

## Redmine - Feature #12097

### Multi Thread Support

2012-10-15 03:32 - Daniel Morgan

<b>Status:</b> Closed	<b>Start date:</b>
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b> Rails support	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> 3.0.0	
<b>Resolution:</b> Fixed	

**Description**

Are there any plans for making Redmine thread safe in the future?  
This is one of very important advantages of using Redmine with JRuby.

I have managed to enable config.threadsafe! with Redmine 1.4.x and gain thread safety to some extent.  
However, I am really struggling to get it working with 2.x. (Mainly related to dependency loading disabled)

Will the Developers please consider thread safety in a future release.  
My understanding is that config.threadsafe! will be a default in Rails 4.

For starters, the following fix was required in "app/models/user.rb" with 1.4.x:

```
def self.current=(user)
  # @current_user = user
  Thread.current[:current_user] = user
end

def self.current
  # @current_user ||= User.anonymous
  Thread.current[:current_user] ||= User.anonymous
end
```

This was to prevent current\_user set by a user performing a bulk edit being overridden by a different user performing a concurrent request.

**Related issues:**

Related to Redmine - Feature #14534: Upgrade to Rails 4.2	<b>Closed</b>
---	---------------

#### Associated revisions

##### Revision 10909 - 2012-12-02 19:06 - Jean-Philippe Lang

Explicitly load dependencies for when running with config.threadsafe! (#12097).

##### Revision 10910 - 2012-12-02 19:07 - Jean-Philippe Lang

Make the tests pass when config.threadsafe! is enabled (#12097).

##### Revision 10911 - 2012-12-02 19:15 - Jean-Philippe Lang

Makes User.current thread safe (#12097).

##### Revision 10912 - 2012-12-02 19:32 - Jean-Philippe Lang

Fixed dependency loading for when running in dev mode (#12097).

##### Revision 10915 - 2012-12-02 20:15 - Jean-Philippe Lang

Load RMagick before the gantt helper (#12097).

##### Revision 10944 - 2012-12-05 22:59 - Jean-Philippe Lang

Prevents redmine/scm/base from being reloaded in dev mode (#12097).

**Revision 12427 - 2013-12-21 12:08 - Toshi MARUYAMA**

make IssuesHelperTest passes when config.threadsafe! is enabled (#12097)

**Revision 12428 - 2013-12-21 12:13 - Toshi MARUYAMA**

make WatchersHelperTest passes when config.threadsafe! is enabled (#12097)

**Revision 13170 - 2014-05-25 16:38 - Toshi MARUYAMA**

fix ProjectsHelperTest fails when config.threadsafe! is enabled (#12097)

**History**

---

**#1 - 2012-12-02 19:15 - Jean-Philippe Lang**

- Target version set to Candidate for next minor release

The dependency loading issue should be fixed by [r10909](#). After that, Redmine starts/runs fine with config.threadsafe!.

**#2 - 2013-02-06 10:06 - Daniel Felix**

[Daniel Lopez](#) Morgan

Can you confirm, that this revisions fix your problem?

Jean-Philippe Lang wrote:

The dependency loading issue should be fixed by [r10909](#). After that, Redmine starts/runs fine with config.threadsafe!.

**#3 - 2013-02-07 04:00 - Daniel Morgan**

@Jean-Philippe Lang

Thank you very much for considering some work on this.

[Daniel Lopez](#) Felix

A quick check hasn't resulted in any issues.

However, I do get migration errors with config.threadsafe! set.

Error details using both [r10909](#) and [r10944](#):

```
E:\torquebox\redmine-trunk>jruby -S rake --trace db:migrate RAILS_ENV=production

** Invoke db:migrate (first_time)
** Invoke environment (first_time)
** Execute environment
** Invoke db:load_config (first_time)
** Execute db:load_config
** Execute db:migrate
== Setup: migrating =====
:
:
:

== AddIssueStatusPosition: migrating =====
-- add_column(:issue_statuses, :position, :integer, {:default=>1})
-> 0.0510s
-> 0 rows
rake aborted!
An error has occurred, all later migrations canceled:

uninitialized constant AddIssueStatusPosition::IssueStatus
org/jruby/RubyModule.java:2677:in `const_missing'
org/jruby/RubyMethod.java:134:in `call'
E:/torquebox/redmine-trunk/db/migrate/019_add_issue_status_position.rb:4:in `up'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:370:in `up'
org/jruby/RubyKernel.java:2069:in `send'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:410:in `migrate'
E:/torquebox/jruby/lib/ruby/1.8/benchmark.rb:293:in `measure'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:410:in `migrate'
```

```

E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/connection_adapters/abstract/connection_pool.rb:129:in `with_connection'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:389:in `migrate'
org/jruby/RubyKernel.java:2073:in `send'
E:0:in `migrate'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:720:in `migrate'
org/jruby/RubyProc.java:261:in `call'
org/jruby/RubyProc.java:209:in `call'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:777:in `ddl_transaction'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:719:in `migrate'
org/jruby/RubyArray.java:1613:in `each'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:700:in `migrate'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:570:in `up'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/migration.rb:551:in `migrate'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/activerecord-3.2.9/lib/active_record/railties/databases.rake:179:in `(root)'
org/jruby/RubyProc.java:261:in `call'
org/jruby/RubyProc.java:209:in `call'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/task.rb:228:in `execute'
org/jruby/RubyArray.java:1613:in `each'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/task.rb:223:in `execute'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/task.rb:166:in `invoke_with_call_chain'
E:/torquebox/jruby/lib/ruby/1.8/monitor.rb:191:in `mon_synchronize'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/task.rb:159:in `invoke_with_call_chain'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/task.rb:152:in `invoke'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:143:in `invoke_task'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:101:in `top_level'
org/jruby/RubyArray.java:1613:in `each'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:101:in `top_level'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:110:in `run_with_threads'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:95:in `top_level'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:73:in `run'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:160:in `standard_exception_handling'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/lib/rake/application.rb:70:in `run'
E:/torquebox/jruby/lib/ruby/gems/shared/gems/rake-10.0.3/bin/rake:33:in `(root)'

org/jruby/RubyKernel.java:1041:in `load'
E:/torquebox/jruby/bin/rake:23:in `(root)'
Tasks: TOP => db:migrate

E:\torquebox\redmine-trunk>

