RubyGems 1.5.0 was released yesterday.

I attempted to install Redmine 1.1.1 using RVM, Ruby 1.8.7p330 and Rails 2.3.5.

rake db:migrate failed with the following error message:

rake aborted!
undefined local variable or method `version_requirements' for #<Rails::GemDependency:0xb7351b0c>

Here's the relevant Rails ticket:


Related issues:
Related to Redmine - Defect # 8325: undefined method `name' for "actionmailer... Closed 2011-05-09
Blocked by Redmine - Feature # 6887: Upgrade to Rails 2.3.11 Closed 2010-11-13

History

#1 - 2011-02-03 02:00 - Anonymous
I have the exact same issue - can I somehow use an older ruby gems version or is there some other work around to get the redmine update until this is fixed?

#2 - 2011-02-03 09:47 - Oto Brglez
I have the same problem. Redmine 1.1, Rails 2.3.5, Ruby 1.8.7, RubyGems 1.5.0.

Perhaps work around is to downgrade RubyGems?

#3 - 2011-02-03 09:51 - Oto Brglez
Phillip Koebbe from Ruby on Rails suggested inserting following code between bootstrap and initialize section of enviroment.rb. And it works.

```ruby
if Gem::VERSION >= "1.3.6"
  module Rails
    class GemDependency
      def requirement
        r = super
        (r == Gem::Requirement.default) ? nil : r
      end
    end
  end
end
```
#4 - 2011-02-07 21:28 - Andrew Vit

Confirmed that this hack fixes the problem. Would it make sense to check this into the codebase for the time being, until the updates for rails 2.3.latest are resolved?

#5 - 2011-02-19 09:51 - Jose Torres

This problem persists through Ruby-gems 1.5.2 with latest git commit: https://github.com/edavis10/redmine/commit/824adaf30c55695c08ebd5456923ba573344b3e9

```ruby
atomic redmine # RAILS_ENV=production rake db:migrate

rake aborted!
undefined local variable or method `version_requirements' for #<Rails::GemDependency:0x7f1bdb0f24d8>
```

(See full trace by running task with --trace)

```ruby
atomic redmine # RAILS_ENV=production rake db:migrate --trace

rake aborted!
undefined local variable or method `version_requirements' for #<Rails::GemDependency:0x7f6f5ec058a8>
```

```
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/rails/gem_dependency.rb:119:in `requirement'
/usr/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/rails/gem_dependency.rb:73:in `add_load_paths'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:301:in `add_gem_load_paths'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:301:in `each'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:301:in `add_gem_load_paths'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:132:in `process'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:113:in `sendf'
/usr/local/lib64/ruby/gems/1.8/gems/rails-2.3.5/lib/initializer.rb:113:in `run'
/var/www/redmine/config/environment.rb:20
```

```
2020-06-10
2/8
```
With trunk updated to Rails 2.3.11, I upgraded my rubygems to 1.5.2 and everything works fine. Does anybody still have issues with current trunk, or can we close this issue?

Could you please back-port to redmine-1.1.1, or should I use 1.2.0 or current?

thank you very much!

This error has been resolved with rails upgraded to 2.3.11. Thanks Jean-Baptise Barth.

@moto: you'll have to wait 1.2.0 unless you're working with trunk. 1.2.0 should be released in about 2 months, not that far from now :) If you can't wait,

---

2020-06-10

3/8
you can try Oto's patch.

@Jose: no problem, we should thank Jean-Philippe for that though.

I close the issue, please re-open if you have problems with rubygems 1.5+ on trunk or after 1.2.0 release.

#10 - 2011-04-06 16:21 - Alex Q

Jean-Baptiste Barth wrote:

With trunk updated to Rails 2.3.11, I upgraded my rubygems to 1.5.2 and everything works fine. Does anybody still have issues with current trunk, or can we close this issue?

