

Créer une boîte d'onglet à la mode web 2.0 avec CSS et jQuery Partie HTML / CSS

par [Didier Mouronval](#)

Date de publication : 25 octobre 2008

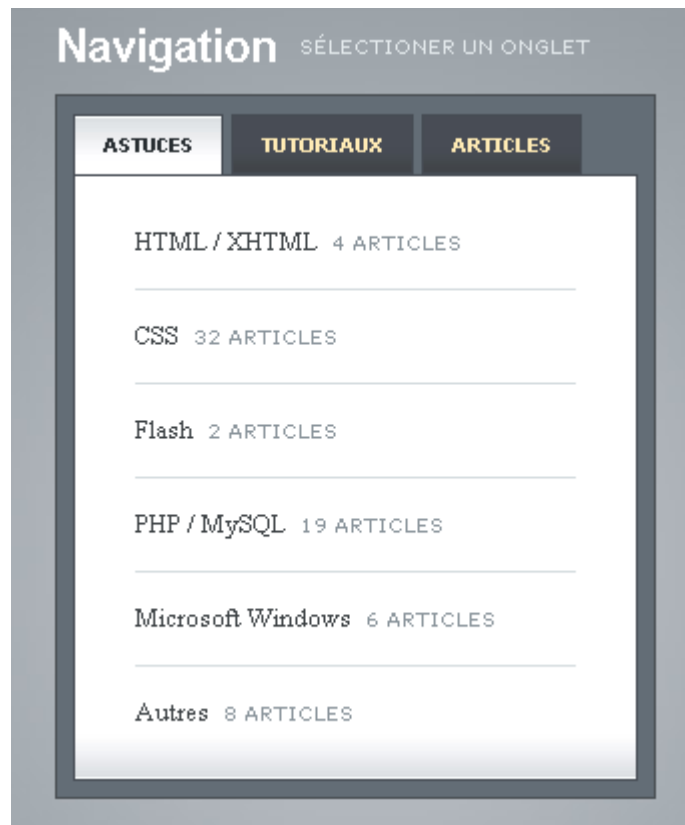
Dernière mise à jour :

Un des plus grands défis d'un concepteur web est de trouver un moyen de placer beaucoup d'informations sans perdre de lisibilité. Les onglets sont un bon moyen de gérer ce type de problème. Depuis l'avènement du Web 2.0, on voit une quantité phénoménale de blogs utilisant cette technique. Nous allons donc voir comment créer une boîte d'onglets en HTML et en y ajoutant des fonctionnalités à l'aide de CSS, JavaScript et jQuery.

Traduit et inspiré de :  [Source](#).

Visualisation.....	3
Etape 1.....	3
Etape 2.....	3
Etape 3.....	4
Etape 4.....	4
Etape 5.....	5
Etape 6.....	6
Etape 7.....	6
Etape 8.....	7
Etape 9.....	7
Etape 10.....	7
Etape 11.....	8
Etape 12.....	8
Etape 13.....	9
Etape 14.....	10
Etape 15.....	10

Visualisation



Etape 1

Tout d'abord, un voyage rapide dans Photoshop afin de créer un petit design Web 2.0 et le tour est joué. Nous avons une belle maquette de ce que notre boîte à onglets devrait ressembler. C'est assez simple! Vous pouvez récupérer les fichiers **Photoshop PSD** pour cette image si vous souhaitez.

Etape 2

La première chose à faire lors de la construction est d'obtenir une idée d'ensemble de ce que vous allez faire. Cela devient plus facile par la suite de générer votre boîte à onglets. En regardant cette image, je dirais que la meilleure chose à faire est:

- 1 Avoir un conteneur principal <div> qui contiendra tout le reste. De cette façon, si l'on désire déplacer notre boîte à onglets, il sera facile de le faire. Il sera aussi plus facile de mettre en forme cette boîte par la suite.
- 2 Ensuite, nous aurons la zone d'en-tête, probablement avec un tag <h>
- 3 Ensuite, juste en dessous, nous aurons un second conteneur <div> qui contiendra les onglets et le contenu. C'est la boîte gris foncé dans l'image.
- 4 A l'intérieur, nous allons utiliser une liste à puces où chaque élément est un lien pour les onglets. Cela nous permettra d'utiliser le et d'obtenir une mise en forme plus flexible.
- 5 Puis en dessous, nous allons créer un dernier conteneur <div> qui contiendra le contenu de chaque onglet. Nous aurons besoin d'un conteneur pour chaque onglet et nous allons les afficher ou les masquer en fonction de l'onglet sur lequel on a cliqué.

