

Redmine - Defect #10179

Wiki diff crashes after upgrade from 0.8 to 1.3

2012-02-08 17:12 - Alexander Kuznecov

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Wiki	Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:	Invalid		

Description

We have upgraded our Redmine from 0.8 to 1.3 version.

Now we got problem with old diffs in wiki history:
Processing WikiController#diff (for xxx.xxx.xxx.xxx at 2012-02-08 11:05:59) [GET]

```
Parameters: {"commit"=>"View differences", "project_id"=>"xxxxxxxx", "action"=>"diff", "id"=>"Wiki", "version"=>"7", "controller"=>"wiki", "version_from"=>"6"}
```

```
Zlib::DataError (incorrect header check):  
  
app/models/wiki_content.rb:96:in `inflate'  
app/models/wiki_content.rb:96:in `text'  
app/models/wiki_page.rb:181:in `initialize'  
app/models/wiki_page.rb:114:in `new'  
app/models/wiki_page.rb:114:in `diff'  
app/controllers/wiki_controller.rb:195:in `diff'  
unicorn (4.1.1) lib/unicorn/http_server.rb:528:in `process_client'  
unicorn (4.1.1) lib/unicorn/http_server.rb:600:in `worker_loop'  
unicorn (4.1.1) lib/unicorn/http_server.rb:485:in `spawn_missing_workers'  
unicorn (4.1.1) lib/unicorn/http_server.rb:135:in `start'  
unicorn (4.1.1) bin/unicorn_rails:209  
  
/home/platform/.rvm/gems/ruby-1.8.7-p352@redmine/bin/unicorn_rails:19:in `load'  
  
/home/platform/.rvm/gems/ruby-1.8.7-p352@redmine/bin/unicorn_rails:19
```

Rendering /export/redmine/redmine13/public/500.html (500 Internal Server Error)

Wiki content versions stored as gzip compressed blobs.
I found that decompression behavior was changed for Ruby 1.9:
https://github.com/edavis10/redmine/commit/1dab1cd5cc9d46724b8d722d994737806ee22bb9#app/models/wiki_content.rb
This code also appeared in 1.3 and 1.3.1

But 1.9 supported with trunk only and it not works with stable versions running on Ruby 1.8.x
Should be other solution here :(

Redmine: 1.3
Ruby: 1.8.7

History

#1 - 2012-02-08 22:10 - Toshi MARUYAMA

[source:tags/1.3.1/app/models/wiki_content.rb#L97](#) is no effect on Ruby 1.8.
It seems your database or environment problem.

#2 - 2012-02-10 08:11 - Jean-Philippe Lang

- *Resolution set to Cant reproduce*

Indeed. This change is just a ruby 1.9 compatibility fix and does not change to the uncompression method.
It looks like your compresses data has been altered.

#3 - 2012-02-10 09:30 - Etienne Massip

Jean-Philippe Lang wrote:

Indeed. This change is just a ruby 1.9 compatibility fix and does not change to the uncompression method.
It looks like your compresses data has been altered.

BTW, couldn't we extend core by adding String#to_utf8 for 1.8 rather than declaring a new method?

#4 - 2012-02-11 17:53 - Jean-Philippe Lang

Etienne Massip wrote:

BTW, couldn't we extend core by adding String#to_utf8 for 1.8 rather than declaring a new method?

Which new method?

#5 - 2012-02-11 18:03 - Etienne Massip

Jean-Philippe Lang wrote:

Which new method?

I was thinking about [source:trunk/lib/redmine/codeset_util.rb#L38](#) which seems redundant with 1.9 String#to_utf8, but maybe I'm out of topic.

#6 - 2012-02-26 09:45 - Alexander Kuznecov

Unfortunately for me, I confirm that problem related to database.

Issue can be marked as invalid or be left for 1.8/1.9 code conversation

#7 - 2012-02-26 11:09 - Etienne Massip

- *Status changed from New to Closed*

- *Resolution changed from Cant reproduce to Invalid*