Redmine - Feature #13064

Use of "SELECT ... LOCK IN SHARE MODE" in ActiveRecord

2013-02-02 14:01 - Toru Haraguchi

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Rails support		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Wont fix			

Description

Symptom

- In following code, our Redmine 2.2.2 instance with a home-made plugin to create a set of tickets, sometimes fails by deadlock detection.
- Not only our system, also other high-loaded "off the shelf" systems may be the case.

```
trunk/lib/plugins/awesome_nested_set/ lib/awesome_nested_set/awesome_nested_set.rb @r10930
414:421
---
      # on creation, set automatically lft and rgt to the end of the tree
      def set_default_left_and_right
        highest_right_row = nested_set_scope(:order => "#{quoted_right_column_name} desc").limit
(1).lock(true).first
        maxright = highest_right_row ? (highest_right_row[right_column_name] || 0) : 0
        # adds the new node to the right of all existing nodes
        self[left_column_name] = maxright + 1
        self[right_column_name] = maxright + 2
      end
---
```

- The code above issues a SELECT ... FOR UPDATE SQL statement, and the statement means an **updatable lock request** against the SQL engine.
- Because the self is not a record already on the table, or at least is not the highest_right_row , updatable lock is not necessary.

Feature Request

- **If a feature is available to express SELECT ... LOCK IN SHARE MODE as like lock(read) in ActiveRecord, please use it, or if not, please interact with whom authoring the superb ActiveRecord and awsome_nested_set to consider having it**.
- Because this issue seems be ActiveRecord and the plugin dependent, I posted this ticket as a feature request.

- However the error is harmless because it just cause an "Internal Error" on screen of the user, and actual table records are protected to be atomic (create all or nothing) by issuing transaction statement, I afraid of complaints of users.

History

#1 - 2013-10-23 21:38 - Thomas Pihl

+1

We seems to have the same issue

#2 - 2021-06-27 03:32 - Go MAEDA

- Status changed from New to Closed

- Resolution set to Wont fix

Since awesome_nested_set was removed in [r13841](#) (Redmine 3.0.0) and the code no longer exists, I am closing this issue.

Please open a new issue if you find problems in the current code (lib/redmine/neseted_set/*).