

Redmine - Feature #31500

Ruby 2.7 support

2019-06-03 02:56 - Go MAEDA

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Go MAEDA	% Done:	0%
Category:	Ruby support	Estimated time:	0.00 hour
Target version:	4.2.0		
Resolution:	Fixed		

Description

Ruby 2.7.0 is going to be released on 2019-12-25.
<https://bugs.ruby-lang.org/versions/35>

Related issues:

Related to Redmine - Patch #32527: Fix ruby 2.7 warning: given argument is ni...	Closed
Related to Redmine - Patch #32542: Fix ruby 2.7 warning: The last argument is...	Closed
Related to Redmine - Defect #32752: Ruby 2.7: Remove deprecated URI.escape/un...	Closed
Related to Redmine - Patch #32906: Update i18n (~> 1.8.2)	Closed
Related to Redmine - Patch #32907: Update capybara (~> 3.31.0)	Closed
Related to Redmine - Defect #33021: [v.4.0.5-stable] Internal Server Error 50...	Closed
Related to Redmine - Defect #35135: FrozenError when new LDAP users try to login	Closed
Related to Redmine - Defect #35441: Inline image in Textile is not displayed ...	Closed
Copied from Redmine - Feature #30118: Ruby 2.6 support	Closed

Associated revisions

Revision 19324 - 2019-11-30 06:40 - Go MAEDA

Ruby 2.7: Regexp#match and Regexp#match? with a nil argument are deprecated (#32527, #31500).

Revision 19454 - 2020-01-23 16:42 - Go MAEDA

Fix ruby 2.7 warning: The last argument is used as the keyword parameter (#32542, #31500).

Patch by Seiei Miyagi.

Revision 19455 - 2020-01-24 00:28 - Go MAEDA

Ruby 2.7: Remove deprecated URI.escape/unescape (#32752).

Patch by Go MAEDA.

Revision 20150 - 2020-10-19 02:34 - Go MAEDA

Ruby 2.7 support (#31500).

History

#1 - 2019-06-03 02:56 - Go MAEDA

- Copied from Feature #30118: Ruby 2.6 support added

#2 - 2019-11-28 23:06 - Go MAEDA

- Related to Patch #32527: Fix ruby 2.7 warning: given argument is nil; this will raise a TypeError in the next release added

#3 - 2019-12-02 05:48 - Seiei Miyagi

With ruby 2.7, following tests failed.

```
bin/rails test test/functional/timelog_report_test.rb:262
bin/rails test test/functional/timelog_report_test.rb:308
```

I send a pull request to ruby/csv to fix it.

<https://github.com/ruby/csv/pull/111>

The pull request is merged but not released yet.

#4 - 2019-12-03 09:17 - Go MAEDA

- Related to Patch #32542: Fix ruby 2.7 warning: The last argument is used as the keyword parameter added

#5 - 2020-01-04 05:09 - Go MAEDA

- Related to Defect #32752: Ruby 2.7: Remove deprecated URI.escape/unescape added

#6 - 2020-01-28 22:28 - Marius BĂLTEANU

- Related to Patch #32906: Update i18n (~> 1.8.2) added

#7 - 2020-01-29 09:20 - Go MAEDA

- Related to Patch #32907: Update capybara (~> 3.31.0) added

#8 - 2020-02-19 10:45 - Go MAEDA

- Related to Defect #33021: [v.4.0.5-stable] Internal Server Error 500 when accessing 'repository' tab added

#9 - 2020-05-05 05:22 - Jeremy Bailey

<https://github.com/ruby/csv/pull/111> was merged in November 2019, but it seems that ruby/csv [hasn't yet been updated](#) since 3.1.2 in October 2019.

#10 - 2020-05-10 14:49 - Ludovic Andrieux

ruby/csv 3.1.3 is released with the fix : <https://github.com/ruby/csv/releases/tag/v3.1.3>

#11 - 2020-08-31 17:54 - Anatol Pomozev

Hi folks,

Ruby 2.7 released almost a year ago. It would be great if useful tools redmine finally became supported at this latest stable version of Ruby.

Could anyone please give an update on the current status of this activity?

