

Première Partie

HTML 4.0

Historique

- Conçu par Tim Berners-Lee au CERN en 1990
- Rendu populaire en 1993 par Mosaic, développé au NCSA
- Suivi de nombreuses extensions diverses
- Novembre 1995: HTML 2.0 défini par IETF
- Janvier 1997: HTML 3.2 défini par W3C
- Décembre 1997: HTML 4.0 défini par W3C
- Janvier 2000: XHTML 1.0 est une recommandation du W3C
- HTML+ (1993) et HTML 3.0 (1995) sont non standards

HTML 4.01

- Internationalisation (d'avantage de caractères, spécification de la langue, de la direction)
- Accessibilité (meilleur support d'agents non graphiques)
- Nouveau modèle de tableaux (meilleur contrôle de la structure et du formatage)
- Documents composés (mécanisme général de composition)
- Feuilles de style (séparation structure/présentation)
- Scripting (documents dynamiques/interactifs)
- Impression (collections de documents)

Spécification W3C depuis le 24 décembre 1999

Spécification à: <http://www.w3.org/TR/html4>

SGML - Standard Generalized Markup Language

Marquage de documents permettant d'enrichir le contenu:

- Structure
- Présentation
- Sémantique

HTML est un exemple de marquage

Un exemple de document HTML

```
<!DOCTYPE HTML PUBLIC
"-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
```

```

<HEAD>
  <TITLE>My first HTML document</TITLE>
</HEAD>
<BODY>
  <IMG src="logo.gif">
  <P>Hello world!
</BODY>
</HTML>

```

Définition d'un langage de marquage

- Déclaration SGML: caractères, "delimiters", ...
- Document Type Definition: syntaxe, constructions, "character entities"
- Spécification: sémantique du marquage, restrictions supplémentaires
- Identificateur public de la DTD

Eléments et Attributs

Eléments

- Structure (chapitre, section, paragraphe, ...)
- Définis par: nom, étiquette "tag" de début, contenu, étiquette de fin
- Les tags peuvent être optionnels (P, HEAD)
- Si contenu vide, pas de tag de fin (IMG)
- Le nom est "case-insensitive"

Attributs

- Propriétés associées aux éléments
- Définis par: nom, type de la valeur, valeur par défaut
- Le nom est "case-insensitive"
- La valeur peut-être "case-insensitive" ou "case-sensitive"

Notes:

- "Tags" et Eléments ne sont pas synonymes!!!
- **La structure d'un document est un arbre**
-> <I>foobar</I> est invalide

Document Type Definition (DTD)

```

<!ELEMENT UL - - (LI)+>
<!ELEMENT IMG - O EMPTY>

<!ATTLIST MAP
  name      CDATA      #REQUIRED
>

  id        ID          #IMPLIED    -- document-wide unique id --
  rowspan   NUMBER 1     -- number of rows spanned by cell --

```

Caractères

- "Character Set": Unicode 2.0
- Plusieurs "encoding" possibles: ISO-8859-1 (Latin-1), EUC-JP, ...
- "numeric character references": "å" et "å" pour "å"
- "character entity references": "å"
 - saisie de caractères non disponible au clavier ou dans l'encoding utilisé
 - échappement de caractères: '<' pour '<', '"' pour '"'

Commentaires

```
<!-- ceci est un commentaire -->  
  
<!-- Attention -- ceci n'est pas valide! -->  
  
<P -- ceci non plus! -->
```

Structure Globale

Information sur la version d'HTML utilisée (DOCTYPE)

```
<!DOCTYPE HTML PUBLIC  
"-//W3C//DTD HTML 4.0//EN"  
"http://www.w3.org/TR/REC-html40/strict.dtd">  
  
<!DOCTYPE HTML PUBLIC  
"-//W3C//DTD HTML 4.0 Transitional//EN"  
"http://www.w3.org/TR/REC-html40/loose.dtd">  
  
<!DOCTYPE HTML PUBLIC  
"-//W3C//DTD HTML 4.0 Frameset//EN"  
"http://www.w3.org/TR/REC-html40/frameset.dtd">
```

Une section de déclaration (HEAD, TITLE, META, ...)

```
<HEAD>  
  <TITLE>HTML 4.0</TITLE>  
  <META name="Author" content="Arnaud Le Hors">  
</HEAD>
```

Le corps du document (BODY or FRAMESET)

Deux catégories d'éléments: "block-level" et "inline" (ou "text-level")

=> caractéristiques de présentation prédéfinies

Texte

- Paragraphe (P)
- Éléments textuels (EM, STRONG, CODE, ABBREV, ...)

