

# Deutsche Dokumentation für CMSimple (v. 2.x)

CMSimple ist ein einfaches und dennoch mächtiges Content Management System, welches keine MySQL-Datenbank benötigt und auf jedem Server funktioniert, der PHP unterstützt.

Entwickler und Rechteinhaber ist  
peter@harteg.dk  
<http://www.cmsimple.dk/>

Lizenz: <http://www.cmsimple.dk/?License>

Dieser Text ist im Wesentlichen die deutsche Übersetzung der englischen Anleitung (<http://www.cmsimple.dk/?Help>) und wurde von Jean-Christophe Haimb (jc@haimb.de – <http://www.haimb.de/>) am 11.11.2003 erstellt.

Inhalt:

- 1.) Installation
- 2.) Sicherheit
- 3.) Das Layout verändern
  - 3.1) template.htm
  - 3.2) stylesheet.css
- 4.) Der Online-Editor
- 5.) Neue Seiten erstellen
- 6.) CMSimple Scripting
  - 6.1) Meta-Tags
  - 6.2) Letztes Update
  - 6.3) geturl()
  - 6.4) Fotogalerie
  - 6.5) Seite verstecken
  - 6.6) Seite entfernen
  - 6.7) Ersetzen
  - 6.8) Zähler
  - 6.9) Einfügen aus einer Datei
  - 6.10) Einfügen aus einer PHP-Datei
  - 6.11) Gesamter Inhalt
  - 6.12) Banner-Rotation
- 7.) Variablen und Funktionen
- 8.) Support
- 9.) FAQ

## 1.) INSTALLATION

Lade zunächst die ZIP-Datei von der Download-Seite (<http://www.cmsimple.dk/?Download>) herunter, entpacke sie und übertrage die Dateien und Verzeichnisse auf den Webserver, auf dem Du CMSimple installieren willst. Normalerweise also in den Root-Pfad.

Hier eine Beschreibung der Dateien und Verzeichnisse:

**index.php** – ein kleines Script, welches die Datei cmsimple/cms.php aufruft.

**downloads/** – das Verzeichnis für Dateien, die Du zum Download anbieten möchtest. Alle dort enthaltenen Dateien stehen automatisch in einer Auswahl-Box im Editor zur Verfügung.

**images/** – das Verzeichnis für die Bilder der Website. Alle dort enthaltenen Dateien stehen automatisch in einer Auswahl-Box im Editor zur Verfügung. Das darin enthaltene Unterverzeichnis „/flags“ enthält Länderflaggen für mehrsprachige Installationen. Ist in „images/“ ein Unterverzeichnis-Editor eingerichtet, wird der Editor versuchen seine Bilder dort zu finden, ansonsten benutzt er die in „cmsimple/image.php“ definierten.

**cmsimple/** – Das Herzverzeichnis, hier werden auch Backups eingestellt.

**.htaccess** – mit dieser Datei kann das Verzeichnis auf Linux-Systemen gesperrt werden.

**cms.php** – das Content Management System selbst.

**image.php** – die für Menü und Editor verwendeten Bilder.

**log.txt** – die Log-Datei für das Login.

**config.php** – die Konfigurations-Datei für die Website.

**cmsimple/languages/** – Das Verzeichnis für die verschiedenen Sprachen. Jede Sprachdefinition hat einen Dateinamen aus der Länderidentifikation und „.php“. Deutsch ist also „de.php“. Voreinstellung ist Englisch und kann im Konfigurationsmenü angepasst werden.

**content/** – Das Verzeichnis für die Inhalte

**.htaccess** – mit dieser Datei kann das Verzeichnis auf Linux-Systemen gesperrt werden.

**content.htm** – in dieser Datei werden die Inhalte der Website gespeichert.

**guestbook.txt** – in dieser Datei werden die Einträge in das Gästebuch gespeichert.

**templates/** – Das Verzeichnis für die Layout-Vorlagen. Um ein eigenes Layout zu entwerfen, kopiere das Unterverzeichnis „default“ und gib' ihm einen eigenen Namen – im Konfigurationsmenü kann diese Vorlage dann eingestellt werden.

**templates/default/** Das Verzeichnis mit dem Standard-Layout

**stylesheet.css** – die Datei, mit der die Cascading Style Sheets der Website zentral

definiert werden.

