

Création de site web

Langage (X)HTML

Damien Nouvel

Langage (X)HTML

Plan



- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

Langage (X)HTML

Plan



- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

Langage (X)HTML

Architecture client / serveur



- Des **statistiques** pour le web ?
 - Savoir quels **outils** utilisent les internautes
 - Connaître la **fréquentation** d'un site
 - Type d'utilisateurs
 - Mode de fréquentation
 - Déterminer le **public visé**
 - Connaître les **tendances**
- À déterminer **pour chaque site** :
 - L'architecture technique
 - La présentation du contenu
 - (...)

Langage (X)HTML

Quelques statistiques du web



- Moteurs de recherche :
 - 85% : Google
 - 7% : Yahoo
 - 3% : Baidu
 - 3% : Bing (MS)
- Systèmes clients (OS) :
 - 90% : Windows
 - 5% : Mac
 - 2% : iOS
 - 1% : Linux
- Clients web (logiciels) :
 - 57% : Internet Explorer
 - 23% : Firefox
 - 10% : Chrome
 - 6% : Safari
 - 2% : Opera
- Serveurs web (logiciels) :
 - 70% : Apache
 - 20% : Microsoft IIS
 - 6% : Nginx
 - 1% : Litespeed

Langage (X)HTML

Quelques statistiques du web



- Top 500 website (Alexa) :
 - Google
 - Facebook
 - Youtube
 - Yahoo!
 - Windows Live
 - Baidu.com
 - Blogger.com
 - Wikipedia
 - QQ.com
 - Twitter
- MSN
- Yahoo! (Japon)
- Taobao.com
- Google (Inde)
- Amazon.com
- Sina.com.cn
- Google.de
- Google.com.hk
- Wordpress.com
- LinkedIn
- Bing

Langage (X)HTML

Plan

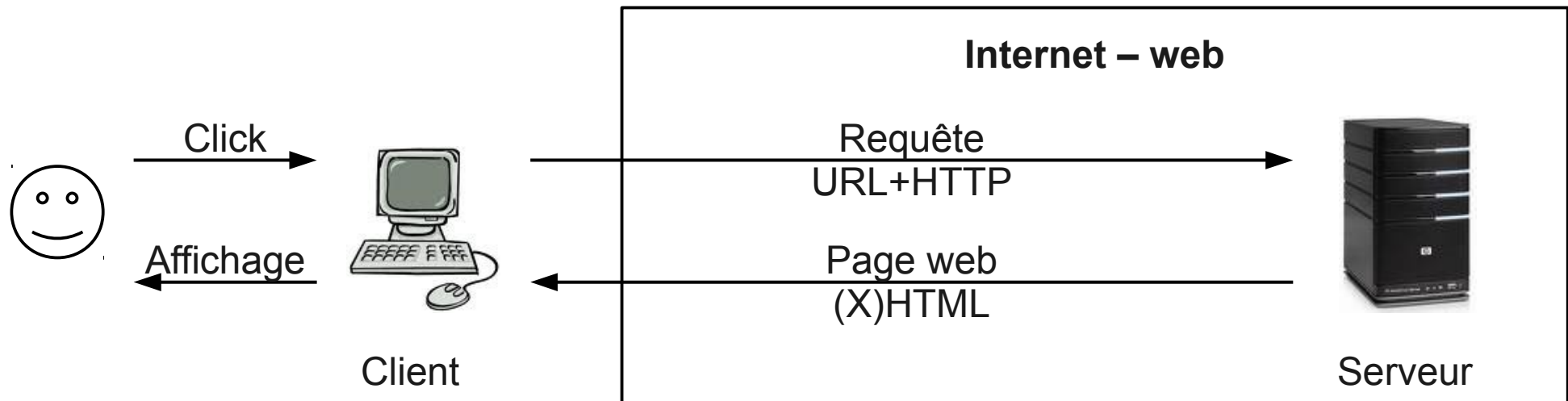


- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

Langage (X)HTML

Architecture client / serveur

- Principe général de la **requête** :
 - L'utilisateur (final) est sur le « **poste client** »
 - Les données à accéder sont sur le « **serveur** »
 - Le client effectue une « **requête** » et le serveur lui renvoie en réponse des données (fichier, flux de données, etc.)



