

ApacheCon US 2004

Apache mod_rewrite

the Swiss Army Knife of URL manipulation

by Mads Toftum
mads@apache.org

Intro

"The great thing about mod_rewrite is it gives you all the configurability and flexibility of Sendmail.

The downside to mod_rewrite is that it gives you all the configurability and flexibility of Sendmail."

-- Brian Behlendorf
Apache Group

Intro

"Despite the tons of examples and docs, mod_rewrite is voodoo.
Damned cool voodoo, but still voodoo. "

-- Brian Moore

Intro

But all this functionality and flexibility has its drawback: complexity. So don't expect to understand this entire module in just one day.

docs/mod/mod_rewrite.html

Intro – agenda

- The main directives
 - RewriteRule
 - RewriteCond
- Comparing mod_rewrite to other solutions
- The directives
 - Main directives revisited
 - Other directives
- mod_rewrite solutions to common problems
- Questions

Intro – not covered

- Regular Expressions see:
 - perldoc perlre
 - "Mastering Regular Expressions" by Jeffrey Friedl

Directives – RewriteRule

Syntax: RewriteRule *Pattern Substitution*

Default: *None*

Context: server config, virtual host, directory, .htaccess

Override: *FileInfo*

- RewriteRule is the main directive in mod_rewrite
- RewriteRules are checked first
- RewriteRules are executed in the *URI translation* or *fixup* phase

Directives – RewriteCond

Syntax: RewriteCond *TestString CondPattern*

Default: *None*

Context: server config, virtual host, directory, .htaccess

Override: *FileInfo*

- RewriteCond is used to add conditions to RewriteRules
- One or more RewriteConds can precede a RewriteRule
- Multiple RewriteConds are joined by an implicit and unless [OR] is used
-

Imitation – Redirect

- The mod_alias version:

```
Redirect /test/ http://example.com/test/
```

```
RedirectPermanent /test2/ http://example.com/test2/
```

- The mod_rewrite version:

```
RewriteEngine On
```

```
RewriteRule ^/test/(.*) http://example.com/test/$1 [R]
```

```
RewriteRule ^/test2/(.*) http://example.com/test2/$1 [R=301]
```

Imitation – ProxyPass

- The mod_proxy version

```
ProxyPass /example/ http://example.com/
```

- The mod_rewrite version

```
RewriteRule ^/example/(.*) http://example.com/$1 [P]
```

Imitation – UserDir

- The mod_userdir version

UserDir /usr/web -> /usr/web/joe/one/two.html

UserDir /home/*/web -> /home/joe/web/one/two.html

- The mod_rewrite version

RewriteRule ^/~(.*)/(.*)\$ /usr/web/\$1/\$2

RewriteRule ^/~([a-z][a-z0-9]+)/(.*)\$ /home/\$1/web/\$2

Imitation – mod_vhost_alias

The vhost_alias version:

```
VirtualDocumentRoot /vhosts/%0/docs/
```

- The mod_rewrite version1:

```
RewriteEngine On
```

```
RewriteRule ^(.+)$ /vhosts/%{HTTP_HOST}/docs$1
```

- The mod_rewrite version2:

```
RewriteCond %{HTTP_HOST} ^(.*)$
```

```
RewriteRule ^(.*)$ /vhosts/%1/docs$1
```

Imitation – trailing slash

- Why does accessing directories only work when I include the trailing "/"

```
ServerName example.com
```

- The mod_rewrite solution:

```
RewriteCond    %{REQUEST_FILENAME}    -d  
RewriteRule    ^(.+[^/])$            $1/    [R,L]
```

Directives – RewriteRule 2

- RewriteRule *Pattern Substitution [Flags]*

[flags]:

- **'redirect|R [=code]'** (force redirect)
- **'proxy|P'** (force proxy)
- **'forbidden|F'** (force URL to be forbidden)
- **'gone|G'** (force URL to be gone)
- **'type|T=MIME-type'** (force MIME type)
- **'noescape|NE'** (no URI escaping of output)
- **'env|E=VAR:VAL'** (set environment variable)
- **'cookie|CO=NAME:VAL:domain[:lifetime[:path]]'** (set cookie)

Directives – RewriteRule 2b

- RewriteRule *Pattern Substitution [Flags]*

[flags]:

- '**last|L**' (last rule)
- '**next|N**' (next round)
- '**chain|C**' (chained with next rule)
- '**skip|S=num**' (skip next rule(s))
- '**passthrough|PT**' (pass through to next handler)
- '**nosubreq|NS**' (used only if no internal sub-request)
- '**qsappend|QSA**' (query string append)
- '**nocase|NC**' (no case)

Directives – RewriteCond 2

- **Syntax:** `RewriteCond TestString CondPattern [Flags]`
- *CondPattern* is matched like perl regular expression, + these:
 - Lexical (string) compares (< > =)
 - '-d' (is directory)
 - '-f' (is regular file)
 - '-s' (is regular file with size > 0)
 - '-l' (is symbolic link)

Directives – RewriteCond 2b

- **Syntax: RewriteCond *TestString CondPattern [Flags]***
 - '-F' (is existing file via subrequest)
 - '-U' (is existing URL via subrequest)

All of these can be negated with !
- [Flags]
 - 'nocase|NC' (no case)
 - 'ornext|OR' (or next condition)

Directives – RewriteOptions

- **Syntax:** RewriteOptions *Options*
- **Default:** RewriteOptions MaxRedirects=10
 - MaxRedirects
 - From 2.0.45 / 1.3.28
 - Example: RewriteRule ^(.*)\$ /test/\$1 [R]
 - Inherit
 - Inherit configuration from parent
 - NB! eu.a.o