As `<CITE>`Harry S. Truman`</CITE>` said,
`<Q>`The buck stops here.`</Q>`

- Contrôle de cassure (BR,)

```
<P>Voici un  
paragraphe<BR>avec une cassure, écrit  
par Arnaud Le&nbsp;Hors.
```

Voici un paragraphe
avec une cassure, écrit par Arnaud Le Hors.

- Paragraphe préformaté (PRE)

```
<PRE>Voici un  
paragraphe preformaté.  
</PRE>
```

Voici un
paragraphe preformaté.

- Support pour l'internationalisation: (lang, dir, BDO)

```
<P><q lang="en">To be or not to be?</q>  
est de l'anglais.
```

Listes

- Liste ordonnée et non ordonnée (OL, UL et LI)

```
<OL>  
  <LI>item 1  
  <LI>item 2  
</OL>
```

```
1. item 1  
2. item 2
```

- Liste de définitions (DL, DT et DD)

```
<DL>  
  <DT>Cracker  
  <DD>hacker on the Internet  
  
  <DT>Nerd  
  <DD>male so into the Net that he forgets  
    his wife's birthday  
</DL>
```

```
Cracker  
  hacker on the Internet  
Nerd  
  male so into the Net that he forgets his  
  wife's birthday
```

Tableaux

- Distribution principale des éléments par rangées (TR)

- Deux types de cellules: données (TD) et entêtes (TH)
- Groupes de rangées (THEAD, TFOOT et TBODY), permet le "scrolling" de TBODY
- Association d'un titre (CAPTION)
- Spécification de propriétés (taille) liées aux colonnes (COL et COLGROUP)

Note

A ne pas utiliser simplement pour contrôler le formatage d'un document!!! (cf. CSS Positioning)

Tableaux - Exemple

```
<TABLE border="1"
  summary="This table gives some statistics about fruit
          flies: average height and weight, and percentage
          with red eyes (for both males and females).">
  <CAPTION><EM>A test table with merged cells</EM></CAPTION>
  <TR><TH colspan="2"><TH colspan="2">Average
    <TH rowspan="2">Red<BR>eyes
  <TR><TH>height<TH>weight
  <TR><TH>Males<TD>1.9<TD>0.003<TD>40%
  <TR><TH>Females<TD>1.7<TD>0.002<TD>43%
</TABLE>
```

A test table with merged cells

	Average		Red eyes
	height	weight	
Males	1.9	0.003	40%
Females	1.7	0.002	43%

Liens

- Liens hypertexts (A):

You'll find a lot more in [chapter two](chapter2.html).

See [section 2](#section2) for details.

...

Section 2

... section 2 ...

See [section 2](#section2) for details.

...

Section 2

... section 2 ...

- "Meta" liens (LINK):

```
<HEAD>
  <TITLE>Chapter 2</TITLE>
  <LINK rel="Index" href="../index.html">
  <LINK rel="Next" href="Chapter3.html">
  <LINK rel="Prev" href="Chapter1.html">
</HEAD>
```

Images, Applets et Objets

- Inclusion d'une image (IMG)

```
<P>I just returned from vacation! Here's a photo  
of my family at the lake:  
<IMG src="vacation/family.png"  
alt="A photo of my family at the lake.">
```

Note:

alt est obligatoire -> accessibilité

- Inclusion d'une "applet" java (APPLET)

```
<APPLET code="Bubbles.class" width="500" height="500">  
Java applet that draws animated bubbles.  
</APPLET>
```

- Mécanisme général d'inclusion (OBJECT)

```
<P>I just returned from vacation! Here's a photo  
of my family at the lake:  
<OBJECT data="vacation/family.png" type="image/png">  
<OBJECT data="vacation/family.gif" type="image/gif">  
A photo of my family at the lake.  
</OBJECT>  
</OBJECT>
```

