Tips für Webmaster

Linkdesign - Apaches mod_rewrite

Das Design von Links wird immer wichtiger - denn dynamische Seiten haben einen großen Nachteil: Die oftmals langen und komplizierten URLs sind bei Suchmaschinen sehr unbeliebt und lassen sich oftmals schlecht weitergeben oder in Print-Medien abdrucken.

Abhilfe schafft dort eine Technik des Apache-Webservers namens "mod_rewrite". Hiermit lassen sich virtuelle, nicht existierende Adressen erstellen, die auf die realen und komplizierten Adressen zurückgreifen. Das Prinzip ist ganz einfach: Aus Adressen wie seite.php?rubrik=news&id=4&modus=2 wird news,4,2.html oder news.4.2.html. Dem Aussehen der neuen Links sind dabei kaum Grenzen gesetzt.

Vorraussetzungen
Benötigt wird ein Apache-Webserver mit installierten mod_rewrite-Modul. Falls es sich hierbei nicht um den eigenen Webserver handelt, muss der Anbieter zudem die Nutzung von .htaccess-Dateien erlauben.

Verfahrensweise
Als erstes muss eine Datei namens ".htaccess" in das Verzeichnis des Webservers gelegt werden, in welchem wir arbeiten möchten. Ist bereits eine vorhanden, nutzen wir diese. Über allen weiteren Anweisungen zum Thema mod_rewrite muss zunächst die sogenannte RewriteEngine aktiviert werden.

```
RewriteEngine on
```

Nehmen wir nun an, unsere bisherige Seite weist die folgende Struktur auf: seite.php?rubrik=[VARIABLE 1]&id=[VARIABLE 2]&modus=[VARIABLE 3]. Wobei die Variablen beliebige Werte enthalten können.
Unser Ziel ist es, daraus eine kurze URL in der Forum [VARIABLE 1],[VARIABLE 2],[VARIABLE 3].html zu machen.

> Beispiel:
> Aus seite.php?rubrik=news&id=4&modus=2 würde dann news,4,2.html

Der Code für diese Umwandlung sähe wie folgt aus:

```
RewriteRule ^(.*),(.*),(.*).html$ seite.php?rubrik=$1&amp;id=$2&amp;mo
dus=$3
```

Zunächst folgt hinter dem Befehl Rewrite Rule das neue Design der Links, beginnend mit einem "^", abschließend mit dem Dollar-Zeichen. "(.*)" steht dabei für eine beliebige Variable. Dahinter folgt die URL der "reellen" Datei. Die Variablen (.*) der neuen Datei werden hier in der Reihenfolge von Links nach Rechts mit $1 bis $3 (in unserem Beispiel) angegeben.

(c) 2024 Support Team <support@suleitec.de> | 2024-03-29 13:02
URL: https://www.webhosting-handbuch.de/content/10/25/de/linkdesign-_-apaches-mod_rewrite.html

In der Adress-Leiste des Browsers bleibt beim Aufruf von news,4,2.html diese Adresse stehen. Der Server liefert jedoch den Inhalt von seite.php?rubrik=news&id=4&modus=2 aus. Sowohl dem Browser bzw. Surfer als auch Suchmaschinen bleibt die wahre Adresse des Dokumentes verborgen.

Solche Befehle lassen sich nun in beliebiger Anzahl nacheinander setzen. Wichtig ist, dass nach jeder RewriteRule ein Zeilenumbruch erfolgt. Hinter der letzten Anweisung sollte ein [L] folgen, um dem Apache-Server zu signalisieren, dass an dieser Stelle die Anweisungen zu Ende sind. Dies lässt sich mit beliebig vielen Variablen erweitern.

Beispiel

Inhalt der .htaccess-Datei:

```
RewriteEngine on
RewriteRule ^seite(.*).html$ daten/seite.php?id=$1
RewriteRule ^artikel/(.*)/seite(.*).html$ artikel.php?id=$1&amp;seite=$2
RewriteRule ^forum(.*)_(.*).html$ index.php?forum=$1&amp;start=$2 [L]
```

So wird z.B. aus daten/seite.php?id=14 einfach: seite14.html
So wird z.B. aus artikel.php?id=23&seite=2 einfach: artikel/23/seite2.html
So wird z.B. aus index.php?forum=2&start=450 einfach: forum_2_450.html

Möglichkeiten
Die oben beschriebene Möglichkeit der erzeugung virtueller URLs ist nur eine von vielen Möglichkeiten, die durch das Apache-Modul mod_rewrite ermöglicht werden. Eine gesamte Beschreibung von mod_rewrite findet sich in der (englischen) Dokumentation für den Apache-Server. Nutzer von Apache 2.0 finden eine Anleitung zu mod_rewrite in der Apache 2.0 - Dokumentation.

Links zum Thema:
Apache 1.3 Dokumentation: mod_rewrite
Apache 2.0 Dokumentation: mod_rewrite

Eindeutige ID: #1024
Verfasser: Michael
Letzte Änderung: 2017-12-18 13:59

(c) 2024 Support Team <support@suleitec.de> | 2024-03-29 13:02
URL: https://www.webhosting-handbuch.de/content/10/25/de/linkdesign-_-apaches-mod_rewrite.html