I still I've this issue

Opensuse11.4 x86_64

```
/srv/www/htdocs/redmine # rpm -qa |grep ruby
rubygem-rake-0.8.7-8.4.x86_64
rubygem-activerecord-2_3-2.3.11-1.2.4.x86_64
rubygems-1.5.2-22.1.x86_64
ruby-1.8.7.p334-1.2.2.x86_64
rubygem-rack-1.1.0-7.4.x86_64
rubygem-activeresource-2_3-2.3.11-1.2.4.x86_64
rubygem-actionpack-2_3-2.3.11-1.2.4.x86_64
rubygem-actionpack-2_3-2.3.11-1.2.4.x86_64
rubygem-rails-2_3-2.3.11-1.2.4.x86_64
ruby-devel-1.8.7.p334-1.2.2.x86_64
rubygem-activesupport-2_3-2.3.11-1.2.4.x86_64
rubygem-actionmailer-2_3-2.3.11-1.2.4.x86_64
```

```
/srv/www/htdocs/redmine # rpm -qa |grep gem
rubygem-rake-0.8.7-8.4.x86_64
rubygem-activerecord-2_3-2.3.11-1.2.4.x86_64
rubygems-1.5.2-22.1.x86_64
rubygem-rack-1.1.0-7.4.x86_64
rubygem-activeresource-2_3-2.3.11-1.2.4.x86_64
rubygem-actionpack-2_3-2.3.11-1.2.4.x86_64
rubygem-actionpack-2_3-2.3.11-1.2.4.x86_64
rubygem-rails-2_3-2.3.11-1.2.4.x86_64
rubygem-activesupport-2_3-2.3.11-1.2.4.x86_64
rubygem-actionmailer-2_3-2.3.11-1.2.4.x86_64
```

gem list

- LOCAL GEMS ***

- actionmailer (2.3.11)
- actionpack (2.3.11)
- activerecord (2.3.11)
- activeresource (2.3.11)
- activesupport (2.3.11)

2020-06-10
My error:

```
/srv/www/htdocs/redmine # RAILS_ENV=production rake db:migrate --trace
```

```
 rake aborted!
undefined local variable or method `version_requirements' for #<Rails::GemDependency:0x7f75411a2490>
```

```
/rails (2.3.11)
rake (0.8.7)
```

2020-06-10
#11 - 2011-04-15 18:24 - Etienne Massip
- Category set to Gems support

#12 - 2011-05-16 20:58 - Evgeny Seliverstov
- Status changed from Closed to Reopened

 Doesn't work for me with trunk (r5798), rails 2.3.11, rubygems 1.8.2, RedCloth (4.0.4, 3.0.4).
 It is very similar to #8325.

Failed with rake db:migrate and rake gems tasks:

```bash
$ rake gems RAILS_ENV=production --trace
  ** Invoke gems (first_time)
  ** Invoke gems:base (first_time)
  ** Execute gems:base
  ** Invoke environment (first_time)
  ** Execute environment
  rake aborted!
undefined method `name' for "RedCloth":String
```

/usr/local/www/.gem/ruby/1.8/gems/rails-2.3.11/lib/rails/gem_dependency.rb:277:in `=='
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:217:in `matching_specs'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/custom_require.rb:36:in `find_all'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/specification.rb:403:in `each'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:216:in `find_all'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/specification.rb:402:in `each'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:238:in `to_specs'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:256:in `to_spec'
/usr/local/lib/ruby/site_ruby/1.8/rubygems.rb:1182:in `gem'
/usr/local/www/.gem/ruby/1.8/gems/rails-2.3.11/lib/rails/gem_dependency.rb:75:in `add_load_paths'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:301:in `each'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:301:in `add_gem_load_paths'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:132:in `send'
/usr/local/lib/ruby/site_ruby/1.8/rubygems/dependency.rb:113:in `run'
/usr/local/www/redmine/config/environment.rb:20
```

2020-06-10
I too am having the same problem with the same setup. This is a piece of Rails 2.x code on a newly built osx platform:

- rails (2.3.11, 2.3.8)
- RedCloth (4.2.7, 4.0.2)
- rake (0.8.7)

I tried the code above. no joy.

Related RubyGems bug tracker entry is here.

# Status changed from Reopened to Closed
Closing this ticket since the problem was about rubygems 1.5.0, not 1.8.