DIV et SPAN

- Mécanisme général de groupement d'éléments et d'augmentation de la structure
- DIV pour "block-level" et SPAN pour "inline"
- Pas de sémantique ni de présentation prédéfinies

```
<DIV id="section1">  
<H1>...</H1>  
<P>...  
</DIV>  
<DIV id="section2">  
<H1>...</H1>  
<P>...  
</DIV>
```

Style

- Collection d'éléments et attributs de présentation (CENTER, I, B, FONT, align, ...)
=> dépréciés
- Support pour feuilles de style (LINK, STYLE, style, class, id)

```
<HEAD>  
...  
<LINK href="mystyle.css" rel="stylesheet"  
type="text/css">  
</HEAD>
```

```
<HEAD>
```

```

...
<STYLE type="text/css">
...
</STYLE>
</HEAD>

<P style="...">Aren't style sheets wonderful?
</P>

<P class="important">Attention: ...</P>

```

- Mécanisme "ouvert": support différents langages (type, META)

```

<META http-equiv="Content-Style-Type"
content="text/css">

```

Frames

- Définition de documents multi-fenêtres
- Spécification de l'emplacement des sous-fenêtres (FRAMESET - **remplace** BODY)
- Spécification du contenu de chaque fenêtre (FRAME)
- Possibilité de cibler les liens (target)

```

<A target="frame1" ...">...</A>

```

Note:

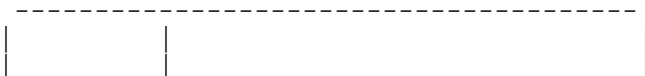
A éviter! Trop orienté vers les agents graphiques, pas de "bookmark" possible, pas de mécanisme de synchronisation.

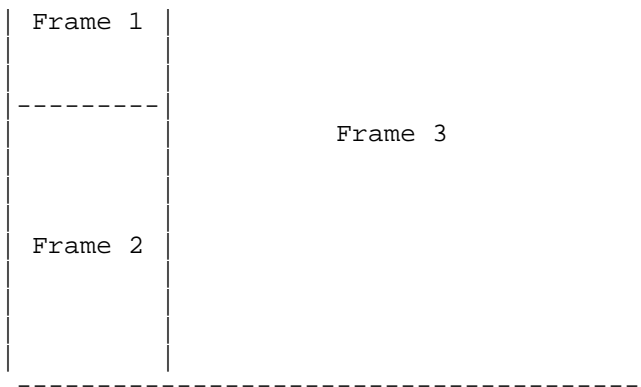
Frames - Exemple

```

<!DOCTYPE HTML PUBLIC
"-//W3C//DTD HTML 4.0 Frameset//EN"
"http://www.w3.org/TR/REC-html40/frameset.dtd">
<HTML>
<HEAD>
<TITLE>A simple frameset document</TITLE>
</HEAD>
<FRAMESET cols="20%, 80%">
<FRAMESET rows="100, 200">
<FRAME name="frame1" src="contents_of_frame1.html">
<FRAME src="contents_of_frame2.gif">
</FRAMESET>
<FRAME src="contents_of_frame3.html">
<NOFRAMES>
<P>This frameset document contains:
<UL>
<LI><A href="contents_of_frame1.html">Some neat contents</A>
<LI><IMG src="contents_of_frame2.gif" alt="A neat image">
<LI><A href="contents_of_frame3.html">Some other neat contents</A>
</UL>
</NOFRAMES>
</FRAMESET>
</HTML>

```





Formulaires

- Formulaires de saisie (FORM) associés a un programme CGI
- Divers types d'objets de saisie: boutons, zone de saisie, etc... (INPUT, SELECT, OPTION, ...)

```
<FORM action="http://somesite.com/prog/adduser" method="post">
<P>
  <LABEL for="firstname">First name: </LABEL>
  <INPUT type="text" id="firstname"><BR>
  <LABEL for="lastname">Last name: </LABEL>
  <INPUT type="text" id="lastname"><BR>
  <LABEL for="email">email: </LABEL>
  <INPUT type="text" id="email"><BR>
  <INPUT type="radio" name="sex" value="Male"> Male<BR>
  <INPUT type="radio" name="sex" value="Female"> Female<BR>
  <INPUT type="submit" value="Send"> <INPUT type="reset">
</P>
</FORM>
```

