

Redmine - Defect #117

\"(char code 0x5c) handling is improper in PDF output

2007-10-01 05:30 - Go MAEDA

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Affected version:	
Resolution:			
Description When createing PDF file, \" may have to be escaped like \". Redmine perhaps does not escape \", so we sometimes get malformed output. Because of this problem, issues which contains specific Japanese multi-byte character whose second byte character is 0x5c are not properly exported PDF file (gets malformed output). [How to reproduce] 1. Create new issue. Description of the issue is following. <pre>printf("Hello\n"); printf("World.\n");</pre> 2. Export it to PDF. You will see malformed output like this. <pre>printf("Hello "); printf("World.\n");</pre>			

Associated revisions

Revision 813 - 2007-10-06 16:27 - Jean-Philippe Lang

Fixed: improper 0x5c char handling in PDF output (Go Maeda).

Revision 5337 - 2011-04-06 02:40 - Toshi MARUYAMA

PDF: create new unit lib pdf test and add '0x5c'(backslash) handling test (#61, #117).

Japanese Shift_JIS and Traditional Chinese Big5 have '0x5c'(backslash) problem.

Revision 5339 - 2011-04-06 04:41 - Toshi MARUYAMA

PDF: set language 'ja' in '0x5c'(backslash) handling unit lib test (#61, #117).

Revision 5341 - 2011-04-06 06:27 - Toshi MARUYAMA

PDF: fix \"(double backslashes) handling of FPDF ANSI (#61, #117).

Revision 5342 - 2011-04-06 06:28 - Toshi MARUYAMA

PDF: add \"(double backslashes) handling of FPDF ANSI test in unit lib test (#61, #117).

Revision 5343 - 2011-04-06 07:38 - Toshi MARUYAMA

PDF: Ruby 1.9 compatibility of '0x5c'(backslash) handling in FPDF ANSI (#61, #117).

Japanese Shift_JIS and Traditional Chinese Big5 have '0x5c'(backslash) problem.

Revision 5344 - 2011-04-06 07:39 - Toshi MARUYAMA

PDF: add test of Ruby 1.9 compatibility of '0x5c'(backslash) handling in FPDF ANSI (#61, #117).

Japanese Shift_JIS and Traditional Chinese Big5 have '0x5c'(backslash) problem.

History

#1 - 2007-10-02 00:00 - Aruo Miura

Hi.
I make patch for this problem.
This one can resolve 2 problems:
A:Iconv can't convert multi-byte string include '\'.
B:multi-byte character include 0x5c can't show in PDF.

I check only environments:
WindowsXP SP2/Japanese
ruby 1.8.6-p110
rails 1.2.3
redmine rev.781

```
Index: ifpdf_helper.rb
=====
==
--- ifpdf_helper.rb      (Revision 781)
+++ ifpdf_helper.rb      (Working copy)
@@ -53,7 +53,13 @@
     def Cell(w,h=0,txt='',border=0,ln=0,align='',fill=0,link='')
       @ic ||= Iconv.new(l(:general_pdf_encoding), 'UTF-8')
       txt = begin
-         @ic.iconv(txt)
+         txtar = txt.split('\\\\')
+         txtar << '' if txt[-1] == ?\\
+         txtar2 = []
+         txtar.each{|x|
+           txtar2 << @ic.iconv(x)
+         }
+         txtar2.join('\\\\').gsub(/\\/ , "\\\\\\\\\\\\\\\\")
       rescue
         txt
       end
```

#2 - 2007-10-06 10:28 - Jean-Philippe Lang

Your fix (slightly edited) is committed in [r813](#).
Thanks for your help.

#3 - 2007-10-06 20:04 - Go MAEDA

Thanks for your work. The fix is of great benefit to MBCS users.

Although the patch's author is Aruo Miura, my name is
written in svn commit log instead of Miura.

#4 - 2007-10-07 10:06 - Jean-Philippe Lang

Oh, I'm sorry. I didn't notice that the patch was from someone
else.

Big thanks to Aruo Miura :-)

#5 - 2007-10-07 17:35 - Aruo Miura

No problem.
Thanks for your work.