

Redmine - Defect #2775

Internal server error when db is locked

2009-02-18 08:58 - Damien Couderc

Status:	Closed	Start date:	2009-02-18
Priority:	Urgent	Due date:	
Assignee:	Eric Davis	% Done:	0%
Category:	Database	Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:	Invalid		

**Description**

Hi,  
I use Redmine 0.8.0 with SQLite3 on a Ubuntu based host.  
When using "Repository.fetch\_changesets" as mentioned in RedmineSettings but i come across the following problem :

```
Processing IssuesController#new (for 126.53.112.209 at 2009-02-18 08:30:49) [POST]
  Session ID: 1a0b65e0c83f5211d42c66db007b1416
  Parameters: {"commit"=>"Créer", "project_id"=>"XXX", "action"=>"new", "issue"=>{"start_date"=>"2009-02-18", "estimated_hours"=>"", "priority_id"=>"4", "fixed_version_id"=>"16", "done_ratio"=>"0", "assigned_to_id"=>"8", "subject"=>"XXX", "category_id"=>"13", "tracker_id"=>"4", "due_date"=>"", "description"=>"XXX", "status_id"=>"2"}, "controller"=>"issues", "attachments"=>{"1"=>{"file"=>"", "description"=>""}}}

SQLite3::BusyException (database is locked):
  /usr/lib/ruby/1.8/sqlite3/errors.rb:62:in `check'
  /usr/lib/ruby/1.8/sqlite3/resultset.rb:47:in `check'
  /usr/lib/ruby/1.8/sqlite3/resultset.rb:39:in `commence'
  /usr/lib/ruby/1.8/sqlite3/resultset.rb:29:in `initialize'
  /usr/lib/ruby/1.8/sqlite3/statement.rb:135:in `new'
  /usr/lib/ruby/1.8/sqlite3/statement.rb:135:in `execute'
  /usr/lib/ruby/1.8/sqlite3/database.rb:182:in `execute'
  /usr/lib/ruby/1.8/sqlite3/database.rb:157:in `prepare'
  /usr/lib/ruby/1.8/sqlite3/database.rb:181:in `execute'
  /usr/lib/ruby/1.8/sqlite3/database.rb:581:in `commit'
  /vendor/rails/activerecord/lib/active_record/connection_adapters/sqlite_adapter.rb:160:in `commit_db_transaction'
  /vendor/rails/activerecord/lib/active_record/connection_adapters/sqlite_adapter.rb:372:in `catch_schema_changes'
  /vendor/rails/activerecord/lib/active_record/connection_adapters/sqlite_adapter.rb:160:in `commit_db_transaction'
  /vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/database_statements.rb:78:in `transaction'
  /vendor/rails/activerecord/lib/active_record/transactions.rb:79:in `transaction'
  /vendor/rails/activerecord/lib/active_record/transactions.rb:98:in `transaction'
  /vendor/rails/activerecord/lib/active_record/transactions.rb:106:in `save'
  /vendor/rails/activerecord/lib/active_record/transactions.rb:118:in `rollback_active_record_state!'
  /vendor/rails/activerecord/lib/active_record/transactions.rb:106:in `save'
  /app/controllers/issues_controller.rb:144:in `new'
  /vendor/rails/actionpack/lib/action_controller/base.rb:1166:in `send'
  /vendor/rails/actionpack/lib/action_controller/base.rb:1166:in `perform_action_without_filters'
  ,
  /vendor/rails/actionpack/lib/action_controller/filters.rb:579:in `call_filters'
  /vendor/rails/actionpack/lib/action_controller/filters.rb:572:in `perform_action_without_benchmark'
  /vendor/rails/actionpack/lib/action_controller/benchmarking.rb:68:in `perform_action_without_rescue'
  /usr/lib/ruby/1.8/benchmark.rb:293:in `measure'
  /vendor/rails/actionpack/lib/action_controller/benchmarking.rb:68:in `perform_action_without_rescue'
  /vendor/rails/actionpack/lib/action_controller/rescue.rb:201:in `perform_action_without_caching'
```

```
g'
/vendor/rails/actionpack/lib/action_controller/caching/sql_cache.rb:13:in `perform_action'
/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/query_cache.rb:33:in
`cache'
/vendor/rails/activerecord/lib/active_record/query_cache.rb:8:in `cache'
/vendor/rails/actionpack/lib/action_controller/caching/sql_cache.rb:12:in `perform_action'
/vendor/rails/actionpack/lib/action_controller/base.rb:529:in `send'
/vendor/rails/actionpack/lib/action_controller/base.rb:529:in `process_without_filters'
/vendor/rails/actionpack/lib/action_controller/filters.rb:568:in `process_without_session_manag
gement_support'
/vendor/rails/actionpack/lib/action_controller/session_management.rb:130:in `process'
/vendor/rails/actionpack/lib/action_controller/base.rb:389:in `process'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:149:in `handle_request'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:107:in `dispatch'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:104:in `synchronize'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:104:in `dispatch'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:120:in `dispatch_cgi'
/vendor/rails/actionpack/lib/action_controller/dispatcher.rb:35:in `dispatch'
/usr/lib/ruby/1.8/mongrel/rails.rb:76:in `process'
/usr/lib/ruby/1.8/mongrel/rails.rb:74:in `synchronize'
/usr/lib/ruby/1.8/mongrel/rails.rb:74:in `process'
/usr/lib/ruby/1.8/mongrel.rb:159:in `process_client'
/usr/lib/ruby/1.8/mongrel.rb:158:in `each'
/usr/lib/ruby/1.8/mongrel.rb:158:in `process_client'
/usr/lib/ruby/1.8/mongrel.rb:285:in `run'
/usr/lib/ruby/1.8/mongrel.rb:285:in `initialize'
/usr/lib/ruby/1.8/mongrel.rb:285:in `new'
/usr/lib/ruby/1.8/mongrel.rb:285:in `run'
/usr/lib/ruby/1.8/mongrel.rb:268:in `initialize'
/usr/lib/ruby/1.8/mongrel.rb:268:in `new'
/usr/lib/ruby/1.8/mongrel.rb:268:in `run'
/usr/lib/ruby/1.8/mongrel/configurator.rb:282:in `run'
/usr/lib/ruby/1.8/mongrel/configurator.rb:281:in `each'
/usr/lib/ruby/1.8/mongrel/configurator.rb:281:in `run'
/usr/bin/mongrel_rails:129:in `run'
/usr/lib/ruby/1.8/mongrel/command.rb:212:in `run'
/usr/bin/mongrel_rails:282
```