Langage (X)HTML

Plan

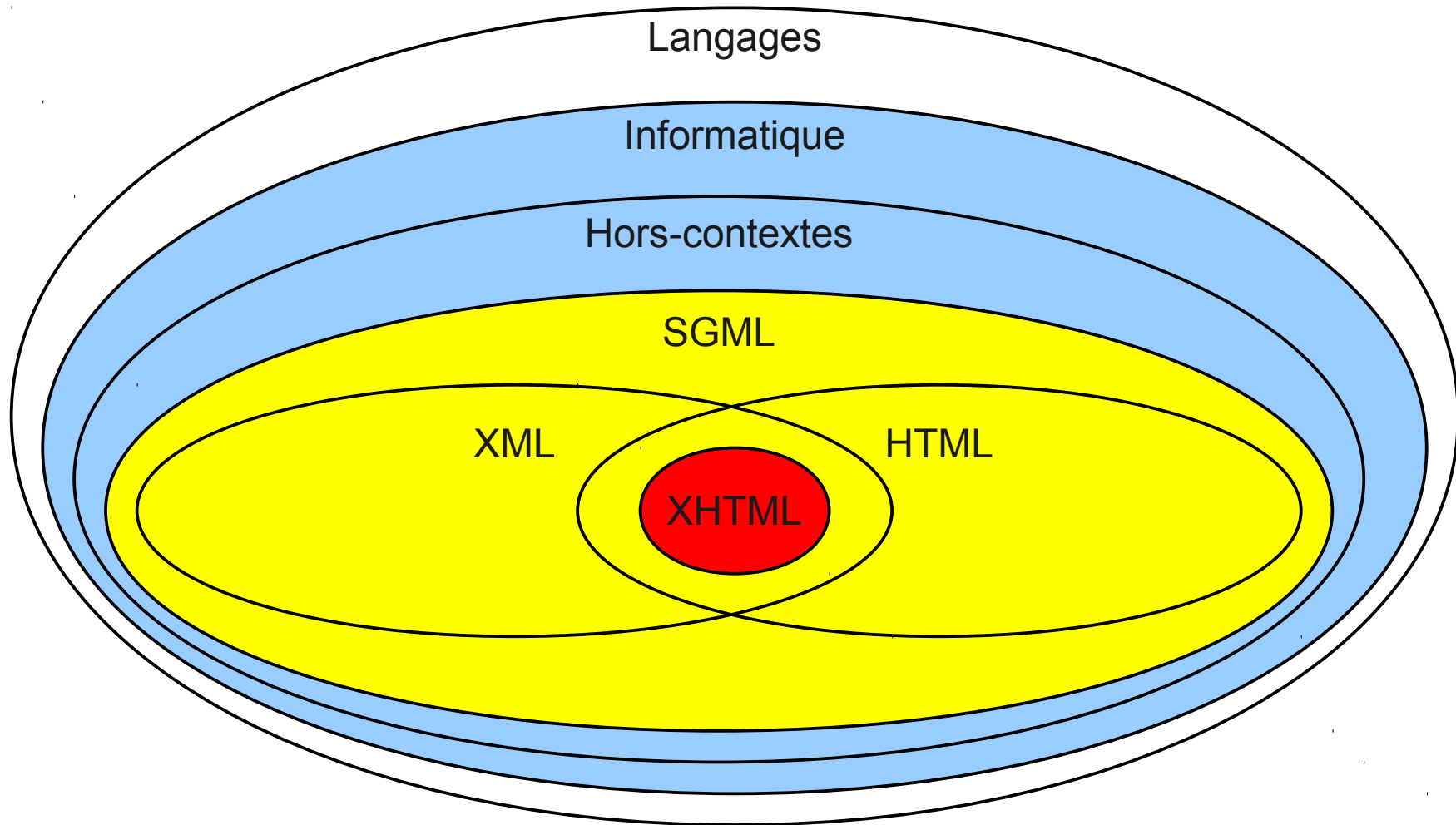


- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

- **eXtensible Markup Language (1996) :**
 - Structuration (encodage) de documents : langage de « balisage »
 - eXtensible : prévoir que le langage **évoluera** au fil des évolutions technologiques et des besoins
 - Markup : principe de la marque, de la **balise**, de l'annotation
 - Language : permet l'analyse par des **grammaires** (informatiques) de type **hors-contexte**, qui peuvent être définies de manière récursive :
 - **XML → Balise XML Balise**
 - Basé sur le **SGML** (Standard Generalized Markup Language, 1969, Goldfarb) pour les échanges électroniques
 - Le **HTML** est un langage SGML

Langage (X)HTML

Langage XML



Langage (X)HTML

Langage XML



- XML, un langage :
 - **Déclaratif** : les données sont utilisées par un programme (navigateur web, base de données, etc.)
 - « Mélangé » au texte
- Syntaxe :
 - Les **chevrons** contiennent des « instructions » XML, les **balises** :
 - « < » (inférieur) : début de balise
 - « > » (supérieur) : fin de balise
 - Par ex. : **balise** ouvrante : « **<mabalise>** »
- Une balise est reconnue par un **langage régulier**
- Commentaires compris entre « <!-- » et « --> »

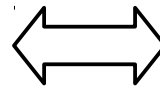
- Grands principes de la balise :
 - Marquent le **début** ou la **fin** d'un « **élément** » :
 - Début : **<element>** (par ex. : **<gras>**, **<titre>**...)
 - Fin : **</element>** (par ex. : **</gras>**, **</titre>**...)
 - **Délimitent** d'autres choses :
 - Par ex. : je vais **<gras> délimiter </gras>** ce mot
 - Pour délimiter un élément, il faut que le début et la fin (correspondante) aient le même nom d'élément
- Par ex. :
 - « Voici un **<gras> texte </gras>** dans lequel on a **<vert> délimité un ensemble de choses </vert>** pour faire des **<grand> effets </grand>** lors de l'affichage... »

