

Rake task for converting from Textile to Markdown

2022/05/09 17:53 - Admin Redmine

ステータス:	New	開始日:	2022/05/09
優先度:	高め	期日:	
担当者:		進捗率:	0%
カテゴリ:	Text formatting_26	予定工数:	0.00時間
対象バージョン:		作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/22005	status_id:	1
category_id:	26	tracker_id:	2
version_id:	0	plus1:	0
issue_org_id:	22005	affected_version:	
author_id:	126812	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	10		

説明

It would be neat to have a [[redmine:RedmineRake]] to convert a redmine installation from *Textile* to *Markdown*. Here is a starter which use "Pandoc": <http://pandoc.org/> and works well but only for the *wiki* module: attachment:convert_textile_to_markdown.rake.

It is taken from "this post":

<http://stackoverflow.com/questions/9782121/how-to-convert-existing-redmine-wiki-from-textile-to-markdown> and only slightly modified to work with newer versions of *Pandoc*. We think it would be a great feature if such a converter existed for all modules : issue tracking, wiki, document, boards, etc.

Apologies if such a tool already exist, but we weren't able to find it.

journals

Yes, this really is a must have. Once this rake task is implemented, there should be even a button for managers or anyone with the proper permissions to convert any given page to MD (from Textile) because ATM I have to:

- open a wiki page
- click edit
- ctrl+a, ctrl+c
- open a text file on my comp
- ctrl+v, ctrl+s
- run my converter batch file which calls pandoc
- open the output of my batch file
- ctrl+a, ctrl+c
- go back to wiki page
- ctrl+a, delete, ctrl+v
- save

And I have to do this for EVERY SINGLE WIKI PAGE. Such a pain in the arse.

BUT first they should fix Markdown rendering. See: "Markdown newline rendering broken":
<http://www.redmine.org/issues/22323>

FWIW, I went through this a while back - wrote a small custom script to scan through all wiki and forum posts in the MySQL database for each project. Automating this is not very hard, and solved the issue for a dozen projects and thousands of entries for me. Having this implemented in Ruby as Rake task would indeed be useful.

It was written in Tcl at the time, but here's the code if it's of any use:

```
#!/usr/bin/env tclsh
```

```

set pw {blahblahblah}

package require mysqltcl

set db [mysql::connect -user xured -password $pw -db redmine]
mysql::encoding $db utf-8
mysql::exec $db "set names 'utf8'"

proc convert {table rowid field} {
    global db
    puts $table:
    foreach row [mysql::sel $db "select $rowid,$field from $table" -list] {
        lassign $row id old

        set fd [open tmp.txt w]
        fconfigure $fd -encoding utf-8
        puts -nonewline $fd $old
        close $fd

        exec pandoc -f textile -t markdown_github -o tmp2.txt tmp.txt

        set fd [open tmp2.txt r]
        fconfigure $fd -encoding utf-8
        set new [read $fd]
        close $fd

        puts "    id $id text [string length $old] => [string length $new]"
        set quoted [mysql::escape $db $new]
        #puts 1<[string range $old 0 150]>
        #puts 2<[string range $new 0 150]>
        #puts 3<[string range $quoted 0 150]>
        mysql::exec $db "update $table set $field = '$quoted' where $rowid = $id"
    }
}

convert wiki_contents id text
convert comments id comments
convert issues id description
convert messages id content
convert news id description

```

PS. I don't think this issue is related to #22323, btw.

