

Introduction à HTML5

Etat des lieux sur HTML5

Les technologies autour d'HTML5 sont le futur du Web mais rien n'est encore finalisé : certaines balises sont utilisables, d'autres sont expérimentales, d'autres encore sont propriétaires (spécifique à un seul navigateur).

Les différentes phases des spécifications au W3C :

- First Public Working Draft
- Working Draft (**Phase en cours**)
- Candidate Recommendation
- Proposed Recommendation
- Recommendation

Actuellement les spécifications du W3C sur HTML5 sont à l'état d'étude (Working Draft) et la première version des standards (Candidate Recommendation) est prévue en 2012. Ce type de spécifications est long à mettre en place ; la dernière phase (Recommendation) étant prévue pour 2022 ... Néanmoins, certains balises sont déjà définies comme standard et peuvent déjà être utilisées.

Liste des nouveautés avec HTML5

HTML5 est juste un markup, il faut donc plutôt parler d'HTML5 et ses technologies associées :

- HTML5 : Le markup de la future génération d'application Web
- CSS3 (Cascade Style Sheet) : Permet d'appliquer des styles au document
- SVG 1.1 (Scalable Vector Graphics) : Moteur de dessin vectoriel
- WOFF
- Géolocalisation
- WebForms
- Etc.

Le Markup HTML5

Le DOCTYPE

La balise DOCTYPE autrefois compliqué sans raison, a été largement simplifiée.

Avant :

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

Après :

```
<!DOCTYPE html>
```

Les balises structurelles

Plusieurs balises ont été ajoutées pour simplifier la structure des sites. Les balises du type `<div class=header>` ou `<div class=footer>` peuvent désormais être remplacées par `<header>`, `<nav>`, `<footer>`, `<section>`, `<article>` et `<aside>`.

Voici un découpage possible pour un site web avec ces nouvelles balises :



La balise `<audio>`

Permet de lire un fichier audio sans nécessiter un plug-in (Flash, Quicktime, Silverlight, ...). Le lecteur est fourni par défaut par le navigateur.

Voici les paramètres :

- Autoplay : lecture automatique
- Loop : lecture en boucle
- Controls : affichage des touches de lecture, pause, volume
- Src : permet de spécifier la source du media
- Preload : permet au navigateur de pré-charger le média : none, auto, metadata (seulement l'entête)

Le paramètre « preload » est inutile si « autoplay » est activé.

Compatibilité

Le W3C a laissé libre choix au navigateur pour le format de lecture. C'est pourquoi les 3 principaux navigateurs ne supportent pas les mêmes formats d'encodage. Le format WAV n'étant pas un format compressé, il est généralement à éviter.

| Navigateur \ Format | WAV | MP3 | OGG | AAC | M4A |
|---------------------|-----|-----|-----|-----|-----|
| Internet Explorer 9 | X | X | | X | X |
| Firefox 3.6 | X | | X | | |
| Google Chrome | X | X | X | | X |

La balise `<source>` permet de pallier à ces incompatibilités en spécifiant plusieurs sources pour un seul media. La lecture des sources est séquentielle : si le navigateur n'arrive pas à lire la première source, il lira la suivante, etc.

Voici un exemple :

```
<audio controls>
<source src="audio.mp3" />
<source src="audio.ogg" />
</audio>
```

Voici un lien permettant de tester les différents formats sur votre navigateur :

<http://jojaba.free.fr/html5/audio.html>

La balise <video>

Permet de lire un fichier vidéo sans nécessiter un plug-in (Flash, Quicktime, Silverlight, ...). Le lecteur est fourni par défaut par le navigateur. A savoir, il n'y a actuellement aucun support DRM et le streaming adaptatif n'est pas supporté.

