

le **cnam**

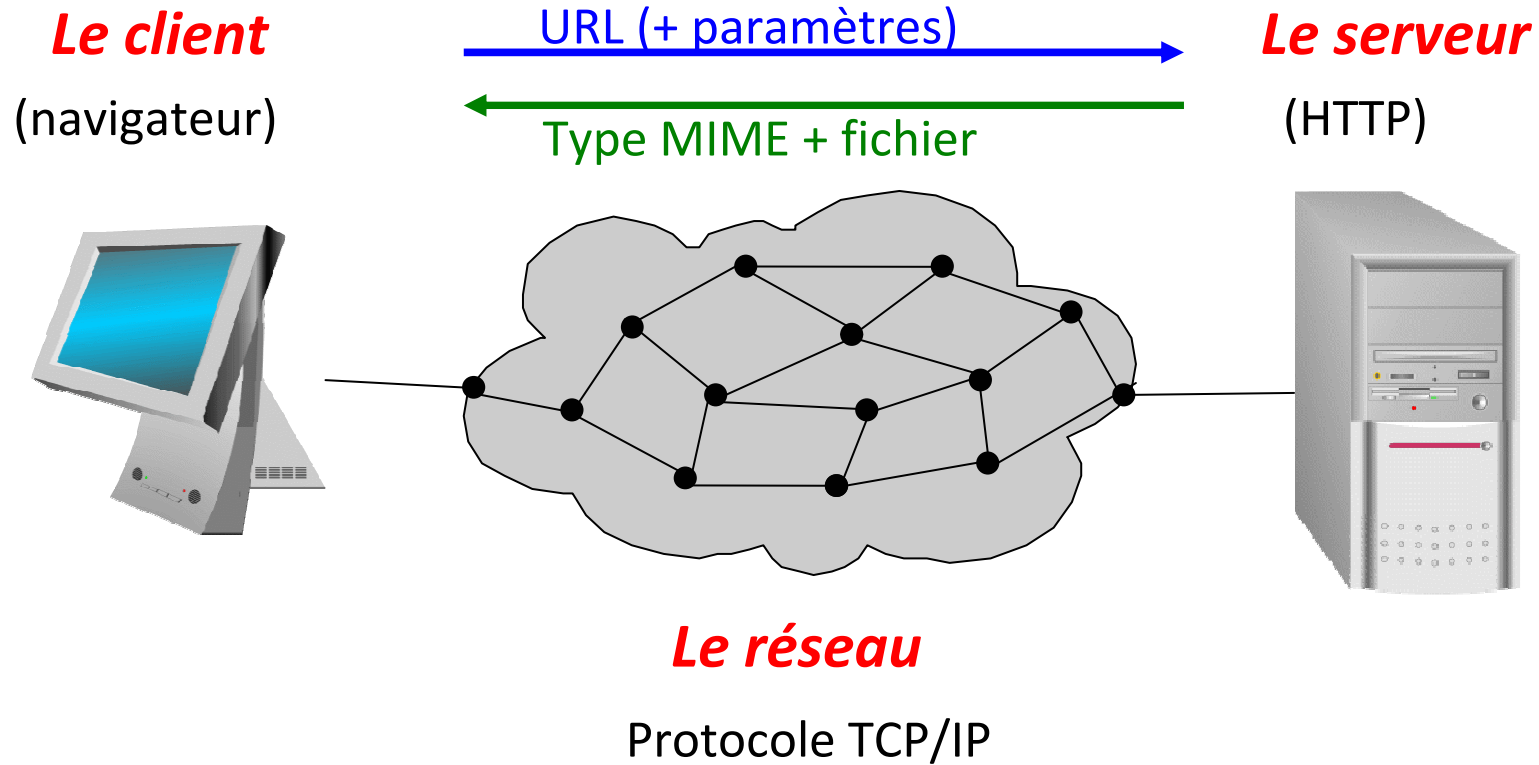
HTML, XHTML



# HTML

- Le langage HTML (HyperText Markup Language) naît en 1989 sous l'impulsion de Tim Berners Lee, "inventeur" du Web
- C'est un langage à balises permettant de structurer et représenter des informations ... pour créer des pages web
- Les documents deviennent interactifs grâce à la présence de liens hypertexte
- En cliquant sur une zone de texte (ou une image) mise en évidence, on peut accéder à un nouveau document situé sur un autre ordinateur en n'importe quel point du globe

# Architecture du Web



# URL et MIME

- **URL** : Uniform Resource Locator
  - Identifie sur le Web un document ou un fragment

[http://www.cnam.fr/~bgates/fr/Enseignement/index.html#web\\_bd](http://www.cnam.fr/~bgates/fr/Enseignement/index.html#web_bd)

protocole machine

répertoire

fichier fragment

↳ (http, https, ftp, news, mailto, telnet...)

