

Les technologies du Web (suite)

Outils complémentaires

Mettre en ligne

1. Disposer d'un hébergement
 - Serveur personnel
 - Espace chez un hébergeur commercial
2. (Éventuellement) disposer d'un nom de domaine
3. Copier les fichiers (X)HTML, CSS et JS sur le serveur
 - Transfert de fichiers : FTP, etc ...
 - Le fichier magique « index.html »
4. Tester intensivement (avec différents navigateurs) !

Pour mettre en ligne un site Internet, il faut disposer d'un serveur HTTP. HTTP est le protocole d'échange de documents Web, il faut donc une machine capable de parler ce protocole et accessible depuis Internet pour que celle-ci puisse diffuser le site en question. Le site ne sera visible sur Internet que tant que le serveur sera capable de le diffuser à qui le lui demandera, une machine fréquemment éteinte est donc un très mauvais candidat pour être un serveur HTTP. Il est possible d'avoir son propre serveur HTTP chez soi mais cela exige des compétences en administration de serveur qui n'ont rien à voir avec la conception de document Web. Il est aussi possible de s'adresser à un hébergeur qui dispose de serveurs à lui et échange avec intérêt un peu d'espace sur ses disques et le service de ses machines contre de l'argent. Ainsi, c'est un tiers qui stocke les données et se charge de faire parler HTTP à ses machines, ne reste qu'à lui envoyer les données en question.

En général, les hébergeurs commerciaux proposent des forfaits incluant un nom de domaine. Le nom de domaine est une adresse qui se place dans les URI pour désigner une machine. En l'occurrence, à moins de ne vouloir accéder au site qu'en précisant explicitement l'adresse du serveur, un nom de domaine permet d'enregistrer un nom pour cette machine. Par exemple, un site est hébergé sur la machine d'adresse « 194.109.137.218 » (tous les ordinateurs sur Internet ont une adresse de cette forme). Cette adresse n'est pas facile à retenir et n'est pas intéressante pour le site en question, donc ses propriétaires ont réservé le nom « www.debian.org » et pour visiter le site qui y est hébergé il est possible de contacter l'URL « http://www.debian.org/ » plutôt que « http://194.109.137.218/ » (les deux fonctionnent).

Ensuite, il faut envoyer les données du site sur le serveur. En général les serveurs commerciaux parlent aussi un autre protocole, le FTP. Ce protocole sert à l'échange de fichiers. Grâce à un identifiant et un mot de passe, le propriétaire d'un site peut copier des fichiers sur le serveur. L'utilisation de FTP nécessite un logiciel destiné à cet usage. De nombreux logiciels existent et leur utilisation est souvent assez simple.

Parmi les conventions usuelles des serveurs HTTP, il en est une qui fait que lorsque l'URL demandée indique un répertoire, le document servi par le serveur est le document de ce répertoire appelé « index.html ». Ceci évite de devoir spécifier impérativement le nom du fichier dans l'URL et les pages d'accueil ont donc souvent le nom d'« index.html ».

Une fois téléchargé les fichiers composant le site sur le serveur, celui-ci peut les partager via HTTP, et donc en faire un vrai site Web accessible depuis tout Internet. Cependant pour qu'il soient vraiment accessibles, mieux vaut s'assurer que tout fonctionne bien et que leur accessibilité ne dépend pas trop du moyen avec lequel l'utilisateur tente d'y accéder. C'est très important, surtout si le site repose beaucoup sur des technologies récentes.

Avant le CMS, le préprocesseur

- (X)HTML : format « lourd »
 - **Balises** manuel contraignant
 - Site Web : multiples documents
 - **Toute** l'information dans **tous** les documents
 - Nombreuses **portions communes**, modifications difficiles à propager
- **Préprocesseurs** pour (X)HTML
 - Génération des documents (X)HTML (parfois aussi CSS et JS)
 - **Modèles** de documents
 - Approche **générale** du contenu du site
 - Distinction entre documents et fichiers

Proposer des documents auto-suffisants en contenu conduit à répéter une portion de l'information d'un document à l'autre (en général, au moins la navigation). Comme un site évolue en général en augmentant en nombre de pages, cette information doit être copiée un grand nombre de fois, souvent avec des modifications mineures pour s'adapter à chaque page qui peuvent être réalisées par un traitement automatisé. De plus, la syntaxe à balise du (X)HTML destinée à une compréhension correcte tant pour l'Homme que pour les machines s'avère être peu adaptée à la rédaction directe par un humain pour produire de grandes quantités de données.