Donc, pour résumer, cela ressemblera à ceci :

```
<div>
  <h4>En-tête</h4>
```

```
<div>
  <ul>
    <li><a>Tab 1</a></li>
    <li><a>Tab 2</a></li>
    <li><a>Tab 3</a></li>
  </ul>

  <div>Contenu de la Tab 1</div>
  <div>Contenu de la Tab 2</div>
  <div>Contenu de la Tab 3</div>
</div>
</div>
```

Ne vous inquiétez pas si, en regardant cette portion de code, vous ne voyez pas immédiatement le rendu final. En outre, il est bon de réfléchir à la structure de ce type avant d'avoir beaucoup trop de classes, de noms et d'identifiants, car il peut devenir difficile de voir l'ensemble clairement lorsque tout est terminé. Le squelette de votre boîte est plus facile à lire que son rendu final.

Maintenant que nous avons une image dans l'esprit, à savoir comment construire notre structure, nous allons entrer dans le vif du sujet.

Etape 3

Par souci du détail, avant même de commencer par la boîte à onglets, créons un design particulier!

Ouvrez Photoshop, créez un nouveau document 1000px x 1000px et tirez un radial dégradé semblable à celui ci-dessus. Notez que j'ai tiré du centre / haut et fait en sorte que le dégradé se termine au moment où l'on atteint le bord du document. Cela nous permettra plus tard de définir la couleur d'arrière-plan en HTML et de l'associer à la couleur plus foncée de votre image.

Etape 4

Pour commencer, créez un nouveau répertoire pour le projet, ensuite créez un second répertoire appelé « images » à l'intérieur et enregistrez l'image créée précédemment dans ce dossier sous le nom « background.jpg ». Lors de la sauvegarde, utilisez « Fichier > Enregistrer pour le Web » et sélectionnez JPG avec un paramètre de qualité d'environ 70.

Maintenant, nous allons créer le document HTML :

```
<!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" xml:lang="en" lang="en">
<head>
  <title>Boîte à onglets - Web 2.0</title>
  <link rel="stylesheet" href="/style.css" type="text/css" media="screen" />
</head>
<body>
</body>
</html>
```

Ceci sera notre base HTML. Nous allons maintenant attaquer la base de notre fichier de style nommer « style.css » :

```
body {
  background-image:url(images/background.jpg);
  background-repeat:no-repeat;
  background-position:top center;
  background-color:#657077;
  margin:40px;
}
```

Quelques trucs à noter avant de continuer :

- 1 Il est possible d'écrire ce même CSS en utilisant des raccourcis et de réduire le nombre de lignes utilisées, mais il est beaucoup plus clair de cette manière afin de vous démontrer ce que nous faisons !
- 2 Nous avons une image d'arrière-plan en dégradé que nous avons mis à no-repeat, car nous voulons qu'elle n'apparaisse qu'une seule fois, elle est centrée et la couleur d'arrière-plan (#657077) est la couleur plus foncée que nous avons utilisé dans Photoshop.
- 3 J'ai ajouté une marge de 40px. Question de positionnement !

Etape 5

Maintenant, nous allons ajouter notre structure à notre page :

```
<!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" xml:lang="en" lang="en">
<head>
  <title>Boîte à onglets - Web 2.0</title>
  <link rel="stylesheet" href="/style.css" type="text/css" media="screen" />
</head>

<body>
<div id="tabbed_box_1" class="tabbed_box">
  <h4>Navigation <small>Sélectionner un onglet</small></h4>
  <div class="tabbed_area">
    <ul class="tabs">
      <li><a href="/" id="tab_1" class="active">Astuces</a></li>
      <li><a href="/" id="tab_2">Tutoriaux</a></li>
      <li><a href="/" id="tab_3">Articles</a></li>
    </ul>

    <div id="content_1" class="content">Contenu pour l'onglet 1</div>
    <div id="content_2" class="content">Contenu pour l'onglet 2</div>
    <div id="content_3" class="content">Contenu pour l'onglet 3</div>
  </div>
</div>
</body>
</html>
```

Comme vous pouvez le voir, j'ai utilisé essentiellement la même structure que mentionnée à l'étape 2. J'ai ajouté des identifiants et des classes et un vrai contenu.

