



Architecture du web

Formateur : Maxime LEROY
maxime.leroy@maximeleroy.fr



Définition : Internet

Ensemble de réseaux mondiaux interconnectés qui permet à des ordinateurs et à des serveurs de communiquer efficacement au moyen d'un protocole de communication commun (IP). Ses principaux services sont :

le Web : Ensemble de site ou page accessible depuis un navigateur

le FTP : Un moyen d'échanger des fichiers entre ordinateurs

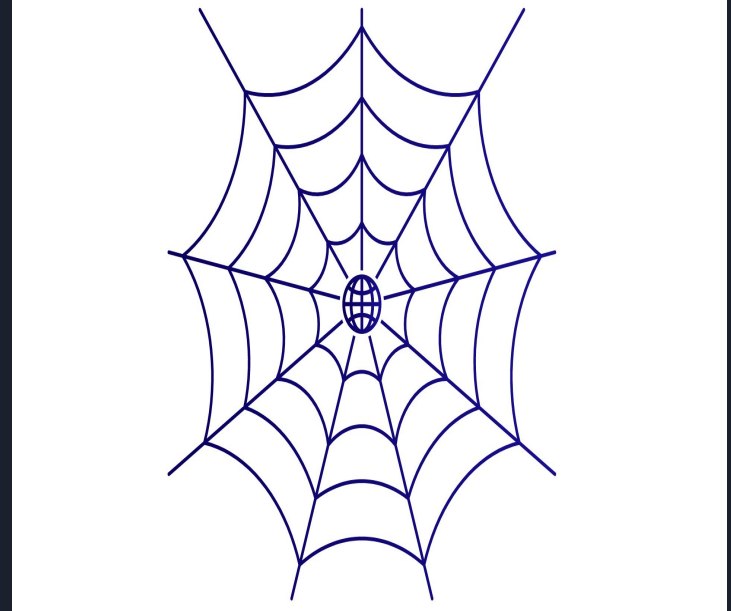
la messagerie : Pour échanger des emails

les groupes de discussion : L'ancêtre des forums

Définition : Web

Le World Wide Web (« la toile mondialement étendue »), est comparable à une toile d'araignée.

Il s'agit d'un réseau d'informations constitué de milliards de documents dispersés sur des millions d'ordinateurs serveurs dans le monde et reliés les uns aux autres selon le principe de l'hypertexte (liens).





Différence entre Web et Internet

Attention à ne pas confondre le web et Internet, internet est un réseau informatique que le Web utilise, le web est donc un service d'internet



L'histoire du web

- 1969 : création de l'ancêtre d'Internet, appelé alors **ARPAnet**. C'est un réseau militaire qui se veut décentralisé (il n'a pas de lieu de commande central). Le réseau a ensuite évolué pour devenir un lieu d'échange universitaire avant de devenir progressivement grand public sous le nom d'Internet.
- 1972 : apparition des **e-mails** pour échanger des messages.
- 1991 : apparition du **Web**, pour afficher des pages d'information.

L'inventeur du Web

Tim Berners-Lee, un Britannique. Il est le premier à présenter un concept de "page web" avec des "liens hypertextes" qui permettent de naviguer d'une page à une autre. Il est aussi à l'origine des bases du langage HTML que l'on utilise toujours aujourd'hui pour concevoir des pages web.





L'inventeur du web

Tim Berners-Lee était en 1991 chercheur au CERN (un important centre de recherche scientifique à la frontière franco-suisse). C'est là qu'il a eu l'idée du Web.

Il a par la suite créé un organisme, le W3C (World Wide Web Consortium) qui a pris le relais pour faire évoluer les technologies du web (HTML, CSS, PNG, XML et autres noms barbares).



Le Dark Web



Définition du Dark Web

Le Dark Web est un ensemble caché de sites Internet accessibles uniquement par un navigateur spécialement conçu à cet effet. Il est utilisé pour préserver l'anonymat et la confidentialité des activités sur Internet, ce qui peut être utile aussi bien pour les applications légales que pour les applications illégales. Si certains l'utilisent pour échapper à la censure gouvernementale, d'autres s'en servent également pour mener des activités hautement illégales.



Que sont le Dark Web, le Deep Web et le Web de surface ?

Internet est vaste. Il compte des millions de pages Web, de bases de données et de serveurs qui fonctionnent tous 24 heures sur 24. Cependant, l'Internet « visible » (également appelé Web de surface ou Web visible), composé de sites accessibles à l'aide de moteurs de recherche, comme Google et Yahoo, ne constitue que la partie émergée de l'iceberg.

Il existe un certain nombre de termes pour qualifier le Web non visible, mais mieux vaut connaître leurs différences.



Le web de surface ou le web visible

Le Web visible, ou le Web de surface, est la couche de surface « visible ». Si nous continuons à visualiser l'ensemble du Web comme un iceberg, le Web visible serait la partie supérieure qui se trouve au-dessus de l'eau. D'un point de vue statistique, cet ensemble de sites Web et de données représente moins de 5 % du volume total d'Internet.