Rendering /var/lib/redmine/redmine-0.8.0/public/500.html (500 Internal Server Error)

This happens not only in issues but also in wiki edition. Of course you lose all the stuff you've typed before submitting which is fun when you just done 3 or 4 screens ;)

## History

### #1 - 2009-02-18 10:00 - Damien Couderc

I forgot to say i suspected the change set fetching because the problem occurred at 8:30 and that i have a crontab that do the fetching every 30 minutes (00 and 30).

And as far as i know, it could also happen with autofetch enabled if somebody is submitting changes when someone else is entering the repository browser (as fetch\_changesets is called in the same manner).

### #2 - 2009-02-19 06:52 - Eric Davis

- Status changed from New to Closed

- Resolution set to Invalid

This is a common occurrence with Sqlite because it has to lock the database whenever it writes. In development this is fine but once there is more than one process writing to the database, you'll get these errors. I'm guessing your Redmine was in use when the crontab triggered so you had 2 processes and caused the deadlock.

I'd suggest moving to MySQL or postgres.

### #3 - 2009-02-19 08:02 - Damien Couderc

- Status changed from Closed to Reopened

- Assignee set to Eric Davis

Ok but i can't find any documentation about database migration in Redmine.  
How to do that ?

**#4 - 2009-02-19 20:04 - Jean-Philippe Lang**

- Status changed from Reopened to Closed

The problem was mentionned in this old Rails ticket:  
<http://dev.rubyonrails.org/ticket/6126>

You can try to use the timeout option in your database configuration file to see if it solves your problem.

Concerning the migration to MySQL, it's in no way specific to Redmine. You should be able to find some usefull information using Google.  
I suggest you to create your MySQL database structure using rake db:migrate then load your data from your SQLite database.

**#5 - 2009-02-20 02:07 - Eric Davis**

Damien Couderc wrote:

Ok but i can't find any documentation about database migration in Redmine.  
How to do that ?

I started with Sqlite3 also and used yamldb by Adam Wiggins to move to MySQL. It might work for you:

[http://blog.heroku.com/archives/2007/11/23/yamldb\\_for\\_databaseindependent\\_data\\_dumps/](http://blog.heroku.com/archives/2007/11/23/yamldb_for_databaseindependent_data_dumps/)

**#6 - 2009-02-20 08:43 - Damien Couderc**

Thank you both for you help.

Jean-Philippe, i know that migration problems are not specific to Redmine but the data model is.  
I also want to add that i was doing the same thing with Trac and SQLite without locking problem so there must be a way to do it.

Anyway i'll try to move to a real db and i hope to get some time to make a documentation if you get some interest for it.

Have fun !

Cheers,  
Damien