



# 2

## Le marquage HTML5

Nous avons exploré au cours du Chapitre 1 beaucoup de nouveaux éléments structurels associés au HTML5. Les balises structurales permettent de décrire précisément et en détail les différentes parties d'un document. Vous avez également appris que de nombreux attributs associés à ces balises existent. Certains d'entre eux sont propres à certaines balises, alors que d'autres sont globaux et s'appliquent à toutes.

Nous allons revoir la plupart des balises que vous avez probablement utilisées dans des versions antérieures du HTML. Cependant, vous constaterez qu'elles ont été considérablement enrichies en HTML5. Vous apprendrez également comment, grâce au HTML5, vous pourrez relier votre application au monde extérieur et, de manière plus importante, au multimédia. Tentons de résoudre ici quelques problèmes.

### Solutions 2.1 – Utiliser la balise `<hr>` en HTML5

Dans la version précédente du HTML, la balise `<hr>` était utilisée pour créer des lignes horizontales dans une page. En HTML5, sa sémantique a changé.

De prime abord, elle semble faire la même chose que précédemment. Cependant, le but du HTML est de décrire les différentes parties du document. Auparavant, la balise `<hr>` désignait une ligne horizontale et, bien que cela soit pratique, elle ne décrivait rien d'autre qu'une ligne horizontale et non une partie du document.

La sémantique de la balise `<hr>` a été modifiée par le W3C. Maintenant, son but est officiellement de "définir la fin d'une section et le début d'une autre". C'est ici qu'une confusion surgit :

comme nous l'avons vu au Chapitre 1, le HTML5 dispose d'une balise appelée `<section>`, prévue pour séparer les sections. C'est ici un vrai sujet de débat parmi les développeurs. Au moment de l'écriture de cet ouvrage, un consensus apparaît sur le fait que la balise `<hr>` peut être utilisée pour séparer les sujets au sein d'une section. Le HTML5 étant encore en construction et, probablement pour un long moment encore, une définition plus précise se fera sûrement jour.

## Tour d'horizon

Supposons que vous souhaitiez séparer plusieurs paragraphes par une ligne horizontale (voir Figure 2.1).



**Figure 2.1**

*Utilisation de la balise `<hr>`.*

Si vous êtes perplexe et que vous vous demandiez quelle est ici la différence, sachez que vous n'êtes pas seul et que c'est le cas de beaucoup de développeurs ! Alors qu'à la Figure 2.1, l'élément `<hr>` semble avoir la même fonctionnalité que dans les versions précédentes du HTML, elle sépare ici véritablement les sujets contenus dans la page ("donner un goût relevé" et "servir vos viandes de manière variée"). À la Figure 2.1, nous l'utilisons également pour séparer les descriptions et le titre ("Comment faire de la viande séchée").

## Mise en œuvre

L'exemple de la Figure 2.1 a été produit avec le code suivant :

```

<!DOCTYPE html>
<html>
<head>
<title>Utiliser la balise <hr> </title>
</head>
<body>
<h1>Comment faire de la viande séchée</h1>
Cette page décrit les différentes manières de préparer de la viande séchée.
<h2>Épices</h2>
<hr>
Les épices donnent à votre viande un goût relevé.
<hr>
Elles permettent de servir vos viandes de manière variée.
</body>
</html>
  
```

Selon le W3C, les sections devraient être séparées plutôt par la balise `<hr>` que par la balise `<p>`. Nous n'avons pas utilisé cette dernière dans cet exemple afin de nous conformer aux standards du HTML5, mais elle peut bien évidemment être employée pour séparer les paragraphes au sein d'une section.

## Conseil d'expert

Afin d'adapter la nouvelle définition sémantique de la balise `<hr>`, vous pouvez utiliser les CSS pour modifier son apparence. Notez l'exemple suivant :

```
hr{
    height: 15px;
    background: url('decorative.gif');
    no-repeat 50% 50%;
}
```

Mais vous devez être prudent avec cette démarche. Au moment où nous écrivons cet ouvrage, si Safari et Firefox remplacent bien la ligne horizontale avec l'élément graphique de ce code, ce n'est pas le cas d'Internet Explorer, lequel entoure l'élément graphique de lignes horizontales.