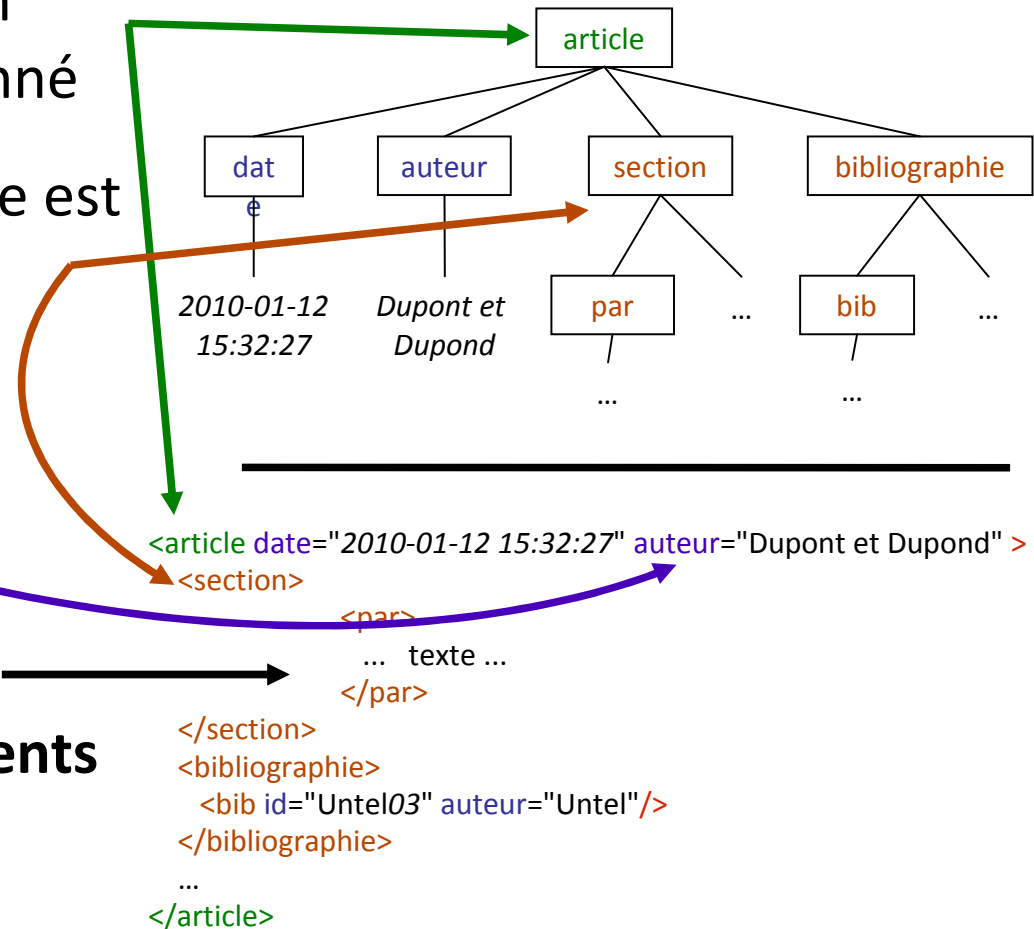
- **MIME** : Multipurpose Internet Mail Extensions
- Type des documents transférés par le protocole HTTP
- La navigateur décide ainsi comment afficher le fichier
  - text/html, text/plain, application/pdf, application/zip, image/jpeg, video/mpeg, audio/x-wav, etc.