Directives – Other 1

- **RewriteEngine** *on|off* (default off)
- **RewriteLog** *file*
- **RewriteLogLevel** *Level* (default 0)
- **RewriteBase** *URL-path* (context: dir, .htaccess)

Directives – RewriteMap

Syntax: RewriteMap *MapName* *MapType:MapSource*

Default: not used per default

Context: server config, virtual host

- Defines a mapping function for key lookups
- MapType in one of:
 - int (toupper,tolower,escape,unescape)
 - prg (external program)
 - txt
 - rnd (like txt but makes random choice from | seperrated list)
 - dbm[=type] (one of gdbm, sdbm, ndbm or db – chek valid types with: *bin/apxs -q AP_LIBS*)

Directives – RewriteMap 2

- `#{ MapName : LookupKey }`
- `#{ MapName : LookupKey | DefaultValue }`

Example (crude load balancer/service level selection):

```
RewriteMap service    rnd:/some/where/map.txt
```

```
RewriteRule ^/(.*)/(.*)$    http://#{service:$1}/$2 [P]
```

/some/where/map.txt:

```
gold  big1|big2|big3|big4|big5|big6    # gold service
```

```
silver small1                        # min service
```

Example 1

- No deep links / disallow external referers:

```
RewriteCond %{HTTP_REFERER} !=""
```

```
RewriteCond %{HTTP_REFERER} !^http://example.com/.*$ [NC]
```

```
RewriteRule .* - [F]
```

- Disallow UserAgent

```
RewriteCond %{HTTP_USER_AGENT} EvilBot.*
```

```
RewriteRule /nobot/.* http://%{REMOTE_ADDR}/ [R=301]
```

Example 2

- How to restructure homedirs to avoid getting too many directories

RewriteEngine on

```
RewriteRule ^/~((([a-z])[a-z0-9]+))/(.*)  
/data/$2/$1/$3
```

or

```
RewriteMap LC int:tolower
```

```
RewriteRule ^/~((([a-z])[a-z0-9]+))/(.*) /data/  
/data/${LC:$2}/${LC:$1}/$3 [NC]
```

Example 3

- How to redirect a logged in user to a specific page:

RewriteEngine on

RewriteMap user-to-page txt:/path/to/map.user-to-page

RewriteCond %{LA-U:REMOTE_USER} ^(.+)\$

RewriteRule ^/\$ \${user-to-page:%1|/notfound.html}

/path/to/map.user-to-page

user1 /some/where/user1.html

user2 /some/where/else/user2.html

...

Example 4

- Webmail redirector

RewriteEngine on

RewriteCond %{HTTP_HOST} ^webmail\.(.*)\.dk\$ [NC]

RewriteRule ^/\$ http://mail.dk/login.x?dom=%1 [R]

Example 5

- Moving from index files to autoindex – a regular Redirect / download/index.htm /dir/ would loop

```
/download/.htaccess
```

```
RewriteEngine on
```

```
RewriteBase /download/
```

```
RewriteCond    %{REQUEST_FILENAME}    !-d
```

```
RewriteRule    ^(.*)/index\.html*    $1/    [R=301]
```

Examples 6

- Making sure dav users can't upload "evil" php

RewriteEngine on

RewriteCond %{REQUEST_METHOD} ^PUT\$ [OR]

RewriteCond %{REQUEST_METHOD} ^MOVE\$

RewriteRule ^/dav/(.*)\.php\$ /dav/\$1.nophp

Example 7

Problem: Ugly URLs.

`http://www.server.com/cgi-bin/program.cgi?A=arg1&B=arg2`

`http://www.server.com/program/arg1/arg2`

`RewriteEngineOn`

`RewriteRule ^/program/([^/]+)/([^/]+) \`

`/serverroot/cgi-bin/program.cgi?A=$1&B=$2`

(`http://www.server.com/cgi-bin/program.cgi?A=$1&B=$2`)

Example 8

- Failed client cert validation errors are difficult to decipher in the browser

SSLVerifyClient optional

RewriteEngine on

RewriteCond %{SSL_CLIENT_VERIFY} !="SUCCESS"

RewriteRule .* /path/client-cert-error.html [L]

Note: Currently broken in 2.0.x

Example 9

- setting mime type based on directory
place files in directories named after their mime type

```
RewriteRule ^/files/(.*)/(.*)/ - [T=$1/$2]
```

Ex:

```
/files/application/pdf/somefile
```

```
/files/image/jpeg/someotherfile
```

```
RewriteRule ^/test/(.*) http://www.cnn.com/ [R]
```

```
[rid#815de90/initial] (2) init rewrite engine with requested uri /test/
```

```
[rid#815de90/initial] (3) applying pattern '^/test/(.*)' to uri '/test/'
```

```
[rid#815de90/initial] (2) rewrite '/test/' -> 'http://www.cnn.com/'
```

```
[rid#815de90/initial] (2) explicitly forcing redirect with http://www.cnn.com/
```

```
[rid#815de90/initial] (1) escaping http://www.cnn.com/ for redirect
```

```
[rid#815de90/initial] (1) redirect to http://www.cnn.com/ [REDIRECT/302]
```

RewriteRule ^/\$ /index.txt

(2) init rewrite engine with requested uri /

(3) applying pattern '^/\$' to uri '/'

(2) rewrite '/' -> '/index.txt'

(2) local path result: /index.txt

(2) prefixed with document_root to /opt/run/httpd-2.1/htdocs/index.txt

(1) go-ahead with /opt/run/httpd-2.1/htdocs/index.txt [OK]

(2) init rewrite engine with requested uri /somewhere/

(3) applying pattern '^/\$' to uri '/somewhere/'

(1) pass through /somewhere/

Questions



Latest version:

<http://cvs.apache.org/~mads/ac2004/>

mads@apache.org