Langage (X)HTML

Langage XML

- Utilisation de **balises** :
 - De **haut en bas** : structurer un document (document, titre...)
 - De **bas en haut** : délimiter des portions pour lesquelles on souhaite un traitement particulier (gras, souligne...)

Aujourd'hui, nous voyons comment fonctionne le langage **XML** ! Ce n'est pas forcément évident, mais les principes sont à la portée de jeunes étudiants ?



```
<document> Aujourd'hui, nous  
Voyons comment fonctionne le  
<titre> Langage <gras> XML  
</gras> ! </titre>  
Ce n'est pas forcément évident,  
mais les principes sont à la  
portée de jeunes  
<souligne> étudiants  
</souligne> ?  
</document>
```

- Possibilités et limites des balises :
 - Les éléments ont des noms très variés, tant qu'ils ne sont formés que de caractères **alphanumériques** ($[a-Z] \cup [0-9]$) :
 - Par ex. : `<monelement> </monelement>`
 - Contre ex. : ~~`<un element> </un element>`~~
 - **Peuvent être imbriquées** les unes dans les autres :
 - Par ex. : un `<g> <v> deux </v> trois </g>` quatre
 - **Ne peuvent pas se chevaucher** :
 - Contre ex. : un `<g> <v> deux </g> trois </v>` quatre
- Ces limites déterminent la validité du fichier XML
 - Mais d'autres contraintes peuvent être ajoutées (HTML)

- Les attributs XML :
 - Permettent de **préciser des caractéristiques** sur un élément délimité, en l'ajoutant **dans la balise ouvrante** :
 - Par ex. : texte en `<bleu ton="marine">bleu marine</bleu>` ici
- Disposent de deux parties :
 - **Nom de l'attribut** (par ex. « ton ») : doit être composé de caractères alphanumériques ($[a-Z] \cup [0-9]$)
 - **Valeur de l'attribut** (par ex. « "marine" ») : doit être compris en guillemets doubles "
- Il peut y en avoir **plusieurs par balise** :
 - Par ex. : voilà une police `<police couleur="bleu" style="gras"> spéciale </police>` pour tester le XML