# Historique

- 1994 : Tim Berners-Lee quitte le CERN pour créer le World WideWeb Consortium (W3C)
- 1997 : le W3C publie HTML 3.2 puis HTML 4.0
- HTML 4.0 introduit 2 variantes de format :
  - la variante stricte (strict) exclut des éléments et attributs dits « de présentation », destinés à être remplacés par les styles CSS
  - la variante transitoire (transitional) plus souple
- 1998 : apparition de XML
- 2000 : le W3C crée le XHTML 1 = HTML + XML (eXtended HyperText Markup Language)
- 2007 : XHTML 2 est abandonné par le W3C (suite aux résistances de l'industrie et du HTML Living Standard)
- 2012 : livraison du HTML5 du W3C

# L'essentiel du langage XML (1/3)

- Un **document** XML = un **arbre** étiqueté et ordonné
- Chaque **nœud** de l'arbre est un **élément**
- Un élément peut avoir des **attributs**
- Les **feuilles** de l'arbre peuvent être des **éléments textuels**



# *L'essentiel du langage XML (2/3)*

## Terminologie

- Racine = document = arbre
- Nœud = élément (entouré par des balises)
- Balise de début (`<title>`), balise de fin (`</title>`), balise d'élément vide (`<br/>`)
- Attribut : `nom="valeur"` dans la balise de début
- Prologue : version de XML et type du document (ex : XHTML)  

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

## *L'essentiel du langage XML (3/3)*

- La structure des documents XML est imposée par une **DTD** ou un **schéma** XML
- Pour analyser des documents XML, on utilise deux méthodes :
  - **DOM** : construction de l'arbre en mémoire (puissant mais lent)
  - **SAX** : parcours linéaire du fichier (limité mais rapide)
  - Les analyseurs XML implémentent les **API** DOM ou SAX
- Un élément est défini de façon unique par son chemin **XPath** :  
`/article[1]/section[2]/par[5]`



# Structure d'un document XHTML

```
<!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" lang="fr" xml:lang="fr" >
```

```
<head >
  <!-- En-tête du document -- >
</head >
```

```
<body >
  <!--
    Corps du document
  -- >
```

```
</body >
```

```
</html >
```

Attributs de langue (redondant !)

Version de XHTML utilisée  
(pour les validateurs notamment)

# Structure d'un document HTML4

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd" >
```

```
<html lang="fr" >
```

```
<head >
```

```
<!-- En-tête du document -- >
```

```
</head >
```

```
<body >
```

```
<!-- Corps du document -- >
```

```
</body >
```

```
</html >
```

Version de HTML utilisée  
(pour les validateurs notamment)

# Structure d'un document HTML5

`<!DOCTYPE html >`

`<html lang="fr" >`

*Ca suffit!*

`<head >`

`<!-- En-tête du document -- >`

`</head >`

`<body >`

`<!-- Corps du document -- >`

`</body >`

`</html >`

le **cnam**

XHTML



# Entêtes XHTML

```
<html >
```

```
<head >
```

```
<!-- Le jeu de caractères de la page (encodage) -->
```

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

```
<!-- Pour le navigateur, l'historique et les moteurs de recherche -->
```

```
<title>Cours de Programmation Web</title>
```

```
<!-- Pour les moteurs de recherche -->
```

```
<meta name="description" lang="fr" content="cours de prog web" />
```

```
<meta name="keywords" lang="fr" content="web,XHTML" />
```

```
<!-- Pour plus tard ! -->
```

```
<link rel="stylesheet" type="text/css" href="./style.css" />
```

```
<!-- ... -->
```

```
</head >
```

```
...
```

# Les titres de section (h1-h6)

```
<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h1>Introduction</h1>
    <h1>(X)HTML</h1>
    <h2>Les bases</h2>
    <h3>Rappels de XML</h3>
    <h2>Les balises</h2>
  </body >
</html>
```



*Pour l'instant, c'est le navigateur qui décide de la forme...*

# Emphases et compagnie

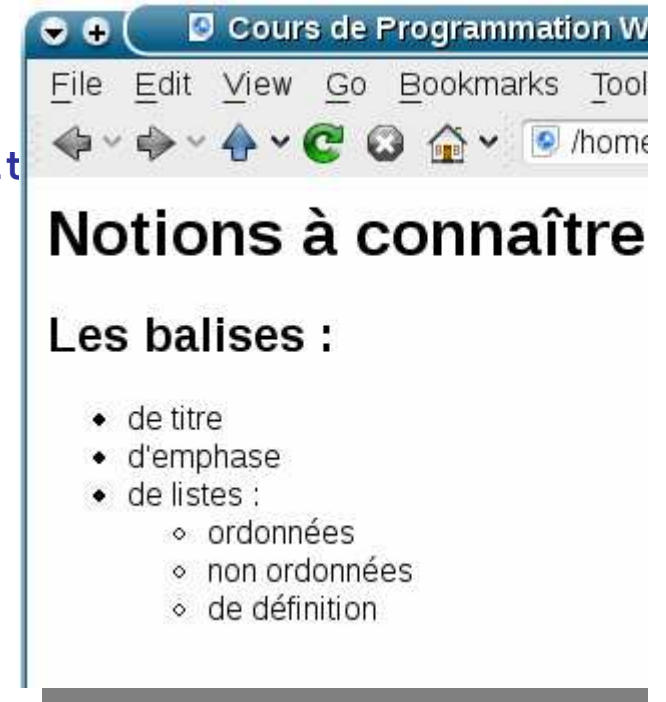
```
<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h1>Introduction</h1>
    <p><em>XHTML</em> est le
      successeur du HTML. <br />
      Il respecte le standard
      <strong>XML</strong>
    </p>
    <hr />
    <strong><em>Voilà !</em></strong>
  </body >
</html>
```

*Pour l'instant, c'est le navigateur qui décide de la forme...*



# Listes non ordonnées