- 1 Le conteneur <div> a un id = "tabbed_box_1" et une class = "tabbed_box". J'ai fait cela parce que nous pouvons utiliser ce code plusieurs fois sur la même page. Si nous avions plus d'une boîte à onglets, nous pourrions utiliser l'identifiant.
- 2 J'ai aussi ajouté les liens et le contenu. Les identifiants ont aussi été ajoutés au contenu parce que nous aurons besoin d'utiliser le JavaScript pour les manipuler plus tard.
- 3 Enfin, nous avons associé une classe à l'élément . En fait, nous pourrions modifier le style sans classe, mais cela pourrait se confondre avec les autres . Il est donc préférable pour elle d'avoir un nom de classe auquel se référer.



Etape 6

Maintenant, travaillons le style ! Il est toujours préférable de travailler de l'extérieur vers l'intérieur. C'est ce que nous allons faire et commencer par le conteneur principal : `tabbed_box` :

```
#tabbed_box {
  margin: 0px auto 0px auto;
  width: 300px;
}
```

Etape 7

Maintenant, passons à l'en-tête :

```
.tabbed_box h4 {
  font-family: Arial, Helvetica, sans-serif;
  font-size: 23px;
  color: #ffffff;
  letter-spacing: -1px;
  margin-bottom: 10px;
}
.tabbed_box h4 small {
  color: #e3e9ec;
  font-weight: normal;
  font-size: 9px;
  font-family: Verdana, Arial, Helvetica, sans-serif;
  text-transform: uppercase;
  position: relative;
  top: -4px;
  left: 6px;
  letter-spacing: 0px;
}
```

Quelques notes avant de continuer :

- 1 Au lieu de définir le style pour les éléments `<h4>`, j'ai défini « `.Tabbed_box h4` ». Ceci est important dans un grand document HTML, car vous avez peut-être un autre style pour les éléments `<h4>` à un autre endroit. Donc, vous voulez vous assurer que vos styles ne se chevauchent pas.
- 2 Vous remarquerez que j'ai également ajusté la marge du bas de l'élément `<h4>` à 10px. Il en est ainsi pour l'espacement entre les caractères. Il est important de savoir que de nombreux éléments ont des valeurs par défaut.
- 3 Aussi vous remarquerez dans l'élément `<small>` que j'ai donné une `position: relative`. Elle me permettra d'ajuster facilement son positionnement.
- 4 Enfin, vous remarquerez que pour l'élément `<h4>`, nous avons donné -1 comme valeur à l'espacement des caractères et 0 px à l'élément `<small>`. Ceci à cause de l'héritage CSS. C'est-à-dire que l'enfant hérite du

parent. Dans notre cas, l'élément `<small>` aurait hérité de l'espacement de -1 px de son parent. Comme nous ne le voulons pas, il a fallu que nous lui donnions une valeur.

Etape 8

Stylisons ensuite le conteneur interne :

```
.tabbed_area {  
  border:1px solid #494e52;  
  background-color:#636d76;  
  padding:8px;  
}
```

Etape 9

Maintenant, la partie la plus difficile, les onglets. Si vous ajoutez cette petite portion de code, vous aurez une bonne idée du produit final.

```
ul.tabs {  
  margin:0px; padding:0px;  
}  
ul.tabs li {  
  list-style:none;  
  display:inline;  
}
```

Ce code dit que les éléments `<ul>` avec la classe « tabs » n'ont pas de marge ni d'espacement. Pour ce qui est des éléments `<li>`, ceux-ci n'ont pas de puces. Finalement, nous avons changé la valeur du `display` de « block » par défaut, à « inline ». Voici les explications :

- 1 Les éléments d'une page ont généralement 3 valeurs possibles comme `display` : block, inline et none.
- 2 La valeur « none » rend l'élément invisible.
- 3 La valeur « inline » permet aux éléments de suivre le flux. C'est donc dire que les éléments seront à la suite les uns des autres au lieu d'un en dessous de l'autre.
- 4 La valeur « block » permet de créer un rectangle imaginaire, où chaque élément est en dessous de l'autre.

