











# L'environnement web : serveur et client

# Sommaire

<b>I. Les serveurs</b> .....	3
A. Qu'est-ce qu'un serveur?.....	3
B. Les types de serveurs.....	4
<b>II. Les clients</b> .....	5
A. Qu'est-ce qu'un client? .....	5
B. Les différents types de client .....	6
<b>III. Les licences</b> .....	7
A. Les licences propriétaires.....	7
B. Les licences ouvertes (ou « Open Source »).....	7

Crédits des illustrations:  
© Skill and You

## Les repères de lecture

 Retour au chapitre	 Définition	 Objectif(s)	 Espace Élèves	 Vidéo / Audio
 Point important / À retenir	 Remarque(s)	 Pour aller plus loin	 Normes et lois	 Quiz

# I. Les serveurs

## A. Qu'est-ce qu'un serveur ?

Nous allons nous concentrer sur la notion de serveur « logiciel », appliquée au domaine du Web.

**?** Le « **serveur** » : il représente, d'une part, le matériel, la machine ayant pour rôle de gérer un réseau d'ordinateurs. D'autre part, on l'emploie pour désigner des logiciels, des programmes ayant pour rôle de gérer plusieurs « requêtes » d'un ensemble de systèmes clients.

Une « **requête** » : c'est une demande adressée par un client à un serveur. Cette demande, réalisée suivant un protocole précis, permet à différents programmes de communiquer.

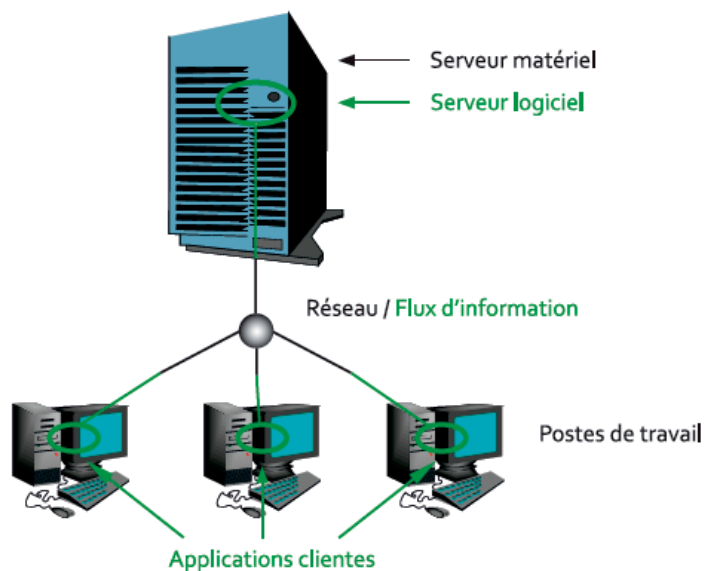


Fig. 1 Schéma montrant un serveur « matériel » et le principe des programmes

Le serveur est un des éléments matériels avec lequel vous serez le plus amené à travailler lors de vos futures réalisations de sites web.

Trois grandes familles de serveurs se dessinent, chacune ayant un rôle bien précis dans l'architecture d'un site web.

Il s'agit de notions dont vous avez peut-être déjà entendu parler :

- les serveurs web ;
- les serveurs applicatifs ;
- les serveurs de base de données.

## B. Les types de serveurs

### 1. Serveur web



Un **serveur web** : programme qui a pour rôle de distribuer à des applications tierces des pages HTML.

Ces pages HTML seront distribuées au travers d'un réseau vers différentes applications capables de les interpréter : « les navigateurs ».



Un **navigateur** : logiciel créé pour consulter le World Wide Web et, au minimum, un client HTTP. Il est composé d'un moteur de rendu des standards du Web avec une interface utilisateur. Il propose également un gestionnaire d'extensions appelées plugins. Les navigateurs les plus utilisés sont Mozilla Firefox, Google Chrome, Internet Explorer (IE)/Microsoft Edge, Safari et Opera.

Afin d'améliorer l'ergonomie d'un site web, ces pages HTML peuvent comprendre des instructions d'autres langages client, tels que le CSS ou le JavaScript.

### 2. Serveur d'application

Un serveur d'application, quant à lui, a pour rôle de comprendre et de transformer des langages avancés, tels que le C# ou le Java, en pages HTML qui pourront ensuite être distribués par des serveurs web aux navigateurs.

