Redmine - Defect #8616

Circular relations

2011-06-15 12:08 - Max Prokopiev

Status:	Closed	**Start date:**	2011-06-15
Priority:	Normal	**Due date:**	
Assignee:	Jean-Baptiste Barth	**% Done:**	100%
Category:	Issues	**Estimated time:**	0.00 hour
Target version:	1.2.1		
Resolution:	Fixed	**Affected version:**	1.2.0

Description

I think I found a bug in redmine, correct me if I'm wrong.

3 issues: 1, 2, 3

two cases:

1. 1 blocked_by 2,3
   2 blocks 1, blocked_by 3
   3 blocks 1,2
2. 1 blocks 3, blocked_by 2
   2 blocks 1, blocked_by 3
   3 blocks 2, blocked_by 1

In second case we have an obvious circular dependency. 1 blocks 3, 3 block 2 and 2 block 2, so we can't close any ticket, but redmine allows me to create this relations.

First case - there isn't any circular relations. My steps in this case:

1. create all tickets
2. create relation 1 blocked_by 2
3. create relation 2 blocked_by 3
4. create relation 1 blocked_by 3 => get an error
   BUT
   I'm still allowed to create relation 3 blocks 1, which is equivalent to relation 1 blocked_by 3

ruby 1.8.7 (2010-06-23) patchlevel 299
rails 2.3.11 (using bitnami redmine stack: http://bitnami.org/stack/redmine)

Associated revisions

Revision 6124 - 2011-06-26 14:47 - Jean-Baptiste Barth

Fixed circular dependencies possibly introduced when using reverse relations, for instance "blocked by" relations (#8616).

History

#1 - 2011-06-23 14:48 - Etienne Massip

- Target version set to Candidate for next minor release

Confirmed on demo site.

Also, after closing one of the 3 issues, its status is still set to New and the new statuses allowed for all of the 3 are the status usually allowed when the issue is closed (and no error message).

Looks like db corruption.

#2 - 2011-06-26 11:17 - Jean-Baptiste Barth

- Status changed from New to 7

- Assignee set to Jean-Baptiste Barth

For "2" : confirmed and wrote a test for it. It's because reverse relations (such as "blocked by") are reversed to their corresponding relation (in this case "blocks") in a before_save callback, which is called **after** validations. The problem is it seems relations should not be reversed if validations failed, there's a specific test for this (test_follows_relation_should_not_be_reversed_if_validation_fails). I'll have a deeper look at it, then at "1" too.

#3 - 2011-06-26 14:47 - Jean-Baptiste Barth

- % Done changed from 0 to 50

The problem comes from r3191 but there isn't any related ticket. The attached test only ensures relations are not reversed if not valid, without saying why they shouldn't. So I committed a minimal fix in r6124.

#4 - 2011-06-26 14:58 - Jean-Baptiste Barth

- Status changed from 7 to Closed

- Target version changed from Candidate for next minor release to 1.2.1

- % Done changed from 50 to 100

- Resolution set to Fixed

The fix is merged for next minor release. I didn't read the 1st case carefully, it should be fixed too. Please re-open if it's not the case, and thanks for reporting.

#5 - 2011-06-27 17:11 - Etienne Massip

Indeed, seems fixed, all went fine.