Script

- Documents dynamiques (SCRIPT)

```
<TITLE>Test Document</TITLE>
<SCRIPT type="text/javascript">
  document.write("<p><b>Hello World!\</b>")
</SCRIPT>
```

```
<TITLE>Test Document</TITLE>
<SCRIPT type="text/javascript" src="myscript.js">
</SCRIPT>
```

- Document interactifs: "event-handlers" (onclick, onload, ...)

```
<INPUT NAME="userName"
  onblur="validUserName(this.value)">
```

- Mécanisme "ouvert": support différents langages (type, META)

```
<META http-equiv="Content-Script-Type"
  content="text/tcl">
```


Validation HTML

- Documents valides = garantie de résultats; cas d'erreurs non spécifiés
- Spécification officielle: <http://www.w3.org/TR/REC-html40>
- Outils et services: nsgmls (<http://www.jclark.com>), <http://validator.w3.org>

"Dynamic HTML"

Terme marketing

Comprend:

- HTML +
- Style (CSS) +
- Scripting +
- Document Object Model (<http://www.w3.org/DOM>)

Evolution future d'HTML

- XHTML

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

Deuxième Partie

CSS level 2

CSS level 2

Language de feuilles de style pour documents HTML

Permet une séparation claire de la structure et du style
=> plus grande indépendance avec le media utilisé.

Recommandation W3C depuis le 12 mai 1998

Spécification à: <http://www.w3.org/TR/REC-CSS2>

Règles de style

Exemple:

```
H1 { color: blue }  
^^^^^^^^^^^^^^^^^ déclaration  
^^ sélecteur
```

Une règle comprend deux parties:

1. un sélecteur: le lien entre HTML et CSS
2. une déclaration: une propriété et une valeur

Intégration de style dans HTML

Utilisation de LINK, STYLE, et/ou style:

```
<!DOCTYPE HTML PUBLIC
"-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>Titre</TITLE>
<LINK rel="stylesheet" type="text/css"
href="http://style.com/cool" title="Cool">
<STYLE type="text/css">
@import url(http://style.com/basic);
H1 { color: blue }
</STYLE>
</HEAD>
<BODY>
<H1>Titre en bleu</H1>
<P style="color: green">Un paragraphe vert.
</BODY>
</HTML>
```

Groupements

Sélecteurs:

```
H1, H2, H3 { font-family: helvetica }
```

Déclarations:

```
H1 {
font-weight: bold;
font-size: 12pt;
line-height: 14pt;
font-family: helvetica;
font-variant: normal;
font-style: normal;
}
```

Autre:

```
H1 { font: bold 12pt/14pt helvetica }
```

Héritage

- En général les propriétés sont héritées du père
- Par exemple EM est bleu dans:

```
H1 { color: blue }
```

```
<H1>Le titre <EM>est</EM> important!</H1>
```

- Style sur BODY == "style par défaut":

```
BODY {  
  color: black;  
  background: url(texture.gif) white;  
}
```

- Exceptions: background, pourcentages...
- Possible de forcer l'héritage: inherit

Le sélecteur class

Exemple:

```
H1.important { color: #FF0000 }  
  
<H1 class="important">Attention!</H1>
```

Sélection de tous les éléments de même classe:

```
.important { color: #FF0000 }
```

Le sélecteur ID

Exemple:

```
#z98y { letter-spacing: 0.3em }  
  
<P ID="z98y">Il fait vraiment un temps  
incroyable.</P>
```

Equivalent à (ID est unique!):

```
P#z98y { letter-spacing: 0.3em }
```

Sélecteur contextuel

Exemple 1:

```
H1 EM { color: red }
```

- Affecte uniquement les éléments EM descendants d'un H1

Exemple 2:

```
H1 > EM { color: red }
```

- Affecte uniquement les éléments EM fils d'un H1

Exemple 3:

```
UL LI { font-size: small }
UL UL LI { font-size: x-small }
```

- Règle 1: éléments LI ayant au moins un ancêtre UL
- Règle 2: éléments LI ayant au moins deux ancêtres UL
- Conflit: le sélecteur le plus spécifique l'emporte