**template.htm** – das HTML-Layout der Website.

**images/** – das Unterverzeichnis für vom Layout abhängige Bilder. Wird das Unterverzeichnis gelöscht, werden die in „cmsimple/image.php“ definierten Bilder verwendet.

**2lang/** Dieses Verzeichnis wird für mehrsprachige Installationen verwendet. Um dies zu erreichen, kopiere es in das Root-Verzeichnis Deiner CMSimple-Installation und benenne es entsprechend der zu verwendeten Sprache um, z. B. „en“. Überprüfe, ob es im Verzeichnis „cmsimple/languages/“ eine entsprechende Sprachdatei und im Verzeichnis „images/flags/“ eine entsprechende Graphik gibt.

## **WICHTIG!**

Alle Verzeichnisse und ggf. Dateien, die mit CMSimple bearbeitet werden sollen, müssen schreibbar gemacht werden. Das oder die content-Verzeichnisse müssen ausführbar sein, um Backups erstellen zu können.

Bei einer vollen Installation mit „chmod 777“ folgenden Verzeichnisse vorbereiten:

cmsimple/

cmsimple/languages/

content/ (ggf. auch die Unterverzeichnisse bei einer mehrsprachigen Installation)

templates/default/

images/ (optional, wenn Bilder über CMSimple verwaltet werden sollen)

downloads/ (optional, wenn Downloads über CMSimple verwaltet werden sollen)

Unter Apache muss „cmsimple/config.php“ durch die .htaccess-Datei geschützt sein!

**NICHT VERGESSEN:** Die Login-Einstellungen werden in der „settings.php“ oder über CMSimple (Option „SETTINGS“ bzw. „EINSTELLUNGEN“) bestimmt.

Voreingestellt ist das Passwort „test“ (JavaScript) bzw. User „admin“ mit Passwort „test“ wenn „\$wwwaut“ auf „true“ eingestellt wurde (auf Linux-Systemen) – beides sollte für den Betrieb verändert werden.

## **ANMERKUNG ZUR RECHTEVERGABE:**

Bei vielen Providern ist ein Telnet- oder SSH-Zugang nicht verfügbar, die Rechte von Dateien müssen also mit dem FTP-Programm – etwa WS-FTP (Win32) oder RBrowser (MacOS X) – gesetzt werden. Der Unix-Befehl chmod 777 macht Dateien für alle Anwender les-, schreib- und ausführbar. Bei WS-FTP auf die Datei bzw. das Verzeichnis rechtsklicken und die entsprechende Option auswählen. Bei RBrowser Datei auswählen und Apfel-3 drücken und die entsprechenden Rechte setzen.

## 2.) SICHERHEIT

Für Apache-Benutzer: Kontrolliere, ob die „cmsimple/config.php“ durch die .htaccess-Datei geschützt wird!

Unter Win32 muss Apache so eingerichtet werden, dass es .htaccess-Dateien berücksichtigt. Unter MS IIS ist es NICHT möglich, diese Dateien zu nutzen.

NICHT VERGESSEN:

Die Zugangsdaten in der „config.php“ verändern! Das kann in der Konfiguration von CMSimple unter „SETTINGS“ bzw. „EINSTELLUNGEN“ geschehen. Voreingestellt sind „admin“ und „test“.

Auf Linux-Systemen „\$wwwaut“ auf „true“ stellen, um den höchsten Schutz zu erreichen.

Aus Sicherheitsgründen wäre es überlegenswert, den „login“-Link aus der HTML-Vorlage zu entfernen. Die Administration bleibt dabei erreichbar:

<http://www.deinedomain.de/&login>

Derzeit wird die Login-Information unverschlüsselt in einem Cookie gespeichert und könnte somit zumindest theoretisch durch jemanden erschlichen werden, der Deine Internet-Verbindung überwachen würde. In späteren Versionen von CMSimple soll u. a. eine Verschlüsselung hierfür eingebaut werden.

Vorschläge zur Verbesserung der Sicherheitsfragen nimmt der Autor von CMSimple gerne entgegen!

### 3.) DAS LAYOUT VERÄNDERN

Das Layout kann ohne Probleme an Deine Bedürfnisse angepasst werden. Einfach nur die Dateien „template.htm“ und „stylesheet.css“ editieren und Dein eigenes Layout steht.

Der Autor von CMSimple bittet darum, diese Dateien auch anderen Benutzern zugänglich zu machen. Ein Weg der Veröffentlichung ist, ihm die Dateien zu mailen.