## Solution 2.2 – Utiliser la balise `<iframe>`

Vous pouvez choisir la balise `<iframe>` pour créer un navigateur au sein d'un autre navigateur et ouvrir ainsi des documents provenant d'autres localisations (voir Figure 2.2).



**Figure 2.2**

*Ouverture de deux sites web externes avec la balise `<tag>`.*

À la Figure 2.2, nous avons ouvert deux sites web séparés avec deux balises `<iframe>`. Cette balise a une importance particulière car le HTML5 ne prend pas en charge les

attributs `scrolling`, `frameborder` et `marginheight`, alors qu'ils l'étaient tous dans les versions antérieures.

## Tour d'horizon

L'une des spécifications proposées de la balise `<iframe>` est l'attribut `seamless`. Il permet à une source extérieure d'être intégrée dans le document hôte sans qu'il y ait de bordure ou de barre de défilement. Cela signifie que la source extérieure apparaîtra comme si elle appartenait au document hôte. Malheureusement, aucun navigateur ne prend cet attribut en charge au moment où nous écrivons cet ouvrage.

## Mise en œuvre

Le code suivant a été utilisé pour créer l'exemple visible à la Figure 2.2 :

```
<!DOCTYPE html >
<html>
<head>
  <title>Utilisation de la balise <iFrame></title>
</head>
<body>
<strong> Utilisation de la balise iFrame</strong><hr>
  <iframe src="http://www.apress.com/" width="600" height="250" seamless>
  </iframe><br>
  <iframe src="http://www.friendsofed.com/" width="600" height="250" seamless>
  </iframe><br>
</body>
</html>
```

Notez que, bien que l'attribut `seamless` ait été utilisé dans ce code, vous voyez toujours les barres de défilement ainsi que les bordures. Cela confirme que le navigateur ne le prend pas en charge.

## Conseil d'expert

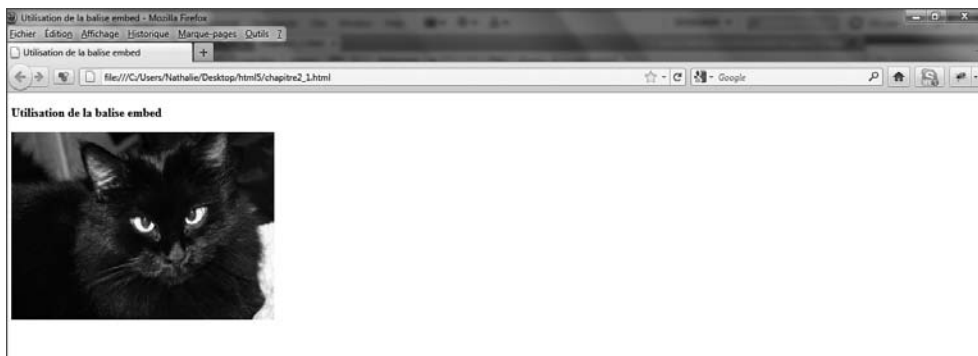
Dès lors qu'un contenu a une source extérieure, la sécurité est un facteur important. En HTML5, la balise `<iframe>` dispose d'un bac à sable qui lui permet d'améliorer sa sécurité. Les attributs du bac à sable incluent :

- **allow-scripts.** Bloque ou autorise l'exécution d'un script, y compris en JavaScript.
- **allow-forms.** Bloque ou autorise la soumission de formulaire.
- **allow-same-origin.** Force le contenu du document extérieur à provenir la même origine que celui qui l'affiche. Ainsi, une page ne peut recevoir de données depuis une origine et écrire ensuite une donnée vers une autre origine.
- **allow-top-navigation.** Bloque ou autorise la navigation vers l'élément parent `'_top'`.

