

 eBook Gratuit

# APPRENEZ

---

# mod-rewrite

eBook gratuit non affilié créé à partir des  
**contributeurs de Stack Overflow.**

#mod-  
rewrite

# Table des matières

À propos.....	1
<b>Chapitre 1: Démarrer avec mod-rewrite.....</b>	<b>2</b>
Remarques.....	2
Versions.....	2
Exemples.....	2
Installation.....	2
Debian / Ubuntu.....	2
Cas général.....	2
Utiliser mod_rewrite dans le fichier de configuration statique.....	3
Utiliser mod_rewrite à partir des fichiers de configuration dynamiques.....	3
<b>Chapitre 2: Contextes de règles de réécriture.....</b>	<b>4</b>
Remarques.....	4
Exemples.....	4
Réécrire les règles dans un contexte par répertoire.....	4
Réécrire des règles dans un contexte d'hôte virtuel.....	5
<b>Chapitre 3: Directives fournies par mod-rewrite dans Apache 2.4.....</b>	<b>6</b>
Syntaxe.....	6
Exemples.....	6
Liste des directives disponibles dans Apache 2.4.....	6
RewriteBase et RewriteEngine.....	7
<b>RewriteBase.....</b>	<b>7</b>
<b>RewriteEngine.....</b>	<b>7</b>
RéécrireCarte.....	8
<b>Chapitre 4: Réécritures internes de base.....</b>	<b>9</b>
Exemples.....	9
URL de fantaisie au script PHP.....	9
URL avec la chaîne de requête au script PHP.....	9
<b>Crédits.....</b>	<b>10</b>

---

# À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [mod-rewrite](#)

It is an unofficial and free mod-rewrite ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official mod-rewrite.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

---

# Chapitre 1: Démarrer avec mod-rewrite

## Remarques

`mod_rewrite` est un module pour Apache. Ce module est utilisé pour les réécritures internes (requêtes externes devant charger une ressource différente) et les redirections externes (requêtes externes qui doivent faire que le client demande une URL différente).

`mod_rewrite` fournit un contrôle plus fin sur les réécritures internes que `mod_alias`, car ce dernier ne peut mapper que les demandes aux noms de fichiers. `mod_rewrite` fournit des moyens de `contrôle d'accès`, mais cela est généralement mieux fait avec `mod_authz_core` et `mod_authz_host`. `mod_rewrite` fournit une intégration avec `mod_proxy`, mais pour des raisons de performances, cette intégration ne devrait pas être utilisée et à la place, `ProxyPass` et `ProxyPassMatch` de ce dernier module devraient être utilisés.

`mod_rewrite` peut être configuré de manière à ce que les directives soient placées dans les fichiers de configuration dynamiques (`.htaccess`). Pour des raisons de performances, il faut toujours utiliser le fichier de configuration statique (`httpd.conf`) chaque fois que cela est possible.

## Versions

Version	Date de sortie
2.2	2015-07-17
2.4	2016-07-05

## Exemples

### Installation

`mod_rewrite` doit être activé avant d'être utilisé sur un serveur Apache.

### Debian / Ubuntu

Exécuter `a2enmod rewrite`

Puis redémarrez Apache avec le `service apache2 restart`

### Cas général

Ajoutez ou décommentez la ligne suivante dans le fichier de configuration statique (tel que `httpd.conf`):

```
LoadModule rewrite_module modules/mod_rewrite.so
```

Puis redémarrez Apache.

## Utiliser mod\_rewrite dans le fichier de configuration statique

Ajoutez la directive suivante *avant d'*utiliser toute autre directive mod\_rewrite (RewriteRule, RewriteCond, RewriteBase ou RewriteMap).

```
RewriteEngine on
```

Par défaut, le moteur est désactivé. Les directives mod\_rewrite trouvées lorsque le moteur est désactivé sont ignorées. Activez-le dans le contexte de l'hôte virtuel lorsque vous utilisez des hôtes virtuels ou à partir de contextes de répertoires spécifiques, le cas échéant.

## Utiliser mod\_rewrite à partir des fichiers de configuration dynamiques

**Important:** L'utilisation des fichiers de configuration dynamiques (.htaccess) est un gros problème de performances. Lorsque vous avez accès au fichier de configuration statique (httpd.conf ou quelque chose de similaire), vous devez l'utiliser à la place.

Dans le fichier de configuration statique, autorisez les fichiers de configuration dynamiques à remplacer "FileInfo" à l'aide de `AllowOverride`. Cette directive doit être placée dans le contexte du répertoire:

```
AllowOverride FileInfo
```

Le nom de fichier utilisé pour les fichiers de configuration dynamiques est régi par la directive `AccessFileName`. Par défaut, les fichiers de configuration dynamiques sont des fichiers masqués appelés `.htaccess`.