#### 3.1) template.htm

Diese Vorlagendatei besteht aus reinem HTML mit PHP-Aufrufen an die Datei „cms.php“.

Einige Funktionen haben ihre eigene Stylesheet-Klasse, die in der Zelle einer Tabelle (<TD>) oder in einem Link (<A HREF>) bestimmt wird. Die Klassen werden im Stylesheet definiert.

Funktionen:

Einige Funktionen sind wichtig und dürfen nicht ausgelassen werden, andere können nach Belieben verwendet werden.

Unerlässlich: head(), onload(), toc(), editmenu() und content()

head() : erstellt den Link zum Stylesheet, die Meta-Tags und den Seitentitel.

onload() : wird für die Funktion des Editors verwendet und muss im Body-Tag eingetragen sein.

sitename() : schreibt den Namen der Website

searchbox() : erstellt die Such-Box.

toc() : schreibt das Inhaltsverzeichnis.

sitemaplink() : erstellt den Link zur automatischen Sitemap.

printlink(): erstellt den Link für die Darstellung in einem Drucker-Freundlichen Format.

loginlink() : erstellt den Link um als Administrator einzuloggen (nur sichtbar, wenn Du nicht eingeloggt bist).

mailformlink() : erstellt den Link zum Mail-Formular.

guestbooklink() : erstellt den Link zum Gästebuch.

languagemenu() : erstellt Links für verschiedene Sprachversionen.

lastupdate() : schreibt Datum und Zeit des letzten Updates.

editmenu() : erstellt im Admin-Modus das Admin-Menü.

content() : schreibt den Inhalt der Seite.

submenu() : erstellt bei Verfügbarkeit den Link zum Untermenü.

previouspage() : erstellt bei Verfügbarkeit den Link zur vorherigen Seite.

top() : erstellt einen Link zu #TOP

nextpage() : erstellt bei Verfügbarkeit den Link zur folgenden Seite.

locator() : schreibt die Navigation.

### 3.2) stylesheet.css

Das Stylesheet liegt im Vorlagenverzeichnis für das Layout. Hier werden alle Klassen der Stile zentral definiert (CCS, Cascading Style Sheet).

Über CSS werden die Formate der Schriften für die verschiedenen Funktionen und Inhalte gesteuert.

#### 4.) DER ONLINE-EDITOR

Mit dem Online-Editor ist es möglich die Seiten-Inhalte WYSIWYG (What you see is what you get – Was Du siehst ist was Du bekommst) zu erstellen. Dabei ist es möglich Inhalte samt Formatierung aus anderen Programmen – wie z. B. Office – zu kopieren. Weiterhin ist es möglich, im HTML-Modus direkt in den Code einzugreifen.

Hierzu muss der verwendete Browser allerdings kompatibel sein. Der Editor funktioniert mit IE ab Version 5.5 sowie Mozilla ab Version 1.3 (also auch der aktuellen Netscape-Version). JavaScript muss aktiviert sein.

Der Editor funktioniert nicht mit IE Version 5 und kleiner sowie Opera oder Safari.

Alle Funktionen des Editors werden durch Buttons gesteuert und entsprechen denen gängiger Editoren und Textverarbeitungen. Die Beschreibung einer jeden Funktion wird beim „berühren“ des Buttons mit der Maus angezeigt.

Um bei der Eingabe anstatt der P-Tags mit BR-Zeilenumbrüchen zu arbeiten, [Shift]-[Enter] benutzen.

Für die Verlinkung interner Seiten wird die entsprechende Auswahl-Box verwendet. Text zum Verlinken markieren, Seitenüberschrift in der Box auswählen und bestätigen. Vorsicht: wird die Überschrift einer Seite später verändert, funktioniert ein solcher Link dann nicht mehr. Um solche „toten“ Links zu finden, dient die Funktion [VALIDATE LINKS] bzw. [LINKS PRÜFEN].

Analog funktioniert die Erstellung von Links auf Dateien zum Download bzw. der Einbau von Bildern in die Seite – sofern die Dateien / Bilder in den dafür vorgesehenen Verzeichnissen liegen (Upload per WWW oder FTP).

CSS-Stile werden über die „stylesheet.css“ definiert.

## 5) NEUE SEITEN ERSTELLEN

Um neue Seiten und auch Unterseiten zu erstellen, nutzt CMSimple die sogenannten Headings. Dieses Konzept mag ungewohnt sein, aber einmal begriffen, ist es sehr leicht damit neue Haupt- und Unterseiten zu erstellen.