Il est assez rare de voir aujourd'hui des systèmes web sans serveur d'application, mis à part dans des domaines particuliers. La plupart du temps, ces derniers embarquent des serveurs web, ce qui simplifie grandement le travail des concepteurs.

### 3. Serveur de base de données

Les serveurs de base de données donnent la possibilité d'enregistrer, de classer et de restituer différentes informations utiles à un site.

Ces données seront accessibles via des applications disposant d'interfaces allant des plus simples aux plus complexes ; ou encore via des langages dédiés comme par exemple le SQL (pour Structured Query Language) qui permet d'accéder à ces informations à partir de tout autre programme en interrogeant la base de façon structurée.

Voici un tableau récapitulatif sur les fonctions des différents serveurs :

Tableau n°1

TYPE DE SERVEUR	FONCTION	EXEMPLE DE LOGICIEL / LANGAGE
Serveur base de données	Stocke les informations	<ul style="list-style-type: none"><li>– Mysql</li><li>– Postgresql</li><li>– Oracle</li><li>– MSSQL</li></ul>
Serveur applicatif	<ul style="list-style-type: none"><li>– Reçoit des requêtes et de l'information de la part du navigateur web.</li><li>– Les transmet au serveur d'application.</li><li>– Transmet la réponse du serveur applicatif au navigateur web.</li></ul>	<ul style="list-style-type: none"><li>– PHP</li><li>– JAVA</li><li>– C#</li></ul>
Serveur Web	<ul style="list-style-type: none"><li>– Décode les réponses du serveur HTML et les présente graphiquement à l'internaute.</li><li>– Envoi des requêtes et de l'information au serveur web.</li></ul>	<ul style="list-style-type: none"><li>– Apache</li><li>– Nginx</li><li>– IIS</li></ul>

## II. Les clients

Nous venons d'aborder les serveurs qui ont pour rôle de diffuser de l'information. Voyons maintenant quels programmes sont destinés à communiquer avec les serveurs.

### A. Qu'est-ce qu'un client ?

Tout comme pour les serveurs, lorsque l'on parle de réseaux, un client représente aussi bien le matériel que le logiciel. Ils ont tous deux les mêmes rôles par rapport à leurs serveurs respectifs.

Lorsque l'on parle de « client » ici, on fait référence à une application interfacée avec un serveur.

Un client envoie des requêtes au serveur auquel il est rattaché. Ces requêtes sont des demandes d'information auxquelles le serveur va répondre par l'envoi de données dans un format particulier préalablement défini.

## B. Les différents types de client

### 1. Client lourd



**Un client lourd :** programme qui s'installe sur un poste de travail et est connecté avec un serveur d'application.

Le point définissant le mieux un client lourd est que la grande majorité des actions qu'il exécute se font sur la machine qui l'héberge.

Mis à part pour certaines applications comme les outils de courrier électronique par exemple – même si de nos jours de plus en plus d'internautes utilisent des services comme Gmail de Google – les clients lourds sont aujourd'hui très peu utilisés dans le monde du Web.

*Exemple : Outlook de Microsoft, ou Thunderbird de Mozilla sont des clients lourds.*

### 2. Client léger

À la différence des clients lourds, un client léger n'exécutera sur le poste sur lequel il est installé que les actions de présentation : affichage de texte, d'images et de contenu dynamique.

Ce mode de fonctionnement s'adapte particulièrement bien aux besoins d'un site web.

Étant donné que les actions à réaliser côté client sont limitées, l'accès aux sites web développés est plus flexible.

Ce qui nous permet de visualiser un site aussi bien, ou quasiment aussi bien quel que soit le navigateur utilisé.

Les navigateurs les plus courants sur le Web sont : Internet Explorer (IE)/ Microsoft Edge de Microsoft, Firefox de Mozilla, Opera d'Opera Software et Chrome de Google. Cette liste n'est pas exhaustive, il existe énormément d'autres navigateurs dont les fonctionnements sont similaires, voire identiques.

Tableau n°3

TYPE DE CLIENT	FONCTION	EXEMPLE DE LOGICIEL / LANGAGE
Léger : Navigateur	<ul style="list-style-type: none"><li>– Décode les réponses du serveur HTML et les présente graphiquement à l'internaute.</li><li>– Envoi des requêtes et de l'information au serveur web.</li></ul>	<ul style="list-style-type: none"><li>– Firefox</li><li>– Chrome</li><li>– Safari</li><li>– Opera</li><li>– Internet Explorer</li><li>– Edge</li></ul>
Lourd	<ul style="list-style-type: none"><li>– traite les données du serveur d'application sur le poste de travail</li></ul>	<ul style="list-style-type: none"><li>– Outlook</li><li>– Thunderbird</li></ul>

### III. Les licences

Une autre notion très importante dans le monde du Web est la **notion de licence**. Cette notion doit être prise en compte afin d'être capable de choisir toujours la solution la plus adaptée au besoin.

