

QUELQUES CONSEILS AUX APPRENTIS WEBMESTRES

DÉNOMMER LES FICHIERS

1/ Des minuscules

N'utiliser que des minuscules pour dénommer vos fichiers est certainement le plus simple et le meilleur des conseils que l'on puisse vous donner. Cette recommandation s'étend également au nom des répertoires. Certes, les majuscules sont techniquement utilisables, mais sources d'erreurs de toutes sortes.

Classiquement, une erreur dans la casse des liens unissant les fichiers de votre site ne sera pas détectée pendant la phase de conception du site sur votre PC, mais rendra impossible la navigation en ligne... Pourquoi ? Parce que Internet Explorer est assez conciliant sur les erreurs d'URL et cherche à rétablir l'enchaînement des pages, sans attirer votre attention sur l'approximation de votre demande. Par contre, les serveurs des fournisseurs d'accès sont sensibles à la casse des noms et renvoient une erreur 404 'page non trouvée' ! En résumé: si vous ne voulez pas tout réviser pour que ce bel édifice testé hors connexion daigne également fonctionner en ligne, obligez-vous à n'utiliser que des minuscules !

De plus, les moteurs de recherche risquent fort de proposer aux internautes des liens entièrement convertis en minuscules... rendant impossible la connexion sur votre site ! Dans ce cas, c'est le robot d'indexation qui n'a pas fait correctement son travail: mais allez donc le lui dire ! Un bon conseil: n'utilisez que des minuscules !

2/ Sans espaces

Traditionnellement, les espaces dans le nom des fichiers et des répertoires sont remplacés par les webmestres par un soulignement (_). Pourquoi ? Parce que ça évite les erreurs d'URL pour les visiteurs désirant saisir directement dans la barre d'adresse le lien en direction de votre site.

L'intérêt ? Eviter les confusions. Ceux qui ont pensé s'en affranchir y sont revenus un jour ou l'autre !

3/ Les caractères spéciaux

Compatibles avec certains navigateurs mais pas tous, je vous déconseille de recourir aux caractères accentués et autres caractères spéciaux pour dénommer les fichiers et les répertoires. Vous m'en remercieriez et vos visiteurs également.

4/ Un nom évocateur

Nullement obligatoire, mais tellement pratique ! Non seulement pour la maintenance régulière exercée par le webmestre, mais également pour le confort du visiteur qui apprécie de pouvoir s'orienter par ce moyen.

De plus, les robots d'indexation donnent quelquefois un poids plus intéressant et un meilleur positionnement aux pages au nom et au chemin évocateurs.

5/ L'extension

La recommandation de n'utiliser que des minuscules s'étend également à l'extension qui affecte le nom des fichiers (cf supra).

Quelle différence entre .htm et .html ? Aucune, sinon que le nom du fichier inclus son extension. La page ne sera affichée qu'à la condition que son nom soit correctement et complètement saisi. Un bon conseil: adoptez l'une ou l'autre de ces deux extensions et restez-y fidèle. Certains logiciels donnent le choix au webmestre, d'autres non.

Ces recommandations s'appliquent également aux autres familles de fichiers: images (gif ou GIF, jpg ou JPG) etc.

RÉPERTOIRES ET SOUS-RÉPERTOIRES

1/ Organisez votre site

Comme sur votre disque dur, répertoires et sous-répertoires sont indispensables au webmestre et au visiteur pour s'y retrouver. Certes, un 'petit' site pourrait s'en dispenser. Un 'gros' site également. Mais c'est tellement plus simple quand c'est rangé que ce serait bien dommage de passer à côté. De toute façon, je ne connais pas de webmestre durablement mal organisé....

De plus, l'indexation de vos pages par les moteurs de recherche et leur positionnement en cas de requête d'un visiteur, tiennent compte des noms évocateurs éventuellement inclus dans l'URL de la page, laquelle comporte le chemin (path) composé de la succession

des répertoires et sous-répertoires y conduisant ainsi que le nom donné à la page.

2/ Quelle profondeur d'arborescence ?

A mon avis, la racine doit contenir quelques fichiers seulement et diriger rapidement le visiteur vers l'un des répertoires principaux (de 2 à 10 selon le cas). Si nécessaire, un niveau inférieur sera mis en place pour certains des répertoires les plus importants.

Techniquement, rien n'empêche le webmestre de poursuivre en multipliant les niveaux inférieurs, mais je vous le déconseille fortement, pour garder les idées claires !

3/ Quelle importance pour chaque répertoire ?

A mon humble avis, un répertoire ou un sous-répertoire ne devrait pas dépasser 50-75 pages html pour faciliter la gestion et la mise à jour du site.

