

Visualisation professionnelle de vos données avec Flot et jQuery

par Dave Lizotte ([PcKULT.NET](#))

Date de publication : 17/11/2008

Il existe une multitude de guides afin d'afficher sous forme graphique des données à l'aide de CSS. Par contre, il arrive que de simples histogrammes ne soient pas suffisants dans des cas où vos données changent en fonction du temps par exemple. Un histogramme ne peut représenter ce genre de données. C'est ici que le plugin **Flot** pour jQuery entre en jeu. Flot est un plugin permettant l'affichage de graphiques à l'allure très professionnelle.

Dans un monde toujours plus informatisé, il arrive de plus en plus d'avoir besoin d'afficher une grande quantité de données sur le Web. Généralement, un simple tableau d'en-têtes et de valeurs fera l'affaire si la quantité de données est petite. Par contre, si l'on se fie à l'être humain, ce dernier aura plus de facilité à assimiler des données si elles sont affichées graphiquement. Nous allons donc voir ici comment utiliser *Flot* afin d'ajouter une touche professionnelle à l'affichage de vos données.

Présentation des données.....	3
Présentation de la librairie.....	4
Utilisation de la librairie.....	4
Tweak.....	7
Remerciements.....	8

Présentation des données

Avant toutes choses, voici le code HTML contenant les données à représenter sous forme de graphique.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Flot Tutorial</title>

</head>

<body>

<div id="plotarea">
  <table>
    <caption>Valeur en millions</caption>
    <tr>
      <td></td>
      <th scope="col">2003</th>
      <th scope="col">2002</th>
      <th scope="col">2001</th>
      <th scope="col">2000</th>
      <th scope="col">1999</th>
      <th scope="col">1998</th>
    </tr>
    <tr>
      <th scope="row">US</th>
      <td>10,882</td>
      <td>10,383</td>
      <td>10,020</td>
      <td>9,762</td>
      <td>9,213</td>
      <td>8,720</td>
    </tr>
    <tr>
      <th scope="row">EU</th>
      <td>10,970</td>
      <td>9,040</td>
      <td>8,303</td>
      <td>8,234</td>
      <td>8,901</td>
      <td>8,889</td>
    </tr>
    <tr>
      <th scope="row">UK</th>
      <td>1,765</td>
      <td>1,564</td>
      <td>1,430</td>
      <td>1,438</td>
      <td>1,460</td>
      <td>1,423</td>
    </tr>
    <tr>
      <th scope="row">Chine</th>
      <td>1,575</td>
      <td>1,434</td>
      <td>1,345</td>
      <td>1,252</td>
      <td>1,158</td>
      <td>1,148</td>
    </tr>
    <tr>
      <th scope="row">Inde</th>
      <td>599</td>
      <td>510</td>
      <td>479</td>
      <td>457</td>
    </tr>
  </table>

```

```

        <td>447</td>
        <td>414</td>
    </tr>
</table>
</div>

<p>Valeur en millions</p>

</body>
</html>
    
```

Présentation de la librairie

Avant toute chose, il est important de référencer les 2 librairies requises (**jQuery**, **Flot**) de plus qu'une librairie pour le support d'Internet Explorer (**ExCanvas**).

Attention à l'ordre, car il vous faut la respecter. Soit, la librairie jQuery en premier, suivie de Flot et de ExCanvas. Flot utilise les éléments canvas afin de générer graphiquement les données. Ces derniers ne sont pas disponible sous Internet Explorer. C'est pourquoi nous allons utiliser les commentaires conditionnels afin de ne pas charger la librairie inutilement si vous êtes sous Firefox ou Safari par exemple.

```

<script src="jquery.js" type="text/javascript"></script>
<script src="jquery.flot.pack.js" type="text/javascript"></script>
<!--[if IE]>
<script type="text/javascript" src="excanvas.pack.js"></script>
<![endif]>
    
```

Utilisation de la librairie

Créer un graphique avec *Flot* est très simple car la majorité des options ont des paramètres par défaut. C'est-à-dire que vous pourrez rapidement créer des graphiques avec un minimum de travail. Mais d'un autre côté, vous pourrez améliorer la base si vous en avez envie. Afin de générer un graphique, il vous faut tout d'abord un élément conteneur ainsi que les données à afficher. Le conteneur doit avoir une hauteur et une largeur spécifique. Nous allons donc utiliser jQuery afin d'attribuer les valeurs à notre conteneur `<div>`.

