

Lire un fichier XML avec jQuery

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

Date de publication : 7 janvier 2009

Dernière mise à jour : 21 novembre 2009

Encore une fois nous allons parler de jQuery et nous allons nous attarder à la lecture d'un fichier ou d'une source XML à partir de jQuery.

| | |
|--------------------------------|---|
| Introduction..... | 3 |
| Étape préliminaire..... | 4 |
| La lecture du fichier XML..... | 5 |
| Code final..... | 7 |
| Remerciements..... | 8 |

Introduction

Eh oui, encore une fois nous allons parler de jQuery, sans aucun doute une des meilleures librairies JavaScript existantes. Bref, nous allons aujourd'hui nous attarder à la lecture d'un fichier ou d'une source XML à partir de jQuery. Alors qu'AJAX prend de plus en plus de place sur la toile voici comment vous aussi vous pouvez ajouter un petit plus à vos développements.

Étape préliminaire

Avant toute chose il faut construire notre page HTML et y inclure notre librairie afin que le tout fonctionne. Voici le code du corps de notre page que nous utiliserons.

```
<body>
  <div id="Div_XML"></div>
</body>
```

La lecture du fichier XML

```
<sites>
  <site id="0">
    <url>http://www.pckult.net</url>
    <desc>
      <title>PC Kult</title>
      <brief>Résumé</brief>
      <long>Description longue</long>
    </desc>
  </site>
  <site id="2">
    <title>Microsoft</title>
    <url>http://www.microsoft.ca</url>
    <desc>
      <brief>Résumé</brief>
      <long>Description longue</long>
    </desc>
  </site>
  <site id="3">
    <title>Intel</title>
    <url>http://www.intel.com</url>
    <desc>
      <brief>Résumé</brief>
      <long>Description longue</long>
    </desc>
  </site>
</sites>
```

Comme toute programmation AJAX, on commence toujours par l'inclusion des tags suivants :

```
$(document).ready( function()
    {
    });
```

À l'intérieur même de ces tags, nous allons créer notre requête AJAX. La requête AJAX demande 4 paramètres distincts :

- Un type
- Une source (url)
- Un type de source
- L'opération à effectuer si la requête est valide

La partie la plus importante est bien sur la partie opération. C'est cette dernière qui va permettre de manipuler le fichier XML.

```
$.ajax( {
  type: "GET",
  url: "Fichier.xml",
  dataType: "xml",
  success: function(xml) { }
});
```

Une fois le fichier chargé, il nous faut maintenant lire son contenu. Nous allons donc utiliser la méthode **find()** afin de récupérer tous les éléments (dans le cas présent, les éléments « **site** ») de notre arbre XML.

```
$(xml).find('site').each( function() { } );
```

Une fois que tous les éléments sont récupérés, il nous faut alors récupérer les données de chaque élément grâce à la fonction **attr()** et la fonction **find()**.

```
var id = $(this).attr('id');  
var title = $(this).find('title').text();  
var url = $(this).find('url').text();  
$('<div class="items" id="link_' + id + '></div>').html('<a href="' + url + '"' + title + '</a>').appendTo('#Div_XML');
```

Code final

```
$(document).ready(
  function()
  {
    $.ajax( {
      type: "GET",
      url: "sites.xml",
      dataType: "xml",
      success: function(xml)
      {
        $(xml).find('site').each(
          function()
          {
            var id = $(this).attr('id');
            var title = $(this).find('title').text();
            var url = $(this).find('url').text();
            $('<div class="items" id="link_' + id + '"></div>').html('<a href="' +
            url + '">' + title + '</a>').appendTo('#Div_XML');
            $(this).find('desc').each(
              function()
              {
                var brief = $(this).find('brief').text();
                var long = $(this).find('long').text();
                $('<div class="brief"></div>').html(brief).appendTo('#link_'+id);
                $('<div class="long"></div>').html(long).appendTo('#link_'+id);
              }
            );
          }
        );
      }
    );
  }
);
```

Remerciements

Tous mes remerciements à **RomainVALERI** pour sa relecture.