```
<html >
  <head >
    <title>Cours de Programmation Web</tit
  </head >
  <body >
    <h1>Notions à connaître</h1>
    <h2>Les balises<nbsp;:</h2>
    <ul>
      <li>de titre</li>
      <li>d'emphase</li>
      <li>de listes :
        <ul><li>ordonnées</li>
          <li>non ordonnées</li>
          <li>de définition</li></ul>
      </li>
    </ul>
  </body >
</html>
```

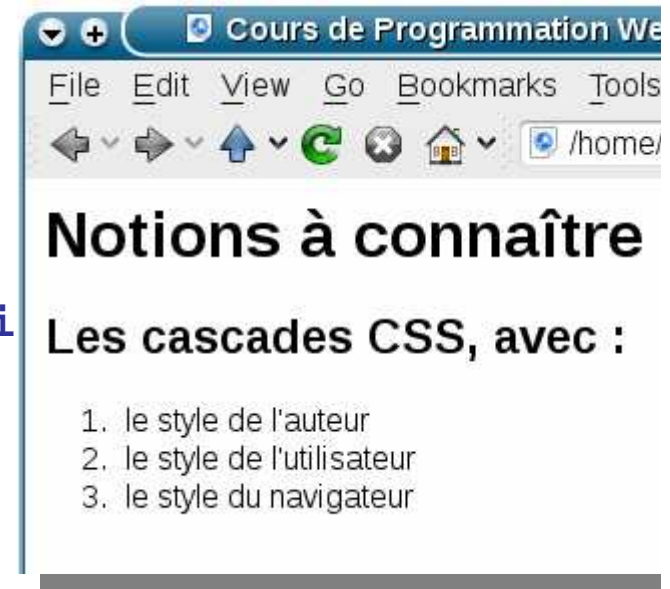


*En XHTML on ferme toutes les balises !*



# Listes ordonnées

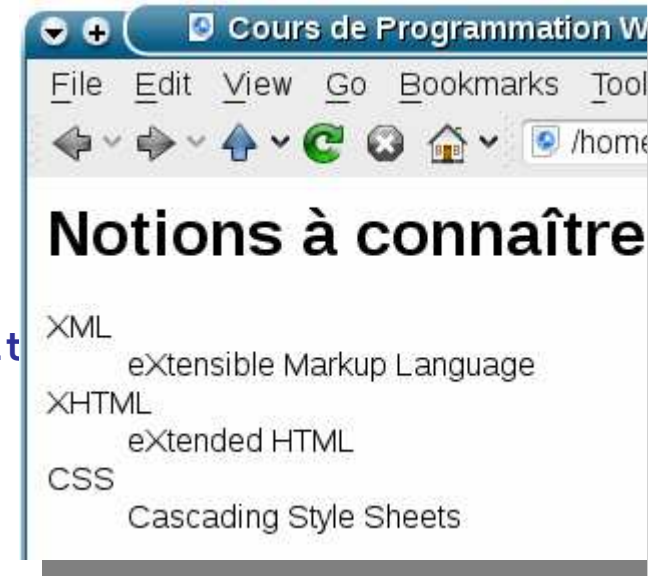
```
<html >
  <head >
    <title>Cours de Programmation Web</ti
  </head >
  <body >
    <h1>Notions à connaître</h1>
    <h2>Les cascades CSS, avec :</h2>
    <ol>
      <li>le style de l'auteur</li>
      <li>le style de l'utilisateur</li>
      <li>le style du navigateur</li>
    </ol>
  </body >
</html>
```



*Attributs de ol : type (valeurs "1", "a", "A", "i" ou "I")  
start (numéro de démarrage)*

# Listes de définitions