## Solution 2.3 – Intégrer un média dans une page

Tout comme la balise `<iframe>` de la section précédente, la balise `<embed>` sert à intégrer un contenu externe dans une page web hôte. Cependant, l'accent est porté ici sur les médias tels

que des photos, des vidéos, des sons et, malgré les nombreuses rumeurs affirmant le contraire, du contenu Flash. La Figure 2.3 est un exemple du résultat que l'on peut obtenir (en supposant qu'une photographie ait été appelée depuis une source extérieure).



**Figure 2.3**

*Utilisation de la balise <embed>.*

## Tour d'horizon

Vous pensez probablement que vous pourriez arriver au même résultat avec la balise `<img>`, mais la balise `<embed>` est appropriée pour inclure une variété de médias dans vos pages web, par exemple, des vidéos, des fichiers audio, etc. En ce qui concerne la vidéo, les choses sont un peu en transition à l'heure actuelle.

Au moment de l'écriture de cet ouvrage, il semble qu'il y ait une mouvance montante contre l'utilisation de plugins de navigateur tels qu'Adobe Flash Player. Beaucoup de technologies mobiles ne les prennent pas en charge. Par exemple FLV, l'un des formats vidéo Flash les plus utilisés, nécessite Adobe Flash Player. La balise `<embed>` semble donc *a priori* apporter une solution à ce problème, mais il faut faire attention. Les vidéos peuvent être de formats divers – FLV, MOV, H.264 etc. –, et chacun de ces types MIME nécessite d'encoder la vidéo originale. Concrètement, cela veut dire l'entourer d'un conteneur, un Codec (compression/décompression). De plus, la personne souhaitant visionner la vidéo doit avoir, sur son ordinateur, une copie installée du codec en question. Dans le cas de la vidéo Flash, celui-ci est contenu au sein même du Flash Player.

Actuellement, deux codecs prennent en charge la vidéo HTML5 : H.264 et Theora. Sachant cela, voici la position des différents navigateurs (là encore, au moment où nous écrivons cet ouvrage) :

- Firefox 3.5 et supérieur : accepte Theora et pas H.264.
- Internet Explorer 9 : accepte H.264 et pas Theora.
- Safari 3 et supérieur : accepte H.264 et pas Theora.
- Chrome 3 : accepte H.264 et Theora.
- Opera 10.5 : accepte Theora et pas H.264.

### Mise en œuvre

Le code suivant a été utilisé pour créer l'exemple que vous voyez Figure 2.3 :

```
<!DOCTYPE html>
<html>
<head>
<title> Utiliser la balise embed </title>
</head>
<body>
<p><strong>Utiliser la balise embed</strong></p>
<embed type="jpg" src="002.jpg" height="250" width="350" />
</body>
</html>
```

Ce qui distingue cette balise est son attribut `type`, que vous associez probablement avec des balises telles que `<input>`. Mais, en HTML5, l'utilisation des attributs est devenue beaucoup plus cohérente sur l'ensemble des balises. Dans la plupart des cas, le type MIME du média (l'extension de fichier) sert comme `type`. Comme vous allez très vite le voir, il existe une autre façon d'identifier les types de vidéo.

L'autre attribut unique est `src`. Il peut s'agir de n'importe quelle URL nécessaire pour se connecter à la source de l'élément.

### Conseil d'expert

Comme nous l'avons vu, les navigateurs prennent en charge différents encodages pour la vidéo en HTML5. Pour cette raison, il peut être nécessaire de réaliser deux encodages : un pour H.264 et un pour Theora. Pour cela, utilisez l'attribut `type` de la balise `<embed>` afin d'identifier la vidéo encodée à ouvrir.

Lorsque vous employez l'attribut `type` avec de la vidéo, le type de vidéo peut être identifié avec la syntaxe `video/type`. Par exemple, pour une vidéo QuickTime, vous l'identifierez ainsi : `type= "video/QuickTime"`.

## Solution 2.4 – Utiliser la balise `<area>`

Vous connaissez déjà la balise `<area>` présente dans les anciennes versions du HTML, mais celle implémentée en HTML5 apporte de nouvelles possibilités intéressantes. Nous allons voir lesquelles.