There is a bug in pandoc which doesn't recognize raw Textile URLs. Pandoc escapes chars in these URLs. Here is a quick fix:
<https://github.com/jgm/pandoc/pull/2970>

You can use modified tcl script:

```

#!/usr/bin/env tclsh

set pw {blahblahblah}

package require mysqltcl

set db [mysql::connect -user xured -password $pw -db redmine]
mysql::encoding $db utf-8
mysql::exec $db "set names 'utf8'"

proc convert {table rowid field} {
    global db
    puts $table:
    foreach row [mysql::sel $db "SELECT $rowid,$field FROM $table WHERE $field IS NOT NULL AND $field != ''"] {
        lassign $row id old

        set fd [open /tmp/tmp.txt w]
        fconfigure $fd -encoding utf-8

```

```
puts -nonewline $fd $old
close $fd
```

```
exec docker exec -ti pandoc pandoc -f textile -t markdown_github -o /tmp/tmp2.txt /tmp/tmp.txt
```

```
set fd [open /tmp/tmp2.txt r]
fconfigure $fd -encoding utf-8
set new [read $fd]
close $fd
```

```
puts " id $id text [string length $old] => [string length $new]"
set quoted [mysql::escape $db $new]
#puts 1<[string range $old 0 150]>
#puts 2<[string range $new 0 150]>
#puts 3<[string range $quoted 0 150]>
mysql::exec $db "update $table set $field = '$quoted' where $rowid = $id"
}
```

```
convert wiki_contents id text
convert comments id comments
convert issues id description
convert messages id content
convert news id description
convert journals id notes
```

With running docker container which has updated pandoc version:

```
docker run -d --name pandoc -v /tmp:/tmp kayrus/pandoc bash -c 'while true; do sleep 1; done'
```

I built upon what you suggested here and came up with a solution that I think is much more complete. First of all it migrates all content (comment, wiki, issue, message, news, document, project and journal), and then it fixes several incompatibility between Redmine's Textile and pandoc's. Please have a look over there: https://github.com/Ecodev/redmine_convert_textile_to_markown

Also feel free to re-use the code for Redmine core or anything else.

That's really a coincidence. I was working at the same time on the same thing. However I decided against using pandoc since it was creating undesired artefacts (I guess Adrien fixed this) and markdown which caused redmine 3.3.0 to crash.

I've chosen a direct approach which avoids using pandoc: https://github.com/akohlbecker/textile_to_markdown.rake

Fell free to also use this code for redmine core or to base redmine code on it.

That interesting... would you have an example of textile or markdown that lead to Redmine crash ? also do you know which version of pandoc you tried ?

Before starting working on this, I was sure that pandoc was the way to go, because it's a real parser, not "only" regexp. But as I worked with it I came to realize that pandoc has a few shortcomings, and that Redmine custom syntax does not help. OTOH by using regexp you are likely to miss edge cases. For instance I have some @pre@ tags that are not at the beginning of the line, but preceded by words (I know, weird), I think your solution would miss this case as it is now. I am bit torn to know which is the best approach...

I stripped it down to the text snippet which is actually crashing the markdown formatter:

```
[areas.php" and "points.php]()
```

It is most probably the missing URI which processed in a not nil save manner.

For details please refer to #23395.

I am bit torn to know which is the best approach...

Honestly, I am feeling the same. Both of the methods have their pros and cons. At least for my specific task I found the solution which hurt less.

Andreas Kohlbecker wrote:

I stripped it down to the text snippet which is actually crashing the markdown formatter:

The source textile snippet from which this invalid link has been created is:

```
* *rest_gen.php* (version 1 developped from May 2011): Merging "areas.php" and "points.php": color theme bot
s (polygons) and distribution points. Output in JSON format is also available.
```

Latest version of pandoc does not output empty links the same way anymore, "see for yourself":

http://pandoc.org/try/?text=Merging+%22areas.php%22+and+%22points.php%22%3A+color+theme&from=textile&to=markdown_github :

Merging “ areas.php ” and [points.php] color theme

[points.php]:

So I guess it will not crash anymore, but I wouldn't call it a correct output either... So I "created an issue for that empty link bug":<https://github.com/jgm/pandoc/issues/3036>. I'd argue that it might be the best advantage to using pandoc, to improve the product for everybody else too. So far @jgm has been very responsive to all issues I opened. Hopefully, in a not so distant future, we'll be able to entirely rely on pandoc.

related_issues

relates,Closed,22323,Markdown newline rendering broken

履歴

#1 - 2022/05/10 17:07 - Admin Redmine

- カテゴリ を Text formatting_26 にセット