

# Cours Web n°1

## Introduction au World Wide Web et à XHTML

Sandrine-Dominique Gouraud (gouraud@lri.fr)  
Pierre Senellart (pierre@senellart.com)



Semaine du 19 septembre 2005

# Plan du cours

- 1 Internet et le Web
- 2 Le langage XHTML
- 3 XHTML et HTML
- 4 Références

- Qu'est-ce que le Web (ou **World Wide Web**, Toile, WWW, W3) ?
  - ▶ Système *hypertexte* public i.e. système contenant des documents liés entre eux par des hyperliens permettant de passer automatiquement d'un document à l'autre.
- Différence entre le Web et l'Internet ?
  - ▶ Internet : réseau mondial d'ordinateurs permettant aux utilisateurs de communiquer (courrier électronique), de publier des informations (Web), de transférer des données (FTP), de travailler à distance (telnet et ssh), etc...
  - ▶ Web : un aspect de l'Internet

Le client (navigateur : Internet Explorer, Mozilla, Firefox...)

- demande au serveur des informations
- affiche des pages pour l'utilisateur

Le serveur (Apache, Microsoft IIS...)

- reçoit en permanence les requêtes du client
- renvoie les documents correspondants

Le réseau Internet

- basé sur le protocole TCP/IP
- chaque machine dispose d'un numéro IP et d'un nom associé grâce aux serveurs de nom du domaine (DNS)

### Exemple

Nom de machine	<code>www.ifips.u-psud.fr</code>
Adresse IP	<code>129.175.251.5</code>

**Protocole** Ensemble normalisé de règles décrivant la manière de transmettre des informations, par exemple sur un réseau comme Internet entre un client et un serveur.

**HTTP** HyperText Transfer Protocol, le plus utilisé des protocoles de communication sur le World Wide Web. Permet à un client Web d'indiquer quelle page il veut obtenir, et au serveur Web de lui répondre en lui donnant cette page.

- URL : *Uniform Resource Locator*
- Identifie l'**endroit** où se trouve une **ressource** sur le Web.
- Dans le cas du Web, ressource = **document** ou **fragment**
- `http:// lea-linux.org /reseau/secu/ firewall.html #intro`  
protocole machine répertoire fichier fragment
- Principaux protocoles utilisés dans les URL : ftp, http, https, news, nntp, mailto, telnet, etc.

**Un serveur Web** : hôte sur lequel fonctionne un serveur HTTP. Il héberge les ressources qu'il sert.

**Un navigateur Web** : logiciel client HTTP conçu pour accéder aux ressources du Web. Sa fonction de base est de permettre la consultation des documents HTML disponibles sur les serveurs HTTP.

**Une page Web** : document destiné à être consulté avec un navigateur Web. Elle est constituée d'une ressource centrale et d'éventuelles ressources liées automatiquement.

**Un éditeur HTML/Web** : logiciel conçu pour faciliter l'écriture de documents HTML et de pages Web en général.

**Un site Web** : ensemble de pages Web et d'éventuelles autres ressources, liées dans une structure cohérente, publiées par un propriétaire (une entreprise, une administration, une association, un particulier, etc.) et hébergées sur un ou plusieurs serveurs Web.

**Un hébergeur Web** : entreprise de services informatiques hébergeant (mettant en ligne) sur ses serveurs Web les ressources constituant les sites Web de ses clients.

**Hyperlien** : lien logique, unidirectionnel (codé par une balise HTML) reliant une page Web à une autre.

**Surfer, Naviguer** : parcourir le Web en suivant les hyperliens, de page Web en page Web.

- Allumez l'ordinateur et connectez-vous.
- Lancez le navigateur de votre choix.
- Allez sur le site du LRI : `http://www.lri.fr/`
  - ▶ regardez le source de la page principale
- Allez sur votre site préféré (ou `http://www.w3.org/Markup/`)
  - ▶ regardez le source
  - ▶ comparez
- Remarques ?

# Plan du cours

- 1 Internet et le Web
- 2 Le langage XHTML**
- 3 XHTML et HTML
- 4 Références

## Caractéristiques :

- successeur de HTML (langage simple utilisé pour générer des documents portables),
- conforme aux normes XML (eXtended Markup Language),
- développé par le W3C (World Wide Web Consortium) regroupant industriels (Microsoft, IBM, Apple...) et académiques (INRIA, MIT...)

