

# Optimisation de site web



Comment et pourquoi optimiser son site web ?

Le 01 Janvier 2017  
Auteurs : Benjamin LARCHER

# DROITS D'AUTEURS

## Licence

Ce document est sous licence « GNU Free documentation 1.3 », hormis les dessins, ceci n'étant pas la production de Nâga.

L'objet de cette Licence est de rendre tout manuel, livre ou autre document écrit « libre » au sens de la liberté d'utilisation, à savoir : assurer à chacun la liberté effective de le copier ou de le redistribuer, avec ou sans modifications, commercialement ou non. En outre, cette Licence garantit à l'auteur et à l'éditeur la reconnaissance de leur travail, sans qu'ils soient pour autant considérés comme responsables des modifications réalisées par des tiers.

Cette démarche est effectuée dans un souci de transparence, de réutilisations et de modifications de données pouvant être sujettes à débat.

# SOMMAIRE

Introduction à l'optimisation.....	1
Qu'est-ce que c'est ?.....	1
Comment optimise t-on ?.....	2
Autres optimisations.....	2
<i>Images</i> .....	2
<i>Polices embarquées</i> .....	2
En pratique.....	4
Modules de compression.....	4
<i>Modules de compression activés sur le serveur</i> .....	4
<i>Modules de compression non activés sur le serveur</i> .....	4
Dans un site « fait maison » hors CMS.....	4
<i>.htaccess</i> .....	4
Dans un CMS (Spip, Joomla, Drupal, Wordpress.....)	6
<i>Interface administration</i> .....	6
<i>Plugins</i> .....	6
<i>.htaccess</i> .....	7

# INTRODUCTION À L'OPTIMISATION

## Qu'est-ce que c'est ?

L'optimisation est l'étape qui suit le développement d'un site et permet d'utiliser moins de ressources.

Les avantages sont multiples :

- Affichage plus rapide, gain de 70 à 90 %,
- Meilleur référencement,
- Diminution de consommation énergétique des serveurs.

*La consommation énergétique des infrastructures nécessaires à Internet (centre de données, liaisons et matériel réseaux, etc.) était estimée à 0,8 % de la consommation mondiale en 2005. En 2012, elle excédait déjà 2 %, autant que l'aérien civil ! C'est l'équivalent de la production de 40 centrales nucléaire... Et tous les quatre ans cette consommation double.*

*Pour les pays industrialisés, les plus connectés, une bonne partie de leur énergie est utilisée pour Internet : au Royaume-Uni c'est 16% de la consommation globale du pays, indique une étude la prestigieuse Royal Society britannique qui a notamment organisé un symposium scientifique mi-mai 2015 sur la question du « capacity crunch » (la crise de capacité) d'Internet.*

*Andrew Ellis, de l'Université Aston de Birmingham, expliquait dans le Daily Mail : "la demande rattrape l'offre. On s'en est très bien sorti pendant des années, mais on arrive au point où on ne peut pas continuer pour toujours".*

*A ce rythme, Gerhard Fettweis de l'Université de Dresde estimait en 2008 que la consommation énergétique liée au web atteindrait, en 2030, l'équivalent de la consommation énergétique mondiale de 2008, tous secteurs confondus ! Autant dire que la consommation du réseau des réseaux va poser de sérieux défis pour les ingénieurs. C'est pourquoi Andrew Ellis, tirait la sonnette d'alarme : "Nous allons arriver à court d'énergie dans une quinzaine d'années."*

Source : [notre-planete.info](http://www.notre-planete.info), <http://www.notre-planete.info/actualites/4328-consommation-energie-web-saturation>

Nous abordons ici uniquement la partie concernant les webmasters. En effet, on peut également diminuer la consommation en utilisant Firefox qui consomme actuellement (en 2016) moins que Chrome.

## Comment optimise t-on ?

On compresse plusieurs éléments à travers des plugins ou dans le fichier .htaccess (fichier caché à la racine du site qu'il faut faire apparaître pour le modifier).

On compresse entre autre :

- Le javascript
- Le html
- Le CSS
- Le XML
- ...

