

Ein Renderer-Plugin für DokuWiki

...das mir leider nicht geglückt ist 😞

Mich stört, dass im Sourcecode von DokuWiki die einfachen und doppelten Anführungszeichen fest codiert sind. Zur Zeit behebe ich das durch einen Flicker im verwendeten  ACH-Template:

```
<?php
    ob_start();
    tpl_content();
    $myhtml = ob_get_contents();
    ob_end_clean();
    echo wfModiOut('screen', $myhtml);
?>
```

Dabei ist wfModiOut() eine Funktion, die im Wesentlichen aus zwei Befehlen besteht:

```
$retval = str_replace('&ldquo;', '&raquo;', $retval);
$retval = str_replace('&rdquo;', '&laquo;', $retval);
```

Ich dachte mir, dass ein Renderer-Plugin so etwas überflüssig macht, und ging nun daran, mir die Freckelei im Template mit folgendem Stück Code zu ersparen:

```
<?php

/**
    MODiFied RENDerer for XHTML output
    @license    GPL 2 (http://www.gnu.org/licenses/gpl.html)
    @author Werner Flamme <w.flamme@web.de>
**/
// the following lines are stolen from the s5 plugin of Harry Fuecks and
Andi Gohr
// must be run within Dokuwiki
if(!defined('DOKU_INC')) die();

// we inherit from the XHTML renderer instead directly of the base renderer
require_once DOKU_INC.'inc/parser/xhtml.php';
// stolen code ends here

if (!defined('DOKU_PLUGIN'))
    define('DOKU_PLUGIN', DOKU_INC . 'lib/plugins/');
require_once DOKU_PLUGIN . 'modirend/conf/default.php';

/**
    This class allows to change the double and single quotation marks,
    something I always wanted but what never got implemented.
    Instead of modifying 4 lines in DokuWikis source code
    I have to write a plugin :-)
```

```
/**/  
class Doku_Renderer_modirend extends Doku_Renderer_xhtml {  
  
    /**  
        Return some info  
    **/  
    function getInfo(){  
        return array(  
            'author' => 'Werner Flamme',  
            'email'  => 'w.flamme@web.de',  
            'date'   => '2007-01-13',  
            'name'   => 'MODIfied RENDerer',  
            'desc'   => 'Changes quote marks',  
            'url'    =>  
            'http://www.wernerflamme.name/users:werner:modirend',  
        );  
    } // function getInfo  
  
    /**  
        add opening single quotation mark to document's code  
    **/  
    function singlequoteopening()  
    {  
        global $conf;  
        $default = '&lsquo;'; // from the standard renderer  
        $insert  = $conf['modirend_sq_open'];  
        $this->doc .= (strlen($insert) > 0) ? $insert : $default;  
    } // function singlequoteopening  
  
    /**  
        add closing single quotation mark to document's code  
    **/  
    function singlequoteclosing()  
    {  
        global $conf;  
        $default = '&rsquo;'; // from the standard renderer  
        $insert  = $conf['modirend_sq_close'];  
        $this->doc .= (strlen($insert) > 0) ? $insert : $default;  
    } // function singlequoteclosing  
  
    /**  
        add opening double quotation mark to document's code  
    **/  
    function doublequoteopening()  
    {  
        global $conf;  
        $default = '&ldquo;';  
        $insert  = $conf['modirend_dq_open'];  
        $this->doc .= (strlen($insert) > 0) ? $insert : $default;  
    } // function doublequoteopening
```

```

/**
    add closing double quotation mark to document's code
**/
function doublequoteclosing()
{
    global $conf;
    $default = '&rdquo;';
    $insert = $conf['modirend_dq_close'];
    $this->doc .= (strlen($insert) > 0) ? $insert : $default;
} // function doublequoteclosing
} // class Doku_Renderer_modirend

//Setup VIM: ex: et ts=4 enc=utf-8 :

```

Ich benutze also den Standard-XHTML-Renderer und überschreibe 4 seiner Methoden. Wenn ich ein `&do=export_modirend` manuell an die URL anhängen (oder über `exportlink($ID, 'modirend')` die Daten sehen will), erscheinen „meine“ Anführungszeichen. Das ist die Theorie.

In der Praxis führt das `&do=export_modirend` dazu, dass der Dokumenteninhalte zwar angezeigt wird, aber ohne meine Anführungszeichen.

Des Weiteren führte das dazu, dass mein Apache auch auf anderen Seiten und ohne `&do=export_modirend` nur noch Fragmente ausgab – im Error-Log fand ich nur einen Hinweis auf einen Segfault, keinerlei PHP-Fehler.

Ebenso erhalte ich einen Segfault, wenn ich folgendes in mein Template einbaue:

```

if ($fpo = fopen('old.txt', 'w')) {
    fwrite($fpo, wfModiOut('screen', $myhtml));
    fclose($fpo);
} // if ($fpo = fopen('old.txt', 'w'))
if ($fpo = fopen('new.txt', 'w')) {
    $newlink = 'http://' . $_SERVER['SERVER_NAME'] . exportlink($ID,
'modirend');
    echo "$newlink<br />\n";
    $newlink = str_replace('&', '&', $newlink);
    fwrite($fpo, "$newlink\n");
    if ($newc = fopen($newlink, 'r')) {
        while (!feof($newc)) {
            $newtxt = fread($newc, 1024);
            fwrite($fpo, $newtxt);
        } // while (!feof($newc))
        fclose($newc);
        fclose($fpo);
    } // if ($newc = fopen($newlink, 'r'))
} // if ($fpo = fopen('new.txt', 'w'))

```

Die Datei `old.txt` enthält den Text wie er angezeigt wird, `new.txt` enthält keine

Anführungszeichen – wenn überhaupt etwas passiert, denn meist segfaultet der Apache  Er tut

das übrigens oft unabhängig von diesem Codestück. 😞

Ich dachte zuerst, dass – aus welchem Grund auch immer – innerhalb der Funktionen die von mir gesetzten Zeichen nicht bekannt sind, deshalb das Setzen von `$default` und die Abfrage von `strlen($insert)`. Aber in `new.txt` erscheint an den jeweiligen Stellen *gar kein* Anführungszeichen.

Wenn mein Renderer *gar* nicht aufgerufen würde, würde entweder *gar* nichts passieren (bzw. eine Meldung kommen, dass der Renderer nicht zur Verfügung steht) oder die Default-Zeichen müssten erscheinen. Aber *gar nichts*?

Frage: warum stellt sich das erwartete Ergebnis – also ein Text, der genau so aussieht wie der standardmäßig produzierte, nur mit anderen Anführungszeichen – nicht ein? Wie benutze ich das Plugin sachgerecht?

From:

<http://wernerflamme.name/> - **Werners Wiki**

Permanent link:

<http://wernerflamme.name/doku.php?id=users:werner:modirend>

Last update: **2007-01-13 21:30**