- A l'intérieur d'un élément XML peut se trouver :
 - Du texte
 - Par ex. : un **<gras> texte en gras </gras>** ici
 - D'autres éléments XML (imbrication, sans chevauchements) :
 - Par ex. : voilà **<bleu> des <gras> éléments </gras> imbriqués**
les < souligne> uns dans les </souligne> autres </bleu> !
 - Rien : l'élément pourra alors être **abrégé** dans une forme plus concise en ne mettant qu'une fois la balise qui comporte alors un **/** avant sa fermeture :
 - Par ex. : **<vide></vide> → <vide />**
- Souvent, **par commodité et pour la lisibilité**, on effectue des **sauts de lignes** entre les éléments, ce qui ne **modifie pas l'interprétation du document**

Langage (X)HTML

Langage XML

- Exemple de fichier (structure) XML :

```
<voiture>
  <pneus>Michelin 135 neige</pneus>
  <moteur type="diesel">1.6 TDI</moteur>
  <habitacle>
    <volant forme="rond">
      <klaxon />
    </volant>
    <sieges confort="cuir" nombre="4">top classe</sieges>
  </habitacle>
  <parechoc>abimé mais tient encore</parechoc>
</voiture>
```

- La structure d'un document XML peut-être définie par la **DTD (Document Type Definition)** :
 - Quels **éléments** peuvent apparaître à quel endroit ?
 - Quels **attributs** acceptent un élément ?
- Déterminent la **validité** d'un document, à définir :
 - **En amont** : permet de spécifier le document attendu
 - **En aval** : permet de vérifier automatiquement que le document soit conforme
- DTD définies par le **W3C** (World Wide Web Consortium) pour les documents de sites web (HTML, Sparkle, VoiceXML, XPath, Xquery, etc.)