### Tour d'horizon

On utilise la balise `<area>` pour créer des hyperliens sur des zones d'image. En d'autres termes, elle prend une ressource graphique et la découpe en sections disposant chacune d'hyperliens uniques. Cependant, le HTML5 considère les liens différemment des versions précédentes du HTML. Il les divise en trois catégories :

- **Les hyperliens.** Ils permettent à l'utilisateur de se rendre vers une ressource donnée.

- **Les ressources externes.** Les liens vers des ressources, qui sont traitées automatiquement pour agrémenter le document actuel.
- **Les annotations.** Les modifications des ressources automatiques utilisées dans les ressources externes. Elles peuvent aussi modifier le fonctionnement des hyperliens.

Dans les versions antérieures du HTML, l'attribut `rel` servait à montrer la relation existant entre une ressource et celle à laquelle elle est liée. Regardez par exemple ce qui suit :

```
rel= "stylesheet"
```

Ce code indique que le document auquel la ressource est liée est une feuille de style. Bien que cet aspect semble négligeable, il peut avoir d'énormes implications pour un moteur de recherche qui comprend les relations existant entre les différents documents à l'intérieur d'un site web.

L'attribut `rel` n'est pas nouveau, mais le HTML5 étend les définitions relationnelles.

Le Tableau 2.1 montre la relation entre le document externe et la balise `<area>` (au moment de l'écriture de ce livre). Cette spécification est encore en transition, et les informations contenues dans le tableau peuvent changer. Cependant, vous serez étonné de voir que les valeurs de référence représentent maintenant des relations aussi diverses que familiales, professionnelles et même amoureuses.

**Tableau 2.1. Les relations à un document référencé depuis la balise `<area>`**

<i>Valeur rel</i>	<i>Utilisé avec la balise <code>&lt;area&gt;</code></i>	<i>Description</i>
acquaintance	oui	Indique que la personne représentée dans le document référencé est une connaissance de la personne représentée dans le document en cours.
alternate	oui	Indique que le document référencé doit être utilisé avec le média spécifié. En HTML5, l'attribut <code>media</code> peut être utilisé avec la balise <code>&lt;area&gt;</code> , ce qui n'était pas le cas des versions précédentes.
archives	oui	Un lien vers un document ayant un intérêt historique.
author	oui	Indique que le document référencé donne des informations supplémentaires sur l'auteur.
bookmark	oui	Indique que le document référencé est plus ancien que le document hôte.
child	oui	Indique que la personne représentée dans le document référencé est un enfant de la personne représentée dans le document en cours.
colleague	oui	Indique que la personne représentée dans le document référencé a une relation professionnelle avec la personne représentée dans le document en cours.
contact	oui	Information de contact.
co-resident	oui	Indique que la personne représentée dans le document référencé partage une adresse avec la personne représentée dans le document en cours.

## 24 Solutions HTML5

<i>Valeur rel</i>	<i>Utilisé avec la balise &lt;area&gt;</i>	<i>Description</i>
co-worker	oui	Indique que la personne représentée est un collègue de travail de la personne représentée dans le document en cours.
crush	oui	Indique que la personne représentée dans le document référencé a un désir amoureux pour la personne du document en cours.
date	oui	Indique que la personne représentée dans le document référencé sort avec la personne du document en cours.
external	oui	Montre que le document référencé ne fait pas partie du site du document hôte.
first	oui	Indique que le document référencé est le premier document d'une série.
friend	oui	Indique que la personne représentée dans le document référencé est un ami de la personne représentée dans le document courant.
help	oui	Montre que le document courant est un document d'aide.
icon	non	Importe une icône pour représenter le document référencé.
index	oui	Lien vers un index ou une table des matières.
kin	oui	Indique que la personne représentée dans le document référencé est un membre de la famille, ou de la famille élargie, de la personne du document courant.
last	oui	Indique que le document référencé est le dernier document d'une série.
licence	oui	Est un lien vers une information de licence du document en cours.
me	oui	Indique que le document en cours et le document référencé vous représentent.
met	oui	Indique que la personne représentée dans le document référencé et la personne du document en cours se sont rencontrées.
muse	oui	Indique que la personne représentée dans le document référencé est une inspiration pour la personne représentée dans le document en cours.
next	oui	Indique que le document référencé est le document suivant sur une liste de documents référencés.
neighbor	oui	Indique que la personne représentée dans le document référencé est un voisin de la personne représentée dans le document en cours.
nofollow	oui	Indique que le document référencé n'est pas approuvé par l'auteur du document hôte mais qu'il est utilisé comme référence ou comme relation.
noreferrer	oui	Signifie que l'en-tête HTTP n'utilise pas la propriété <code>referrer</code> dans l'hyperlien.
parent	oui	Indique que la personne représentée dans le document référencé est parente avec la personne représentée dans le document courant.