Compatibilité

Comme pour la balise audio, tous les formats ne sont pas supportés par tous les navigateurs :

| Navigateur \ Format | H264 | WebM | Theora |
|---------------------|------|------|--------|
| Internet Explorer 9 | X | X* | |
| Firefox 4 | | X | X |
| Google Chrome | | X | X |
| Safari 5 | X | | |
| Opera 11 | | X | X |

* Nécessite d'installer le plugin.

La balise <source> fonctionne également avec la balise <video>. Voici un exemple :

```
<video controls>
  <source src="mavidéo.ogg" type="video/ogg" />
  <source src="mavidéo.mp4" type="video/mp4" />
  <object src="mavidéo.swf"
    type="application/x-shockwave-flash"
    width="800" height="500"
    allowscriptaccess="always"
    allowfullscreen="true" />
</video>
```

Dans cet exemple, si le navigateur ne support pas OGG et MP4, il lancera le plugin flash SWF.

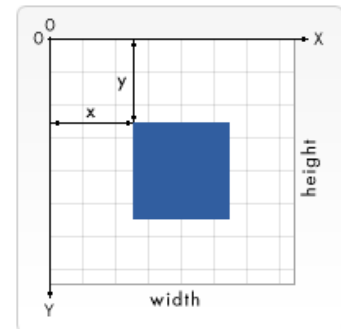
Voici un lien permettant de tester les différents formats sur votre navigateur :

<http://jojaba.free.fr/html5/video.html>

Utilisation des Canvas

La balise <canvas> permet de dessiner dans une Bitmap des formes et primitives simples au travers d'APIs JavaScript. Cette fonctionnalité notamment des créer des animations simples et des applications pouvant interagir avec l'utilisateur (par exemple, dessiner avec la souris).

Une présentation est disponible sur le « testdrive » de Microsoft :
<http://ie.microsoft.com/testdrive/Graphics/CanvasPad/Default.html>



Voici un exemple d'utilisation

```
<canvas id="monDessin" width="200" height="200">
<p>Votre navigateur ne supporte pas Canvas.</p>
</canvas>

<script language="javascript">
  // Récupération de l'élément Canvas
  var canvas = document.getElementById('monDessin');

  // Pour être sûr que le navigateur supporte Canvas
  if (canvas.getContext){

    // Utilisation de getContext pour dessiner sur le Canvas
    var ctx = canvas.getContext('2d');

    // Dessin de la bouche
    ctx.beginPath();
    ctx.arc(75,75,45,0,Math.PI,false);
    ctx.lineWidth = 10;
    ctx.strokeStyle = "rgb(255 ,215 ,0)";
    ctx.stroke();
    ctx.closePath();

    // Dessin des yeux
    ctx.beginPath();
    ctx.arc(35,40,10,0,Math.PI*2,true);
    ctx.moveTo(125,40);
    ctx.arc(115,40,10,0,Math.PI*2,true);
    ctx.fillStyle = "rgb(255 ,215 ,0)";
    ctx.fill();
    ctx.closePath();
  } else {
    alert('Votre navigateur ne supporte pas Canvas.');
```

Le rendu :



Quelques nouveautés de CSS3

CSS3 Media Queries

Permet de s'adapter aux différentes résolutions/périphériques par style :

- Résolution limitée pour les Smartphones
- Résolution normale pour les netbooks
- Large et haute résolution sur PC/Mac

Exemple, pour changer de fond suivant la taille de l'écran :

```
/* Fond par défaut */
.box
{
    Background-image : url('image1.jpg') ;
}

/* Fond pour les mobiles */
@media (max-width:480px)
{
    .box
    {
        Background-image : url('image1.jpg') ;
    }
}

/* Fond pour les netbooks */
@media (min-width:481px) and (max-width:1024px)
{
    .box
    {
        Background-image : url('image2.jpg') ;
    }
}

/* Fond pour les postes de travail */
@media (min-width:1025px)
{
    .box
    {
        Background-image : url('image3.jpg') ;
    }
}
```

