

JavaScript & AJAX

JavaScript : un langage incompris

JavaScript a été conçu par **Brendan Eich** en **1995** pour le navigateur Netscape Navigator 2. Tout d'abord nommé *LiveScript*, il fut renommé *JavaScript* en décembre 1995 pour des raisons commerciales : les concepteurs pensaient ainsi obtenir la popularité du langage Java alors très en vogue à l'époque. Malheureusement mis à part le nom, ces 2 langages n'ont absolument rien en commun. Ce changement de nom entraîna d'ailleurs une forte confusion et est malheureusement toujours d'actualité.

Internet Explorer de Microsoft a lui aussi son langage de Script : JScript proposé à partir de la version 3 du navigateur. En 1997 Netscape propose son langage à l'ECMA afin de le faire normaliser, il en résulte un nouveau langage ayant pour nom *ECMAScript*. Ces 3 langages ont heureusement très peu de différences, seul le nom change, on peut très bien faire du JavaScript initialement prévu pour Netscape et donc Firefox qui s'exécutera parfaitement dans Internet Explorer.

Comme son nom l'indique, JavaScript est un langage de Script, c'est-à-dire qu'il est prévu pour s'exécuter dans un environnement hôte. La plupart du temps on l'exécute dans un navigateur mais on le retrouve aussi dans Adobe Reader, la technologie XUL, Flash avec ActionScript ou encore les gadgets de Windows Vista.

JavaScript est un langage orienté prototype : une forme de langage orienté objet mais sans classe. Tout se fait grâce à un prototype. Un prototype est un tableau qui contient des méthodes, variables, événements,... on peut ajouter ou supprimer des méthodes d'un prototype mais aussi, les instancier, combiner deux prototypes pour n'en former qu'un seul, voir même de modifier le prototype une fois l'objet instancié. Toutes ces possibilités font que JavaScript est un langage extrêmement puissant mais compliqué à prendre en main rapidement.

Utilité du JavaScript

JavaScript est principalement utilisé par les navigateurs pour interagir dynamiquement avec l'utilisateur. Avec la prise en charge du DOM (Document Object Model), JavaScript prend une nouvelle dimension, puisqu'il devient possible de manipuler les balises du document HTML, de s'abonner à des événements, etc.... DOM est standardisé par le W3C et est implémenté par tous les navigateurs récents. Malgré cela JavaScript et la manipulation du DOM reste impopulaire, notamment à cause des problèmes de compatibilité qu'ont connu les développeurs avant la standardisation du DOM.

Depuis quelques mois on parle de Web 2.0, comme beaucoup de développeurs je n'aime pas ce terme purement marketing, le « web 2.0 » désigne seulement l'évolution naturelle du Web, il ne s'agit en rien d'une nouvelle version. Cette évolution met en avant une technologie oubliée AJAX ainsi que des concepts avancés de JavaScript

AJAX

AJAX (Asynchronous JavaScript And XML) est la technologie qui rend à JavaScript ses titres de noblesse. Cette technologie a été conçue par Microsoft pour Internet Explorer 4 en 1998 et est en cours de validation par le W3C. Elle permet de faire des requêtes HTTP via JavaScript c'est-à-dire qu'on peut envoyer des informations vers le serveur, les récupérer et traiter la réponse via JavaScript sans aucun rechargement de la page.

Malgré tout ce qu'on peut entendre, Ajax n'est pas une révolution technologique ! Ce n'est qu'un concept qui permet de faire des requêtes http via JavaScript. JavaScript souffrant d'une mauvaise image, les commerciaux ont préféré utiliser le

terme AJAX pour désigner toutes ces redécouvertes. Le « Web 2.0 » n'est rien d'autre que l'utilisation de JavaScript combiné avec la technologie Ajax.

Et le futur ?

De base, JavaScript est très pauvre, pour faire la moindre chose il faut écrire de nombreuses lignes de code. C'est pour cela qu'est apparu différentes bibliothèques JavaScript permettant de réaliser de nombreuses actions complexes en un minimum de code. Parmi les plus populaires on retrouve entre autres les bibliothèques Script.aculo.us, Dojo, Yahoo ! et Atlas de Microsoft qui est pour l'instant en version bêta mais déjà très prometteur. En effet, Atlas est le compagnon indispensable de ASP.net 2. Cette bibliothèque offre un nombre impressionnantes de fonctionnalités, le cœur possède plus de 12 000 lignes de JavaScript, elle introduit une nouvelle façon de faire du JavaScript grâce au xml-script : un langage XML permettant de déclarer des objets et de s'abonner à leurs événements de façon déclarative.

Code

Voici un Hello World avec une notion d'objet fait en JavaScript.

```
var monObjet = function(){
    this.helloWorld = function(){
        alert('Retrouver de nombreux codes JavaScript sur www.JavaScriptFR.com');
    }
}

var instance = new monObjet();
instance.helloWorld();
```

Voici un Autre Hello world pour une page Atlas fait en XML-Script

```
<button id="btn1" runat="server">Click me</button><br />
<span id="result"></span>
...
<script type="text/xml-script">
    <page xmlns:script="http://schemas.microsoft.com/xml-script/2005">
        <components>
            <button id="btn1">
                <click>
                    <setProperty target="result" property="text"
                        value="Retrouver de nombreux codes Atlas sur www.aspfr.com" />
                </click>
            </button>
            <label id="result"></label>
        </components>
    </page>
</script>
```

Fiche signalétique: JavaScript

| | |
|------------------------------------|--|
| Les + | Très puissant une fois pris en main, disponible sur tous les navigateurs récents. |
| Les - | Très Impopulaire à cause de son historique, difficile à prendre en main, le concept objet est très différent des autres langages objets. |
| Pourquoi choisir JavaScript | Pour réaliser une interface utilisateur agréable dans une page web |
| IDE principal | Notepad2, Visual Web Dev Express |
| Portabilité | Total : La plupart des navigateurs supporte JavaScript |
| Avenir | La démocratisation du « web 2.0 » ainsi que son exclusivité dans les navigateurs assurent un très joli avenir pour ce langage. |

Pour en savoir plus sur les langages orienté prototype :

[http://fr.wikipedia.org/wiki/Programmation orientée prototype](http://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_prototype)

Cyril DURAND

Membre de l'équipe de développement CodeS-SourceS

<http://blogs.developpeur.org/cyril/>