```

<script language="javascript" type="text/javascript">
$( function()
{
    var plotarea = $("#plotarea");
    plotarea.css("height", "250px");
    plotarea.css("width", "500px");
    $.plot( plotarea , data );
}
);
</script>
    
```

Le premier paramètre est un objet jQuery du conteneur. Le second est un tableau tridimensionnel où le premier enfant représente les données pour un pays par exemple et le second représente un point cartésien (X,Y) (Années,Valeur) où la donnée sera affichée sur le graphique. Commençons donc par les données des USA par exemple.

```

<script language="javascript" type="text/javascript">
$( function ()
{
    var data = [ [[2003, 10882],
                  [2002, 10383],
                  [2001, 10020],
                  [2000, 9762],
    
```

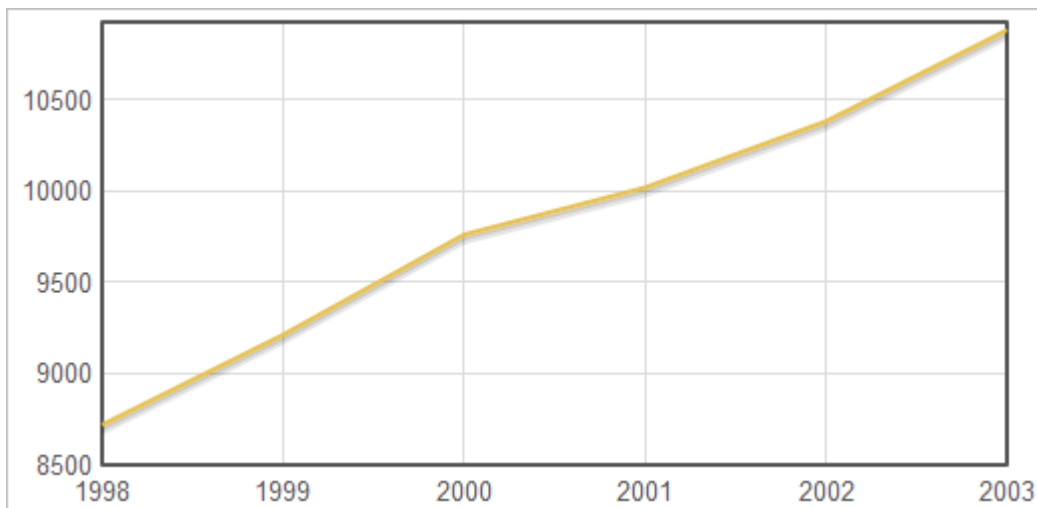
```

        [1999, 9213],
        [1998, 8720]] ];

var plotarea = $("#plotarea");
plotarea.css("height", "250px");
plotarea.css("width", "500px");
$.plot( plotarea , data );
    }
);
</script>

```

Le tableau que nous avons au départ est maintenant remplacé par un joli graphique.



Comme vous pouvez le constater, le tout est très simple. Ajoutons donc les autres pays afin de compléter notre graphique.

```

var data = [
  [[2003, 10882],
   [2002, 10383],
   [2001, 10020],
   [2000, 9762],
   [1999, 9213],
   [1998, 8720]],

  [[2003, 10970],
   [2002, 9040],
   [2001, 8303],
   [2000, 8234],
   [1999, 8901],
   [1998, 8889]],

  [[2003, 1795],
   [2002, 1564],
   [2001, 1430],
   [2000, 1438],
   [1999, 1460],
   [1998, 1423]],

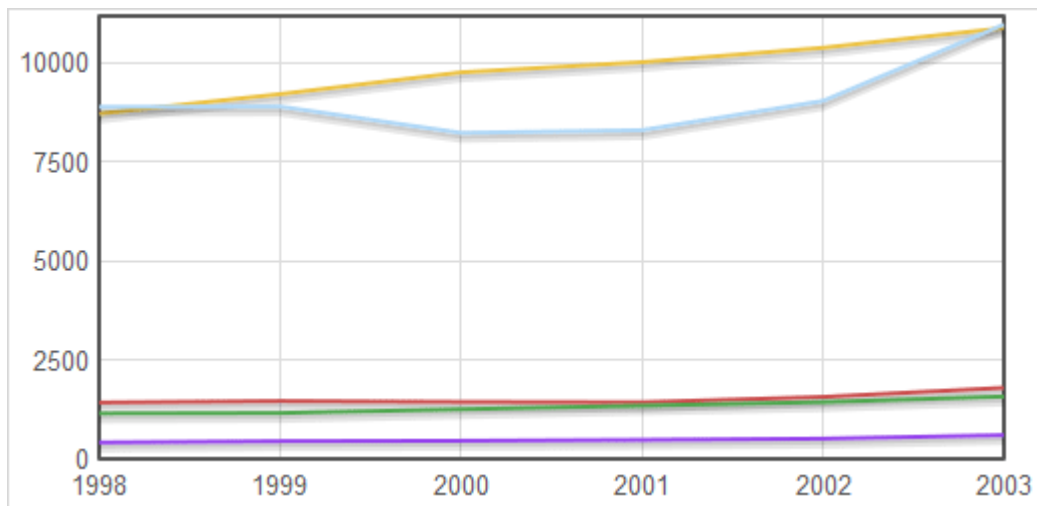
  [[2003, 1575],
   [2002, 1434],
   [2001, 1345],
   [2000, 1252],
   [1999, 1158],
   [1998, 1148]],