```

#### #4 - 2013-04-24 22:06 - Djordjije Crni

Maybe this can help:

[Redmine code cleaning for config.threadsafe!](#)

#### #5 - 2013-08-09 18:22 - Numa Schmeder

I have a working threadsafe version running on Jruby since a few months. It's working like a charm and is very fast.

I had to modify many, many files to make it work. I still have the sources and the modified files. If anyone is ready to build a patch, I can post the modified files (I don't have any time right now to do it).

One would have to compare each file with the current trunk version and create patches from it or at least a change log.

Regards,

#### #6 - 2013-08-19 02:50 - Daniel Morgan

[Numa Droz](#) Schmeder

Could you possibly upload your modified version of redmine onto GitHub or somewhere similar?  
(with information of which revision you based your modifications on)

**#7 - 2013-12-21 03:30 - Toshi MARUYAMA**

- *Blocks Feature #14534: Upgrade to Rails 4.2 added*

**#8 - 2013-12-21 03:44 - Toshi MARUYAMA**

- *File unit-test-r12424.log.txt added*

- *File config.threadsafepatch-r12424.diff added*

These are patch and unit test log.  
Functional tests have many errors.

**#9 - 2014-05-02 10:11 - Christopher Mann**

- *File rails4threadsafepatch.diff added*

Please disregard previous file.

The current attached file has changes to about 20 files. This is Numa's correct Diff.

Here is Numa Schmeder's modification to make Redmine work in threadsafe mode in Rails 4 (see attached file).

It mainly has to do with added require statements.

**#10 - 2014-05-02 13:04 - Christopher Mann**

- *File threadsafe\_redmine\_patch.diff added*

Please disregard previous patch file and consider this one for Rails 4 threadsafe implementation.

**#11 - 2014-05-26 06:23 - Toshi MARUYAMA**

- *File thread-r13171.diff added*

This is patch for [r13171](#).

Tests on Travis.

<https://travis-ci.org/marutosi/redmine/builds/25998010>

Only one test fails.

```
1) Failure:
test_macro_include(Redmine::WikiFormatting::MacroTest)
  [test/test_helper.rb:185:in `assert_include'
   test/unit/lib/redmine/wiki_formatting/macros_test.rb:199:in `test_macro_include']:
"This is a link to a ticket" not found in "<p><div class="flash error">Error executing the <strong>include</strong> macro (Circular inclusion detected)</div></p>".
<false> is not true.
```

Christopher Mann wrote:

Please disregard previous patch file and consider this one for Rails 4 threadsafe implementation.

Sorry, I cannot understand your patch means.  
Rails4 porting issue is [#14534](#).

**#12 - 2014-05-26 08:13 - Toshi MARUYAMA**

- *File thread-r13171-1.diff added*

Sorry, note-11 patch has duplicate code.  
This is new patch.

Test results are same with note-11.

<https://travis-ci.org/marutosi/redmine/builds/26022443>

**#13 - 2014-10-22 09:34 - Toshi MARUYAMA**

- Blocks deleted (Feature #14534: Upgrade to Rails 4.2)

**#14 - 2014-10-22 09:35 - Toshi MARUYAMA**

- Related to Feature #14534: Upgrade to Rails 4.2 added

**#15 - 2014-10-22 09:35 - Toshi MARUYAMA**

- Target version changed from Candidate for next minor release to 3.0.0

**#16 - 2015-02-18 13:14 - Jean-Philippe Lang**

- Assignee set to Toshi MARUYAMA

Can we close this?

**#17 - 2015-02-18 13:20 - Toshi MARUYAMA**

- Status changed from New to Closed

- Assignee deleted (Toshi MARUYAMA)

[http://edgeguides.rubyonrails.org/4\\_0\\_release\\_notes.html#general](http://edgeguides.rubyonrails.org/4_0_release_notes.html#general)

Thread safe by default - Rails can run in threaded app servers without additional configuration

**#18 - 2015-02-18 18:18 - Toshi MARUYAMA**

- Resolution set to Fixed

I forgot to change resolution.

**Files**

---

config.threadsafepatch-r12424.diff	1.38 KB	2013-12-21	Toshi MARUYAMA
unit-test-r12424.log.txt	7.66 KB	2013-12-21	Toshi MARUYAMA
rails4threadsafepatch.diff	2.98 KB	2014-05-02	Christopher Mann
threadsafepatch_redmine_patch.diff	27.3 KB	2014-05-02	Christopher Mann
thread-r13171.diff	2.77 KB	2014-05-26	Toshi MARUYAMA
thread-r13171-1.diff	2.29 KB	2014-05-26	Toshi MARUYAMA