## Principes :

- lecture possible dans des conditions correctes et sans contrainte matérielle ou logicielle
- accessible par tout le monde y compris les daltoniens, malvoyants, aveugles, etc.

## Conséquence :

votre code doit pouvoir être validé par un validateur tel que <http://validator.w3.org/>

- XHTML permet de décrire la **structure** d'un document.
- XHTML est un langage qui alterne textes et *balises* (`<blabla>` ou `</blabla>`)
  - ▶ Les balises définissent les propriétés de chaque partie du document et sont analysées par le navigateur pour réaliser la mise en page du document.
- Les fichiers XHTML
  - ▶ sont des fichiers texte
  - ▶ ont (conventionnellement) pour extension *.html*
  - ▶ sont structurés en deux parties principales : l'en-tête (`<head>...</head>`) et le corps (`<body>...</body>`)
- En XHTML, les blancs (espace, tabulations, retours à la ligne) sont en général équivalents et servent juste à délimiter mots, balises... Leur nombre n'a pas d'importance.

- Leur syntaxe est

`<balise attributs>contenu de la balise </balise>`

ou

`<balise />`

où

**balise** mot clé associé à une propriété particulière

le **contenu** peut représenter du texte ou d'autres balises

**attributs** représentent les différents paramètres associés à la balise. Ils s'expriment sous la forme d'une liste de `type="valeur"` ou `type='valeur'` séparés par des espaces

- Les noms des balises et les types des attributs sont toujours écrits en minuscule.
- Les balises sont ouvertes et refermées dans l'ordre (`<b><i></i></b>` et non `<b><i></b></i>`).
- Il y a des règles strictes qui déterminent quelles balises peuvent être mises à l'intérieur de quelles balises.

## Exemples

- `<br />` pour obtenir un saut de ligne
- `<title>coucou</title>` pour attribuer le titre *coucou* au document
- `<em>cuicui</em>` pour mettre en *emphase* le texte *cuicui* (cela sera rendu, le plus souvent, par une mise en italique).
- `<strong>cuicui</strong>` pour indiquer que le texte **cuicui** est important (cela sera rendu, le plus souvent, par une mise en gras).
- `<!--zut.-->` pour écrire un commentaire, ici *zut.*, dans le document, qui ne sera pas affiché par le navigateur.

## Contre-exemple

```
<strong> <em>bouh</strong> </em>
```

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

  <html xmlns="http://www.w3.org/1999/xhtml">
    En-tête { <head>
              :
              </head>
    Corps    { <body>
              :
              </body>
    </html>
```

La partie DOCTYPE précise en particulier la version d'(X)HTML utilisée.

- Cette partie du document est délimitée par la balise `head` qui ne possède pas d'attributs.
- L'en-tête contient des informations concernant le document telles que l'auteur, le titre, l'encodage, la langue, etc.
- L'information la plus importante (et la seule requise) est le titre de la page qui est affiché dans la barre de titre.

```
<title> Bienvenue sur lavache.com : le site des  
adorateurs de la vache </title>
```

- Les balises `meta` sont facultatives mais très utiles :
  - ▶ aux navigateurs car elles donnent des renseignements sur la manière d'afficher votre document
  - ▶ aux moteurs de recherche (Google) qui s'en servent pour indexer les pages

- L'auteur de la page

```
<meta name="author" content="Jean Dupont" />
```

- La langue de la page (sert pour les synthétiseurs vocaux, les moteurs de recherche, les moteurs de traduction...)

```
<meta http-equiv="Content-Language" content="fr" />
```

### Remarque

Il faut également préciser la langue avec les attributs lang et xml:lang dans la balise html :

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="fr"
xml:lang="fr">
```

- Le jeu de caractères de la page, à mettre **tout au début** de l'en-tête

```
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
```

**Unicode** : *répertoire de caractères*, assignant à chaque caractère, de quelque langue que ce soit, un nombre entier.

## Exemples

A	→	65		ε	→	949
é	→	233		ℵ	→	1488

**Jeu de caractères** : moyen de représenter concrètement, par une suite de 0 ou de 1, un caractère Unicode.

## Exemples (é)

iso-8859-1	11101001		Seulement pour certains caractères
utf-8	11000011	10101001	
utf-16	11101001	00000000	

On utilisera uniquement **utf-8**.