```
<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h1>Notions à connaître</h1>
    <dl>
      <dt>XML</dt><dd>eXtensible Markup Language</dd>
      <dt>XHTML</dt><dd>eXtended HTML</dd>
      <dt>CSS</dt><dd>Cascading Style Sheets</dd>
    </dl>
  </body >
</html>
```



# Images

```
<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h1>Images</h1>
     <br />
    
</html>
```

infobulle

Attribut **alt** *indispensable*, pour remplacer l'image lorsque le texte ne s'affiche pas ou pour les lecteurs (humains ou machines) n'ayant pas accès aux images.

Format d'images possibles : **jpeg** (photos)

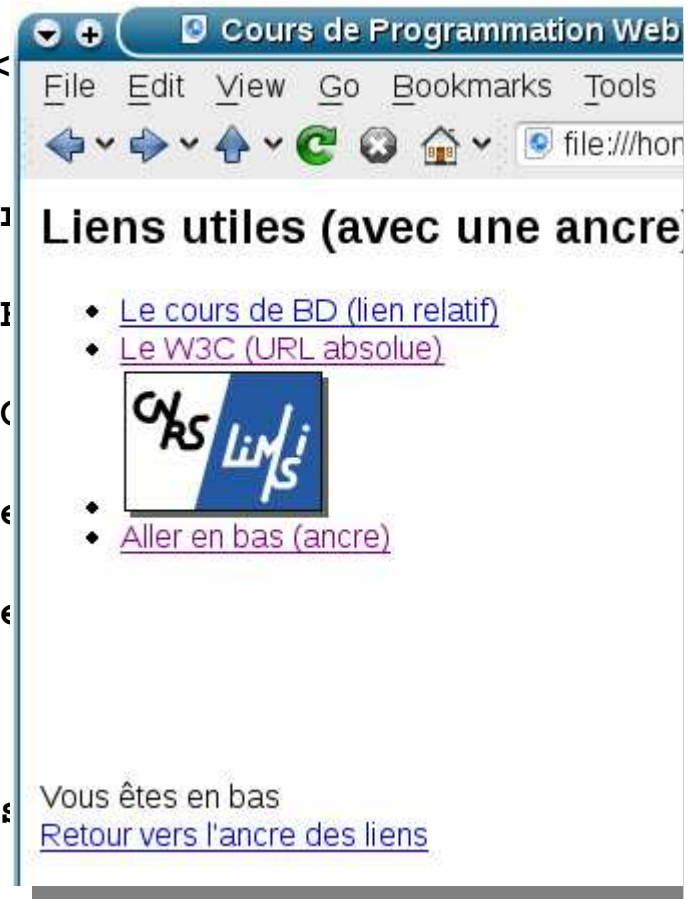
**gif** (animations)

**png** (transparence)



# Liens

```
<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h2 id="liens">Liens utiles (avec une ancre)</h2>
    <ul>
      <li><a href='../BD.html'>Le cours de BD (lien relatif)</a></li>
      <li><a href="http://www.w3.org">Le W3C</a></li>
      <li><a href="http://www.limsi.fr">Le site du LIMSI (image)</a></li>
      <li><a href="#bas">Aller en bas (ancrer)</a></li>
    </ul>
    <br /><br /><br /><br /><br />
    <a id="bas">Vous êtes en bas</a><br />
    <a href="#liens">Retour vers l'ancrer des liens</a>
  </body >
</html >
```



# Tableaux

```

<html >
  <head >
    <title>Cours de Programmation Web</title>
  </head >
  <body >
    <h2>Tableau</h2>
    <table>
      <caption>Comparaison XHTML / HTML</caption>
      <tr><th>XML</th><th>XHTML</th><th>HTML</th></tr>
      <tr><th>forme</th><td>non</td><td>oui</td></tr>
      <tr><th>structure</th><td colspan="2">oui</td></tr>
      <tr><th>&lt;br&gt;</th><td rowspan="2">non</td><td>oui</td></tr>
      <tr><th>&lt;hr&gt;</th></tr>
    </table>
  </body >
</html >

```

légende

ligne

cellule d'entête

cellule de donnée

cellules multi-colonnes  
et multi-lignes

Cours de Programmation Web

Tableau

Comparaison XHTML / HTML

	XHTML	HTML
XML	oui	non
structure	oui	
forme	non	oui
 	non	oui
<hr>		oui

En trichant !  
(feuille de style)



le **cnam**

# Les formulaires XHTML



# Les formulaires

