**Redmine - Defect #24616**  
**Should not replace all invalid utf8 characters (e.g in mail)**  
2016-12-15 02:13 - Pavel Rosický

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
<th>Start date:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Category:</td>
<td>I18n</td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version:</td>
<td>3.4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resolution:</td>
<td>Fixed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Affected version:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

Hello,
I've an email, that is encoded in utf8, but it contains an invalid character. In this case, redmine converts the content to us-ascii and then to utf8. This step will replace non-ascii compatible chars to "?". Why?

1) Failure:
```
MailHandlerTest#test_invalid_utf8 [/test/unit/mail_handler_test.rb:548]:
  Expected: "Здравствуйте?"
  Actual: "?????????????"
```

I changed Redmine::CodesetUtil.replace_invalid_utf8(str) and Redmine::CodesetUtil.to_utf8(str, encoding)

```ruby
```

to

```ruby
```

all tests are passing with this change.

Redmine version                3.3.1.stable
Ruby version                   2.1.5-p273 (2014-11-13) [x64-mingw32]
Rails version                  4.2.7.1
Environment                   production
Database adapter               Mysql2
SCM:
   Git                          2.10.1
   Filesystem                   
Redmine plugins:
   no plugin installed

**Associated revisions**

Revision 16271 - 2017-01-28 06:43 - Toshi MARUYAMA

add more tests (#24616)

Revision 16272 - 2017-01-28 06:43 - Toshi MARUYAMA
unify duplicate codes (#24616)

Revision 16273 - 2017-01-28 07:24 - Toshi MARUYAMA

do not replace all invalid utf8 (#24616)

Revision 16274 - 2017-01-28 07:25 - Toshi MARUYAMA

additional test for mail by Pavel Rosický (#24616)

Revision 16275 - 2017-01-28 07:29 - Toshi MARUYAMA

svn propset svn:eol-style native test/fixtures/mail_handler/invalid_utf8.eml (#24616)

History

#1 - 2016-12-29 08:14 - Go MAEDA
- File defect-24616.diff added
- Target version set to 3.3.2

Looks good to me.

# valid UTF-8 string
  text = "こんにちは"
  p text.valid_encoding? # => true

# making invalid UTF-8 string
  text.force_encoding("ASCII-8BIT")
  text[-1] = 0xff.chr
  text.force_encoding("UTF-8")
  p text.valid_encoding? # => false
  p text # => "こんにち\xE3\x81\xFF"

# Current code of Redmine
  # => "???????"

# Fixed code by Pavel Rosický
  # => "こんにち??"

#2 - 2016-12-29 09:31 - Toshi MARUYAMA
- Target version deleted (3.3.2)

Did you run whole tests?
Especially this test.

2020-04-11
Pavel Rosický wrote:

Hello,

I've an email, that is encoded in utf8, but it contains an invalid character. In this case, redmine converts the content to us-ascii and then to utf8.

This step will replace non-ascii compatible chars to "?". Why?

You can see this function purpose.

Thanks Toshi, I rechecked it again and all tests are passing.

In this case, my change has no effect on the result, because the string contains just one invalid utf-8 character.

```
s1.encode('us-ascii', :invalid => :replace, :undef => :replace, :replace => '?').encode('utf-8')
"Texte encodé en ISO-8859-1."
```

# patched
```
s1.encode('utf-8', :invalid => :replace, :undef => :replace, :replace => '?')
"Texte encodé en ISO-8859-1."
```

but a combination of valid and invalid utf-8 chars (non-ascii-compatible) will result both characters are stripped. Try out GO Media's example.

```
$ irb
1.9.3-p551 :001 > text = "こんにち\xE3\x81\xFF"
=> "こんにち\xE3\x81\xFF"
=> "こんにち\xE3\x81\xFF"
1.9.3-p551 :003 > text.valid_encoding?
=> false
```

```
$ irb
2.3.3 :001 > text = "こんにち\xE3\x81\xFF"
=> "こんにち\xE3\x81\xFF"
2.3.3 :002 > text =  text.encode("UTF-8", :invalid => :replace, :undef => :replace, :replace => '?')
=> "こんにち??"
2.3.3 :003 > text.valid_encoding?
=> true
```
Pavel Rosický wrote:

```markdown
Hello,
I've an email, that is encoded in utf8, but it contains an invalid character. In this case, redmine converts the content to us-ascii and then to utf8. This step will replace non-ascii compatible chars to "?". Why?
```

Because of Ruby 1.8.7 behavior compatibility.

source:tags/2.6.9/lib/redmine/codeset_util.rb

---

#7 - 2017-01-28 07:09 - Toshi MARUYAMA
- Subject changed from encoding error if email contains an invalid utf8 character to Should not replace all invalid utf8 characters
- Category changed from Email receiving to I18n

#8 - 2017-01-28 07:10 - Toshi MARUYAMA
- Subject changed from Should not replace all invalid utf8 characters to Should not replace all invalid utf8 characters (e.g in mail)

#9 - 2017-01-28 08:56 - Toshi MARUYAMA
- Status changed from New to Closed
- Target version set to 3.4.0
- Resolution set to Fixed

I have committed r16273 to pass on Ruby 1.9.3.
I don't want to change behavior on stable.

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>invalid_utf8_test.patch</td>
<td>1.49 KB</td>
<td>2016-12-15</td>
<td>Pavel Rosicky</td>
</tr>
<tr>
<td>defect-24616.diff</td>
<td>1.73 KB</td>
<td>2016-12-29</td>
<td>Go MAEDA</td>
</tr>
</tbody>
</table>