- Les balises `<body> . . . </body>` délimite le corps de votre document.
- Pour faciliter sa lecture, il faut le structurer en rubriques, paragraphes, listes, etc.
- Il existe 6 balises permettant de représenter les différentes rubriques et leur importance :
  - ▶ `<h1>` Rubrique numérotée A `</h1>`
  - ▶ `<h2>` Rubrique numérotée A.l `</h2>`
  - ▶ `<h3>` Rubrique numérotée A.l.a `</h3>`
  - ▶ `<h4>` Rubrique numérotée A.l.a.1 `</h4>`
  - ▶ `<h5>` Rubrique numérotée A.l.a.1.i `</h5>`
  - ▶ `<h6>` Rubrique numérotée A.l.a.1.i.α `</h6>`
- Les balises `<p> . . . </p>` permet de délimiter un paragraphe. La mise en page correspondante est généralement celle des anglo-saxons : texte sans alinéa, aligné à gauche, mais cela peut être changé avec CSS.
- La balise `<hr />` (*horizontal rule*) permet de tracer une ligne horizontale.

- Ouvrez un éditeur simple
- Créez la page donnée en modèle
- Enregistrez la sous le nom *essai.html*
- Validez à l'aide de `http://validator.w3.org/`
- Ouvrez le composer du navigateur Mozilla
- Créez (à la manière d'un document word) la page donnée en modèle
- Enregistrez la sous le nom *essai2.html*
- Comparez les sources des fichiers *essai.html* et *essai2.html*

- XHTML possède plusieurs balises permettant de présenter le texte sous forme de listes.
  - On en distingue trois types :
    - ▶ les listes non numérotées,
    - ① les listes numérotées,
- les listes de définitions (ou lexiques)
- Ces listes sont des blocs et peuvent donc toutes être emboîtées.

- Les listes classiques :
  - ▶ Les listes non-numérotées sont délimitées par les balises `<ul>...</ul>`
  - ▶ Les listes numérotées sont délimitées par les balises `<ol>...</ol>`.
  - ▶ Tous les éléments d'une liste numérotée ou non sont délimités par les balises `<li>...</li>`
- Les lexiques sont délimités par les balises `<dl>...</dl>` et leurs entrées par les balises `<dt>`, `<dd>...</dt>`, `</dd>`

## Exemples

```
<ol> <li>un</li> <li>deux</li> </ol>
```

```
<dl> <dt>lapin</dt> <dd>animal</dd> </dl>
```

En utilisant l'éditeur simple (resp. le composeur), rajoutez dans votre fichier *essai.html* (resp. *essai2.html*) :

- dans la rubrique *Exemples* :
  - ▶ une liste non-numérotée de 4 pays
  - ▶ pour chaque pays, une liste numérotée de 2 villes
  - ▶ une liste numérotée de 2 fruits
  - ▶ pour chaque fruit, une liste non-numérotée de 2 prix possibles au kilo
- une rubrique *Annexe*, contenant un lexique de 5 balises définies depuis le début du cours.
- Validez !

- Les tableaux sont délimités par les balises `<table>...</table>`
- Les balises `<tr>...</tr>` délimitent les lignes
- Les balises `<td>...</td>` délimitent les cellules
- **Attention !!** On déclare dans l'ordre le tableau, les lignes puis les cellules

## Exemple

```
<tr> <td> l1, c1 </td> <td> l1, c2 </td> </tr>
```

## Contre-exemple

```
<td>  
<tr> c1, l1 </tr>  
<tr> c1, l2 </tr>  
</td>
```

- Hiérarchiser un tableau en :
  - ▶ donnant une légende au tableau en utilisant les balises `<caption>...</caption>` juste après la balise ouvrante `<table>`.
  - ▶ donnant un titre aux colonnes en utilisant les balises `<th>...</th>`

En utilisant l'éditeur simple, rajoutez dans la rubrique *Exemples* de votre fichier *essai.html* une sous-rubrique contenant la table de multiplication de 1 à 4 (avec légende). Validez !

