

## Redmine - Defect #644

### Unable to delete issue statuses

2008-02-12 19:52 - Mario Moreira

<b>Status:</b> Closed	<b>Start date:</b> 2008-02-12
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> 0.7	<b>Affected version:</b>
<b>Resolution:</b>	
<b>Description</b>	
<p>Hi,</p> <p>I'm still using redmine 0.5.0 and wanted to give a try to redmine 0.6.3. I'm using PostgreSQL as the redmine database. So I did a dump of the database from one server, and imported to the test server. Configured the version 0.6.3 to use the new database, and did a</p> <pre>rake db:migrate RAILS_ENV="production"</pre> <p>Everything went ok and I started testing the new version (running with webbrick).</p> <p>If I go to Administration and then Issue statuses and create a new issue status, I cannot delete the issue status I just created. I get the message "Unable to delete issue status".</p> <p>Looking at the logs, i find the following:</p> <pre>IssueStatus Load (0.000132)   SELECT * FROM issue_statuses WHERE (issue_statuses."id" = 11) SQL (0.000047)   BEGIN Issue Load (0.000383)   SELECT * FROM issues WHERE (status_id=11) LIMIT 1 Workflow Delete all (0.000195)   DELETE FROM workflows WHERE (old_status_id = 11) IssueStatus Update (0.000125)   UPDATE issue_statuses SET position = (position - 1) WHERE (1 = 1 AND position &gt; 8) IssueStatus Update (0.000000)   PGError: ERROR: null value in column "position" violates not-nul l constraint : UPDATE issue_statuses SET "is_closed" = 'f', "name" = 'klklkl', "position" = NULL, "is_default" = 'f' WHERE "id" = 11 SQL (0.000052)   ROLLBACK Redirected to <a href="http://10.112.64.1:3000/issue_statuses/list">http://10.112.64.1:3000/issue_statuses/list</a></pre> <p>Perhaps this only happens with PostgreSQL.</p>	

#### History

##### #1 - 2008-02-12 20:39 - Thomas Lecavelier

Could you try to reuse the update SQL line in a postgresql terminal, please?

```
UPDATE issue_statuses SET position = (position - 1) WHERE (1 = 1 AND position > 8)
```

I'm rather sure that the problem is (position -1) where position should be already replaced by the old position value. Trying the exact line confirm that Postgresql can't reuse a referential value during an Update command.

Thank you.

##### #2 - 2008-02-12 21:04 - Mario Moreira

No, the problem is not with that statement. If you check the log above you see that the error is with the next statement:

```
UPDATE issue_statuses SET "is_closed" = 'f', "name" = 'klklkl', "position" = NULL, "is_default" = 'f' WHERE "id" = 11
```

Just to confirm that:

```
redmine=# UPDATE issue_statuses SET position = (position - 1) WHERE (1 = 1 AND position > 8);
```

```
UPDATE 0
redmine=# UPDATE issue_statuses SET "is_closed" = 'f', "name" = 'klklkl', "position" = NULL, "is_default" = 'f
' WHERE "id" = 11;
ERROR: null value in column "position" violates not-null constraint
redmine=#
```

### #3 - 2008-02-12 21:32 - Mario Moreira

A bit more of info on this. The rake db:migrate command run without errors, and my schema is in the version 86 as expected:

```
redmine=# SELECT * from schema_info ;
version
-----
      86
(1 row)
```

So i think that migration 76 is not working properly with postgres. If we check table issue\_statuses, we get:

```
redmine=# \d issue_statuses
               Table "public.issue_statuses"
   Column      |          Type          |          Modifiers
-----+-----+-----
 id             | integer                | not null default nextval('issue_statuses_id_seq'::regclass)
 name          | character varying(30) | not null default ''::character varying
 is_closed     | boolean                | not null default false
 is_default    | boolean                | not null default false
 position      | integer                | not null default 1
Indexes:
 "issue_statuses_pkey" PRIMARY KEY, btree (id)
```

I have the last version of the postgres adapter and I'm running PostgreSQL 8.2.4:

```
postgres (0.7.9.2008.01.28, 0.7.1)
  Ruby extension library providing an API to PostgreSQL
```

### #4 - 2008-02-12 22:20 - Jean-Philippe Lang

There was a bug on a few migrations. Some not null constraints were not removed with Postgresql. This is fixed in [r1140](#) and ported in 0.6 stable branch.

You can checkout the trunk or branches/0.6-stable, but you'll have to rerun the migration from your 0.5 database copy. I hope you will enjoy the new features :-)

### #5 - 2008-02-12 22:24 - Jean-Philippe Lang

- Status changed from *New* to *Resolved*

One more thing, you need Rails 2.0. Not null constraint deletion doesn't seem to be handled in previous versions of the activerecord postgresql adapter.

### #6 - 2008-03-30 00:01 - Jean-Philippe Lang

- Status changed from *Resolved* to *Closed*  
 - Target version set to 0.7

### #7 - 2008-06-12 21:08 - Colan Schwartz

- Status changed from *Closed* to *Reopened*  
 - Target version changed from 0.7 to 0.7.1

This still doesn't work on MySQL. I even changed the status on all issues to something other than the one I want to delete, and it still didn't work: "Unable to delete issue status".

### #8 - 2008-06-13 00:06 - Colan Schwartz

After further investigation, it seems as though the status initially named "Closed" (the name can be changed) **cannot** be deleted. I swapped names with a created status, and then I could delete it. Is it really necessary that this status needs to be special (i.e. cannot be deleted)?

### #9 - 2008-06-14 14:15 - Jean-Philippe Lang

- Target version deleted (0.7.1)

Please don't assign ticket to already released versions.

**#10 - 2008-08-25 17:34 - Jean-Philippe Lang**

- *Status changed from Reopened to Closed*

- *Target version set to 0.7*

There is no special behaviour for the "Closed" status.

Please open a new ticket with the information requested in [SubmittingBugs](#) and the exact steps to reproduce if this problem still occurs.