RAPIDITÉ D'AFFICHAGE

1/ Pour quels visiteurs ?

Si votre site est destiné à une consultation hors connexion, pas besoin de trop s'en faire. De même si les visiteurs que vous ciblez ont systématiquement recours à une liaison par câble.

Mais la majorité des internautes est encore connectée via un modem téléphonique dont les modèles récents atteignent 56.000 bauds.

C'est pourquoi il est traditionnel de recommander aux webmestres d'élaborer des pages 'pesant' au maximum 30ko (texte et images compris). Très exceptionnellement, cette taille confortable sera dépassée à la condition de prévenir le visiteur d'être patient en attendant le transfert des fichiers concernés jusqu'à son ordinateur.

2/ Les fichiers images

Ce sont habituellement les fichiers les plus volumineux. Les formats usuels (gif et jpg) représentent un bon compromis entre respect du visuel et poids et sont donc privilégiés par les webmestres. Les gif s'adressent aux dessins et les jpg sont réservés aux images. Les gif animés, juxtapositions de gifs chaînés entre eux, sont souvent plus volumineux mais l'animation qu'ils apportent au site fait pardonner ce léger excès d'embonpoint... Quant aux autres formats graphiques, ils sont bannis en raison de leur poids excessif, même si on peut techniquement y avoir recours (exemple: bmp).

Un bon conseil, retravaillez les images à l'aide d'un des nombreux outils listés par ailleurs sur notre site.

UNE MAINTENANCE SIMPLIFIÉE

1/ Des fichiers communs

Fronton, bannières, outil de navigation sont autant de parties communes à l'ensemble des pages de votre site. Intervenir sur l'ensemble des pages du site pour changer seulement un lien ou un visuel serait une perte de temps inutile que vous pouvez aisément éviter. Ayez recours à un fichier extérieur généralement porteur de l'extension .js et faites-y référence à l'intérieur des pages concernées.

Un seule ligne suffit, du genre:

```
<script src="/chemin/fichier_commun.js"></script>
```

La syntaxe du fichier js lui-même est assez simple. Je vous invite à consulter à cet égard les fichiers placés en cache sur votre ordinateur, après avoir un peu navigué sur www.faitonsite.fr.st par exemple.

Vous pourrez l'écrire avec le bloc-notes et lui attribuer l'extension de votre choix, même s'il est habituel de retenir .js . Ce fichier externe pourra contenir ce que bon vous semble, rédigé en langage html. La seule 'difficulté' consistera à remplacer les guillemets par des apostrophes... pas trop dur !

2/ Une zone de service

Destiné à recevoir les fichiers communs à l'ensemble du site (images, bannières, fichiers js), ce répertoire permettra au visiteur de ne charger qu'une seule fois les éléments communs à l'ensemble des pages de votre site. Une économie de temps pour l'internaute et une facilité de maintenance pour le webmestre !

3/ Des liens relatifs

A chaque fois que possible, pour la navigation à l'intérieur de votre site, utilisez des liens relatifs. En effet, dans l'hypothèse d'un déménagement sur un autre serveur, le travail sera d'autant simplifié... Croyez-moi, un fournisseur défaillant ou soudainement mal embouché, ça n'arrive pas qu'aux autres !

Qu'est-ce qu'un lien relatif ? C'est un lien désigné par référence à la page en cours. Remonter dans l'arborescence s'écrira ../ ou ../../ selon qu'il s'agit d'un ou deux niveaux, suivi du chemin pour 'redescendre' dans un autre répertoire.

Exemple:

```
/
/repertoire01/sousrepertoire01a/index.html
/repertoire01/sousrepertoire01b/index.html
/repertoire02/sousrepertoire02a/index.html
/repertoire02/sousrepertoire02b/index.html
```

La page /repertoire01/sousrepertoire01a/index.html contenant le lien

```
../../repertoire02/sousrepertoire02a/index.html
```

permettra de se rendre dans cette nouvelle page. Ces liens fonctionnent également hors connexion.

La référence relative peut également s'exprimer par référence à la racine du site, notée /. Dans notre exemple, le lien aurait pu être écrit:

```
/repertoire02/sousrepertoire02a/index.html
```

Mais les liens de ce type ne fonctionnent qu'en ligne et ne peuvent donc pas aisément être testés par le webmestre hors connexion.

A l'opposé, un lien absolu comporte une référence <http://www.nomdusite.com/pagesouhaitee.html>

LES FICHIERS SPÉCIAUX

1/ Pages d'erreur

Interceptez les sempiternels messages d'erreur en rédigeant un fichier .htaccess avec un éditeur de texte (le bloc-notes de Windows par exemple). Il suffira de le transférer dans chacun des répertoires du site pour que ses instructions soient prises en compte. Le transfert doit se faire en mode texte, au moyen de votre logiciel FTP habituel. Ce dernier est vraisemblablement configuré pour reconnaître automatiquement le mode de transfert adapté.