Die Headings <H1>, <H2> und <H3> entscheiden darüber, auf welchem Niveau der Navigation eine Seite eingeordnet wird. <H4> wird für normale Überschriften innerhalb einer Seite verwendet.

Beispiel:

H1-Überschrift-1

    H2-Überschrift-A

        H3-Überschrift-a

    H2-Überschrift-B

    H2-Überschrift-C

H1-Überschrift-2

Um zwischen H2-Überschrift-A und H2-Überschrift-B eine neue Seite auf H2-Niveau einzufügen, musst Du im Editor die Seite „H3-Überschrift-a“ öffnen und an ihrem Ende mit der neuen H2-Überschrift-A2 anfangen. Analog müsstest Du am Ende der Seite „H2-Überschrift-C“ mit einer H1-Seite beginnen, die zwischen „H1-Überschrift-1“ und „H1-Überschrift-2“ einsortiert werden soll.

Um eine Seite zu löschen, muss diese im Editor geöffnet werden und alle Inhalte entfernt werden. Eventuell verbleibt noch HTML-Code der ebenfalls zu löschen wäre. Um diese zu entfernen, in die HTML-Ansicht wechseln und verbliebenen Code löschen. Die Löschung durch die „Speicherung“ der nun leeren Seite bestätigen.



## 6.) CMSimple SCRIPTING

Mit CMSimple Scripting kann in den Inhalten gscriptet werden. Bearbeitung wird im Edit-Modus und ausgeführt im normalen Modus.

Der folgende Code ändern den Titel der Seite (und nur dieser) in der er ausgeführt wird:

```
#CMSimple $GLOBALS['cf']['site']['title']='hi dude!'; #
```

Auf diese Art ist es möglich nahezu alle von CMSimple verwendeten Variablen zu verändern!

Normalerweise wird CMSimple-Code nur im Editier-Modus angezeigt. Um ihn auf einer Beispielseite zu darzustellen, muss ein FONT-Tag vor das erste Numsign (#). Wenn Du also solchen Code von einer anderen Website per Copy&Paste kopieren willst, könnte es passieren, dass der Code nicht ausgeführt wird, weil der (unsichtbare) FONT-Tag mitkopiert wurde. In der HTML-Ansicht prüfen und ggf. entfernen.

Mit Ausnahme von „hide“ und „remove“ handelt es sich um normalen PHP-Code, der ausgeführt wird.

Im Normalfall lässt sich auf einer Seite nur ein CMSimple-Script ausführen.

### 6.1) Meta-Tags

Ein häufig verlangtes Feature ist die Möglichkeit einer Seite eigene Meta-Tags zuweisen zu können. Dies kann mit den Variablen \$keywords und \$description erreicht werden:

```
#CMSimple $keywords='einige,woerter'; $description='Eine  
Seitenbeschreibung'; #
```

### 6.2) Letztes Update

Es kann eine Funktion wie „lastupdate()“ aus cms.php eingefügt werden:

```
#CMSimple $output.=lastupdate(); #
```

Siehe auch „6.7) Ersetzen“

### 6.3) geturl()

In CMSimple wurde eine Funktion integriert, mit der der Inhalt einer externen HTML-Seite in eine eigene eingefügt werden kann. Hierbei aber bitte keine fremden Rechte missachten!

```
#CMSimple
$output.=geturl('http://www.cmsimple.dk/index.php?&print');#
```

allow\_url\_fopen muss in der php.ini des Servers aktiviert sein, damit dies funktioniert.

Darüber hinaus gibt es die Funktion `geturlwp()`, die den URL samt Parametern holen kann (z. B. in der Fotogalerie verwendet)

#### 6.4) Fotogalerie

CMSimple kann mit der Autogallery von circle.ch verwendet werden, weiterführende Informationen unter <http://www.circle.ch/scripts/>

```
#CMSimple
$output.=autogallery('http://www.circle.ch/scripts/test/autogallery/');#
```

allow\_url\_fopen muss in der php.ini des Servers aktiviert sein, damit dies funktioniert.

#### 6.5) Seite verstecken

Mit dem folgenden Code kann die bearbeitete Seite aus der Navigation herausgehalten werden, bleibt jedoch erreichbar:

```
#CMSimple hide#
```

Sie wird im Inhaltsverzeichnis nun nur im Editier-Modus angezeigt, sie wird von der Suchmaschine nicht durchsucht und in der Sitemap ebenfalls nicht angezeigt. Der Locator zeigt nur den Titel der Seite.

