Target version:	Affected version:
Resolution:	

Description

When copying an issue with subtasks rgt of root issue is wrong calculating.

Steps to reproduce:

1. Create an issue (let's say issue0)
2. add some subtasks to issue0
3. create new issue from copying issue0 (on copy form mark "Copy subtasks")

Will be created:

- copy of issue0
- copy of issue0's subtasks

And issue0's copy creating with wrong rgt. Redmine then fails to find a tree structure and relate all subtasks to parent (root) issue.

History

#1 - 2019-11-21 02:44 - Go MAEDA

The lft and rgt value of copied issues look good to me. Could you explain more details?

```
sqlite> select id, subject, lft, rgt from issues where root_id = 21;
id      subject    lft    rgt
-----
21      issue0     1      6
22      subtask1   2      3
23      subtask2   4      5
```

#2 - 2019-11-21 08:49 - Dmitry Makurin

- Status changed from New to Resolved

Yeah that looks fine. In our case copied root issue had rgt set to 2 (even thou it has multiple subtasks). Probably something is broken in our environment. I'll try to find out what caused that strange behavior.
Thank you for the reply, Go!

#3 - 2019-11-23 04:16 - Go MAEDA

- Status changed from Resolved to Closed

Actually, parent-child relations are sometimes broken. You can fix it by running the following command in the installation directory of Redmine.

```
bundle e rails r 'Issue.rebuild_tree!' RAILS_ENV=production
```

#4 - 2020-01-24 08:46 - Markus Boremski

Could you please give me a hint? We have the same Problem here.
Tried to rebuild the tree but do get

```
`connect': Access denied for user 'root'@'localhost' (using password: NO)
```

on a bitnami-based installation.

#5 - 2022-01-26 09:07 - Markus Boremski

We have this issue (again) on a redmine 4.2 installation.
Maybe this has to be opened and addressed to someone who can fiddle out the underlying reason.

We observed this while copying a task with multy-level subtasks:

- Task A
 - Task A.a
 - Task A.a.1
 - Task A.a.2
 - Task A.b
 - Task A.b.1
 - Task A.b.2

If we copy this we get this:

- Task B
- Task B.a
 - Task B.a.1
 - Task B.a.2
- Task B.b
 - Task B.b.1
 - Task B.b.2

Task B.a is having Task B as a parent, but is only showing this when in edit-mode.

We decided to delete Task B and replace it by a "handmade" Task C.

~~Now we see even more strange behaviour:~~

~~Task B.a is showing two parent Tasks: Task A and Task C~~

edit: this was a mistake on my side. Task B.a.1 is showing Task B.a and Task C as parrents, what ist totally correct.

Doing a rails runner "Issue.rebuild_tree!" -e production did not solve the situation.

The database-Field root_id still has the id of Task A, what leads to this strange behaviour.