Langage (X)HTML

Plan

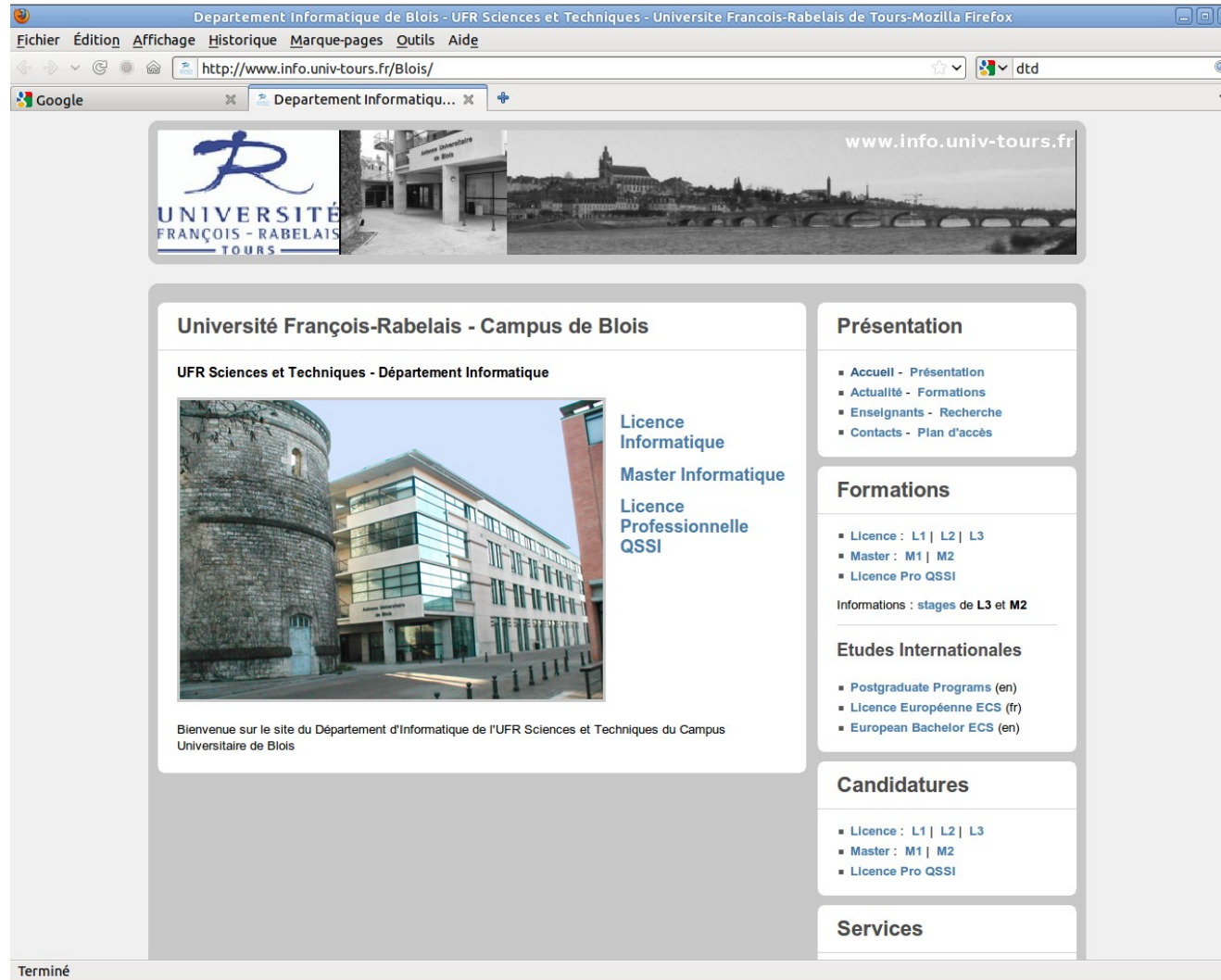


- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

- **HTML : HyperText Markup Language (1992)**
 - A l'origine : mise en forme de documents hypertextes (liens)
 - Aujourd'hui :
 - **Mise en forme**
 - **Structuration**
 - Insertion d'**objets** (Flash, vidéo, Java, etc.)
 - Utilisé pour quasiment **tous les sites web**
 - Evolution (malheureusement) **laborieuse** :
 - Version HTML : 4.0 (1997) **4.0.1 (1999)**
 - Version XHTML : 1.1 (2001)
 - Prochaine version **XHTML5** (?)
- Types de fichiers : « **.html** »

Langage (X)HTML

Mise en forme HTML



- Mise en forme simple :
 - **Gras (Bold)** : ` texte en gras `
 - **Italique** : `<i> texte en italique </i>`
 - **Souligné (Underline)** : `<u> texte souligné </u>`
 - **Saut de ligne** : `
`
- Structuration du texte :
 - **Titre de niveau X** : `<hX> Le titre </hX>`
 - Introduit automatiquement un saut de ligne (bloc)
 - Par ex. : `<h2> Titre de second niveau </h2>`
 - **Paragraphe** : `<p> Mon paragraphe... </p>`
 - Introduit automatiquement un saut de ligne (bloc)

- Listes :
 - **Deux éléments** imbriqués :
 - La **liste** : ` ... `
 - Un **élément** de la liste : ` item `
 - Par ex. :

```
<ul>  
  • <li> élément 1 </li>  
  • <li> élément 2 </li>  
  • ...  
</ul>
```
 - Peut-être **numérotée** en utilisant ` ... `
 - On peut créer des **sous-listes** à l'intérieur de listes

- Tableaux
 - **Trois éléments** imbriqués (dans l'ordre) :
 - Le **tableau** : `<table> ... </table>`
 - Les **lignes** : `<tr> ... </tr>`
 - Les **colonnes** : `<td> ... </td>`
 - Des **entêtes** (à la place) de colonnes : `<th> ... </th>`
 - Par ex. :

```
<table>
  <tr><th> entête 1 </th> <th> entête 2 </th></tr>
  <tr><td> cellule 1 1 </td> <td> cellule 1 2 </td></tr>
  <tr><td> cellule 3 1 </td> <td> cellule 3 2 </td></tr>
  ...
</table>
```

