Redmine - Defect #30211
Crash with the message "undefined method `alias_method_chain'" when connecting to Microsoft SQL Server
2018-12-14 02:24 - Go MAEDA

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>Database</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Resolution:</td>
<td>Fixed</td>
</tr>
<tr>
<td>Start date:</td>
<td></td>
</tr>
<tr>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Affected version:</td>
<td>4.0.0</td>
</tr>
</tbody>
</table>

Description

The problem has been reported in #30209#note-1 by Nigel Wilson.

I ran into this issue as well. I made the following changes in the GemFile to meet all the dependencies. However, I am using MS SQL Server, so you will need to find the corresponding adapter for mysql.

Change: gem "rails", "5.2.2"
To: gem "rails", :git => "https://github.com/rails/rails.git", :branch => "5-2-2"

Change: gem "tiny_tds", "~> 1.0.5"
To: "tiny_tds", "~> 2.1.2"

Change: gem "activerecord-sqlserver-adapter",

Related issues:
Related to Redmine - Defect # 30285: Microsoft SQL server support is broken
Closed

History
#1 - 2018-12-14 02:28 - Go MAEDA
- Related to Defect #30210: Application crash when deleting an issue from issue list with SQLServer added

#2 - 2018-12-14 14:40 - Nigel Wilson

Hi Go MAEDA,

I will try to get you a crash log for this in a day or two. Note that this error (undefined method `alias_method_chain') occurred when trying to install Redmine v4 from git master as an upgrade from v2.5.2. Requirements (Ruby 2.5 and Rails 5.2.2) were confirmed installed.

- replace /ruby19 folder with previously installed version of ruby.
- Go into the redmine web folder created by the Helicon tool and edit the deploy.rb file:
  - set the REDMINE variable to the version of redmine you want to install. e.g.: REDMINE = "master"
  - prepend all the rake commands with bundle exec. e.g: bundle exec rake --trace db:create
- Go back to Helicon installer and click finish.

#3 - 2018-12-15 02:15 - Go MAEDA
- Status changed from New to Confirmed
- Target version set to Candidate for next minor release

2020-05-04 1/3
Confirmed the problem. Redmine crashes during startup.

```
ubuntu@redmine-trunk:~/redmine-trunk$ bin/rails c
Traceback (most recent call last):
 26: from bin/rails:4:in `<main>'
 25: from bin/rails:4:in 'require'
 24: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/commands.rb:18:in `<top (required)>'
 23: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/command.rb:46:in `invoke'
 22: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/command/base.rb:65:in `perform'
 21: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/thor-0.20.3/lib/thor.rb:387:in `dispatch'
 20: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/thor-0.20.3/lib/thor/invocation.rb:126:in `invoke_command'
 19: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/thor-0.20.3/lib/thor/invocation.rb:120:in `run'
 18: from
/home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/commands/console/console_command.rb:95:in `perform'
 17: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/command/actions.rb:17:in `require_application_and_environment'
 16: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/railties-5.2.2/lib/rails/commands/actions.rb:17:in `require'
 15: from /home/ubuntu/redmine-trunk/config/application.rb:5:in `<top (required)'
 14: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler.rb:114:in `require'
 13: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:65:in `require'
 12: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:65:in `each'
 11: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:76:in `block in require'
 10: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:76:in `each'
 9: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:81:in `block (2 levels) in require'
 8: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/2.6.0/bundler/runtime.rb:81:in `require'
 7: from
/home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/activerecord-sqlserver-adapter.rb:1:in `<top (required)>'
 6: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/activerecord-sqlserver-adapter.rb:1:in `<top (required)>'
 5: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter.rb:3:in `<top (required)>'
 4: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter.rb:3:in `require'
 3: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter/core_ext/active_record.rb:145:in `<top (required)>'
 2: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter/core_ext/active_record.rb:145:in `include'
 1: from /home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter/core_ext/active_record.rb:12:in `included'
/home/ubuntu/.rbenv/versions/2.6.0-rc1/lib/ruby/gems/2.6.0/gems/activerecord-sqlserver-adapter-2.3.8/lib/active_record/connection_adapters/sqlserver_adapter/core_ext/active_record.rb:13:in `singleton class': undefined method `alias_method_chain' for #<Class:ActiveRecord::Base> (NoMethodError)
```
We have to wait for the release of the latest version of activerecord-sqlserver-adapter that supports Rails 5.2 to resolve this issue.

https://github.com/rails-sqlserver/activerecord-sqlserver-adapter/issues/636

- Subject changed from Crash when using Microsoft SQL Server to Crash with the message "undefined method `alias_method_chain" when connecting to Microsoft SQL Server

- Related to Feature #23630: Migrate to Rails 5.2 added

- Related to Defect #30285: Microsoft SQL server support is broken added

- Related to deleted (Defect #30210: Application crash when deleting an issue from issue list with SQLServer)

- Related to deleted (Feature #23630: Migrate to Rails 5.2)

- Target version deleted (Candidate for next minor release)

- Status changed from Confirmed to Closed
- Resolution set to Fixed

The issue is going to be fixed in the upcoming Redmine 4.1.1 and 4.0.7 (#30285).