

## Le rôle du JavaScript - 1/2

**Le JavaScript est le langage côté client le plus utilisé dans les pages web... Il permet de manipuler (modifier, créer, effacer, déclencher...) les éléments définis dans une page web...**

Le JavaScript est un langage de script utilisé côté client. C'est à dire que ses lignes de code seront interprétées sur la machine de l'utilisateur final. C'est le *navigateur internet* ou *browser* (Internet Explorer, Netscape, Mozilla, Opera...) qui a téléchargé la page web qui va interpréter le Script. Le rôle du JavaScript est donc de manipuler les éléments de la page. Les exemples les plus courants sont par exemple de changer une image quand elle est survolée (Rollover Images) ou de valider le format des données saisies dans un formulaire *avant même* de l'avoir envoyé. En effet, inutile de faire une requête http supplémentaire si l'utilisateur rentre un email sans arobase (@), on peut lui demander de le resaisir tout de suite.

Le gros intérêt du point de vue de l'utilisateur final, c'est qu'un langage côté client permet de réagir instantanément sur l'affichage. Contrairement à un site uniquement basé sur des technos côté serveur (+HTML seul) pour lequel la moindre modification nécessitera une requête http et donc un aller-retour serveur impliquant un délai d'attente (de l'ordre de la seconde) et un rafraichissement de l'écran pas toujours très esthétique.

Mais le Javascript est capable de bien plus :

Calculs, utilisation d'objets, intervention dans les autres frames du même site, création d'éléments à la volée (tableaux, formulaires, images... N'importe quel composant HTML en fait), tri alphabétique d'un tableau selon telle ou telle colonne, animation d'images (les faire se déplacer, pas en changer le contenu)...

Utilisé conjointement avec un langage côté serveur (PHP, asp,...), vous serez capables d'optimiser l'ergonomie et les fonctionnalités de vos sites.

le rôle des langages côté serveur peut être, en particulier, ce mettre à jour des variables javascript en fonction d'une base de données.

### Confusions fréquentes entre les langages :

Le Javascript initialement développé par *Netscape* (il s'appellait LiveScript au tout début), ne doit surtout pas être confondu avec le Java développé par *Sun* et qui n'a pas le même rôle sur Internet.

La version *Microsoft* du JavaScript s'appelle Jscript et est compatible JavaScript là, on parle bien du même rôle mais avec des fonctionnalités supplémentaires qui ne fonctionneront pas sur les browsers concurrents.

### Compatibilité des scripts :

C'est là que ça se gâte pour le JavaScript... Contrairement aux langages côté serveur qui vont être exécutés/interprétés dans un environnement bien connu (votre serveur, ou celui de votre hébergeur), Dans le cas du JavaScript, c'est chez l'utilisateur final que c'est interprété. Et les différents utilisateurs peuvent très bien avoir des configurations très variées (PC/Mac/Unix/..., MSIE/Netscape/...) le problème, c'est que tous les moteurs de javascript de ces différentes configurations... Sont légèrement incompatibles... le gros est commun mais certaines fonctionnalités de base sont différentes. il faut donc d'une manière ou d'une autre, connaître les spécificités de chaque navigateur que l'on souhaite couvrir... et Tester son application (ses pages, son site...) sur plusieurs configurations. Il est très courant de devoir écrire plusieurs fois un algorithme javascript adapté à tel ou tel browser : *si (netscape) { faire comme ceci; } sinon { faire comme cela; }*

### Limitations de sécurité :

## Le rôle du JavaScript - 2/2

Qui dit langage côté client... dit exécuté chez l'utilisateur final... Et là, si les navigateurs n'étaient pas sécurisés comme on va le voir, ça pourrait présenter de gros risques (piratage, espionnage,...) pour l'utilisateur :

Il est impossible de manipuler des frames d'autres domaines...

Dans une URL, le *domaine*, c'est la partie entre http:// et le slash (/) suivant :

<http://www.google.fr/search?q=javascript>

Au bout du compte, pour résumer la situation, la portée d'un script javascript est limitée aux pages provenant du même site, le domaine étant un identifiant réseau du serveur hébergeant le site.

ça empêche par exemple un site d'espionner trop facilement les habitudes de navigation d'un utilisateur.

Il est impossible sans autorisation expresse de l'utilisateur, de manipuler des données système...

Il est en particulier impossible de lire ou d'écrire des fichiers sur l'ordinateur de l'utilisateur, ou de faire des appels à des ActiveX non signés.

Notons bien que ce ne sont pas des limitations du JavaScript mais des sécurités ajoutées par les navigateurs.

### Adresses utiles :

Pour terminer, si vous voulez souhaiter vous y mettre, je vous conseillerais ces 2 liens :

une référence claire et précise avec exemples sur [devguru.com](http://devguru.com)

et un didacticiel très bien fait sur [w3schools.org](http://w3schools.org)

Bonne chance et... que le web soit avec vous ;o)

NB : Ce qui est expliqué ici est disons... le cas général... Il est également possible d'utiliser le JavaScript côté serveur mais c'est bien plus rare. En fait, le JavaScript est un langage de script générique qui peut être utilisé à bien de fins...