

Le langage XML

JON BOSAK • TIM BRAY

Le réseau Internet s'est considérablement développé grâce à la mise au point de systèmes «hypertexte» qui gèrent les images, les textes, les sons. Le langage XML accentuera ce développement.

Quelques indices nous suffisent pour reconstruire une information : en jetant un coup d'œil à cette page, et en repérant des mots imprimés en gros caractères, suivis de colonnes en petits caractères, nous déduisons que nous avons sous les yeux le début d'un article. De même, un simple coup d'œil sur une liste de produits d'épicerie montre que celle-ci est une liste de courses. En consultant quelques rangées de chiffres, on identifie un relevé de compte bancaire.

Les ordinateurs n'ont pas cette capacité : on doit leur indiquer précisément la nature des objets qu'on leur fait manipuler, les liens entre ces objets et ce qu'ils sont censés en faire. Le langage XML (*eXtensible Markup Language*, soit «langage de balisage extensible») est conçu justement pour rendre l'information autodescriptive. Ce changement de la communication entre les ordinateurs, simple en apparence, pourrait étendre l'utilisation de l'Internet à nombre d'activités humaines, au-delà de la simple transmission d'information. Depuis la création de XML, au début de 1998, par le Consortium W3C (*World Wide Web Consortium*), il s'est répandu comme une traînée de poudre dans les milieux scientifiques et industriels.

Cet engouement s'explique par l'espoir que XML comblera certaines des principales lacunes du réseau Internet. L'une d'elles est la lenteur : sur le réseau Internet, l'information est censée circuler à la vitesse de la lumière, mais, en pratique, elle arrive aux utilisateurs à la vitesse de l'escargot. En outre, bien qu'une quantité phénoménale d'information soit disponible, la recherche

de l'information adéquate est souvent fastidieuse.

Ces deux problèmes sont en grande partie dus à la nature du langage principal du réseau, le langage HTML (l'acronyme de *Hypertext Markup Language*, soit «langage de balisage hypertexte»). Bien que celui-ci soit le langage de publication électronique le plus populaire jamais inventé, il reste superficiel : il décrit seulement comment les programmes de navigation sur le réseau doivent organiser le texte, les images et les boutons sur les pages qu'ils affichent. Cette superficialité de HTML le rend facile à apprendre, mais cette simplicité a un prix.

Notamment, avec HTML, on crée difficilement des sites qui ne se limitent pas à envoyer des documents à ceux qui en font la demande, tels des télécopieurs améliorés. Les particuliers et les entreprises veulent des sites qui enregistrent les commandes des clients, qui transmettent des dossiers médicaux ou, même, qui gèrent des usines ou des instruments scientifiques à l'autre bout du monde. Le langage HTML n'est pas conçu pour de telles tâches.

Aujourd'hui, un médecin peut récupérer un dossier médical à l'aide d'un navigateur, mais il ne peut ensuite l'envoyer électroniquement à un collègue spécialiste afin qu'il l'intègre directement à la base de données de son hôpital. Son ordinateur ne sait pas quoi faire de l'information, qui n'est que du `<H1>bla bla </H1> <BOLD>bla bla bla</p> </BOLD>`. Le problème avec le «Ce Que Vous Voyez Est Ce Que Vous Obtenez» (la traduction française de *What You See Is What You Get* ou WYSIWYG), c'est

que ce que vous voyez n'est que ce que vous obtenez.

Les étiquettes entre crochets, dans l'exemple précédent, sont des balises. Le langage HTML ne possède pas de balise qui signale une réaction allergique à des médicaments, par exemple ; sa raideur est une autre de ses limites. L'adjonction d'un nouveau type de balises impose des démarches administratives qui peuvent prendre des années, si bien que peu s'y aventurent. Pourtant, chaque application, et pas seulement l'échange de dossiers médicaux, nécessite ses propres balises.

Cette raideur explique également la lenteur actuelle des bibliothèques en ligne, des catalogues de vente par correspondance ou des divers sites interactifs. Quand on veut modifier les quantités commandées ou le mode d'acheminement des marchandises, le serveur doit renvoyer une nouvelle page, dont du texte et des graphiques déjà reçus. Pendant ce temps, votre puissant ordinateur attend bêtement, parce qu'il ne connaît que les `<H1>` et les `<BOLD>`, et non les prix et les options de transport.

De là résultent également les recherches frustrantes sur le réseau. Parce qu'il n'existe pas de moyen de signaler qu'une donnée est un prix, vous ne pouvez pas utiliser les informations tarifaires dans vos recherches.

Comment faire du neuf avec du vieux ?

En théorie, le problème serait résolu si des balises décrivaient le contenu de l'information, et non ce à quoi elle ressemble. Par exemple, au lieu de classer les différentes composantes d'une commande d'après les rubriques «caractères gras», «paragraphe», «ligne» et «colonne» (ce que fait le langage HTML), on les dénommerait «prix», «taille»,

