

# HTML



# CSS



## Programmation Web

La **programmation web** est la programmation informatique qui permet d'éditer des Sites web. Elle peut prendre différentes formes : de la simple page statique à la page dynamique avec connexion à une base de données.

Pour un internaute qui fait appel à une URL, la page web correspondante, qu'elle soit statique ou dynamique, consiste en du code « côté client » (HTML, CSS, Javascript, ...) qui est interprété par son navigateur (Google Chrome, Firefox, ...) produisant une interface utilisateur. Ce code, qui est renvoyé par un serveur web, peut ou non avoir été généré par du calcul « côté serveur », via des langages de programmation particuliers, pouvant faire appel à des bases de données, des API...

Il existe différentes technologies et différents langages permettant de construire des sites web par l'intermédiaire de serveurs web, que l'on peut donc classer selon deux principes : des langages de programmation côté client, ou côté serveur. Cette distinction est faite pour séparer les langages « côté serveur » dont le code est exécuté sur le serveur web avant d'arriver sur le navigateur de l'utilisateur, des langages « côté client » dont l'exécution ne nécessite pas de calcul sur le serveur web mais, seulement après le téléchargement de la page, une interprétation par le navigateur de l'utilisateur.

## Table des matières

- [Langages de balisage](#)
  - [Langage HTML ou XHTML](#)
  - [Limites du HTML](#)
- [Programmation client](#)
  - [Principe](#)
  - [Utilisation](#)
  - [Langages de Script](#)
  - [Avantages et inconvénients](#)
- [Programmation serveur](#)
  - [Principe](#)

- [Différents langages et technologies](#)
  - [JavaScript](#)
  - [PHP](#)
  - [Python](#)
  - [Ruby](#)
  - [Java](#)
  - [Technologies Microsoft](#)
- [Technologie Ajax](#)

## Bases de la page web: Les langages de balisage

### Langage HTML ou XHTML

Toute page web comprend une base de langage HTML ou XHTML. Il s'agit d'un langage de balisage qui définit essentiellement la structure de la page web (titres, tableaux, paragraphes, etc.).

C'est un langage qui permet de créer des hyperliens, à savoir des liens d'un document à un autre ou d'un endroit d'un document à un autre endroit du même document (identificateur de fragment).

Au départ ce langage ne définissait que la structure de la page mais très vite se sont rajoutées des balises (ou tags) qui s'occupaient de spécifier les caractéristiques de la page web (hiérarchisation du contenu, attributs donnés aux balises...).

Depuis sa version 4, le langage HTML s'accompagne des CSS qui sont des styles se chargeant de la mise en forme et de l'animation de la page.

Le XHTML est une évolution du HTML basée sur XML.

### Limites du HTML

Une page web créée en HTML est totalement statique et n'offre qu'une faible possibilité d'interaction avec le visiteur. Chaque page doit être créée au préalable et doit être manuellement mise à jour.

Concernant l'aspect dynamique, seules existent deux balises HTML (<marquee> et <blink>), mais sont propriétaires à Microsoft, ne sont aujourd'hui plus supportées par certains navigateurs et ne sont pas reconnues par le [W3C](#).

Quelques fonctionnalités des CSS permettaient de créer un peu de dynamisme dans la page (ex. changement de couleur lors d'un survol). Le CSS 3.0 est en train de changer la donne.

À propos de l'interactivité, le langage HTML permet uniquement de créer le fond d'un formulaire d'interaction avec le visiteur. Il permet de définir les paramètres d'une action, et de renvoyer les données du formulaire vers un 'système' qui gèrera celles-ci, mais ne permet pas de définir l'action elle-même.

### Portabilité

Le langage HTML est multiplateforme. Tous les navigateurs web reconnaissent le langage HTML de base, seuls certaines balises sont propres à certains navigateurs (comme la balise `<marquee>` qui n'est interprétée comme défilement de texte que dans Internet Explorer et Google Chrome. Les CSS, même s'ils sont standardisés par le W3C, ne sont pas toujours reconnus de la même manière par les différents navigateurs web.

## **Mise en place**

Le langage HTML a l'avantage d'être facile à mettre en place, et ne demande pas de compétences techniques très poussées. Ses pages sont d'un traitement très léger et rapide par le serveur.

Il faut donc autre chose pour ajouter de l'animation ou de l'interactivité dans la page web.

