

projekt kivitendo - Unterstützung #107

Formulareingabe auf nicht UTF-8 kodierte Zeichen prüfen

17.11.2015 14:06 - Jan Büren

Status:	Neu	Beginn:	17.11.2015
Priorität:	Normal	Abgabedatum:	
Zugewiesen an:		% erledigt:	0%
Kategorie:		Geschätzter Aufwand:	0:00 Stunde
Zielversion:		Aufgewendete Zeit:	0:00 Stunde
Beschreibung			
Ich hab einmal, zum Ausprobieren folgenden Check beim Speichern von Angeboten eingetragen.			
<pre>use Encode; use Carp; (...) eval { decode("UTF-8", \$form->{"longdescription_\$i"}, Encode::FB_CROAK); die if \$@; 1; } or do { croak "Ungültige Zeichen im Langtext bei Position \$i. Fehlerhafte Stelle innerhalb d ieser Eingabe:" . \$form->{"longdescription_\$i"}; };</pre>			
Und komme dann zu einer Fehlermeldung wie im Anhang beschreiben.			
Ich finde das Verhalten um Längen besser, als die lange Fehlerkette von LaTeX, vom Anwender, "parsen" zu lassen.			
Im Detail kann/sollte die Prüfung auch anders aussehen (neuer Controller?).			
Was ich mir noch wünschen würde, wäre die genaue Stelle des fehlerhaften Zeichens einzugrenzen.			
Das hier beschriebene FB_QUIET oder FB_DEFAULT Fehlerhandling lieferte bei mir immer nur die Meldung Zeichenkette enthält wide character, o.ä. zurück:			
http://perldoc.perl.org/Encode.html#Handling-Malformed-Data			
Offen zur Diskussion ...			

Historie

#1 - 17.11.2015 22:28 - Bernd Bleßmann

Jan Büren schrieb:

Ich finde das Verhalten um Längen besser, als die lange Fehlerkette von LaTeX, vom Anwender, "parsen" zu lassen.

Das Problem hatte ich schon kurz im Chat angesprochen:

Dein problematisches Zeichen war ja ein bestimmter Bindestrich (Hyphen) utf8: 0xE28090: Unicode 0x2010: -

Das ist durchaus ein gültiges UTF-8-Zeichen.

Mit Deinem Workaround fängst du auch andere ab, die so schon (durch special_chars) in Latex gesetzt werden können (z.B. den left-arrow: utf8: 0xE28690: Unicode 0x2190: ←), die also funktionieren.

Wie mosu sagt, ist nicht falsches utf8 das Problem, sondern latex weiß nichts damit anzufangen.

#2 - 23.11.2015 19:01 - Jan Büren

Ich hab die special_chars versuchsweise mal probiert "whitezulisten", allerdings gelingt mir dies auf Anhieb nicht:

Falls der eval-Block einen Fehler wirft, entsprechend noch vorher prüfen, ob wir diese Zeichen nicht in special_chars parsebar zu Verfügung haben:

```
# whitelist versuch an dieser stelle
croak "Ungültige Zeichen im Langtext bei Position $i. Fehlerhafte Stelle innerhalb dieser Eingabe:" . $form->{"longdescription_$i"}
unless $form->{"longdescription_$i"} =~ m/(\x{2}|\x{3}|\x{4}|\x{5}|\x{6}|\x{7}|\x{8}|\x{9}|\x{A}|\x{B}|\x{C}|\x{D}|\x{E}|\x{F}|\x{10}|\x{11}|\x{12}|\x{13}|\x{14}|\x{15}|\x{16}|\x{17}|\x{18}|\x{19}|\x{1A}|\x{1B}|\x{1C}|\x{1D}|\x{1E}|\x{1F}|\x{20}|\x{21}|\x{22}|\x{23}|\x{24}|\x{25}|\x{26}|\x{27}|\x{28}|\x{29}|\x{2A}|\x{2B}|\x{2C}|\x{2D}|\x{2E}|\x{2F}|\x{30}|\x{31}|\x{32}|\x{33}|\x{34}|\x{35}|\x{36}|\x{37}|\x{38}|\x{39}|\x{3A}|\x{3B}|\x{3C}|\x{3D}|\x{3E}|\x{3F}|\x{40}|\x{41}|\x{42}|\x{43}|\x{44}|\x{45}|\x{46}|\x{47}|\x{48}|\x{49}|\x{4A}|\x{4B}|\x{4C}|\x{4D}|\x{4E}|\x{4F}|\x{50}|\x{51}|\x{52}|\x{53}|\x{54}|\x{55}|\x{56}|\x{57}|\x{58}|\x{59}|\x{5A}|\x{5B}|\x{5C}|\x{5D}|\x{5E}|\x{5F}|\x{60}|\x{61}|\x{62}|\x{63}|\x{64}|\x{65}|\x{66}|\x{67}|\x{68}|\x{69}|\x{6A}|\x{6B}|\x{6C}|\x{6D}|\x{6E}|\x{6F}|\x{70}|\x{71}|\x{72}|\x{73}|\x{74}|\x{75}|\x{76}|\x{77}|\x{78}|\x{79}|\x{7A}|\x{7B}|\x{7C}|\x{7D}|\x{7E}|\x{7F}|\x{80}|\x{81}|\x{82}|\x{83}|\x{84}|\x{85}|\x{86}|\x{87}|\x{88}|\x{89}|\x{8A}|\x{8B}|\x{8C}|\x{8D}|\x{8E}|\x{8F}|\x{90}|\x{91}|\x{92}|\x{93}|\x{94}|\x{95}|\x{96}|\x{97}|\x{98}|\x{99}|\x{9A}|\x{9B}|\x{9C}|\x{9D}|\x{9E}|\x{9F}|\x{A0}|\x{A1}|\x{A2}|\x{A3}|\x{A4}|\x{A5}|\x{A6}|\x{A7}|\x{A8}|\x{A9}|\x{AA}|\x{AB}|\x{AC}|\x{AD}|\x{AE}|\x{AF}|\x{B0}|\x{B1}|\x{B2}|\x{B3}|\x{B4}|\x{B5}|\x{B6}|\x{B7}|\x{B8}|\x{B9}|\x{BA}|\x{BB}|\x{BC}|\x{BD}|\x{BE}|\x{BF}|\x{C0}|\x{C1}|\x{C2}|\x{C3}|\x{C4}|\x{C5}|\x{C6}|\x{C7}|\x{C8}|\x{C9}|\x{CA}|\x{CB}|\x{CC}|\x{CD}|\x{CE}|\x{CF}|\x{D0}|\x{D1}|\x{D2}|\x{D3}|\x{D4}|\x{D5}|\x{D6}|\x{D7}|\x{D8}|\x{D9}|\x{DA}|\x{DB}|\x{DC}|\x{DD}|\x{DE}|\x{DF}|\x{E0}|\x{E1}|\x{E2}|\x{E3}|\x{E4}|\x{E5}|\x{E6}|\x{E7}|\x{E8}|\x{E9}|\x{EA}|\x{EB}|\x{EC}|\x{ED}|\x{EE}|\x{EF}|\x{F0}|\x{F1}|\x{F2}|\x{F3}|\x{F4}|\x{F5}|\x{F6}|\x{F7}|\x{F8}|\x{F9}|\x{FA}|\x{FB}|\x{FC}|\x{FD}|\x{FE}|\x{FF})/;
```