Langage (X)HTML

Mise en forme HTML



- Polices :
 - Balise ** ... **
 - Attributs :
 - **Taille** : ` taille 5 `
 - Entre 1 et 7, par défaut 3
 - En relatif, de -3 à +3
 - **Couleur** : ` couleur `
 - La couleur est codée sur trois octets qui indiquent ses composantes RGB : Red Green Blue (#FF0000 = rouge, #00FF00 = vert...)
 - **Police** : ` ma police `
 - Il peut y avoir **plusieurs attributs** pour un élément
 - Par ex. : voici une ` mise en forme ` de police

- Liens :
 - Balise `<a> ... `
 - Attributs :
 - **Adresse du lien** : ` lien `
 - Lien vers le site à atteindre si l'utilisateur clique
 - Doit être une URL (la plupart du temps http://...)
 - **Titre** : ` mon lien `
 - Ce qui s'affiche au-dessus du lien lorsque la souris le survole
 - **Cible** : ` lien pop-up `
 - Permet d'ouvrir le lien dans une nouvelle fenêtre
 - Il peut y avoir **plusieurs attributs** pour un élément
 - Par ex. :
`cliquez`

- Images :
 - Balise **** (élément XML « vide »)
 - Attributs :
 - **Source de l'image** : ``
 - Lien vers l'image à afficher
 - Ca doit être une URL, mais elle peut-être **relative**
 - **Titre** : ``
 - Ce qui s'affiche au-dessus de l'image lorsque la souris le survole
 - **Texte alternatif** : ``
 - Ce qui s'affiche si l'image n'apparaît pas
 - Il peut y avoir **plusieurs attributs** pour un élément :
 - Par ex. :
``

Langage (X)HTML

Mise en forme HTML



- Quelques autres éléments utiles :
 - Bloc (styles CSS) : `<div> ... </div>`
 - Styles en ligne (styles CSS) : ``
 - Citations : `<blockquote> ... </blockquote>`
 - Centrer un texte : `<center> ... </center>`
 - Barrer du texte : `<strike> ... </strike>`

Langage (X)HTML

Plan



- Quelques statistiques du web
- Architecture client / serveur
- Langage XML
- Mise en forme HTML
- Structure d'un document HTML

Langage (X)HTML

Structure d'un document HTML



- Paradigme contenant / contenu :
 - Mettre en forme \approx structurer
 - **Contenu** : le texte seulement
 - **Contenant** : tout ce qui permet
 - D'organiser le texte
 - De présenter la page
 - De mettre en forme le texte
- La structuration est réalisée en **XML / HTML** :
 - **Pas de chevauchements** entre balises
 - **Imbrications** successives
 - Peut se représenter sous forme d'un **arbre (arborescence)**

Langage (X)HTML

Structure d'un document HTML



- Le **DOM (Document Object Model)** :
 - Dépend de la **DTD XML du fichier HTML (4.0.1)**
 - **Élément racine** : `<html> ... </html>`
 - Contient tout le code HTML
 - **Entêtes** : `<head> ... </head>`
 - **Titre** de la page (affiché sur la fenêtre) : `<title> ... </title>`
 - Éléments pour les moteurs de recherche
 - Mots clés, résumé, etc.
 - Fichiers externes de **mise en forme** (feuilles de style CSS)
 - Liens vers des programmes (Javascript)
 - **Corps** : `<body> ... </body>`
 - Contient tout le contenu structuré en HTML de la page