<i>Valeur rel</i>	<i>Utilisé avec la balise &lt;area&gt;</i>	<i>Description</i>
pingback	non	Autorise le serveur référencé à faire un pingback sur le document en cours.
prefetch	oui	Permet au document référencé d'être préchargé.
prev	oui	Comme pour la valeur next, indique que le document appartient à une liste de documents référencés, et qu'il est le précédent.
search	oui	Lie à un outil effectuant des recherches sur le document référencé ou sur le document en cours.
sidebar	oui	Indique le document référencé comme une colonne latérale dans le navigateur.
sibling	oui	Indique que la personne représentée dans le document référencé est le frère ou la sœur de la personne du document en cours.
spouse	oui	Indique que la personne représentée dans le document référencé est mariée à la personne du document en cours.
sweetheart	oui	Indique que la personne représentée dans le document référencé à une relation amoureuse avec la personne du document en cours.
stylesheet	non	Indique que le document référencé est une feuille de style
tag	oui	Indique qu'un tag – ou mot-clé – du document référencé s'applique au document en cours.

Dans tous les liens rel répertoriés dans le Tableau 2.1:

- Les hyperliens sont créés avec les valeurs acquaintance, alternate, archives, author, bookmark, child, colleague, contact, co-resident, co-worker, crush, date, external, first, friend, help, index, kin, license, me, met, muse, neighbor, next, parent, prev, search, sidebar, sibling, spouse, sweetheart et tag.
- Les ressources externes sont référencées avec prefetch.
- Les annotations sont utilisées par nofollow et noreferrer.

De même, depuis le HTML5, la balise <area > peut utiliser l'attribut media afin d'indiquer le média que le lien va cibler. Dans les versions précédentes, ce n'était pas possible. Mais, pour une plus grande cohérence, cela a été réalisé en HTML5.

## Mise en œuvre

Il existe de multiples façons d'utiliser la balise <area>. L'exemple qui suit en montre une. Notez qu'il s'agit d'une démonstration uniquement et non d'un site fonctionnel.

```
<!DOCTYPE html >
<html>
<head>
<title> attribut href de la balise Area</title>
</head>
```

```
<body>

<map name="usamap">
  <area coords="20, 80, 122, 230" href="http://www.friendsofed.html"
    ▶ shape="rect" hreflang="en" rel="license" media="screen" />
  <area coords="123, 82, 288, 160" href="http://www.apress.html" shape="rect"
    ▶ hreflang="en" rel="license" media="print"/>
</map>
</body>
</html>
```

La balise `<area>` est utilisée en conjonction avec la balise `<map>`. Remarquez l'attribut `usemap` dans la balise `<img>`. Le même identifiant apparaît dans l'attribut `name` de la balise `<map>` ce qui associe l'image avec une carte.

L'attribut `media` sert à indiquer quel est le média associé au lien. Comme nous l'avons dit auparavant, la possibilité de l'employer dans la balise `<area>` est une nouveauté du HTML5.

## Résumé

Nous avons vu au cours de ce chapitre de quelle manière le marquage en HTML5 apporte de nouvelles fonctionnalités aux balises HTML traditionnelles. Nous avons aussi examiné comment relier un document HTML5 au monde extérieur à l'aide de balises telles que `<iframe>` et `<area>`.

Nous allons voir, au Chapitre 3, comment le HTML5 utilise l'API DOM et, ce faisant, nous reverrons quelques concepts que vous avez appris ici ainsi qu'au fil du Chapitre 1, mais cette fois dans un contexte plus complexe.