Wenn ich mir den Wert der longdescription in die debug-Datei schreibe und diesen testweise überprüfe, funktioniert alles einwandfrei:

```
my $long = " was hier:<p><br /><test>";
unless ($long =~ m/(\x{2}|\x{3}|\x{4}|\x{5}|\x{6}|\x{7}|\x{8}|\x{9}|\x{A}|\x{B}|\x{C}|\x{D}|\x{E}|\x{F}|\x{10}|\x{11}|\x{12}|\x{13}|\x{14}|\x{15}|\x{16}|\x{17}|\x{18}|\x{19}|\x{1A}|\x{1B}|\x{1C}|\x{1D}|\x{1E}|\x{1F}|\x{20}|\x{21}|\x{22}|\x{23}|\x{24}|\x{25}|\x{26}|\x{27}|\x{28}|\x{29}|\x{2A}|\x{2B}|\x{2C}|\x{2D}|\x{2E}|\x{2F}|\x{30}|\x{31}|\x{32}|\x{33}|\x{34}|\x{35}|\x{36}|\x{37}|\x{38}|\x{39}|\x{3A}|\x{3B}|\x{3C}|\x{3D}|\x{3E}|\x{3F}|\x{40}|\x{41}|\x{42}|\x{43}|\x{44}|\x{45}|\x{46}|\x{47}|\x{48}|\x{49}|\x{4A}|\x{4B}|\x{4C}|\x{4D}|\x{4E}|\x{4F}|\x{50}|\x{51}|\x{52}|\x{53}|\x{54}|\x{55}|\x{56}|\x{57}|\x{58}|\x{59}|\x{5A}|\x{5B}|\x{5C}|\x{5D}|\x{5E}|\x{5F}|\x{60}|\x{61}|\x{62}|\x{63}|\x{64}|\x{65}|\x{66}|\x{67}|\x{68}|\x{69}|\x{6A}|\x{6B}|\x{6C}|\x{6D}|\x{6E}|\x{6F}|\x{70}|\x{71}|\x{72}|\x{73}|\x{74}|\x{75}|\x{76}|\x{77}|\x{78}|\x{79}|\x{7A}|\x{7B}|\x{7C}|\x{7D}|\x{7E}|\x{7F}|\x{80}|\x{81}|\x{82}|\x{83}|\x{84}|\x{85}|\x{86}|\x{87}|\x{88}|\x{89}|\x{8A}|\x{8B}|\x{8C}|\x{8D}|\x{8E}|\x{8F}|\x{90}|\x{91}|\x{92}|\x{93}|\x{94}|\x{95}|\x{96}|\x{97}|\x{98}|\x{99}|\x{9A}|\x{9B}|\x{9C}|\x{9D}|\x{9E}|\x{9F}|\x{A0}|\x{A1}|\x{A2}|\x{A3}|\x{A4}|\x{A5}|\x{A6}|\x{A7}|\x{A8}|\x{A9}|\x{AA}|\x{AB}|\x{AC}|\x{AD}|\x{AE}|\x{AF}|\x{B0}|\x{B1}|\x{B2}|\x{B3}|\x{B4}|\x{B5}|\x{B6}|\x{B7}|\x{B8}|\x{B9}|\x{BA}|\x{BB}|\x{BC}|\x{BD}|\x{BE}|\x{BF}|\x{C0}|\x{C1}|\x{C2}|\x{C3}|\x{C4}|\x{C5}|\x{C6}|\x{C7}|\x{C8}|\x{C9}|\x{CA}|\x{CB}|\x{CC}|\x{CD}|\x{CE}|\x{CF}|\x{D0}|\x{D1}|\x{D2}|\x{D3}|\x{D4}|\x{D5}|\x{D6}|\x{D7}|\x{D8}|\x{D9}|\x{DA}|\x{DB}|\x{DC}|\x{DD}|\x{DE}|\x{DF}|\x{E0}|\x{E1}|\x{E2}|\x{E3}|\x{E4}|\x{E5}|\x{E6}|\x{E7}|\x{E8}|\x{E9}|\x{EA}|\x{EB}|\x{EC}|\x{ED}|\x{EE}|\x{EF}|\x{F0}|\x{F1}|\x{F2}|\x{F3}|\x{F4}|\x{F5}|\x{F6}|\x{F7}|\x{F8}|\x{F9}|\x{FA}|\x{FB}|\x{FC}|\x{FD}|\x{FE}|\x{FF})/) {
    print "blub";
}
```

#3 - 29.11.2023 19:58 - Jan Büren

Noch die Ergänzung von Sven von damals dazu:
Comment (by s.schoeling@...):

Das sind aber zwei verschiedene paar Schuhe Jan.

Der Artikel den Du gelinkt hast geht um invalid UTF8, sprich Bytesequenzen die nicht korrektes UTF8 sind, sondern binärer Müll. Das ist aber nicht das was beim Drucken auftritt.

Was beim LaTeX Drucken Probleme macht sind valide UTF-8 Zeichen, für die LaTeX nur keine Glyphe parat hat. Sehr häufig ist das ' ' (U+00A0 NO-BREAK-SPACE, wird aus generiert), typographische Anführungszeichen wie sie MS-Office ungefragterweise in allem generiert (‘ ’ , “ ” „), irgendwelcher anderer typographischer Müll der in solchen Strings eigentlich nichts zu suchen hat (Œ - LATIN SMALL LIGATURE FF), oder einfach mal Zeichen auf den wilden Weiten von Unicode (◊, ◊, R. ◊, ◊. ◊, ☐)

Ich hab mich da schonmal mit beschäftigt, und es ist sehr schwer a priori rauszufinden was LaTeX kann und was nicht. Locales und Sprachpakete können Zeichen, andere funktionieren einfach so, andere lassen das in jeder Form abkacken.

Dateien

unsichtbare-steuerzeichen-fehler-eingrenzen.png	19,7 KB	17.11.2015	Jan Büren
diff-OE-pm.png	59,9 KB	17.11.2015	Jan Büren