  [[2003, 599],
   [2002, 510],
   [2001, 479],

```

```
[2000, 457],
[1999, 447],
[1998, 414]
];
```

Et voici le rendu correspondant à notre exemple :



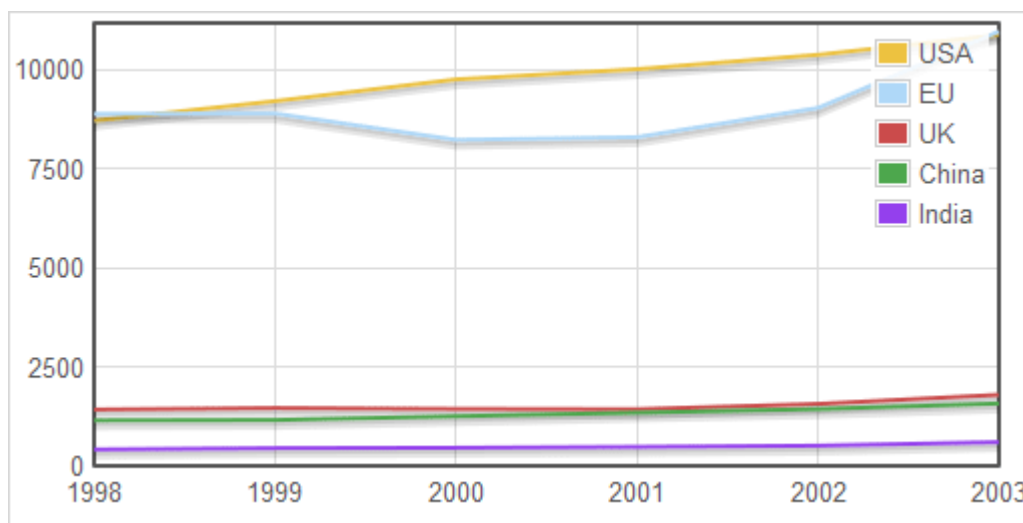
Nous avons maintenant un superbe graphique mais nous ne savons pas quelle ligne est associée à quel pays. Nous allons donc ajouter une légende. Rien de plus simple que d'ajouter un élément label dans les données en faisant comme ceci :

```
var data = [
  {
    label: "USA",
    data: [[2003, 10882],
           [2002, 10383],
           [2001, 10020],
           [2000, 9762],
           [1999, 9213],
           [1998, 8720]]
  },
  {
    label: "EU",
    data: [[2003, 10970],
           [2002, 9040],
           [2001, 8303],
           [2000, 8234],
           [1999, 8901],
           [1998, 8889]]
  },
  {
    label: "UK",
    data: [[2003, 1795],
           [2002, 1564],
           [2001, 1430],
           [2000, 1438],
           [1999, 1460],
           [1998, 1423]]
  },
  {
    label: "Chine",
    data: [[2003, 1575],
           [2002, 1434],
           [2001, 1345],
           [2000, 1252],
```