CSS3 Border-radius

La propriété « border-radius » permet de créer des coins arrondis. Elle est définie en pixel et peut avoir 1 paramètre (si tous les arrondis sont identiques) ou 4 paramètres (haut-gauche, haut-droit, bas-droit, bas-gauche). Cette propriété peut être remplacée par :

- border-top-left-radius
- border-top-right-radius
- border-bottom-right-radius
- border-bottom-left-radius

Test sans border-radius

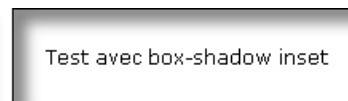
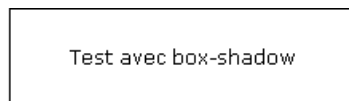
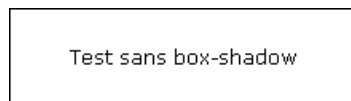
Test avec border-radius

Test avec border-radius élevé

CSS3 Box-shadow

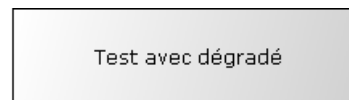
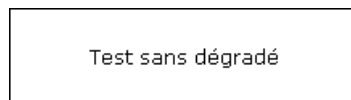
La propriété « box-shadow » permet de créer des ombres portées. Voici la liste des paramètres :

- Le premier paramètre est le décalage horizontal de l'ombre.
- Le second paramètre est le décalage vertical.
- La troisième valeur, toujours positive, est la longueur du dégradé.
- La quatrième valeur étend l'ombre dans la direction opposée.
- Un code de couleur donne la coloration de cette ombre.
- Le paramètre « inset » peut être ajouté pour créer l'ombre à l'intérieur de la boîte et non à l'extérieur.



CSS3 Gradient Background Maker

Dans la propriété « background-image », on peut définir un dégradé linéaire par le paramètre « linear-gradient ». Néanmoins, cette fonctionnalité est encore au stade expérimental, tous les navigateurs n'ont pas encore adopté la même syntaxe.



Pour plus d'informations :

<http://ie.microsoft.com/testdrive/graphics/cssgradientbackgroundmaker/>

CSS3 Multi-column Layout

Avec CSS3, il est désormais possible de définir des colonnes, à la manière des contenus éditoriaux de journaux. Il était auparavant très délicat de reproduire une telle structure de page. La propriété de base « column-count » permet de spécifier le nombre de colonnes.

```
.texte
{
    Column-count :auto ;
    Column-width :250px ;
}
```

L'exemple le code ci-dessus va créer des colonnes de 250 pixels, le nombre de colonnes changera automatiquement en fonction de la taille de l'écran. Cette propriété peut donc être très utile pour rendre compatible un site sur les mobiles.

Fonts WOFF

Les fonts Web Open Font Format permet de spécifier dans une feuille de style une police d'écriture sans se soucier de savoir si elle existe sur le poste client. Le navigateur sait où trouver les fonts WOFF sur le Web et les télécharge automatiquement sur le poste client. La plupart des navigateurs sont déjà compatibles, les autres sont en cours d'implémentation.

Scalable Vector Graphics (SVG)

Permet de créer et dessiner des éléments graphiques vectoriels en 2D avec du XML. Les éléments SVG sont accessibles depuis le DOM et peuvent être manipulés comme des balises normales (mettre du CSS, du JavaScript, etc.).

Exemple 1

Le code source :

```
<svg width="400" height="200" xmlns="http://www.w3.org/2000/svg">  
  <rect fill="red" x="20" y="20" width="100" height="75" />  
  <rect fill="blue" x="50" y="50" width="100" height="75" />  
</svg>
```

Le rendu :