En haut de chaque fichier de configuration dynamique contenant les directives mod\_rewrite, ajoutez la directive suivante:

```
RewriteEngine on
```

Lire Démarrer avec mod-rewrite en ligne: <https://riptutorial.com/fr/mod-rewrite/topic/5687/demarrer-avec-mod-rewrite>

# Chapitre 2: Contextes de règles de réécriture

## Remarques

Cette rubrique décrit les deux contextes dans lesquels `RewriteRule` peut être utilisé. Dans les exemples d'omission de `RewriteEngine on`, il est supposé que cette directive s'est produite avant cet exemple.

## Exemples

### Réécrire les règles dans un contexte par répertoire

Le contexte par répertoire fait partie du fichier de configuration statique entre les balises `<Directory>` et `</Directory>`. L'intégralité du contenu des fichiers de configuration dynamiques se situe dans le contexte par répertoire du dossier dans lequel réside le fichier `.htaccess`.

Le contexte de contexte `RewriteRule` par répertoire correspond à la partie d'une URL après le protocole, le nom d'hôte, le port et le préfixe du répertoire dans lequel ils résident, et avant la chaîne de requête.

#### Dans le fichier de configuration statique

Lorsque la règle suivante est utilisée sur l'URL `http://example.com/foo?id=1`, l'expression régulière du premier argument de `RewriteRule` est associée à `foo`. Le protocole (`http`), le nom d'hôte (`example.com`) et le préfixe de ce répertoire (`/`) sont supprimés. À l'autre extrémité, la chaîne de requête (`?id=1`) est également supprimée.

```
<Directory "/">
  RewriteRule ^foo$ bar [L]
</Directory>
```

Dans l'exemple suivant, à l'aide de l'URL `http://example.com/topic/15-my-topic-name`, le premier argument de `RewriteRule` correspondrait à `topic/15-my-topic-name`:

```
<Directory "/topic/">
  RewriteRule ^topic/([0-9+)-[^\s]*/?$ topics.php?id=$1 [L]
</Directory>
```

#### Dans le fichier de configuration dynamique

Lorsque la règle suivante est placée dans un fichier `.htaccess` qui se trouve dans le dossier `www-root`, puis utilisée dans l'URL `http://example.com/foo?id=1`, la première expression régulière est associée à `foo`.

```
RewriteRule ^foo$ bar [L]
```

Dans un contexte par répertoire, l'URL correspondante **ne** commence **jamais** par un / . Dans un tel contexte, une directive commençant par `RewriteRule ^/` ne correspondra à rien.

## Réécrire des règles dans un contexte d'hôte virtuel

Le contexte d'hôte virtuel fait partie du fichier de configuration statique entre les balises

```
<VirtualHost> et </VirtualHost> .
```

Le contexte de l'hôte virtuel `RewriteRule` correspond à la partie de l'URL après le protocole, le nom d'hôte et le port, et avant la chaîne de requête.

Lorsque la règle suivante est utilisée pour l'URL `http://example.com/foo?id=1` , le regex du premier argument de `RewriteRule` est associé à `/foo` .

```
<VirtualHost 1.2.3.4:80>
  ServerName example.com

  RewriteEngine on
  RewriteRule ^/foo$ /bar [L]
</VirtualHost>
```

Lire Contextes de règles de réécriture en ligne: <https://riptutorial.com/fr/mod-rewrite/topic/6065/contextes-de-regles-de-reecriture>

# Chapitre 3: Directives fournies par mod\_rewrite dans Apache 2.4

## Syntaxe

- RewriteBase URL-path
- RewriteCond TestString CondPattern
- RewriteEngine on | off
- RewriteMap MapName MapType: MapSource
- RewriteOptions Options
- RéécritureRemplacement du motif de règle [drapeaux]

## Exemples

### Liste des directives disponibles dans Apache 2.4

Apache 2.4 fournit les 6 directives suivantes via le module `mod_rewrite` :

1. RewriteBase
2. RéécrireCond
3. Réécrire le moteur
4. RéécrireCarte
5. RewriteOptions
6. RewriteRule

Les directives suivantes, disponibles précédemment dans Apache 2.2, ont été supprimées:

1. RewriteLock
2. RewriteLog
3. RewriteLogLevel

Toutes les directives (à l'exception de `RewriteMap`) définies par `mod_rewrite` peuvent être autorisées à remplacer un `RewriteMap` `.htaccess` par répertoire via `AllowOverride FileInfo`.