```
<html >
  <body >
    <form method="    " enctype="    " action="script.php">
      </form>
    </body >
  </html>
```

*(on verra plus tard...)*

*URL du script auquel sera soumis le formulaire*

*valeur par défaut*

*application/x-www-form-urlencoded  
multipart/form-data  
text/plain*

# Les ensembles de champs

```
<html >
  <body >
    <form method="    " enctype="    " action="script.php">

      <p>                                } Si les champs de formulaires
      </p>                             } sont au milieu d'un texte

      <fieldset>                       }
        <legend></legend>             } Pour regrouper des champs de formulaire
      </fieldset>                     } ayant une sémantique proche

      <div>                             }
      </div>                          } Sinon

    </form>
  </body >
</html>
```

*En XHTML, pas de champs de formulaire "en vrac"*



# Les types de champs

- `input type = " text hidden password checkbox radio file submit reset image button "`
- `label for`
- `textarea name cols rows`

# Étiquette, saisie de texte

```
<form method="    " action="script.php">
  <fieldset>
    <legend>Identité</legend>
    <label for="nom">Nom</label>
    <input type="text" name="nom" id="nom"
      value="Nom par défaut" maxlength="25" /><br />
    <label for="prenom">Prénom</label>
    <input type="text" name="prenom" id="prenom"
      value="Prénom par défaut" maxlength="25" /><br />
    <label for="passwd">Mot de passe</label>
    <input type="password" name="passwd" id="passwd"
      value="12345678" maxlength="10" /><br />
  </fieldset>
</form>
```

Optionnel. Un clic sur le label sélectionne le champ de saisie

Sans feuille de style...

# Saisie multiligne

```
<form method="    " action="script.php">
  <div>
    <label>Description</label>
    <textarea name="desc" cols="40" rows="5">
    </textarea><br />

    <label>Commentaire</label>
    <textarea name="comm" cols="20" rows="10">
      Pas de commentaire...
    </textarea>
    <br />
  </div>
</form>
```

*Texte par défaut*



The screenshot shows a web browser window with a form. The first section is labeled 'Description' and contains an empty text area. The second section is labeled 'Commentaire' and contains a text area with the text 'Pas de commentaire...'. A vertical scrollbar is visible on the right side of the 'Commentaire' text area.

# Champ caché

```
<form method="    " action="script.php">
  <div>
    <input type="hidden" name="monnaie" value="EURO" />
  </div>
</form>
```

- Permet de masquer des champs pour le client tout en envoyant leur contenu avec le formulaire
- Précise des informations name/value
- **Attention**, à utiliser pour "masquer", pas pour "cacher" !  
Le client peut éditer la page à la main pour voir et changer la valeur de ces champs !

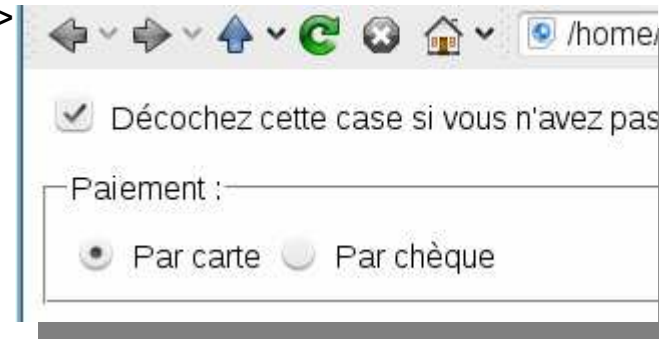
# Choix multiples (1/2)

```

<form method=" " action="script.php">
  <div>
    <input type="checkbox" name="tele" id="tele"
      value="teleOK" checked="checked" tabindex="0" />
    <label for="tele">Décochez cette case si vous n'avez pas de télé</label>
  </div>
  <fieldset><legend>Paielement :</legend>
    <input type="radio" name="paiement" id="carte"
      value="carte" checked="checked" tabindex="1" />
    <label for="carte">Par carte</label>
    <input type="radio" name="paiement" id="cheque"
      value="cheque" tabindex="2" />
    <label for="cheque">Par chèque</label>
  </fieldset>
</form>