Etape 10

Bien sûr, à l'heure actuelle, nos onglets sont affreux. Nous avons stylisé les éléments `<li>` passons maintenant aux liens `<a>` :

```
ul.tabs li a {  
  background-color:#464c54;  
  color:#ffebb5;  
  padding:8px 14px 8px 14px;  
  text-decoration:none;  
  font-size:9px;  
  font-family:Verdana, Arial, Helvetica, sans-serif;  
  font-weight:bold;  
  text-transform:uppercase;  
  border:1px solid #464c54;  
}  
ul.tabs li a:hover {  
  background-color:#2f343a;  
  border-color:#2f343a;  
}  
ul.tabs li a.active {  
  background-color:#ffffff;  
  color:#282e32;  
  border:1px solid #464c54;
```

```
border-bottom: 1px solid #ffffff;
}
```

Ce que nous avons fait ici :

- 1 Tout d'abord, nous avons stylisé l'élément <a>.
- 2 Ensuite nous avons stylisé la pseudo-classe « :hover », c'est à dire que lors du passage de la souris sur l'élément, nous allons modifier la couleur de celui-ci.
- 3 Enfin, nous avons ajouté le style pour l'onglet actif qui permettra de savoir sur quel onglet on se trouve.



Etape 11

Il reste 2 choses à faire afin de rendre la zone de contenu fonctionnelle : ajouter le style des 2 zones de contenu caché et celle de la zone du contenu affichable.

```
.content {
  background-color:#ffffff;
  padding:10px;
  border:1px solid #464c54;
}
#content_2, #content_3 { display:none; }
```

Vous verrez que la première partie du CSS indique au navigateur que tous les éléments avec la classe « content » devraient être de couleur blanche et des bordures de la même couleur que l'onglet. La deuxième partie explique que les éléments dont l'identifiant est « content_2 » et « content_3 » devraient être cachés grâce à la valeur « none » associée à la propriété « display ». Plus tard, cela nous permettra d'utiliser un peu de JavaScript afin de modifier l'onglet qui est affiché lors du clic d'un onglet.

Etape 12

Nous allons maintenant arranger le problème de l'espacement entre les onglets et le contenu. Pour cela, modifions le code css associé à l'élément « ul.tabs » par celui-ci :

```
ul.tabs {
  margin:0px; padding:0px;
  margin-top:5px;
  margin-bottom:6px;
}
```



Etape 13

Nous allons maintenant ajouter un peu de contenu dans la zone appropriée. J'ai attendu avant de faire cette étape afin que le code soit plus clair.

```

<!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" xml:lang="en" lang="en">
<head>
  <title>Boîte à onglet - Web 2.0</title>
  <link rel="stylesheet" href="/style.css" type="text/css" media="screen" />
</head>

<body>
<div id="tabbed_box" class="tabbed_box">
  <h4>Navigation <small>Sélectionner un onglet</small></h4>
  <div class="tabbed_area">
    <ul class="tabs">
      <li><a href="/" id="tab_1" class="active">Astuces</a></li>
      <li><a href="/" id="tab_2">Tutoriaux</a></li>
      <li><a href="/" id="tab_3">Articles</a></li>
    </ul>

    <div id="content_1" class="content">
      <ul>
        <li><a href="/">HTML / XHTML <small>4 Articles</small></a></li>
        <li><a href="/">CSS <small>32 Articles</small></a></li>
        <li><a href="/">Flash <small>2 Articles</small></a></li>
        <li><a href="/">PHP / MySQL <small>19 Articles</small></a></li>
        <li><a href="/">Microsoft Windows <small>6 Articles</small></a></li>
        <li><a href="/">Autres <small>8 Articles</small></a></li>
      </ul>
    </div>
    <div id="content_2" class="content">
      <ul>
        <li><a href="/">HTML / XHTML <small>4 Articles</small></a></li>
        <li><a href="/">Javascript <small>32 Articles</small></a></li>
        <li><a href="/">Autres <small>19 Articles</small></a></li>
      </ul>
    </div>
    <div id="content_3" class="content">
      <ul>
        <li><a href="/">Conception Web <small>4 Articles</small></a></li>
        <li><a href="/">SEO et Référencement <small>32 Articles</small></a></li>
        <li><a href="/">Programmation <small>2 Articles</small></a></li>
        <li><a href="/">Autres <small>19 Articles</small></a></li>
      </ul>
    </div>
  </div>
</div>
</body>
</html>