## **Programmation web côté client**

Les objets ne fournissent pas toutes les solutions aux besoins d'interactivité et de dynamisme des sites ou sont parfois lourds à mettre en place. Il est possible alors de passer à de la programmation permettant une meilleure interaction avec la page HTML.

## **Principe**

Lorsque le visiteur demande une simple page HTML (avec ou sans objets) en tapant une adresse ou URL, le serveur web lui renvoie simplement le contenu de la page demandée, c'est le navigateur web qui interprète le contenu HTML et les plug-ins qui interprètent les objets, en demandant à leur tour le contenu des objets au serveur web.

Si du code, côté client, est inséré dans la page, le serveur web renverra également la page au client, y compris le code. C'est donc à nouveau le navigateur web qui doit interpréter le code et faire ce que le code demande.

## **Utilisation**

La programmation côté client sera utilisée essentiellement dans le cas de validation de formulaires (champs obligatoires, petits calculs...) et dans la création de petites animations (menus déroulants...).

## **Langages de Script**

La programmation côté client utilise des langages de scripts. À la différence d'autres langages de programmation, ces langages ne sont pas compilés. Ils sont interprétés par le navigateur web du visiteur, et leur syntaxe est moins stricte que celle des langages de développeurs (variables non-typées a priori, le point-virgule à la fin des instructions est optionnel...).

- JavaScript est le plus utilisé des langages de programmation web côté client. Il est reconnu par tous les navigateurs (dans les versions actuelles). Il se base sur le langage normalisé ECMAScript.
- JScript, fort proche du JavaScript est simplement une adaptation de celui-ci par Microsoft.

- VBScript est le langage de script développé par Microsoft sur base du Visual Basic. Ce langage n'est interprété que par les navigateurs basés sur ceux développés par Microsoft (soit Internet Explorer et Maxthon). Il est donc peu utilisé sur l'Internet (mais utilisé parfois sur des intranets).
- DHTML n'est PAS un langage de script mais simplement la combinaison, dans une page web, de JavaScript, de HTML 4 ou de XHTML et de CSS.

## Avantages et inconvénients

### Avantages:

La programmation côté client est légère et relativement simple à implémenter (syntaxe pas trop complexe).

### Inconvénients:

Le code est envoyé tel quel au client, et donc visible par celui-ci (s'il regarde les sources de sa page) et donc peu sécurisé. Le bon déroulement du programme dépend entièrement du navigateur du client, de sa configuration et de son niveau de sécurité.

Il n'est donc pas question ici de lier le site web à une base de données.

## Programmation web côté serveur

### Principe

Lorsqu'un visiteur demande (ou requête) le chargement d'une URL sur son navigateur, un appel (le plus souvent via HTTP ou HTTPS) est effectué sur le serveur web qui héberge le site correspondant. Le serveur web va ensuite identifier le ou les scripts (écrits dans un langage « côté serveur ») correspondant à cette URL, et demander leur exécution. L'exécution va alors résulter en un contenu HTML, qui va ensuite être expédié en réponse au visiteur, lisible par son navigateur.

La programmation "côté serveur" consiste donc à produire du code dont l'exécution, du calcul et de la récupération de données externes (bases de données, API, ...), va résulter en un contenu qui sera envoyé au client. Ce contenu peut être du code "côté client".

### Différents langages et technologies



### JavaScript

Il est possible de mettre en place un serveur web en utilisant le programme *Node.js*, exposant des URL, via notamment *Express.js*, à un projet de code développé en Javascript.

## PHP

PHP est un langage de script interprété côté serveur. Le code PHP est lu par le serveur et interprété pour produire une page HTML (ou d'autres types de fichier comme des images ou des documents PDF) à chaque fois que la page sera demandée. Ce langage a été créé initialement dans le cadre des logiciels Open Source. De nombreux développeurs ont ensuite profité de l'Open Source pour développer de nombreux modules prêts à l'usage et paramétrables à volonté.

Les modules serveurs de PHP sont implémentés sur les serveurs Apache et Unix mais PHP peut être implémenté sur d'autres plates-formes via des CGI.

La syntaxe, relativement simple, part d'une base de C.

Frameworks notables : Symfony, Laravel

## Python

Django : Framework open-source largement répandu pour construire et déployer une application côté serveur en Python.

## Ruby

Ruby on Rails : Framework open-source largement répandu pour construire et déployer une application côté serveur en Ruby.

