

# HTACCESS RÉÉCRITURE D'URL

ARNAUD DORMOY

## 02 .HTACCESS : Définition

Les fichiers htaccess sont des fichiers de configurations pour les serveurs web basé sur Apache. Basé à la racine ou dans un répertoire de l'espace d'hébergement, ce fichier définit en général des règles d'accès au contenu du site.

Parmi les utilisations les plus communes, les fichiers htaccess sont utilisés pour les applications suivantes:

- Protéger l'accès d'une page, d'un dossier ou du site dans son ensemble à partir d'un mot de passe (htpasswd).
- Gérer l'URL rewriting (la réécriture des URL).
- Définir des redirections.
- Définir les pages d'erreurs personnalisées.
- Ajouter un mime-type.

# URL REWRITING :

## Définition

Les sites dynamiques génèrent la plupart du temps des pages aux URL longues et complexes, du fait de la présence de variables.

Une fois que la réécriture des URL est mise en place sur un site, les pages seront accessibles grâce à des URL « propres », que ce soit pour les internautes ou pour les robots des moteurs de recherche.

### ADRESSE D'ORIGINE

*<http://www.notre-site.com/articles/article.php?id=12&page=2&rubrique=5>*

### ADRESSE APRÈS URL REWRITING

*<http://www.notre-site.com/articles/article-12-2-5.html>*

# “ URL REWRITING : Principe

Le principe de la réécriture d'URL est donc de mettre en place un « système » sur le serveur pour qu'il sache interpréter ce nouveau format d'URL.

Dans notre exemple, quand un visiteur accède à la page  
***<http://www.notre-site.com/articles/article-12-2-5.html>***,  
le serveur doit renvoyer exactement la même chose que si le visiteur avait demandé à accéder à la page  
***<http://www.notre-site.com/articles/article.php?id=12&page=2&rubrique=5>***.

Pour que ces 2 URL correspondent, on utilise des "**RÈGLES DE RÉÉCRITURE**".

***Chaque règle permet de décrire un format d'URL.***

“

# URL REWRITING : Exemple de règle.

*http://www.notre-site.com/articles/article.php?id=12&page=2&rubrique=5.*



**http://www.notre-site.com/articles/article-12-2-5.html**

Dans l'exemple ci-dessus, la règle de réécriture va indiquer au serveur de prendre *le premier nombre* comme **numéro d'article**, le *deuxième* comme **numéro de page** et le *troisième* comme **numéro de rubrique**.

La technique de réécriture d'URL la plus connue est celle disponible sur les serveurs Apache, le plus souvent utilisés avec le langage PHP.

# URL REWRITING :

## Comment le mettre en place ?

Procédons par étape :

---

1. Vérifier que votre hébergeur permet l'utilisation de l'URL Rewriting.
2. Identifier les pages dynamiques dont l'URL comporte des paramètres, et choisir un nouveau schéma d'URL « propre ».
3. Ecrire les règles de réécriture dans le fichier .htaccess adéquat.
4. Changer tous les liens vers chaque fichier dont l'URL a changé.
5. Mettre à jour votre site et vérifier que tout fonctionne.

# 1 Vérifier que votre hébergeur permet l'utilisation de l'URL Rewriting.

- Si votre site est hébergé sur un serveur dédié, vous avez accès vous-même à la configuration du serveur (comme Apache).
- Si votre site est hébergé sur un serveur mutualisé, il n'est pas garanti que votre hébergeur ait activé le support de la réécriture d'URL, principalement pour des raisons de sécurité.
- Si votre site est fourni par un hébergeur gratuit, il y a peu de chances que la réécriture d'URL soit possible.

# 2 Définir les formats d'URL

Le principe de l'URL rewriting consiste à trouver les schémas des URL à partir de leurs formes communes. Dans notre exemple, les articles sont accessibles selon deux types d'URL (id+rubrique ou id+rubrique+page), suivant que le numéro de page est précisé ou non.

A partir du moment où vous avez identifié ces « schémas d'URL », vous devez choisir un nouveau format d'URL (l'URL « propre »). En général on fait apparaître un nom de fichier avec l'extension .html (ou .htm) mais sachez que vous pouvez mettre ce que vous voulez, cela n'a aucune incidence sur la prise en compte des pages par Google. En effet, quelle que soit l'extension que vous aurez choisie, la page restera une page respectant la norme HTML.



# Rédiger les règles de réécriture

On rédigera nos règles de réécritures dans un fichier .htaccess. (voir page 11)

Chaque répertoire de votre site devra proposer son propre fichier .htaccess.

Les deux premières instructions (**Options +FollowSymLinks** et **RewriteEngine on**) ne doivent être présentes qu'une seule fois par fichier, avant toute règle de réécriture.

La suite du fichier est constituée d'une série de règles de réécriture. Chaque règle est écrite sur une seule ligne (sauf règles complexes) et respecte le format suivant :

# Rédiger les règles de réécriture

- RewriteRule est un mot-clé spécifique au module mod\_rewrite qui indique que la ligne définit une règle de réécriture.
- Ensuite vient l'URL à réécrire, c'est-à-dire l'URL « propre » sans existence physique sur le serveur.
- Enfin vient l'URL réécrite, c'est-à-dire l'URL telle qu'elle sera appelée en interne sur le serveur.

Ces 3 éléments doivent être écrits sur une seule ligne, et séparés par un ou plusieurs espaces à chaque fois

```
#-----  
# Répertoire : /articles/  
#-----  
  
# Le serveur doit suivre les liens symboliques :  
Options +FollowSymlinks  
  
# Activation du module de réécriture d'URL :  
RewriteEngine on  
  
#-----  
# Règles de réécriture d'URL :  
#-----  
  
# Article sans numéro de page :  
RewriteRule ^article-([0-9]+)-([0-9]+).html$ /articles/article.php?id=  
$1&rubrique=$2 [L]  
  
# Article avec numéro de page :  
RewriteRule ^article-([0-9]+)-([0-9]+)-([0-9]+).html /articles/article  
.php?id=$1&page=$2&rubrique=$3 [L]
```

# 4 Modifier tous les liens internes

Maintenant que nous avons défini les schémas d'URL et créé les règles de réécriture, il reste à vérifier que dans tout le site, tous les liens utilisent le bon schéma d'URL.

En effet, les règles de réécriture du fichier .htaccess ne suffisent pas à ce que tout votre site soit au nouveau format, avec des URL propres ! C'est à vous de changer la façon d'écrire les liens, que ce soit dans des pages statiques ou dans des pages dynamiques.

# 5 Mettre à jour et tester

Transférez tous les fichiers modifiés en ligne, y compris le fichier .htaccess, puis rendez-vous dans votre navigateur pour tester si la réécriture fonctionne.

En cas de blocage complet du site (par exemple avec une erreur de type 500), n'oubliez pas qu'il suffit de supprimer le fichier .htaccess (ou d'annuler les dernières modifications) pour que tout revienne dans l'ordre.

Il est conseillé d'utiliser un logiciel de vérification des liens à l'intérieur de votre site. Vous pouvez par exemple choisir Xenu's Link Sleuth, un logiciel à installer sous Windows.