- Pour insérer une image dans un document XHTML, on utilise la balise `img`.
  - ▶ L'attribut `src` permet de préciser où se trouve l'image.
  - ▶ L'attribut `alt` permet de remplacer l'image quand elle n'est pas disponible. Il est impératif de l'utiliser pour que toute personne (aveugles, navigateur texte, incidents techniques) ne pouvant voir votre image puisse avoir un texte alternatif.

```

```

```

```

- Les formats autorisés sont :
  - ▶ Le JPEG (.jpg), un format adapté aux photos
  - ▶ Le GIF (.gif, *non-libre*) et le PNG (.png), des formats adaptés aux logos et aux images ayant peu de couleurs

- Ce qui différencie une page Web (page HyperTexte) d'un banal document : ce sont les liens !
- Ils sont introduits par la balise `a` suivi d'attributs.
- En cliquant sur un lien, on peut se déplacer vers :
  - ▶ un autre serveur ou un fichier du même serveur
  - ▶ une autre partie du même document
- Localisation d'un hyperlien (texte ou graphique) dans un navigateur :
  - ▶ le curseur change de forme en passant sur le lien
  - ▶ le texte est en couleurs et souligné (la couleur change si le lien a déjà été visité)
  - ▶ l'image est encadrée

- Le lien de base utilise l'attribut href :  

```
<a href="http://www.cnrs.fr/">  
    
</a>  
<a href="bio/indexbioinfo.html">Bioinformatique</a>
```
- Les ancres servent à atteindre un endroit précis dans le document.
  - ▶ On commence par définir les ancres, soit sur une balise existant déjà grâce à l'attribut id, soit indépendemment :  

```
<h3 id="tutorials">Tutorials</h3>  
<a id="tutorials" />
```
  - ▶ Ensuite, on fait le lien avec cette ancre.  

```
<a href="#tutorials"> tutorials </a>  
<a href="http://www.w3.org/MarkUp/#tutorials">tutorials</a>
```

- Insérez une image à la fin de *essai.html* pointant vers <http://pierre.senellart.com/ifips.jpg>. Ne pas oublier un texte alternatif approprié.
- Enregistrez ce fichier image dans le même répertoire que *essai.html* et insérez à nouveau une image pointant vers ce fichier local.
- Transformez cette image en lien pointant vers <http://www.ifips.u-psud.fr/>.
- Ajoutez une ancre à la rubrique Annexe, et un lien vers cette ancre au tout début du document.

# Plan du cours

- 1 Internet et le Web
- 2 Le langage XHTML
- 3 XHTML et HTML**
- 4 Références

- HyperText Markup Language
- Dernière version : HTML 4.01
- Décrit par un formalisme très compliqué (SGML)
- En pratique : pages Web ne respectent pas ce formalisme  $\implies$   
navigateurs ne respectent pas ce formalisme  $\implies$  *soupe de balises!*

- Les balises sans contenus (`<br />`), s'écrivent `<br>` en HTML.
- Certaines balises peuvent ne pas être refermées (`<ol> <li> un <li> deux </ol>`).
- Les valeurs des attributs peuvent ne pas être entre guillemets (`<img src=toto.png alt=toto>`).
- Les noms des balises et des attributs sont insensibles à la casse (`<HTML lang=fr>`).
- `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">`
- Pas de `xmlns`, pas de `xml:lang`.
- Et beaucoup d'autres petites subtilités...

### Avantages de HTML 4.01

- Meilleur support par les navigateurs (Internet Explorer 6/7 ne comprennent pas XHTML).
- Moins de contraintes...

### Avantages de XHTML 1.0

- Plus de contraintes... (donc plus simple!).
- Syntaxe claire, sans ambiguïté.
- Familiarité avec XML, utile dans d'autres contextes.
- Facilité d'utilisation dans des contextes XML (p.ex. XSLT).

- Théoriquement, `<p />` et `<p></p>` sont synonymes en XML, donc en XHTML. En pratique, on utilisera la notation `<balise />` pour les balises n'ayant jamais de contenu (p.ex. `<br />`, `<hr />`...).
- Théoriquement, `<br />` et `<br/>` sont synonymes. En pratique, on utilisera toujours `<br />`.
- Théoriquement, un document XHTML peut commencer par une ligne `<?xml version="1.0" encoding="utf-8" ?>`. En pratique, on l'omettra.
- cf <http://www.w3.org/TR/xhtml1/#guidelines>
- On pourra utiliser <http://qa-dev.w3.org/~bjoern/appendix-c/validator/> pour vérifier le respect de ces règles.

# Plan du cours

- 1 Internet et le Web
- 2 Le langage XHTML
- 3 XHTML et HTML
- 4 Références**

- Spécification de XHTML 1.0  
<http://www.w3.org/TR/xhtml1/>
- Spécification de XML 1.0  
<http://www.w3.org/TR/REC-xml/>
- Spécification de HTML 4.01  
<http://www.w3.org/TR/REC-html40/>
- *HTML et XHTML : La Référence*, O'Reilly
- <http://www.tuteurs.ens.fr/internet/>