On utilise également la « mise en cache » qui fait garder en mémoire le contenu à un ordinateur qui est déjà passé sur le site et donc ne recharge pas tout le site.

Tout ceci est à faire à la fin d'un développement, car rend les noms des fichiers difficilement lisibles car compressés. Du coup, l'analyse par des outils de développement type Firebug devient compliqué pour retracer les noms de fichiers. Le rafraîchissement des changements hors interface d'administration ne sont plus pris en compte directement (à moins de forcer le vidage du cache, ce que certains sites type CMS ou plugins permettent).

## Autres optimisations

### *Images*

On peut également optimiser le site en dehors de ce qui sera présenté par la suite.

On peut mettre des images en 72 dpi sur le site (100 dpi maximum), pour alléger l'affichage. Un lien vers un fichier plus volumineux peut être fait dans le cas de site de « photos pro » à télécharger par exemple.

Certains formats d'images sont plus légers. Jpg est plus léger que png, mais ne permet pas le détourage.

Le format svg est très léger avec une qualité optimum, même en zoomant, pour tout ce qui affiche des éléments graphiques hors photos.

### *Polices embarquées*

Certains sites comme google font précise le « load time » soit la rapidité d'affichage. Certaines police sont 20 fois plus lourdes que d'autres !

On peut sélectionner dans une police uniquement les éléments utilisés. Certaines polices ont 3 types de gras quand 1 seul est généralement utile.

# EN PRATIQUE

## Modules de compression

### *Modules de compression activés sur le serveur*

Bien souvent les hébergeurs ont par défaut les modules de compression activés, mais ce n'est pas effectif si l'on ne complète pas le .htaccess (renseignements par la suite); Pour s'en assurer on peut tester l'adresse d'un site sur :

<http://www.whatsmyip.org/http-compression-test/>

### *Modules de compression non activés sur le serveur*

Si ce n'est pas le cas, ça ce complique un peu...

Selon le serveur et les versions , apache est équipé de modules de compression :

dans les anciennes versions 1.3 : mod\_gzip ou mod\_deflate

depuis la version 2.0 et supérieures : du module officiel mod\_deflate qui utilise zlib et remplace mod\_gzip

Il faut les activer si cela n'est pas déjà fait sur le serveur.

Actuellement, nous sommes très souvent en version apache 2.0 et supérieures.

Tout est indiqué ici pour l'activation selon les cas de figures :

<http://www.alsacreations.com/article/lire/914-compression-pages-html-css-gzip-deflate.html>

Ça peut paraître un peu abscon, mais il ne faut l'activer qu'une fois pour un gain important sur tous les sites hébergés.

## Dans un site « fait maison » hors CMS

### *.htaccess*

Dans le fichier caché .htaccess (le faire apparaître dans les propriétés/options du FTP) à la racine du site copier/coller ce qui suit dans le fichier (exemple pour version Apache 2.0 et supérieures). Si il n'existe pas, il faut le créer, des exemples se trouvent facilement sur le net.