Langage (X)HTML

Structure d'un document HTML



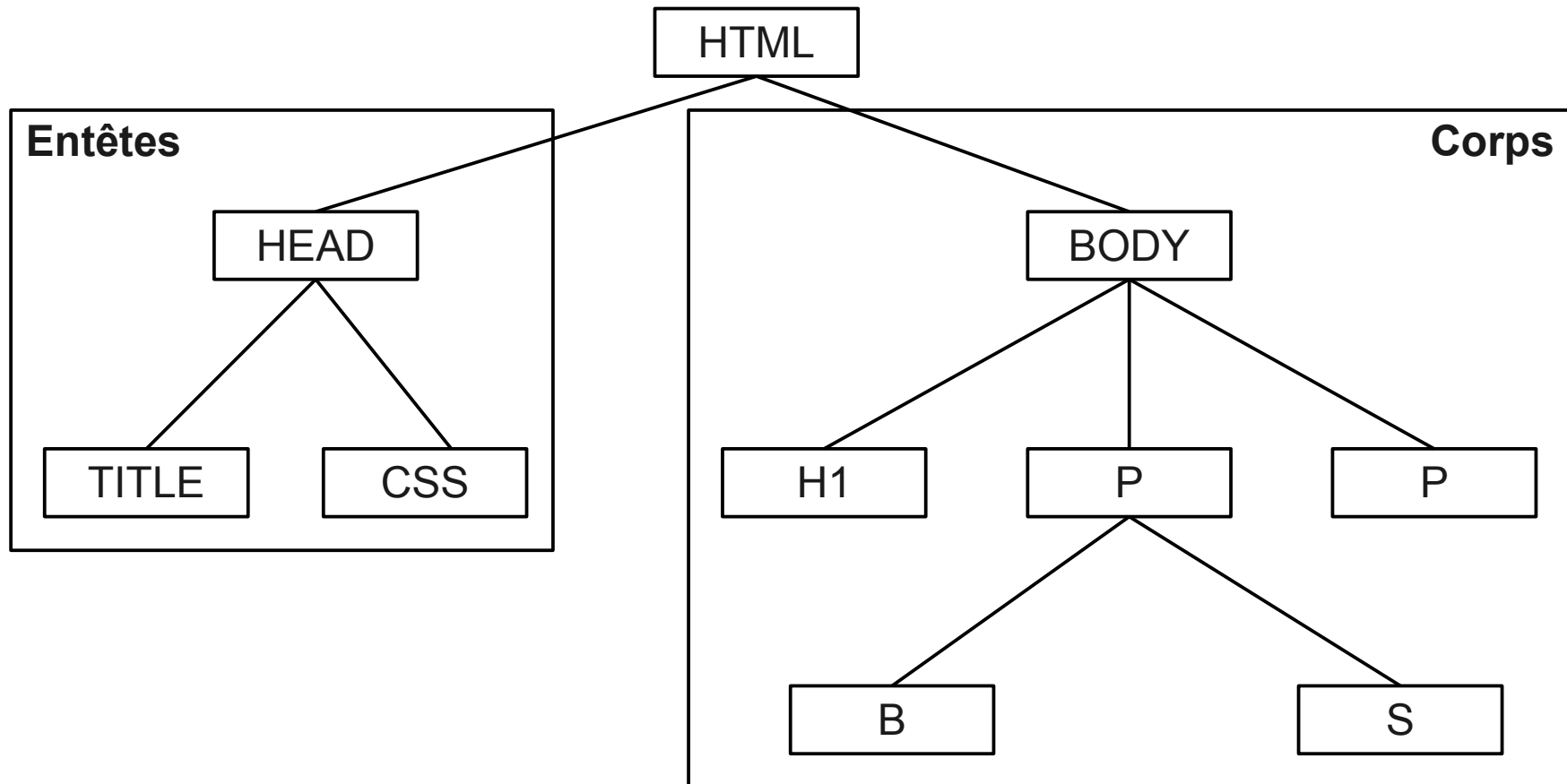
- Par ex., **fichier HTML bien structuré** :

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Titre de page</title>
    <meta http-equiv="Content-Type" content="charset=UTF-8" />
    <link href="css/generic.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
    <h1> Première section </h1>
    <p> Un paragraphe avec un peu de <b> mise en forme </b> plutôt
    traditionnelle pour le <s> coup </s> </p>
    <p> Un autre paragraphe ! </p>
  </body>
</html>
```

Langage (X)HTML

Structure d'un document HTML

- Par ex., **arbre DOM associé** :



Langage (X)HTML

Structure d'un document HTML



- Une structuration du document est essentielle pour :
 - Analyse par le **navigateur**
 - Mise en forme par **feuilles de styles CSS** (sélecteurs)
 - Interactions **Javascript**
- D'où l'intérêt d'une structuration :
 - **Concise** : pas trop de balises
 - **Transverse** : identique dans les pages du site
 - **Personnalisée** : adaptée à l'usage du site