

Introduktion til Pæne URLs med mod_rewrite

Skrevet d. 12/02, 2007 kl 11:11 af Martin Visninger: 1448 [Læs/skriv kommentar \(18\)](#)

Mod_Rewrite (module rewrite) er et modul i Apache webserveren. Modulet muliggør "omskrivning" af forespørgsler til serveren. Dvs. brugeren indtaster en URL men serveren sender denne forespørgsel videre til en anden fil. Dette ses bl.a. her på sitet, hvor man ved kontakt siden kommer ind på <http://martin-nielsen.com/kontakt> - det der så sker er, at serveren sender forespørgslen videre til <http://martin-nielsen.com/index.php?navid=kontakt> På den måde vil man på index.php stadig kunne modtage navid som GET variabel, og dermed kan vi stadig bruge almindelige includes.

I gang med mod_rewrite

Da dette er en introduktion, går vi ikke i dybden med mod_rewrite, men tager det fra bunden og stille og roligt. Trin 1 er at lave en .htaccess fil. Denne fil laver du i fx notepad, og når du gemmer skal du så huske, at vælge "alle filer" i filtype – ellers vil filen blive gemt forkert. (NB husk punktummet foran htaccess.)

For at "starte" mod_rewrite skal den første linje i .htaccess filen være: RewriteEngine on

Efter mod_rewrite er startet, skal man give serveren instrukser, der fortæller den hvornår, og hvordan der skal redirectes, dette sker i følgende format:

```
RewriteRule Mønster Erstatning [Valgfrie tags]
```

Mønster

Vha. regular expressions angives hvornår mod_rewrite skal reagere (det ser vi nærmere på via eksempler om lidt)

Erstatning

Hvad skal der rewrites til

Valgfrie tags

Denne del er ikke nødvendig for mod_rewrite, men man kan nogle ekstra ting her, et par eksempler på nyttige attributter er

- F - Forbudt - her vil brugeren simpelthen få en fejl 403 - not authorized
- L - Last rule - Indikerer, at der ikke skal behandles flere regler i mod_rewrite efter denne
- R[=code] - Brugeren bliver omstillet synligt, fx med koden 302 som er midlertidigt flyttet, eller 301 for permanent flytning af filen - rigtig smart hvis man flytter sit site til et andet domæne
- QSA - Querystring append - Hvis man ønsker, at kunne sende querystring variable med på sin mod_rewrite

I gang med eksempler

Ovenstående virker måske en smule teknisk, og måske også et langt skridt fra den lovet introduktion, men det er rart nok lige at ha' lidt mere viden inden vi går i gang. Men nok om det. Første eksempel:

```
RewriteEngine on
```

```
RewriteRule ^(/[^\./]+)?/$ index.php?side= [L]
```

Første linje starter som sagt mod_rewrite. Dernæst har vi en rewriterule. for at gøre det nemmere gennemgår vi den stykke for stykke.

^ Checker om linjen starter med det efterfølgende - hvis ikke ignoreres regelen ([^\./]+) () indikerer, at alt hvad der bliver matchet imellem parenteserne skal huskes. Inde i parenteserne har vi [^\./]+ Dette betyder i almindeligt sprog, match alt som ikke er en / eller et . (punktum) - (nb her er det vigtigt at . escapes med). + tegnet indikerer at vi ønsker et eller flere tegn imellem. /?\$ \$ indikerer at det sidste der skal matches her vil være en mulig / (den er ikke nødvendig da den efterfølges af ?) index.php?side=\$1 Den side som der skal omskrives til internt, her bliver hvad vi har matchet mellem () indsat via \$1 - hvis vi havde flere () sæt, ville disse blive \$2 \$3 osv osv. [L] Dette stopper mod_rewrite fra at udfører evt følgende rewrite regler

Herefter uploades .htaccess filen til rod biblioteket sammen med din index.php. Og i index.php vil du nu kunne modtage \$_GET['side'] [som min første tutorial gennemgik](#)

Jeg vil også ha' undersider

Ja men ok da! Det skal du få, princippet er det samme, så jeg vil nøjes med et eksempel uden den store forklaring

```
RewriteEngine on
RewriteRule ^(/[^\./]+)?/$ index.php?side= [L]
RewriteRule ^(/[^\./]+)/([^\./]+)/?$ index.php?side=&underside= RewriteEngine on
RewriteRule ^(/[^\./]+)?/$ index.php?side=$1 [L] RewriteRule
^(/[^\./]+)/([^\./]+)/?$ index.php?side=$1&underside=$2 [L] [L]
```

Her har vi simpelthen bare kopieret den første regel, og indsat en ekstra ([^\./]+) - og til slut sender vi det hele videre til index.php?side=\$1&underside=\$2

Betingelser

Du kan lave et hav af betingelser med mod_rewrite, men da målet med denne tutorial simpelthen er en grundlæggende ide om hvad man kan, holder vi os til det vigtigste. Som det er nu med eksemplet, vil alle mapper blive rewriteret også, og det er jo ikke meningen, Derfor laver vi et check om det vi prøver at rewrite er en fil eller mappe:

```
RewriteEngine on
RewriteCond %{REQUEST_FILENAME} -f [OR]
RewriteCond %{REQUEST_FILENAME} -l [OR]
RewriteCond %{REQUEST_FILENAME} -d
RewriteRule ^.* - [L]
RewriteRule ^(/[^\./]+)?/$ index.php?side= [L]
RewriteRule ^(/[^\./]+)/([^\./]+)/?$ index.php?side=&underside= RewriteEngine on
RewriteCond %{REQUEST_FILENAME} -f [OR] RewriteCond %{REQUEST_FILENAME} -l [OR]
RewriteCond %{REQUEST_FILENAME} -d RewriteRule ^.* - [L] RewriteRule
^(/[^\./]+)?/$ index.php?side=$1 [L] RewriteRule ^(/[^\./]+)/([^\./]+)/?$
index.php?side=$1&underside=$2 [L] [L]
```

REQUEST_FILENAME indeholder det vi har indtastet i browseren, -f -L og -d checker alle om det er en fil, link eller mappe, hvis det er det stopper vi rewrite med -

HJ?P min CSS, billeder osv virker ikke mere

Dette løber de fleste ind i. Dette skyldes, at browseren jo fx tror at /kontakt er en mappe, og den kan simpelthen ikke finde dine billeder og CSS filer inde i mappen /kontakt - Dette er nemt fixet, hvis dine billeder ligger inde i mappen images fx så sætter du bare / foran, dette vil fortælle browseren, at den skal lede efter images mappen, i roden af domænet, og vupti så virker det hele igen igen.

NYT: Har skrevet en række nye eksempler her:

http://martin-nielsen.com/blog/nyttige_mod_rewrite_eksempler-182.html

Konklusion

Mod_rewrite kan lave dine URLs meget pænere, og mere semantisk korrekte, dette er ikke bare en fordel for dine besøgende, men også en fordel for dig, i forhold til søgemaskineoptimering mm. Du kan omskrive alt, det er kun fantasien der sætter grænser, fx lad os sige du driver en webshop hvor URLs måske ser sådan her ud: index.php?side=Produkter&katalog=Film&Genre=Action - kunne meget nemt ud fra denne introduktion omskrives til /produktkatalog/film/action - som sagt kun fantasien der sætter grænser! God fornøjelse.

[Download eksempel](#)

[Vis eksempel](#)

[Apache mod_rewrite docs](#)

[The definitive guide to mod_rewrite \(bog\)](#)

[Flere eksempler](#)