Mixage de sélecteurs

Exemple:

```
.reddish H1 { color: red }
#x78y CODE { background: blue }
DIV.sidenote H1 { font-size: large }
H1 B, H2 B, H1 EM, H2 EM { color: red }
```

Commentaires

Identiques à ceux de C:

```
EM { color: red } /* red, really red!! */
```

Pseudo-classes et pseudo-éléments

Sélection indépendante de la structure HTML

Pseudo-classes:

```
A:link { color: red } /* unvisited */
A:visited { color: blue }
A:active { color: lime }
DIV > P:first-child { text-indent: 0 }
:lang(fr) > Q { quotes: '« ' ' »' }
```

Pseudo-éléments:

```
P:first-line { font-style: small-caps }
P:first-letter { font-size: 200% }
P.special:before { content: "Special! " }
```

La cascade

Combinaison de feuilles de style par l'auteur:

```
@import url(http://www.style.org/marine);
/* surcharge du style importé */
H1 { color: red }
```

- Facilite le partage des feuilles de style

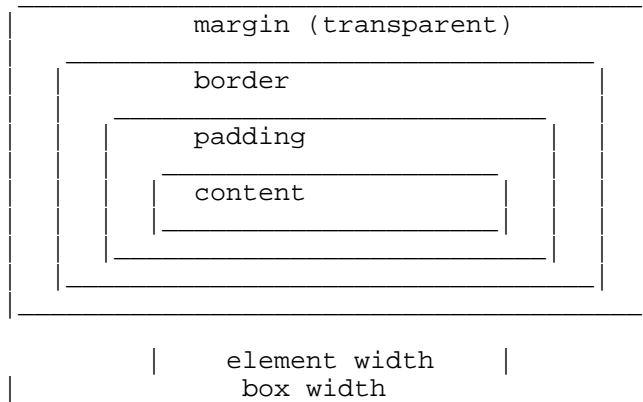
Combinaison des feuilles de style de l'auteur avec celle de "l'utilisateur".

L'ordre normal de precedence peut être modifié:

```
H1 { color: black ! important }
```

Le modèle de formattage

Modèle de boites:



Le modèle de formattage (2)

A chaque élément correspond une ou plusieurs boites.

Deux types principaux d'éléments:

- block-level éléments => changement de flot
- inline éléments => même flot

Les propriétés

- Polices de caractères
- Couleurs et fonds
- Propriétés textuelles (alignement, indentation...)
- Propriétés de boites (marges, encadrement...)
- Propriétés spéciales ('display', 'white-space'...)
- ...

Les unités

- Longueurs:
 - relatives: "0.5em", "1ex", "10px"
 - absolues: "0.5in", "3cm", "12pt"..
- Pourcentages: "120%"
- Couleurs: "black", "#f00", "rgb(255,0,0)"...

Positioning

- flottant
- relatif
- absolu ou fixe

Exemples:

```
IMG.icon {  
float: left;  
margin-left: 0;  
}
```

```
#header {  
position: fixed;  
width: 100%;  
height: 15%;  
top: 0;  
right: 0;  
bottom: auto;  
left: 0;  
}
```

Et d'avantage...

- différents types de media
- pagination (règle @page)
- feuilles de styles orales
- curseurs
- génération de contenu (compteurs)
- ...

Validation CSS

- Feuilles de style valides = garantie de résultats; cas d'erreurs non spécifiés
- Spécification officielle: <http://www.w3.org/TR/REC-CSS2>
- Outils et services: <http://jigsaw.w3.org/css-validator/>

Evolution future de CSS

CSS3 en préparation:

- support pour entêtes, bas de pages, multi-colonnes
- internationalisation: texte vertical et en grille
- ...

La suite: XHTML 1.0

Réécriture de HTML 4 en XML

- Documents bien formés: ``, `<hr />` ... (toutes les balises en minuscules, guillemets autour des valeurs des attributs)
- Possibilité d'utiliser un parser XML Compatible avec la plupart des navigateurs HTML4 en suivant quelques règles simples (<http://www.w3.org/TR/xhtml1/#guidelines>)
-
- Spécification : <http://www.w3.org/TR/xhtml1>

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