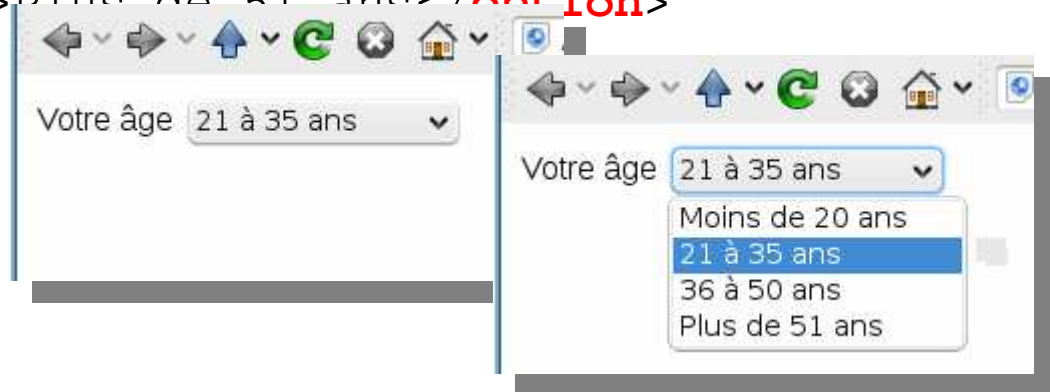
```

*Optionnel mais très conseillé  
pour l'accessibilité*



## Choix multiples (2/2)

```
<form method="    " action="script.php">
  <div>
    <select name="age">
      <label>Votre âge :</label>
      <option value="20">Moins de 20 ans</option>
      <option value="35" selected="selected">21 à 35 ans
        </option>
      <option value="50">36 à 50 ans</option>
      <option value="51">plus de 51 ans</option>
    </select>
  </div>
</form>
```



# Fichier joint

```
<form method=" " enctype="multipart/form-data" action="fic.php">
  <div>
    <label for="fichier">Fichier joint :</label>
    <input type="file" name="fichier" id="fichier"
      accept="text/*" />
  </div>
</form>
```



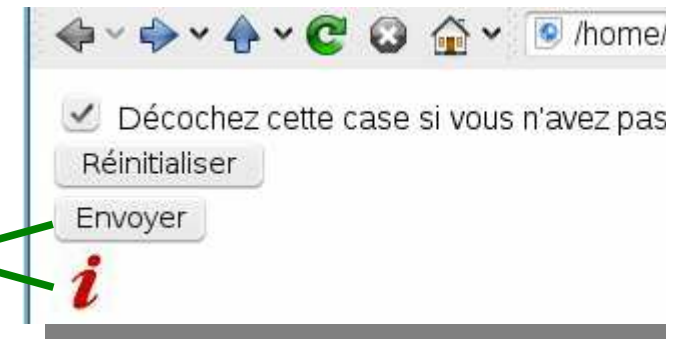
*type MIME (optionnel) :*

- text/plain
- text/html
- image/gif
- video/\*
- ...

# Bouton