```

        [1999, 1158],
        [1998, 1148]]
    },
    {
        label: "Inde",
        data: [[2003, 599],
            [2002, 510],
            [2001, 479],
            [2000, 457],
            [1999, 447],
            [1998, 414]]
    }
];

```



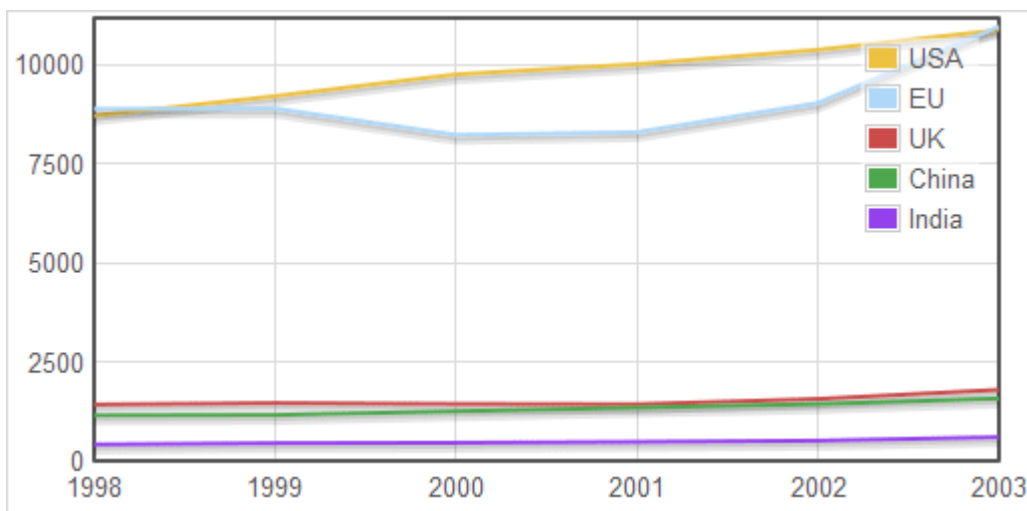
Tweak

J'ai mentionné plus tôt que *Flot* avait beaucoup de valeurs par défaut. Voyons donc comment améliorer notre graphique avec les autres options. Tout d'abord, afin d'améliorer la lisibilité de votre graphique, nous allons afficher la légende et nous lui ajouterons un peu de transparence afin de ne pas perdre les données affichées derrière. Pour ce faire, la fonction de création du graphique peut accueillir un 3ème paramètre `$.plot(plotarea , data, option);`

```

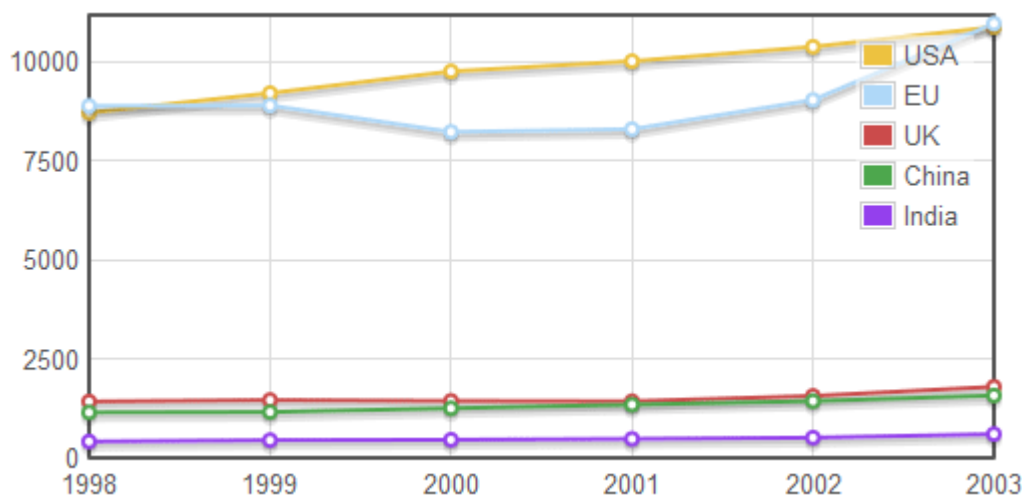
var options = {
    legend: {
        show: true,
        margin: 10,
        backgroundOpacity: 0.5
    }
};

```



Une autre astuce est de faire afficher un point sur la ligne pour chaque valeur. Cela rend selon moi le graphique beaucoup plus visible. Voyons comment faire cela facilement :

```
var options = {
  legend: {
    show: true,
    margin: 10,
    backgroundOpacity: 0.5
  },
  points: {
    show: true,
    radius: 3
  },
  lines: {
    show: true
  }
};
```



Les possibilités de *Flot* sont énormes, nous n'avons ici qu'effleuré les possibilités de ce plugin. Mais déjà, on peut voir que cette librairie est facilement accessible à tous.

Remerciements

Tous mes remerciements à **Bovino** et à **romaintaz** pour sa relecture.