Da der Navigator sie ebenfalls nicht anzeigt, sollten „unsichtbare“ Seiten am Ende der Site-Struktur angebracht werden. Eventuelle Unterseiten einer „unsichtbaren“ Seite sollten ebenfalls versteckt werden.

Im Normalfall ist es nicht möglich, zwei CMSimple-Scripte in einer Seite auszuführen, allerdings ist es möglich eine Seite per Script zu verstecken und ein weiteres Script auszuführen:

```
#CMSimple hide#
#CMSimple $output.=geturl('http://www.harteg.dk').chr(35).'CMSimple hide'.chr(35);#
```

Der letzte Teil des Codes wird hier benötigt um sicherzustellen, dass die Seite auch beim Betrachten aus dem Inhaltsverzeichnis ausgeblendet bleibt.

## 6.6) Seite entfernen

Der folgende Code entfernt eine Seite komplett, ohne sie zu löschen. Sie ist dann nur im Editier-Modus sichtbar.

```
#CMSimple remove#
```

## 6.7) Ersetzen

Um die Ausgabe des CMSimple Scripting an eine spezifische Stelle innerhalb einer Seite anzubringen, muss der Code in der Seite wie folgt ersetzt werden:

```
#CMSimple
$output=preg_replace("/".chr(35)."CMSimple.*".chr(35)."/",lastupdate
(),$c[$s]);#
```

*Last update:*

*November 10. 2003 08:31:09*

In diesem Beispiel wird lastupdate() an die Stelle eingefügt an der sich das CMSimple Script befindet.

## 6.8) Zähler

Um in einer Seite einen seitenspezifischen Zähler einzubauen, kann wie folgt vorgegangen werden:

```
Aufrufe dieser Seite: #CMSimple
$i=0;$t="./count/".$su.".inc";if($fh=fopen($t,"a+")){$i=fread($fh,
filesize($t));$i++;fclose($fh);if($fh=fopen($t,"w")){fputs($fh,$i);f
close($fh);$output=preg_replace("/".chr(35)."CMSimple.*".chr(35)."/"
,$i,$c[$s]);}} #
```

**Ausgabe:**

*Aufrufe dieser Seite: 158*

Zusätzlich muss es im Rootverzeichnis ein Verzeichnis „count/“ existieren und mit chmod 777 les-, schreib- und ausführbar gemacht werden.

## 6.9) Einfügen aus einer Datei

Um den Inhalt einer Datei in eine CMSimple-Seite einzufügen, kann folgender Code verwendet werden:

```
#CMSimple
$t="./index.php";if($fh=fopen($t,"r")){$output=preg_replace("/".chr(35)."CMSimple.*".chr(35)."/",fread($fh, filesize($t)), $c[$s]);fclose($fh);} #
```

(Um mit diesem Beispiel den PHP-Code der „index.php“ anzuzeigen, wurde ein textarea-Tag gesetzt)

## 6.10) Einfügen aus einer PHP-Datei

Um den Output einer PHP-Datei in eine CMSimple-Seite einzufügen, Code wie den folgenden verwenden:

```
#CMSimple
$output=preg_replace("/".chr(35)."CMSimple.*".chr(35)."/",geturl('http://www.cmsimple.dk/demo/?&print'), $c[$s]);#
```

## 6.11) Gesamter Inhalt

Um den gesamten Inhalt Deiner CMSimple-Website in eine einzige Seite auszugeben, kann folgender Code verwendet werden:

```
#CMSimple $output.=rf($pth['file']['content']);#
```

Wenn dabei CMSimple-Code erhalten bleiben soll:

```
#CMSimple
$output.=preg_replace("/".chr(35)."/s",chr(38).chr(35)."35;",rf($pth['file']['content']));#
```

## 6.12) Banner-Rotation

Um in einer Seite eine Banner-Rotation einzubauen, kann ein Code wie der folgende verwendet werden:

```
#CMSimple
$t=array(array("bp_cmsimple.png","http://www.cmsimple.dk"),array("cms_powered.png","http://www.cmsimple.dk/demo/"),array("cmsimple80x20.png","http://www.cmsimple.dk/forum")); $i=rand(0, count($t)-1); $output=preg_replace("/".chr(35)."CMSimple.*".chr(35)."/",chr(60)."a href='". $t[$i][1]."'".chr(62).chr(60)."img src='./images/'". $t[$i][0]."'".chr(62), $c[$s]);#
```