#Activation du Mod Deflate

<IfModule mod\_deflate.c>

# Compresse les fichiers HTML, CSS, JavaScript, Text, XML et les fonts

AddOutputFilterByType DEFLATE application/javascript

AddOutputFilterByType DEFLATE application/rss+xml

AddOutputFilterByType DEFLATE application/vnd.ms-fontobject

AddOutputFilterByType DEFLATE application/x-font

AddOutputFilterByType DEFLATE application/x-font-opentype

AddOutputFilterByType DEFLATE application/x-font-otf

AddOutputFilterByType DEFLATE application/x-font-truetype

AddOutputFilterByType DEFLATE application/x-font-ttf

AddOutputFilterByType DEFLATE application/x-javascript

AddOutputFilterByType DEFLATE application/xhtml+xml

AddOutputFilterByType DEFLATE application/xml

AddOutputFilterByType DEFLATE font/opentype

AddOutputFilterByType DEFLATE font/otf

AddOutputFilterByType DEFLATE font/ttf

AddOutputFilterByType DEFLATE image/svg+xml

AddOutputFilterByType DEFLATE image/x-icon

AddOutputFilterByType DEFLATE text/css

AddOutputFilterByType DEFLATE text/html

AddOutputFilterByType DEFLATE text/javascript

AddOutputFilterByType DEFLATE text/plain

AddOutputFilterByType DEFLATE text/xml

# Ce code est pour assurer une compatibilité avec les anciens navigateurs

BrowserMatch ^Mozilla/4 gzip-only-text/html

BrowserMatch ^Mozilla/4\.0[678] no-gzip

BrowserMatch \bMSIE !no-gzip !gzip-only-text/html

Header append Vary User-Agent



```
</IfModule>
```

```
#Activation du mod Expires qui garde en mémoire les pages visitées auparavant
```

```
<IfModule mod_expires.c>
```

```
ExpiresActive On
```

```
ExpiresByType image/jpg "access plus 1 year"
```

```
ExpiresByType image/jpeg "access plus 1 year"
```

```
ExpiresByType image/gif "access plus 1 year"
```

```
ExpiresByType image/png "access plus 1 year"
```

```
ExpiresByType text/css "access plus 1 month"
```

```
ExpiresByType application/pdf "access plus 1 month"
```

```
ExpiresByType text/x-javascript "access plus 1 month"
```

```
ExpiresByType application/x-shockwave-flash "access plus 1 month"
```

```
ExpiresByType image/x-icon "access plus 1 year"
```

```
ExpiresDefault "access plus 7 days"
```

```
</IfModule>
```

## Dans un CMS (Spip, Joomla, Drupal, Wordpress...)

### *Interface administration*

#### **Spip**

Dans Spip, on peut déjà compresser pas mal de choses sans rien installer mais en configurant dans Configuration / fonctions avancées / Optimisations et compression.

Les expire header n'est pas pris en compte, mais l'on voit cela plus tard (dans la partie .htaccess)

#### **Joomla**

Voir ce tutos pour activer Gzip :

<https://www.agerix.fr/joomla/activer-la-compression-gzip-de-votre-site-joomla>

Les expire header n'est pas pris en compte, mais l'on voit cela plus tard ( dans la partie Plugins )

### *Plugins*

## **Wordpress**

Quelques plugins parmi les mieux noté en version récente 4,7.

Attention ! Il ne faut en installer qu'un seul !

À noter que le premier ne semble pas gérer le expire header, mais les autres oui. Le troisième n'a que peu de notes, mais ça on peut toujours tester !

Les deux derniers semblent tout gérer, rien de plus à faire !

<https://wordpress.org/plugins/jch-optimize/>

<https://wordpress.org/plugins/wp-speed-of-light/>

<https://wordpress.org/plugins/advanced-speed-increaser/>

## **Joomla**

Pour le expire header, voir ce plugin :

<https://extensions.joomla.org/extension/expired-headers>

## **.htaccess**

### **Spip**

Ajouter ce qui suit :

#Activation du mod Expires qui garde en mémoire les pages visitées auparavant

```
<IfModule mod_expires.c>
```

```
ExpiresActive On
```

```
ExpiresByType image/jpg "access plus 1 year"
```

```
ExpiresByType image/jpeg "access plus 1 year"
```

```
ExpiresByType image/gif "access plus 1 year"
```

```
ExpiresByType image/png "access plus 1 year"
```

```
ExpiresByType text/css "access plus 1 month"
```

```
ExpiresByType application/pdf "access plus 1 month"
```

```
ExpiresByType text/x-javascript "access plus 1 month"
```

```
ExpiresByType application/x-shockwave-flash "access plus 1 month"
```

```
ExpiresByType image/x-icon "access plus 1 year"
```

```
ExpiresDefault "access plus 7 days"
```

</IfModule>

### ***Drupal***

Même méthode que celle vues dans la partie *Dans un site « fait maison » hors CMS*

### ***Joomla / Wordpress***

Les plugins et *configurations indiquées auparavant suffisent.*