Il vous faudra également élaborer un page html dénommée par exemple erreur404.html dans laquelle vous afficherez le message d'excuse de votre choix et la redirection qui vous sied. Un formulaire pourra également se montrer utile pour comprendre l'origine du dysfonctionnement, avec l'aide du malheureux visiteur.

Sur votre ordinateur vous ne pourrez pas attribuer le nom .htaccess au fichier que vous aller élaborer. C'est pas grave: il suffira de le renommer directement sur le serveur.

Son contenu ? Quelques lignes toutes simples, demandant au serveur de servir la page erreur404.html de votre cru au lieu de la page standard. Faites de même pour les erreurs 403, 500 et autres. Attention, sur certains serveurs, le nom de la page d'erreur est imposé mais pas toujours dévoilé. Il vous faudra alors provoquer volontairement un message d'erreur en appelant une page erronée, pour découvrir les artifices éventuellement employés par l'hébergeur (exemple: Multimania).

2/ Mot de passe

Les fichiers .htaccess présentent un autre intérêt, celui de pouvoir appeler la procédure de référence en matière de mot de passe sécurisé. Bien plus efficace que tous ces scripts foireux que vous ne manquerez pas de trouver sur les sites les plus divers. Ici, les internautes trop curieux en seront pour leurs frais: pas besoin de fouiller dans le cache en espérant trouver quoi que ce soit pour forcer le filtre d'accès... Tout est géré par le serveur.

Le principe général est commun à l'ensemble des serveurs, mais la syntaxe est un peu différente d'un hébergeur à l'autre: il suffit de consulter les renseignements qu'il diffuse à ce sujet ! Il vous indiquera comment rédiger le fichier texte contenant la liste des mots de passe autorisés. Celle-ci peut être cryptée (Multimania) ou non (Free). Ce fichier secret sera placé par vos soins dans un répertoire au fin fond de votre site, éventuellement protégé par un fichier .htaccess contenant la seule ligne :

```
deny from all
```

3/ Exemples de fichier .htaccess

Exemple pour MULTIMANIA:

```
ErrorDocument 404 /service/erreur/erreur404.html
```

```
ErrorDocument 401 /service/erreur/erreur401.html
AuthUserFile
/data/perso/a/a/b/aabmonlogin/service/secret/.htpasswd
AuthGroupFile /dev/null
AuthName "ICI le nom de votre site ou un message"
AuthType Basic
<Limit GET>
require valid-user
</Limit>
```

qui sera complété par un second fichier .htpasswd contenant les mots de passe autorisés et cryptés. Un brin paranoïaque...

Exemple pour FREE:

```
ErrorDocument 404 /service/erreur/erreur404.html
ErrorDocument 401 /service/erreur/erreur401.html
PerlSetVar AuthFile /service/secret/motsdepasseautorises.txt
AuthName " ICI le nom de votre site ou un message "
AuthType Basic
<Limit GET POST>
require valid-user
</Limit>
```

où les mots de passe ne sont pas cryptés, et ça suffit bien comme ça...

4/ Les fichiers externes

Déjà évoqués précédemment, ces fichiers sont rédigés avec un éditeur de texte (le bloc-notes par exemple) et sont traditionnellement affublés de l'extension .js, mais ce n'est pas une obligation. Détaillés précédemment, ils présentent l'intérêt d'être appelés par une ou plusieurs pages html. Leur mise à jour dispense d'intervenir sur chacune des pages html auxquelles il est lié.

5/ robots.txt

Placé à la racine du site, ce fichier texte est destiné aux robots d'indexation afin de leur interdire de référencer tel répertoire ou tel fichier.

Exemple:

```
# ligne(s) de commentaire à usage interne (facultatif)
# Le fichier robots.txt doit être copié à la racine du site
# Il ne doit pas contenir de ligne vide entre User-agent et
Disallow, tant qu'on s'adresse à un même robot. On passe une
ligne pour s'adresser au robot suivant (si nécessaire).
#
# Cette ligne pour que tous les robots se sentent concernés...
User-agent:*
#
# Désignons la liste des répertoires à écarter. On peut fermer par
un / ou non.
Disallow:/repertoire01/
Disallow:/repertoire03/
#
# Sans oublier quelques pages html de la racine (avec ou sans
l'extension).
Disallow:/page01
Disallow:/page03
Disallow:/page04.html
#
# C'est finalement plus simple que d'utiliser les META tags qu'il
faudrait introduire dans chaque page...
```