## Java

- **JSP** : Technologie développée par Sun Microsystems dans le but de créer des pages dynamiques avec le langage Java. Les JSP permettent d'ajouter du code Java dans la page HTML qui sera interprété par le serveur. On peut distinguer trois ensembles d'instructions JSP :
  - Les instructions de programmation permettent d'insérer des bouts de code Java directement dans la page web, et donnent accès à une grande partie de la bibliothèque de programmation de la version standard de Java (composants JavaBeans, accès aux bases de données via JDBC etc.)
  - Les directives qui permettent de spécifier certaines propriétés de la page, l'inclusion du contenu d'autres fichiers et l'utilisation d'autres classes et bibliothèques de balises personnalisées.
  - Les actions permettent essentiellement d'utiliser des composants JavaBeans existants.
  - **Portabilité** : Très grande portabilité grâce à Java : serveurs Apache mais aussi IIS, etc.
  - **Mise en place** : Utilisation de Java comme langage de programmation.
  - **Performance** : Très variable en fonction de l'implémentation utilisée.
  - **Utilisation** : Tout type de fonctionnalités.
- **Servlets** : Le nom vient d'une analogie possible à faire avec les applets. Il s'agit donc de programmes créés en Java et tournant sur le serveur web. L'exécution du programme génère les pages web renvoyées au client
  - **Langages utilisés** : Java.

- **Portabilité** : Très grande portabilité grâce à Java : serveurs Apache mais aussi IIS, etc.
- **Mise en place** : Utilisation de Java comme langage de programmation.
- **Performance** : Très variable en fonction de l'implémentation utilisée.
- **Utilisation** : Tout type de fonctionnalités.

