

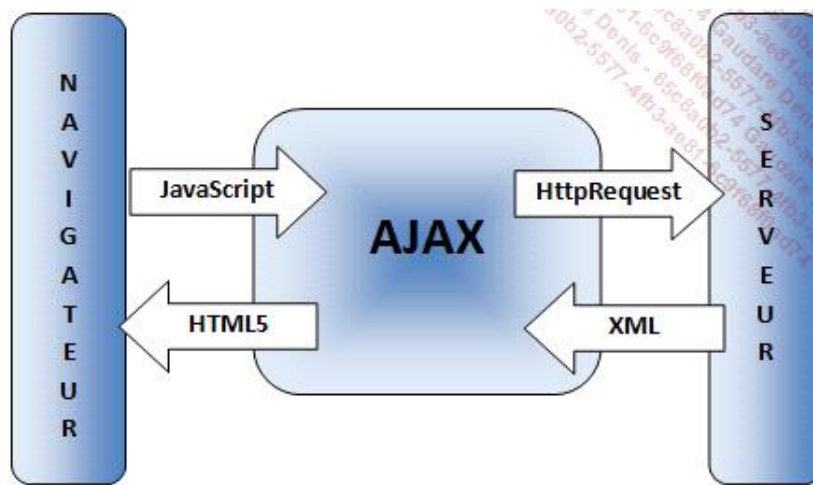
Présentation

L'objet XMLHttpRequest permet de faire des requêtes HTTP au serveur, de recevoir des réponses et de mettre à jour une partie de la page Web. En mode asynchrone, cette mise à jour se réalise sans devoir recharger la page et donc de façon totalement transparente pour l'utilisateur.

L'objet XMLHttpRequest s'utilise donc dans une architecture de type client-serveur.

Détaillons son mode de fonctionnement :

- L'objet XMLHttpRequest est créé par le moteur JavaScript du navigateur.
- Cet objet est alors utilisé pour envoyer une requête HTTP vers le serveur.
- La réponse est fournie par celui-ci au navigateur.
- À l'aide du HTML et des feuilles de style CSS, le résultat est ensuite affiché dans le navigateur.



L'avantage des applications AJAX réside principalement dans la diminution de la bande passante, car seules les données réclamées sont affichées, sans devoir recharger tout le document. D'où une interactivité et une fluidité accrue.

L'objet XMLHttpRequest a initialement été développé en tant que contrôle ActiveX de Microsoft XML Core Services (MSXML), implémenté dans Internet Explorer 5.0, soit en 1999. Il y eu, au fil des années, différentes versions de ce composant. Citons :

- Microsoft.XMLHTTP
- MSXML2.XMLHTTP
- MSXML2.XMLHTTP.3.0
- MSXML2.XMLHTTP.4.0
- MSXML2.XMLHTTP.5.0
- MSXML2.XMLHTTP.6.0

En 2002, l'objet XMLHttpRequest a été repris par le projet Mozilla comme objet natif JavaScript. Il est ainsi reconnu dès les premières versions de Firefox. Les autres navigateurs du marché ont suivi avec Safari, Google Chrome, Opera et, à la surprise générale, Internet Explorer 7. Ainsi la compatibilité de l'objet XMLHttpRequest peut être jugée globalement excellente pour les navigateurs récents.

Cette version, appelée a posteriori XMLHttpRequest 1, présentait quelques inconvénients :


- Les requêtes ne peuvent être envoyées qu'au même serveur que le fichier d'appel.
- La transmission se limite à des données textuelles.
- Peu d'informations sont disponibles durant l'envoi et la réception de données.

Une nouvelle version, XMLHttpRequest 2, a vu le jour. Très rapidement pour Firefox (version 4) et Google Chrome (version 7). Pour Internet Explorer, il a fallu attendre la version 10. Sans surprise, elle pallie les inconvénients présentés et offre de nouvelles possibilités :

- Ajout d'un délai (*timeout*) pour la réponse du serveur.
- Il est maintenant possible de télécharger des données binaires telles des fichiers audio et vidéo.
- Les requêtes peuvent être transmises vers un serveur différent du fichier d'appel (*cross-site*).

Le W3C a publié un premier brouillon (*working draft*) pour la normalisation de l'objet XMLHttpRequest, en date du 5 avril 2006. Le XMLHttpRequest 2 a, quant à lui, été défini pour la première fois le 25 février 2008. Le 5 décembre 2011, les deux versions ont été fusionnées en une seule spécification. Cette dernière n'est toujours pas finalisée à l'heure actuelle (septembre 2014). Heureusement, les éditeurs de navigateurs ont rapidement intégré l'objet XMLHttpRequest, contournant ainsi les lenteurs de ce groupe de travail.

Le passage de la version 1 à la version 2 est transparent pour l'utilisateur.

 Avec ce chapitre, nous abordons des applications client-serveur. Votre seul navigateur ne suffit plus pour mener à bon terme notre étude. Il sera nécessaire de passer par un serveur local. Comme signalé au chapitre L'environnement de travail, l'installation et l'utilisation d'un serveur http local, comme IIS de Windows ou EasyPHP, se révèle indispensable pour traiter les exemples de cet ouvrage. Concrètement, à partir de ce chapitre, tous les fichiers des exemples devront être situés sur ce serveur local. Dans le navigateur, les fichiers seront alors accessibles via l'adresse `http://localhost/exemple.htm` si vous utilisez IIS ou `http://127.0.0.1/exemple.htm` si vous utilisez EasyPHP.