Redmine - Feature #6428

Interim solution for i18n >= 0.4

2010-09-17 20:26 - Felix Schäfer

Status:	Closed		**Start date:**	2010-09-17
Priority:	Normal		**Due date:**	
Assignee:	Eric Davis		**% Done:**	100%
Category:	Translations		**Estimated time:**	0.00 hour
Target version:	1.0.2			
Resolution:	Fixed			

Description

As more users get the i18n gem >= 0.4 pulled into their system, I think we need an interim solution until redmine works with that (probably redmine 1.1). The changes proposed by Martin in #5608 make redmine work with i18n 0.4.1, but it still throws a deprecation warning each time it's loaded with a full trace.

Based on an IRC chat with Eric, I will try to reduce eliminate that trace so that redmine can be somewhat functional with i18n 0.4 without needlessly polluting the logs.

```
[20:10] thegcat: edavis10: ok, so the small fix for redmine/I18n makes everything work, but it wil
l basically throw a deprecation warning each time the i18n gem gets reloaded
[20:10] thegcat: edavis10: (which is on every page load in development xX)
[20:12] thegcat: edavis10: we _could_ silence that warning by forcing @skip_syntax_deprecation to
true in the i18n gem, or even overload the method that generates the warn to make a little less ch
atty (it currently throws the full trace…)
[20:15] edavis10: thegcat: can you see about overloading it to 1) only show the single string with
 error and 2) say "you should install the correct version of the gem.... see redmine.org/issues…."
```

Related issues:

Related to Redmine - Defect #5608: broken with i18n 0.4.0	**Closed**	**2010-05-28**
Related to Redmine - Defect #6496: Remove i18n 0.3.x/0.4.x hack for Rails 2.3.5	**Closed**	**2010-09-26**

Associated revisions

Revision 4183 - 2010-09-26 19:13 - Eric Davis

Workaround for i18n 0.4.x with the old style syntax. #6428 #5608

This will also silence the whole trace with the deprecation warning.

Contributed by Felix Schäfer

History

#1 - 2010-09-17 23:42 - Felix Schäfer

- Status changed from New to 7

- Assignee changed from Felix Schäfer to Eric Davis

- Target version set to 1.0.2

Long story short: http://github.com/thegcat/redmine/commit/ba15e4e96b15a754cbdee16ae8a10bc0175ead0c makes redmine happy with i18n-0.4.1. (for anybody wondering: we've discussed this in some length with Eric on IRC, I will post a link to the logs once they show up/if I remember to)

Eric Hulser: remember to make a ticket to remember to remove this hackishness once we move past rails-2.3.5

Long story: the breakage with i18-0.4 is not (only) i18n's fault. I18n-0.4 as it is *is backwards compatible* with the {{something}} format and will just throw a deprecation warning the first time it encounters the "old" format, the only problem with that deprecation warning is that it throws the whole trace in the warning, which can be somewhat long, that is part the part in config/initializers/10-patches.rb fixes (it's the original function minus the stacktrace plus a reference to #5608).

The real reason all hell broke loose with redmine and i18-0.4 is that rails<=2.3.5 (ActiveRecord to be precise) adds a bit to i18n to warn about the deprecation of %d and %s style interpolation and replaces them with {{count}} and {{string}}. Redmine does have %d in some of its locale strings, but only for date formats, and those didn't trip the deprecation warning from rails with i18n-0.3.7, but do with 0.4.1, in which case the %d in the date format string is replaced by {{count}}, ultimately causing i18n to fail because it doesn't have a clue what count shall be. That's worked around by adding all those :count => date.strftime('%d') to the calls that need them. Funny thing is: said interpolation warning and mangling was removed in rails-2.3.6 to work correctly against i18n-0.4.

Where does that leave us: with a patch to somewhat defuse the overly long i18n deprecation warning firing each first-time-after-getting-loaded i18n encounters a {{something}} formatted locale string, and rails-2.3.5 firing a deprecation warning each an every time a string with %d in it gets into i18n. (I really tried to get around the rails thing, but I wasn't able to defuse the alias_method_chain, if someone knows a way, I'd be happy to hear it! :-))

#2 - 2010-09-22 05:06 - Dmitry Polushkin

Please, don't do that, because monkey-patching is bad, you can forget about it.