## 7.) VARIABLEN UND FUNKTIONEN

### Variablen:

\$c content array  
\$e error output  
\$f function  
\$fh used for file handle  
\$h heading array  
\$i and \$j used in for/next  
\$l level  
\$o output in content  
\$s selected number  
\$sn scriptname  
\$su selected url  
\$t used for temp variable  
\$ts table start  
\$u url of the page

### Funktionen:

gc get cookie  
ml menu link  
rf readfile  
rfc readfile content

## 8.) SUPPORT

Support findest Du unter:

<http://www.cmsimple.dk/forum/>

es wäre auch nett, wenn Du selbst Hilfe geben könntest, wenn Du kannst :-)

## 9.) FAQ

### **Wie verschiebe ich Seiten?**

Alles Markieren, ausschneiden, speichern. Dann an das Ende der Seite gehen, hier die die Seite verschoben werden soll, dort einfügen und wieder speichern.

Siehe auch: 5.) Neue Seiten erstellen

### **Wie kann ich bilder uploaden?**

In CMSimple ab Version 2.x, allerdings nur unter PHP4

Wenn Du in der CMSimple-Konfiguration die FTP-Einstellungen vornimmst, kannst Du IE als FTP-Programm verwenden.

### **Wie kann ich die Bilder im Menü loswerden oder gegen andere tauschen?**

Die verwendeten Bilder sind im Unterverzeichnis des Vorlagenverzeichnisses, also z. B. in `templates/default/menu/`.

Ist dieses Unterverzeichnis gelöscht, holt sich CMSimple die Bilder aus der Datei `cmsimple/image.php`.

Wenn Du diese Bilder verwenden willst, lösche das `menu/`-Unterverzeichnis und ändere die Farben der Icons des Dokuments in der Konfiguration.

Das HTML für diese Bilder ist in der `tr`-Funktion in `cms.php` definiert und kann dort auch verändert werden, z. B. wenn Du keine Bilder haben möchtest.

Mit folgendem PHP-Code kannst Du den binären Wert von gif-Bildern errechnen (im gleichen Verzeichnis wie der Code) und diese Werte in die `image`-Funktion in `image.php` einfügen:

```
$localdir='.';
if ( is_dir($localdir) ) {
$fd = @opendir($localdir);
while ( ($part = @readdir($fd)) == TRUE ) {
if (preg_match("/\.(gif)$/i", $part)) {
$file_handler = fopen($part, "r");
$content = fread($file_handler, filesize($part));
fclose($file_handler);
echo 'if ($image==\''. $part. '\') return \'' . bin2hex($content) . '\';
';}}
if($fd == TRUE) { closedir($fd); }
} else { echo "Could not open dir"; }
```

Das `binimg.php`-Script gibt es hier:

<http://www.cmsimple.dk/?download=binimg.php> (0,4kb)

In diesem Zusammenhang dürfte Dich auch <http://php.holtsmark.no/base64img/> interessieren.

### **Ist es möglich, andere Scripte in meine Seiten einzufügen?**

Es ist möglich, eine Seite aus dem Netz zu „Grabben“ und den Inhalt einer Seite mit CMSimple-Scripting zu ersetzen. Siehe hierzu 6.) CMSimple-Scripting.

### **Kann ich eine Newsbox machen?**

Ja. In cms.php gibt es eine newsbox()-Funktion. Der Aufruf erfolgt mit `<?=newsbox('Die exakte Überschrift der Seite');?>` in der Vorlagendatei (template.htm).

### **Irgendwas klappt nicht! :-)**

#### **Wenn Funktionen in der Vorlagendatei nicht ausgeführt werden:**

Auf manchen Systemen muss die Erweiterung von file\_content von .htm auf .shtml oder .php geändert werden, damit CMSimple funktioniert.

#### **Egal welche Seite ich auswähle, der Inhalt verändert sich nicht!**

Aus irgendeinem Grund, gibt das freie PHP-Hosting bei portland.co.uk `$HTTP_SERVER_VARS['QUERY_STRING']` nicht zurück.

Die Lösung ist folgenden Code an den Anfang der Funktion sv(\$s) anzufügen:  
`if ($s=='QUERY_STRING') return $GLOBALS['QUERY_STRING'];`