Il devient donc assez difficile (même très vite impossible) de maintenir de gros sites Web en éditant manuellement tous les fichiers. Une petite modification peut exiger de modifier tous les documents du site (par exemple l'ajout d'un lien dans un menu de navigation). Un certain nombre de programmes permettent donc de générer le code (X)HTML (et parfois aussi le code CSS et JS) à partir de modèles. Les modèles de document et la

séparation d'un document en plusieurs morceaux indépendants (chacun de ces morceaux pouvant être utilisés dans différents documents) permettent d'organiser l'information du site à l'échelle de l'ensemble et non plus du document. Les portions répétées peuvent donc être éditées de manière unique puis propagées automatiquement dans l'ensemble des documents qui composent le site.

Certains préprocesseurs proposent aussi leur propre syntaxe qui permet de s'affranchir du balisage (X)HTML et permettent donc la composition de documents sous une forme plus proche de la rédaction usuelle de texte. Cela n'empêche pas ces préprocesseurs de bien structurer le contenu, ils tirent juste parti d'informations provenant du contexte ou de règles syntaxiques simplifiées, mais si le document final qu'ils produisent ne contient pas toute l'information alors ce sont de mauvais préprocesseurs :-). Comme leurs règles sont simplifiées, pour ne pas limiter le réalisateur de document, ce type de préprocesseurs autorisent en général l'insertion de code (X)HTML pour pallier leurs manques.

Système de gestion de contenu

- **CMS** : Content Management System
- Exécuté sur le **serveur**
- Modèles de pages
- Traitement automatisé de l'information
 - Manipulation directe de l'information
 - Stockage en **base de donnée**
- **Génération dynamique** des documents à la demande
 - requêtes formulées dans l'URL
- Soumission de contenu directement depuis le site Web

Les systèmes de gestion de contenu (ou **CMS**) sont des programmes (programmes CGI, scripts PHP, etc ...) qui sont exécutés sur le serveur HTTP et qui génèrent les documents à la demande. Ils alimentent leurs documents en contenu depuis une base de donnée et disposent en général de modèles de document complexes pour s'adapter à des besoins multiples. Chaque CMS a son fonctionnement propre et il est difficile de tirer des généralités mais souvent la qualité des documents servis par un CMS n'est conditionnée que par la qualité des modèles qu'il utilise. Ceci signifie que bien que le CMS se charge de générer les documents finaux, le développeur Web qui utilise un CMS doit toujours s'assurer que les documents servis respectent les standards du Web, et doit donc comprendre au moins un minimum le fonctionnement du CMS et de ses modèles. Les CMS actuels proposent de plus en plus un jeu de modèles par défaut respectueux des standards et des bonnes pratiques du Web.

En utilisant une syntaxe particulière pour leurs URLs, les CMS peuvent générer dynamiquement des documents à partir d'une requête codée dans l'URL et peuvent ainsi permettre de retrouver le même document depuis la même URL. La génération dynamique de documents n'est donc pas incompatible avec le rôle des URL d'identifier un document tant qu'une même URL permet de reproduire le même document.

Étant de véritables programmes tournant sur le serveur, il est courant que les CMS intègrent des mécanismes non seulement pour lire une base de donnée mais aussi pour l'alimenter. Dès lors, il est tout à fait possible de modifier/ajouter le contenu d'un site directement depuis une page particulière de ce site (en général avec un accès restreint, mais pas forcément dans le cas de Wiki ou de forums). Il n'est donc plus nécessaire de composer ses documents puis de les envoyer sur le serveur, le site peut vivre de lui-même et être alimenté par ses utilisateurs, ou même si l'ajout de données est réservé à des administrateurs, ceux-ci peuvent utiliser l'interface fournie par le CMS, en générale plus adaptée aux êtres humains que le codage direct des documents.

Exercices pratiques

- Présentation d'outils pratiques
 - Mozilla DOM Inspector
 - CSSed, éditeur de CSS
 - Console d'erreur JavaScript Mozilla
- Créer un document (X)HTML
 - Exemple
- Mise en forme avec une feuille de style
 - Exemple
- Un peu de JavaScript
 - Exemple

Liens

A List Apart

<http://alistapart.com/>

Articles, cours et réflexions sur le bon usage des standards sur le Web

CSS Zen Garden

<http://www.csszengarden.com/tr/francais/>

Démonstration du rôle et de la puissance des feuilles de style.

MeyerWeb

<http://meyerweb.com/eric/css/>

Site personnel d'Eric Meyer. Cours et démonstrations de CSS.

OpenWeb

<http://openweb.eu.org/>

Articles et cours sur les bonnes pratiques du Web, (X)HTML et CSS.

Pompage

<http://www.pompage.net/>

Articles sur le design et la conception de pages Web respectueuses des standards et des règles d'accessibilité.

W3C

<http://www.w3.org/>

Référence du Web. Description des standards, documents normatifs.