I had the same problem in my own rails 2.3 applications. Because I don't want yet to use gemsets (rvm), I tried to find some simple solution for that kind of "bug". And I found it. Just update redmine with bundler; so it will work, and also it will be easier to migrate to rails3.

To do so, just follow the link: http://gembundler.com/rails23.html

Hope this helps.

#3 - 2010-09-26 18:53 - Eric Davis

Dmitry Polushkin wrote:

> I had the same problem in my own rails 2.3 applications. Because I don't want yet to use gemsets (rvm), I tried to find some simple solution for that kind of "bug". And I found it. Just update redmine with bundler; so it will work, and also it will be easier to migrate to rails3.

That's not an option for the stable branch of Redmine (1.0.2). There have been too many issues caused by the wrong i18n version that patching Rails is going to be the easier and most maintainable option at this point.

#4 - 2010-09-26 19:14 - Eric Davis

- Category set to Translations

- Status changed from 7 to Resolved

- % Done changed from 0 to 100

- Resolution set to Fixed

Applied in r4183.

When we get Redmine 1.1.0 ready, we will upgrade to the latest Rails 2.3.x or Rails 3.x so this patch can be removed then.

#5 - 2010-09-27 00:21 - Eric Davis

- Status changed from Resolved to Closed

Merged into 1.0-stable

#6 - 2010-09-27 17:19 - Dmitry Polushkin

Eric Davis wrote:

> Dmitry Polushkin wrote:
>
> > I had the same problem in my own rails 2.3 applications. Because I don't want yet to use gemsets (rvm), I tried to find some simple solution for that kind of "bug". And I found it. Just update redmine with bundler; so it will work, and also it will be easier to migrate to rails3.
>
> That's not an option for the stable branch of Redmine (1.0.2). There have been too many issues caused by the wrong i18n version that patching Rails is going to be the easier and most maintainable option at this point.

Actually I don't see that the commit r4183 is more easier, than to create 2 files (Gemfile and preinitializer.rb).

Anyway thanks for the commit. Can approve that the problem gone.

#7 - 2010-09-27 17:39 - Eric Davis

Dmitry Polushkin wrote:

> Actually I don't see that the commit r4183 is more easier, than to create 2 files (Gemfile and preinitializer.rb).

Writing the code isn't difficult, it's helping all of the users get Bundler setup and running. Also as part of our versioning, we aren't switching library versions for point releases.

For Redmine 1.1.0, we can (and probably will) move to using Bundler. http://www.redmine.org/boards/4/topics/17065

#8 - 2012-03-20 09:04 - Pavel Potcheptsov

redmine Revision: 9237
ruby 1.8.7 (2010-01-10 patchlevel 249) [i486-linux]
Rails 2.3.14

- LOCAL GEMS ***
  actionmailer (2.3.14)
  actionpack (2.3.14)
  activerecord (2.3.14)
  activeresource (2.3.14)
  activesupport (2.3.14)
  cgi_multipart_eof_fix (2.5.0)
  daemons (1.1.8)
  fastthread (1.0.7)
  gem_plugin (0.2.3)
  i18n (0.4.2)
  mongrel (1.1.5)
  mongrel_cluster (1.0.5)
  mysql (2.8.1)
  oniguruma (1.1.0)
  plist (3.1.0)
  rack (1.1.3)
  rails (2.3.14)
  rake (0.8.7)
  rdoc (2.4.2)
  ruby-ole (1.2.11.3)
  rubygems-update (1.8.17, 1.6.2)
  spreadsheet (0.6.5.9)
  textpow (0.10.1)
  ultraviolet (0.10.2)

mongrel.log:
The {{key}} interpolation syntax in I18n messages is deprecated. Please use %{key} instead.
en - Draft saved at {{value}}

after edit http://www.redmine.org/projects/redmine/repository/revisions/4183/diff/trunk/test/unit/lib/redmine/i18n_test.rb
log is clear. thank you peoples.