## Technologies Microsoft

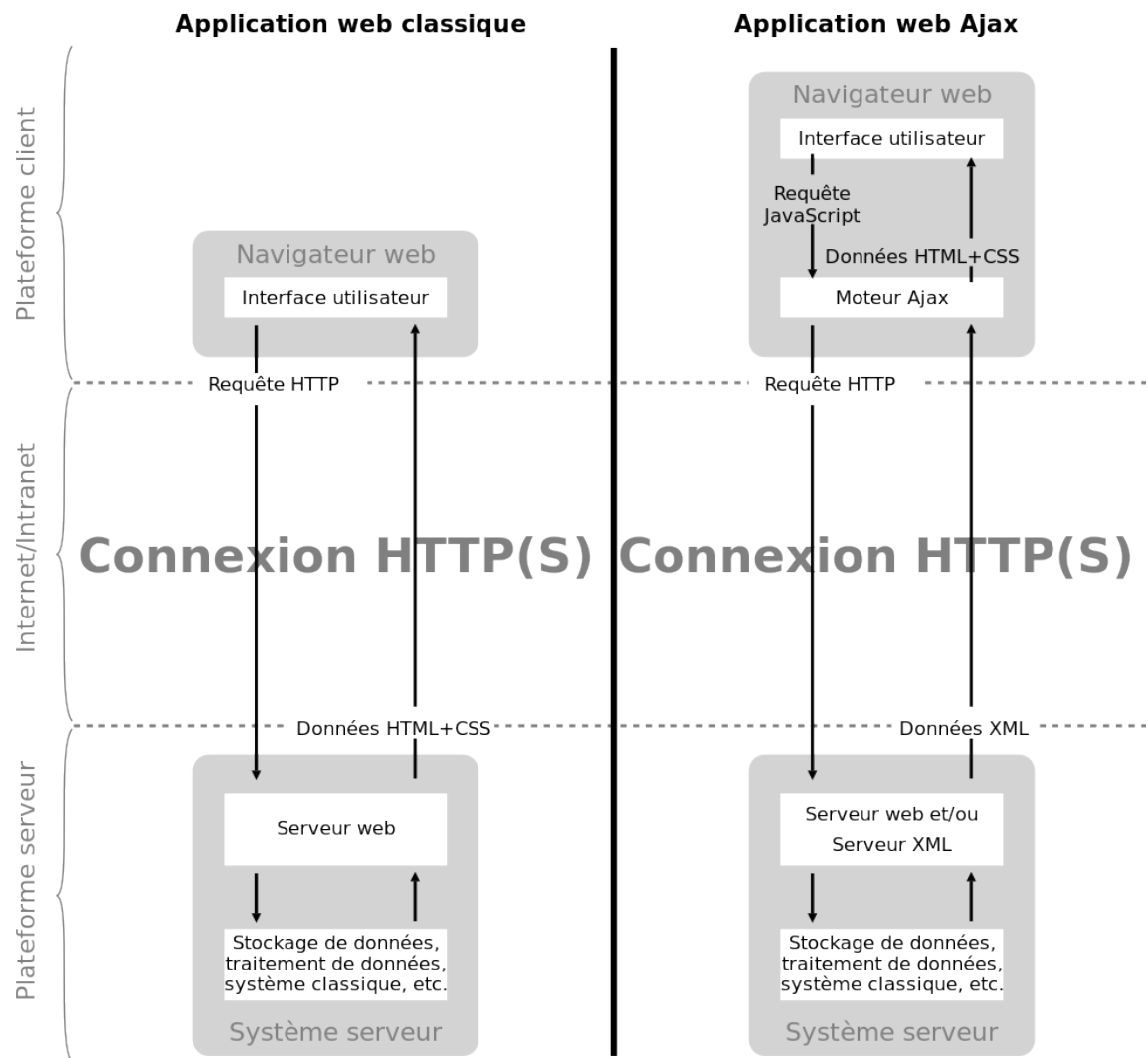
- **ASP**: Technologie développée par Microsoft à partir de 1996 dans le but de créer des pages dynamiques. ASP permet d'ajouter du code dans la page HTML qui sera interprété par le serveur. La partie ADO (ActiveX Data Object) de ASP permet de se connecter à une base de données.
  - **Langages utilisés** : VBScript (langage par défaut) ou JScript.
  - **Portabilité** : La technologie est implémentée sur les serveurs web de Microsoft (IIS et PWS).
  - **Mise en place** : Utilisation de langage de Script donc d'accès relativement aisé (mais se complexifie vite).
  - **Performance** : Bonne performance surtout si l'on utilise du code compilé (dll).
  - **Utilisation** : Tout site web dynamique peut être créé en ASP.
  - **Dernière version** : ASP 3.0.
- **ASP.NET** : Technologie développée par Microsoft sur base du Framework.Net. L'évolution est importante par rapport à ASP et les possibilités deviennent quasi infinies. ASP.NET a le gros avantage, par rapport à d'autres technologies, de bien séparer le contenu de la page web, sa mise en forme et la programmation des aspects dynamiques (travail en *Code Behind*).
  - **Langages utilisés** : C#, VB.Net, JavaScript.Net... et de nombreux autres langages partageant le *Common Runtime Language* du Framework.Net.
  - **Portabilité** : La technologie est implémentée sur les serveurs web de Microsoft (IIS 2003).
  - **Mise en place** : Utilisation de langage de programmation et nécessite donc de bonnes bases en programmation mais le langage importe peu (plusieurs langages possibles). Avec la version 2 (ASP.NET 2.0) et un outil de développement comme Visual Studio, le développement d'applications peut déjà aller très loin sans programmation : beaucoup de contrôles sont gérés par leurs propriétés.
  - **Performance** : Excellentes performances car tout le code est compilé (soit précompilé, soit compilé à la volée).
  - **Utilisation** : Tout type d'application web peut être créé avec ASP.NET.

## Technologie Ajax

Ajax est une technologie, avec ses avantages et inconvénients, qui combine plusieurs éléments :

- Du XHTML pour la structure de la page,
- Les CSS pour la mise en forme de la page,
- Du JavaScript et le DOM pour la partie dynamique,
- XML, XSLT et XMLHttpRequest pour la manipulation des données (ou d'autres formats de fichiers de données).

Cette technologie permet d'effectuer une requête HTTP asynchrone (en parallèle) depuis un client au serveur web (effectuer une action, mettre à jour une information...). Le serveur renvoie la ressource demandée, celle-ci pouvant contenir des informations ou du code pour mettre à jour la page HTML.



## Articles Wikipedia

- [Programmation Web](#)
- [Site web](#)
- [Conception de site Web](#)
- [HTML](#)
- [CSS](#)

[Simon BERNARD](#).

Pour le [Master SID](#) de l'[Université de Rouen Normandie](#)  
UFR Sciences et Techniques

*Avenue de l'Université*  
*76800 Saint-Étienne-du-Rouvray*