Le choix de type de licence définira principalement le coût du site que vous mettrez en œuvre, mais aussi l'accès à diverses ressources : environnements de développement, hotlines, communautés de développeurs, bibliothèques de codes...

Cela dit, aujourd'hui, « **les solutions Open source** » et propriétaires offrent généralement les mêmes possibilités. Ce système de licence s'applique à tous les composants logiciels d'un site web, du serveur au client, en passant par les langages.



**Open source** : terme traduit de l'anglais « source accessible » ou « source libre » et signifiant d'un logiciel qu'il a été développé selon un mode de travail collaboratif.

Ces solutions open source, ou logiciels libres, sont élaborées grâce aux contributions des membres d'une communauté (core team) - composée de contributeurs de haut niveau et souvent portée par des éditeurs ou des sociétés de services spécialisés. Leur travail est alors contrôlé et validé par ces derniers.

**Le but des « Open source »** : produire des logiciels au coût limité et à la technicité élevée.

#### A. Les licences propriétaires

**Les solutions propriétaires ont pour principale particularité d'appartenir à leur société éditrice.**

Ce fait se traduit la plupart du temps par des coûts d'utilisation, généralement non négligeables.

Néanmoins, bien souvent à l'utilisation ces solutions sont fournies avec un service de support aux développeurs.

Généralement, les éditeurs fournissant ce type de solutions les maintiennent plus longtemps. Mais surtout, le support est généralement assuré par des professionnels, salariés ou prestataires de ces sociétés.

Les solutions Microsoft (le langage C# et la base de données SQL Server par exemple) sont propriétaires.

Leur utilisation nécessite l'achat de licences.

#### B. Les licences ouvertes (ou « Open Source »)

Les licences Open source ont un fonctionnement bien différent. Deux points les distinguent principalement des solutions propriétaires :

- le premier est que leur code source est accessible à tous. Dans les faits, n'importe qui peut voir comment la solution est développée ;
- le second est que ce code peut être utilisé et modifié sans restrictions (ou presque pas).

Ce dernier point amène un comportement très intéressant : ces solutions sont, pour la plupart, réalisées et maintenues par des passionnés, souvent développeurs et architectes logiciels professionnels. Ces solutions sont aussi gratuites, et contrairement à certaines idées reçues, très efficaces.

*Exemples de logiciels libres : Java, Unix, MySQL, PHP...*

Voici une présentation des licences des principaux logiciels utilisés dans le Web :

Tableau n°4

LOGICIEL	TYPE DE LICENCE	PLATEFORMES SUPPORTÉES	COÛT
Mysqls	Open Source		Gratuit (support payant si souhaité)
Postgresl	Open Source		Gratuit (support payant si souhaité)
Oracle	Propriétaire		Payant
MSSql	Propriétaire		Payant
Apache	Open Source		Gratuit
Nginx	Open Source		Gratuit
IIS	Propriétaire	Windows serveurs uniquement	Livré avec Windows server
Firefox	Open Source	Toutes plateformes	Gratuit
Chrome	En grande partie Open Source	Toutes plateformes	Gratuit
Safari	En grande partie Open Source	Windows/Mac OS X	Gratuit
Opera	En grande partie Open Source	Toutes plateformes	Gratuit
Edge	Propriétaire	Windows/Mac OS X	gratuit

Tableau n°5

LE CONTEXTE	ÉLÉMENTS DE CHOIX
Le budget de développement est restreint ?	De préférence, privilégier les solutions gratuites.
Les programmes développés pour le site doivent s'interfacer avec d'autres logiciels annexes ?	Dans un premier temps, s'informer sur la nature des logiciels car les logiciels propriétaires disposent souvent de leurs propres interfaces logicielles, souvent performantes (parfois, pas besoin de réinventer la roue).
Les programmes seront hébergés sur des plateformes open source ? (UNIX ou Linux par exemple)	Écarter les logiciels Microsoft qui sont payants. Privilégier toutes les solutions Open Source.