#12 - 2020-09-28 15:21 - Xavier Maquil

Any news about release for 2.7 support (Ubuntu 20.04.1 has it build in)

#13 - 2020-10-16 04:03 - Go MAEDA

Since the latest [Ruby 2.7.2](#) disables all deprecation warnings by default, I think Redmine can support Ruby 2.7 if it ignores 2.7.0 and 2.7.1.

The reason why Redmine still stays in Ruby 2.6 is that huge numbers of deprecation warnings are displayed while running tests. The problem is resolved by the change of Ruby 2.7.2, so I think it is no problem to use Ruby 2.7.2 now.

```
diff --git a/Gemfile b/Gemfile
index 0cfe10e49..24c85b341 100644
--- a/Gemfile
+++ b/Gemfile
@@ -1,6 +1,6 @@
 source 'https://rubygems.org'

-ruby '>= 2.3.0', '< 2.7.0'
+ruby '>= 2.3.0', '< 2.8.0'
 gem 'bundler', '>= 1.12.0'

 gem 'rails', '5.2.4.4'
diff --git a/doc/INSTALL b/doc/INSTALL
index f4374dae1..137088290 100644
--- a/doc/INSTALL
+++ b/doc/INSTALL
@@ -7,7 +7,7 @@ http://www.redmine.org/
```

== Requirements

```
-* Ruby 2.3, 2.4, 2.5, 2.6
+* Ruby 2.3, 2.4, 2.5, 2.6, 2.7 (except for 2.7.0 and 2.7.1)
* Bundler >= 1.12.0
```

```
* A database:
```

#14 - 2020-10-19 02:35 - Go MAEDA

- Status changed from New to Closed
- Assignee set to Go MAEDA
- Resolution set to Fixed

Redmine supports Ruby 2.7 (excluding 2.7.0 and 2.7.1) after [r20150](#).

#15 - 2020-10-19 05:58 - Enziin System

If Ruby 2.7 (**excluding** 2.7.0 and 2.7.1) then Redmine **NOT** support Ruby 2.7 version, still Ruby 2.6x

In the Gemfile:

```
ruby '>= 2.3.0', '< 2.7.0'
```

To

```
ruby '>= 2.3.0', '<= 2.7.0'
```

#16 - 2020-10-19 06:05 - Go MAEDA

Enziin System wrote:

If Ruby 2.7 (**excluding** 2.7.0 and 2.7.1) then Redmine **NOT** support Ruby 2.7 version, still Ruby 2.6x

Although Redmine does not support Ruby 2.7.0 and 2.7.1 but supports 2.7.2 or higher. This is related to the following issues:

- [Feature !#17000: 2.7.2 turns off deprecation warnings by default](#)
- [Feature !#16345: Don't emit deprecation warnings by default.](#)

#17 - 2020-10-22 09:10 - Enziin System

Ok, thanks!

In the above comment, you said "I think it is no problem to use Ruby 2.7.2 now"

Redmine run passed all tests on Ruby 2.7.2, isn't it?

#18 - 2020-10-22 09:14 - Go MAEDA

Enziin System wrote:

Redmine run passed all tests on Ruby 2.7.2, isn't it?

Yes. Redmine on Ruby 2.7.2 passes all tests without warnings.

#19 - 2021-02-26 08:41 - Adrien Crivelli

Thank you for the patch.

Unfortunately the patch was ready 4 months ago, but it has not been released yet, and will only be released as part of 4.2.0. According to the roadmap, 4.2.0 is at [80% with 42 opened issues](#). It could still be weeks, if not months, before we see this patch released.

I would like to suggest to consider to make releases **slightly** more often. Last release was almost [1 year ago](#). In fact with 210 issues 4.2.0 is on the biggest version in recent years. What about splitting versions in two to have **approximately** two releases per year ?

So I'd suggest to release the first half of 4.1.0 right now, and postpone what is not ready in the next version.

What do you think ?

#20 - 2021-04-21 01:37 - Go MAEDA

- Related to Defect #35135: FrozenError when new LDAP users try to login added

#21 - 2021-07-27 11:31 - Go MAEDA

- Related to Defect #35441: Inline image in Textile is not displayed if the image URL contains ampersands added