Directif	Le contexte	La description
RewriteBase	répertoire, <code>.htaccess</code>	Définit l'URL de base pour la réécriture par répertoire
RéécrireCond	Partout	Définit les conditions dans lesquelles l'action de réécriture aura lieu
Réécrire le moteur	Partout	Définit le statut du moteur de réécriture

Directif	Le contexte	La description
RéécrireCarte	configuration du serveur, hôte virtuel	Définit une fonction de recherche de clé
RewriteOptions	Partout	Définit des <i>options</i> spéciales pour le moteur de réécriture
RewriteRule	Partout	Définit des règles spécifiques pour le moteur de réécriture

Le contexte `Everwhere` signifie que la directive peut être définie dans l'un des quatre emplacements suivants:

1. configuration du serveur
2. configuration de l'hôte virtuel
3. contexte d'annuaire
4. fichier `.htaccess`

Les directives `RewriteLog` et `RewriteLogLevel` ont été fusionnées avec la directive globale `LogLevel` et seront utilisées comme:

```
LogLevel rewrite:<level>
```

où `<level>` est une valeur de `trace8` (la moins significative) à `emerg` (la plus significative). Cette liste est disponible [ici](#).

## RewriteBase et RewriteEngine

Directif	Défaut	Le contexte	La description
RewriteBase	Aucun	Répertoire, <code>.htaccess</code>	Définit l'URL de base pour la réécriture par répertoire
Réécrire le moteur	de	partout	Activer ou désactiver le moteur de réécriture à l'exécution

### RewriteBase

La directive spécifie le préfixe d'URL à utiliser pour remplacer les chemins relatifs.

### RewriteEngine

La directive, si elle est `off`, n'effectuera aucun traitement de réécriture à l'exécution. Ces règles ne sont pas héritées par les hôtes virtuels (à partir de la configuration du serveur) et devront être définies individuellement.

## RéécrireCarte

La directive définit une fonction qui recherchera une clé dans la carte définie et la remplacera de la carte par la recherche.

La fonction de mappage est définie avec la directive `RewriteMap` elle-même comme suit:

```
RewriteMap MAPNAME Type:Source
```

et peuvent être référencés dans l'une des directives `RewriteCond` ou `RewriteRule` pour agir comme guide de substitution comme suit:

```
${ MAPNAME : KEY | DEFAULT }
```

Les éléments suivants sont des valeurs valides pour `Type` dans la définition de la carte:

1. `int` - permet `toupper` , `tolower` , `escape` et `unescape` seulement
2. `txt` - recherche un fichier texte
3. `dbd` - recherche dans une base de données à l'aide de l'instruction SQL `SELECT`
4. `rnd` - recherches aléatoires à partir d'un fichier texte
5. `dbm` - similaire à `txt` , sauf que `httxt2dbm` doit convertir les données en hachages
6. `fastdbd` - recherche dans une base de données à l'aide de l'instruction SQL `SELECT` avec mise en cache

Lire Directives fournies par `mod-rewrite` dans Apache 2.4 en ligne: <https://riptutorial.com/fr/mod-rewrite/topic/5981/directives-fournies-par-mod-rewrite-dans-apache-2-4>

---

# Chapitre 4: Réécritures internes de base

## Exemples

### URL de fantaisie au script PHP

Dans cet exemple, nous réécrivons les URL du formulaire `http://example.com/topic/id-seoname` en un script php qui prend un identifiant en entrée. Cet exemple attend que la règle soit dans un contexte "par répertoire" .

```
RewriteEngine on

RewriteRule ^topic/([0-9+)-[^\s]*/?$ /topics.php?id=$1 [L]
```

Dans cet exemple, `topic/` est le préfixe commun de tous les sujets. Il est suivi par un numéro utilisé par le script. Enfin, le nom du référencement est affiché. Ce nom de référencement est ignoré par `mod_rewrite`, car il n'y est que pour des raisons distinctes. Le deuxième argument de `RewriteRule` contient l'URL à réécrire. L'espace réservé `$1` est remplacé par le contenu du premier groupe de capture de l'expression régulière avant celle-ci. Dans ce cas, il sera remplacé par ce qui correspond à `([0-9+)` .

### URL avec la chaîne de requête au script PHP

Pour correspondre à une chaîne de requête, une condition doit être ajoutée à `RewriteRule` . Cela se fait en plaçant les directives `RewriteCond` avant la règle correspondante. Dans l'exemple suivant, nous réécrivons dynamiquement une ancienne URL en une nouvelle URL.

```
RewriteCond %{QUERY_STRING} ^name=([^&]*)$
RewriteRule ^oldscript\.php$ newscript.php?username=%1 [L]
```

S'il vous plaît noter que pour correspondre au point littéral, nous devons y échapper avec une barre oblique. `%1` est remplacé par le premier groupe de capture de la condition précédente. Dans ce cas, il est remplacé par tout ce qui correspond à `([^&]*)` .

Lire Réécritures internes de base en ligne: <https://riptutorial.com/fr/mod-rewrite/topic/6162/reecritures-internes-de-base>

# Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec mod-rewrite	<a href="#">4444</a> , <a href="#">Community</a> , <a href="#">Sumurai8</a>
2	Contextes de règles de réécriture	<a href="#">Sumurai8</a>
3	Directives fournies par mod-rewrite dans Apache 2.4	<a href="#">hjpotter92</a>
4	Réécritures internes de base	<a href="#">Sumurai8</a>