

## Redmine - Defect #10772

### 4-byte utf-8 characters

2012-04-26 04:54 - Richard Lee

<b>Status:</b> Closed	<b>Start date:</b>
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b> Database	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b>	<b>Affected version:</b>
<b>Resolution:</b> Duplicate	

**Description**

My redmine version is 1.2.0

When I update the first name of my account, the internal error appear.

The error description below:

ActiveRecord::StatementInvalid in MyController#account  
Mysql::Error: Incorrect string value: '\xF0\xA0\xAE\x9F' for column 'firstname' at row 1: UPDATE `users` SET `firstname` = '□□'

**Related issues:**

Related to Redmine - Defect #10776: One Japanese character can not be input~!	<b>Closed</b>
Related to Redmine - Defect #10779: redmine cannot deal with 4-byte utf-8 cha...	<b>Closed</b>
Related to Redmine - Feature #31921: Changes to properly support 4 byte chara...	<b>Closed</b>
Is duplicate of Redmine - Defect #18866: MySQL: disappear after 4-Byte UTF-8	<b>Closed</b>
Is duplicate of Redmine - Defect #20143: Mailhandler cannot handle 4-byte cha...	<b>New</b>
Has duplicate Redmine - Defect #34235: Losing data due to unsupported characters	<b>Closed</b>

### History

#### #1 - 2012-04-26 11:01 - Etienne Massip

- Category set to Accounts / authentication

Could you please give more details according to [SubmittingBugs?](#)

#### #2 - 2012-04-26 11:12 - Richard Lee

- Assignee set to Etienne Massip

The character '

#### #3 - 2012-04-26 11:15 - Etienne Massip

- Status changed from New to Closed

- Assignee deleted (Etienne Massip)

- Resolution set to Invalid

#### #4 - 2012-04-26 11:19 - Etienne Massip

- Status changed from Closed to New

- Resolution deleted (Invalid)

Sorry, there was a bug here, your post has been truncated.

Here is the content of the mail:

The character '<problematic character>' make redmine Internal Error

My redmine version is 1.2.0

When I update the first name of my account, the internal error appear.

The error description below:

ActiveRecord::StatementInvalid in MyController#account

Mysql::Error: Incorrect string value: '\xF0xA0\xAE\x9F' for column 'firstname' at row 1: UPDATE `users` SET `firstname` = ' ' <problematic character>, `updated\_on` = '2012-04-26 10:48:31' WHERE `id` = 1

```
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract_adapter.rb:227:in `log'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/mysql_adapter.rb:324:in `execute'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/database_statements.rb:265:in `update_sql'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/mysql_adapter.rb:339:in `update_sql'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/database_statements.rb:49:in `update_without_query_dirty'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/query_cache.rb:19:in `update'
C:/redmine/vendor/rails/activerecord/lib/active_record/base.rb:2936:in `update_without_lock'
C:/redmine/vendor/rails/activerecord/lib/active_record/locking/optimistic.rb:81:in `update_without_dirty'
C:/redmine/vendor/rails/activerecord/lib/active_record/dirty.rb:146:in `update_without_timestamps'
C:/redmine/vendor/rails/activerecord/lib/active_record/timestamp.rb:64:in `update_without_callbacks'
C:/redmine/vendor/rails/activerecord/lib/active_record/callbacks.rb:282:in `update'
C:/redmine/vendor/rails/activerecord/lib/active_record/base.rb:2927:in `create_or_update_without_callbacks'
C:/redmine/vendor/rails/activerecord/lib/active_record/callbacks.rb:250:in `create_or_update'
C:/redmine/vendor/rails/activerecord/lib/active_record/base.rb:2577:in `save_without_validation'
C:/redmine/vendor/rails/activerecord/lib/active_record/validations.rb:1090:in `save_without_dirty'
C:/redmine/vendor/rails/activerecord/lib/active_record/dirty.rb:79:in `save_without_transactions'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:229:in `send'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:229:in `with_transaction_returning_status'
C:/redmine/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/database_statements.rb:136:in `transaction'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:182:in `transaction'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:228:in `with_transaction_returning_status'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:196:in `save'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:208:in `rollback_active_record_state!'
C:/redmine/vendor/rails/activerecord/lib/active_record/transactions.rb:196:in `save'
C:/redmine/app/controllers/my_controller.rb:60:in `account'
```

#### #5 - 2012-04-26 14:04 - Etienne Massip

- Subject changed from *The character* ' to *The character* '

Could you try with the mysql2 database adapter?

#### #6 - 2012-04-26 15:12 - Etienne Massip

-

#### #7 - 2013-09-05 09:59 - Not Relevant

I encountered the same problem and wanted to let you know my findings.