```

Nous avons donc ici ajouté une simple liste `<ul>` dans chacun de nos onglets. Mais vous pouvez ajouter ce que vous désirez. Il ne reste plus qu'à styliser le contenu :

```
.content ul {
  margin:0px;
  padding:0px 20px 0px 20px;
}
.content ul li {
  list-style:none;
  border-bottom:1px solid #d6dde0;
  padding-top:15px;
  padding-bottom:15px;
  font-size:13px;
}
.content ul li a {
  text-decoration:none;
  color:#3e4346;
}
.content ul li a small {
  color:#8b959c;
  font-size:9px;
  text-transform:uppercase;
  font-family:Verdana, Arial, Helvetica, sans-serif;
  position:relative;
  left:4px;
  top:0px;
}
```

Quelques précisions :

- 1 Encore une fois, nous avons supprimé les puces des éléments `<li>` à l'aide de « `list-style:none` »
- 2 Encore une fois, nous avons stylisé l'élément `<small>` qui ici contient le nombre d'articles de chaque catégorie. Ceci permet de le discocier du style de son parent `<a>`.

Etape 14

Le tout semble parfait, exceptée une bordure de trop. La bordure sous le dernier élément peut être supprimée grâce au pseudo-selecteur « `:last-child` » :

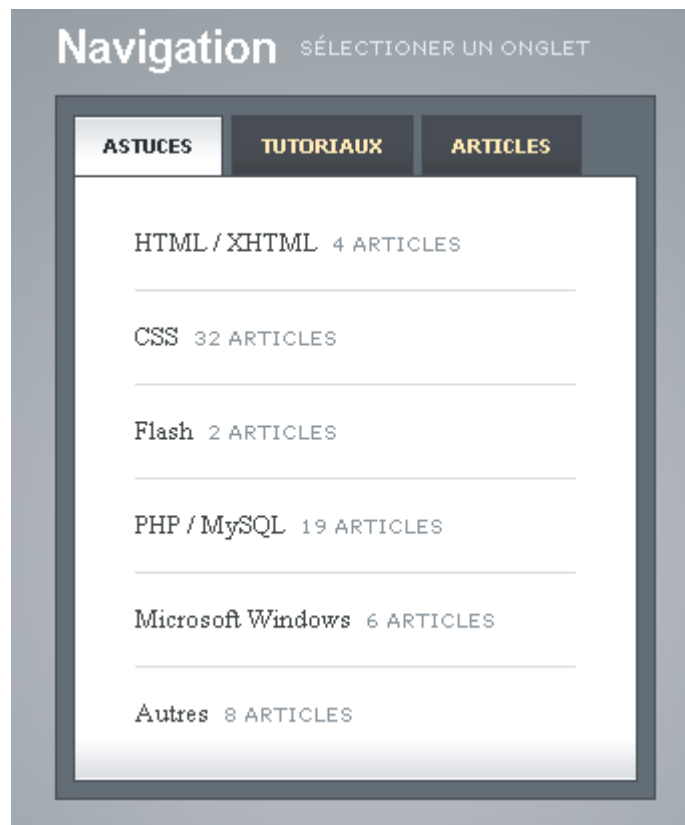
```
.content ul li:last-child {
  border-bottom:none;
}
```

Etape 15

Passons maintenant au style des onglets. Pour styliser les onglets, nous allons rappeler le dégradé de l'image de fond. De plus, il nous faudra styliser les onglets selon 3 styles : actif, inactif et le contenu actif. Nous avons donc extrait de notre fichier PSD 3 "slice" de 1px de large qui seront répétés sur le fond de notre bouton.

```
ul.tabs li a {
  background-image:url(images/tab_off.jpg);
  background-repeat:repeat-x;
  background-position:bottom;
}
ul.tabs li a.active {
  background-image:url(images/tab_on.jpg);
  background-repeat:repeat-x;
  background-position:top;
}
.content {
  background-image:url(images/content_bottom.jpg);
  background-repeat:repeat-x;
  background-position:bottom;
```

}



Et voilà, votre boîte à onglets est maintenant superbe. Il ne manque plus qu'un peu de JavaScript afin de fonctionner. Ce que nous verrons dans la prochaine partie.