```
<form method="" action="script.php">
  <div>
    <input type="checkbox" name="tele"
          value="teleOK" checked="checked" />
    <label>Décochez cette case si vous n'avez pas de télé</label><br />
    <input type="reset" value="Réinitialiser" /> <br />
    <input type="submit" value="Envoyer" /> <br />
    <input type="image" src="./infos.jpg" alt="Envoyer" />
  </div>
</form>
```

Même comportement



type **button** : pas d'action, seulement du javascript. À éviter ! (accessibilité)



le **cnam**

HTML5



# *Qu'est-ce que HTML 5 ?*



- Une révision majeure d'HTML
  - En remplacement de XHTML 2.0
  - Pas encore adoptée (prévu fin 2014)
  - Mais déjà partiellement traitée par les navigateurs récents (mais pas toujours les mêmes éléments...)
- S'appuie sur les standards existants
- Supprime les balises de forme (à gérer par le CSS) :  
`<big>`, `<center>`, `<tt>`, `<u>`, `<basefont>`, `<font>`
- Simplifie la syntaxe
- Ajoute des balises sémantiques de structure
- Définit de nouvelles API

# *Simplification de la syntaxe*

- Déclaration

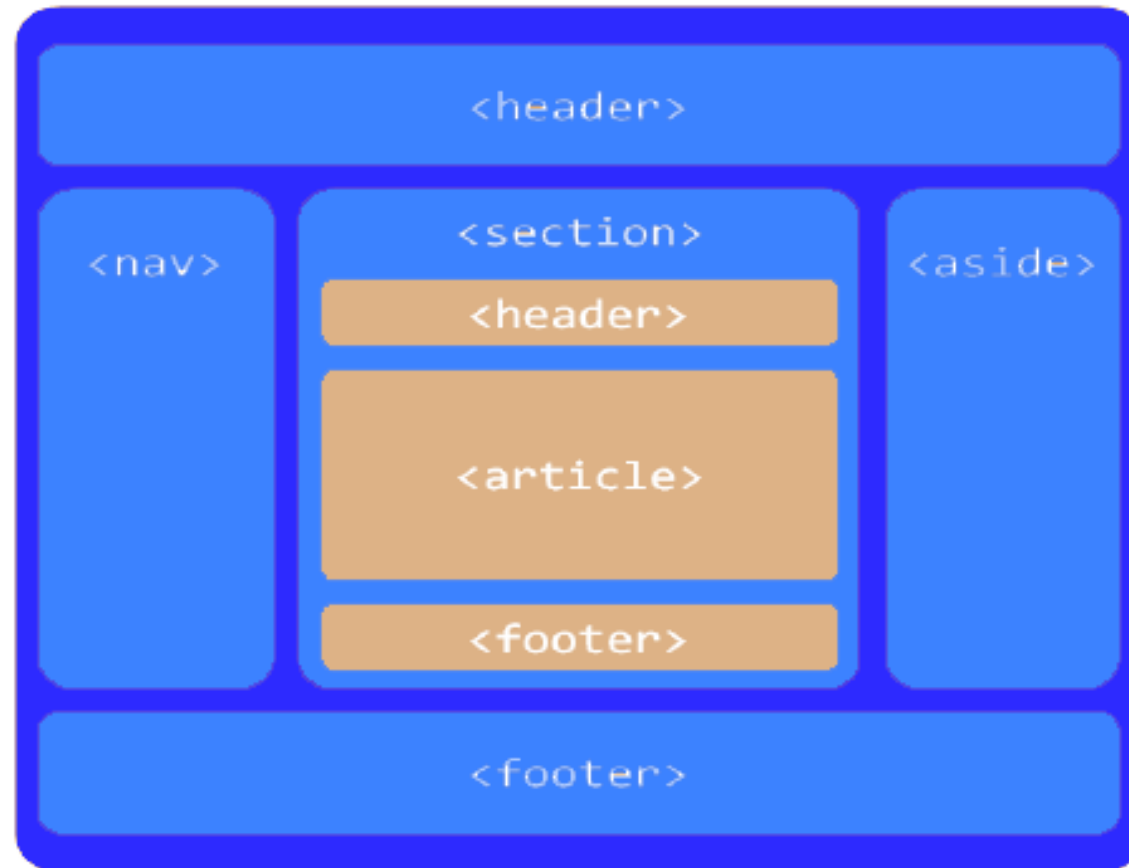
```
<!DOCTYPE html>
```

- Encodage

```
<meta charset="UTF-8">
```

- Insensible à la casse  
(mais la convention est d'utiliser les minuscules)
- Fermer les balises est optionnel  
(mais conseillé)

# Balises sémantiques de structure



(exemple de positionnement,

source : <http://www.codeproject.com/Articles/146409/Semantic-HTML5-Page-Layout>)

# *De nouvelles API*

- Dessin 2D (balise canvas)
- Vidéo/audio (balises video et audio)
- Applications hors-ligne
- Drag and drop
- Accès à l'historique

# Références

- Spécification de XHTML 1.1  
<http://www.w3.org/TR/xhtml11/>
- Spécification de XML 1.0  
<http://www.w3.org/TR/REC-xml/>
- Spécification de HTML 4.01  
<http://www.w3.org/TR/REC-html40/>
- Spécification de HTML5  
<http://www.w3.org/TR/html5/>
- HTML5 Tutoriels et démos  
<http://www.html5rocks.com>, <http://html5demos.com>