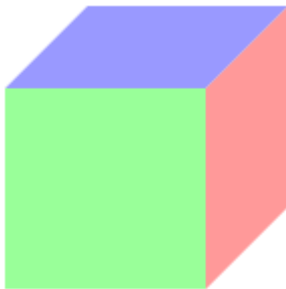


Exemple 2

Le code source :

```
<svg width="500" height="500" xmlns="http://www.w3.org/2000/svg">  
  <rect width="100" height="100" fill="#9F9" x="1" y="42" />  
  <polygon points="1,42 42,1 142,1 101,42 1,42" fill="#99F" />  
  <polygon points="101,42 142,1 142,101 101,142 101,42" fill="#F99" />  
</svg>
```

Le rendu :



Les technologies expérimentales

Certaines technologies proposées pour l'HTML5 sont encore au stade expérimental. Il est donc recommandé de ne pas les utiliser pour des applications professionnelles car elles peuvent évoluer très rapidement. Elles sont généralement préfixées :

- -o (Opéra),
- -ms (Internet Explorer),
- -webkit (Safari et Chrome),
- -moz (Firefox).

Voici un exemple de CSS markup expérimentale avec le paramètre linear-gradient vu précédemment :

```
/* IE10 */
background-image: -ms-linear-gradient(top left, #FFFFFF 0%, #00A3EF 100%);

/* Mozilla Firefox */
background-image: -moz-linear-gradient(top left, #FFFFFF 0%, #00A3EF 100%);

/* Opera */
background-image: -o-linear-gradient(top left, #FFFFFF 0%, #00A3EF 100%);

/* Webkit (Safari/Chrome 10) */
background-image: -webkit-gradient(linear, left top, right bottom, color-stop(0, #FFFFFF), color-stop(1, #00A3EF));

/* Webkit (Chrome 11+) */
background-image: -webkit-linear-gradient(top left, #FFFFFF 0%, #00A3EF 100%);

/* Proposed W3C Markup */
background-image: linear-gradient(top left, #FFFFFF 0%, #00A3EF 100%);
```

Les principales technologies expérimentales :

- WebSockets
- WebGL
- WebWorkers

Certaines de ces technologies ont été implémenté par certains navigateurs, mais ce n'est pas forcément un plus pour ces navigateurs. Pour preuve, WebSockets a été implémenté par Mozilla en début d'année et a donné lieu à une importante faille de sécurité. Depuis, les navigateurs ont tous désactivés par défaut cette fonctionnalité (mais l'utilisateur peut toujours la réactiver dans les options du navigateur).

Actuellement il n'est pas possible de dire si un navigateur supporte mieux HTML5 qu'un autre car une grande partie des technologies associées ne sont pas encore stables et définies précisément par W3C.

Comment savoir si son navigateur support bien HTML 5

Le test <http://html5test.com/> donne un score sur 400. Le score n'est qu'une indication de la façon dont le navigateur prend en charge les spécifications du standard HTML5 et des technologies liées à venir. Sachant que plusieurs technologies sont encore expérimentales et non standardisées, ce score est à prendre à la légère. De plus que nous ne savons pas sur quelle base sont ventilés les points.

Le test <http://acid3.acidtests.org>, créé par Ian Hickson (Google), permet de lancer 100 tests unitaires de base. A savoir que le W3C prévoit plus de 10000 tests unitaires dans les standards à venir.

L'outillage

Majoritairement du Notepad (voir Notepad++)...

Les outils modernes ne proposent pour l'instant que de l'aide via auto-complétion :

Helper HTML5 pour Visual Studio 2008 & 2010 :

<http://visualstudiogallery.msdn.microsoft.com/fr-FR/d771cbc8-d60a-40b0-a1d8-f19fc393127d>

Adobe Dreamweaver CS5 va un peu plus loin avec de la prévisualisation multi-écrans :

<http://labs.adobe.com/technologies/html5pack/>

Certains commencent à le proposer dans les templates comme WebMatrix :

<http://asp.net/webmatrix>

Canvas

Adobe Illustrator propose le plugin Ai->Canvas qui permet d'exporter les vecteurs et illustrations bitmap directement en code pour la balise <canvas> :

<http://visitmix.com/work/ai2canvas/>