My test environment:

```
Environment:
  Redmine version      2.3.1.stable
  Ruby version         1.9.3 (i686-linux)
  Rails version        3.2.14
  Environment          production
  Database adapter     Mysql2
```

The underlying problem is the mysql database. Per default mysql databases and tables are utf8. But only up to 3 characters:

Here is what the mysql manual states:

The character set named utf8 uses a maximum of three bytes per character and contains only BMP characters. As of MySQL 5.5.3, the utf8mb4 character set uses a maximum of four bytes per character supports supplemental characters:

To solve this issue you need to convert your database and tables to utf8mb4

```
ALTER DATABASE redmine_default CHARACTER SET = utf8mb4 COLLATE = utf8mb4_unicode_ci;
ALTER TABLE attachments CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE auth_sources CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE boards CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE changes CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE changeset_parents CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE changesets CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE changesets_issues CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE comments CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE custom_fields CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE custom_fields_projects CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
ALTER TABLE custom_fields_trackers CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

And change the database.yml accordingly

```
adapter: mysql2
database: redmine_default
host: localhost
port:
username: redmine_default
password: secret
encoding: utf8mb4
```

Worst part about this bug is: Everything after chars 4 bytes long will be truncated without warning. That's the reason for your "Sorry, there was a bug here, your post has been truncated."

Some sanity check should be done. Not sure if that's redmine or upstream.

And no, this has nothing to do with Account or i18n, it's a general problem.

**#8 - 2013-09-21 12:30 - Toshi MARUYAMA**

- Category changed from Accounts / authentication to Database

**#9 - 2015-06-22 05:46 - Toshi MARUYAMA**

- Subject changed from The character ' to 4-byte utf-8 characters

**#10 - 2015-06-22 05:48 - Toshi MARUYAMA**

- Related to Patch #19742: RedmineInstall: MySQL: collation\_database added

**#11 - 2015-06-22 05:49 - Toshi MARUYAMA**

- Has duplicate Defect #18866: MySQL: disappear after 4-Byte UTF-8 added

**#12 - 2015-06-22 05:54 - Toshi MARUYAMA**

- Related to deleted (Patch #19742: RedmineInstall: MySQL: collation\_database)

**#13 - 2015-06-22 06:03 - Toshi MARUYAMA**

- Status changed from New to Closed

- Resolution set to Duplicate

I close this issue because description is Ruby 1.8 (Mysql, not Mysql2).

There are duplicate issues.

- lose 4-byte utf8 ([#18866](#))
- Mailhandler raises exception ([#20143](#))

**#14 - 2015-06-22 06:06 - Toshi MARUYAMA**

- Has duplicate deleted (Defect #18866: MySQL: disappear after 4-Byte UTF-8)

**#15 - 2015-06-22 06:08 - Toshi MARUYAMA**

- Is duplicate of Defect #18866: MySQL: disappear after 4-Byte UTF-8 added

**#16 - 2015-06-22 06:09 - Toshi MARUYAMA**

- Is duplicate of Defect #20143: Mailhandler cannot handle 4-byte characters added

**#17 - 2017-12-18 13:40 - Jēkabs Jānis Kalniņš**

Not Relevant wrote:

I encountered the same problem and wanted to let you know my findings.  
My test environment:  
[...]

The underlying problem is the mysql database. Per default mysql databases and tables are utf8. But only up to 3 caracters:

Here is what the mysql manual states:

The character set named utf8 uses a maximum of three bytes per character and contains only BMP characters. As of MySQL 5.5.3, the utf8mb4 character set uses a maximum of four bytes per character supports supplemental characters:

To solve this issue you need to convert your database and tables to utf8mb4

[...]

And change the database.yml accordingly

[...]

Worst part about this bug is: Everything after chars 4 bytes long will be truncated without warning. That's the reason for your "Sorry, there was a bug here, your post has been truncated."

Some sanity check should be done. Not sure if that's redmine or upstream.

And no, this has nothing to do with Account or i18n, it's a general problem.

Thanks this worked for me [#27803](#)

```
ALTER TABLE issues CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci
ALTER TABLE journals CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci
ALTER TABLE wiki_contents CONVERT TO CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci
```

```
In config/database.yml
encoding: utf8
to
encoding: utf8mb4
```

#### **#18 - 2019-08-17 18:32 - Marius BĂLTEANU**

- Related to Feature #31921: Changes to properly support 4 byte characters (emoji) when database is MySQL added

#### **#19 - 2021-01-25 00:29 - Marius BĂLTEANU**

- Related to Defect #34235: Losing data due to unsupported characters added

#### **#20 - 2021-01-25 00:30 - Marius BĂLTEANU**

- Related to deleted (Defect #34235: Losing data due to unsupported characters)

#### **#21 - 2021-01-25 00:32 - Marius BĂLTEANU**

- Has duplicate Defect #34235: Losing data due to unsupported characters added