Tous les sites Web accessibles au public via des navigateurs traditionnels, comme Google Chrome, Internet Explorer et Firefox, sont présentés ici. Les sites Web sont généralement étiquetés avec des opérateurs de registre, comme « .com » et « .org », et peuvent être facilement localisés au moyen des moteurs de recherche populaires.

La localisation de sites Web de surface est possible parce que les moteurs de recherche peuvent indexer le Web au moyen de liens visibles (il s'agit d'un processus appelé « exploration du Web » en raison du fait que le moteur de recherche parcourt le Web comme une araignée).



Le Deep Web

Le Deep Web se trouve sous la surface et représente environ 90 % de tous les sites Web. Il s'agit de la partie d'un iceberg sous l'eau, beaucoup plus grande que le Web de surface. En fait, ce Web invisible est si vaste qu'il est impossible de savoir précisément combien de pages ou de sites Web sont actifs à la fois.

Pour poursuivre l'analogie, les grands moteurs de recherche pourraient être considérés comme des bateaux de pêche qui ne peuvent « attraper » que les sites Web proches de la surface. Tout le reste, des revues universitaires à des bases de données privées en passant par du contenu plus illicite, est hors de portée. Ce Web profond comprend également la partie que nous connaissons sous le nom de **Dark Web**.



Ne pas confondre Deep Web et Dark Web

Bien que de nombreux médias utilisent indifféremment les expressions « Deep Web » et « Dark Web », une grande partie du Web profond dans son ensemble est parfaitement légale et sûre. Parmi les parties les plus importantes du Web profond, on trouve ce qui suit :

- **Bases de données** : collections de fichiers publics et privés protégés qui ne sont pas connectés à d'autres zones du Web, mais qui peuvent être consultés dans la base de données elle-même.
- **Intranets** : réseaux internes d'entreprises, de gouvernements et d'établissements d'enseignement utilisés pour communiquer et contrôler certains éléments à titre privé au sein de leurs organisations.



Ce qu'on trouve dans le Deep Web

- Les comptes financiers comme les comptes bancaires et les comptes de retraite
- Les comptes de messagerie électronique et sociale
- Les bases de données privées des entreprises
- Les renseignements sensibles de la HIPAA, comme les documents médicaux
- Les dossiers légaux



Le Dark Web

Le Dark Web fait référence à des sites qui ne sont pas indexés et qui ne sont accessibles que par des navigateurs Web spécialement conçus à cet effet. Beaucoup plus petit que le minuscule Web de surface, le Dark Web est considéré comme faisant partie du Web profond. Si l'on se base sur notre illustration de l'océan et des icebergs, le Dark Web serait la partie inférieure de l'iceberg submergé.

Le Dark Web, en revanche, est une partie très cachée du Web profond avec laquelle peu de gens interagissent ou que peu de gens découvriront. Autrement dit, le Web profond couvre tout ce qui se trouve sous la surface et qui reste accessible à l'aide du logiciel approprié, *y compris* le Dark Web.



Le Dark Web, un havre d'anonymat

- **Aucune indexation des pages Web** par les moteurs de recherche du Web de surface. Google et d'autres outils de recherche populaires ne peuvent pas détecter ni afficher les résultats des pages du Dark Web.
- Présence de « **tunnels de circulation virtuels** » via une infrastructure de réseau randomisée.
- **Inaccessible par les navigateurs traditionnels** en raison de son opérateur de registre unique. En outre, il est encore plus caché par diverses mesures de sécurité réseau, comme des pare-feu et du chiffrement.

La réputation du Dark Web a souvent été liée à des intentions criminelles ou à des contenus illégaux et à des sites « marchands » où les utilisateurs peuvent acheter des biens ou des services illicites. Cependant, des parties juridiques ont également fait usage de cette infrastructure.



Les langages clients

Avant toute chose, il faut savoir qu'il y a 2 types d'ordinateurs connectés au Web :

- **Des clients** : c'est vous, votre ordinateur qui sert à aller consulter des sites web.
- **Des serveurs** : ce sont des ordinateurs spéciaux (souvent très puissants) qui envoient les sites web aux clients. Les serveurs "possèdent" les sites web et les distribuent à ceux qui veulent les visiter.



Les langages

Pour construire un site web, on a recours à des langages. Ils servent en quelque sorte à établir les plans d'architecte dont on a besoin pour construire les sites. Ces langages sont principalement :



Tous ces langages sont désormais indispensables à la réalisation de tous les sites web. On dit que ce sont des **langages client** ou encore des **langages frontend**, car ils sont lus par les machines des clients.



HTML

L'HyperText Markup Language, **HTML**, désigne un type de langage **informatique** descriptif. Il s'agit plus précisément d'un format de données utilisé dans l'univers d'Internet pour la mise en forme des pages Web.



CSS

CSS est l'un des langages principaux du Web ouvert et a été standardisé par le W3C. Ce standard évolue sous forme de niveaux (*levels*), CSS1 est désormais considéré comme obsolète, CSS2.1 correspond à la recommandation et CSS3, qui est découpé en modules plus petits, est en voie de standardisation.



Javascript

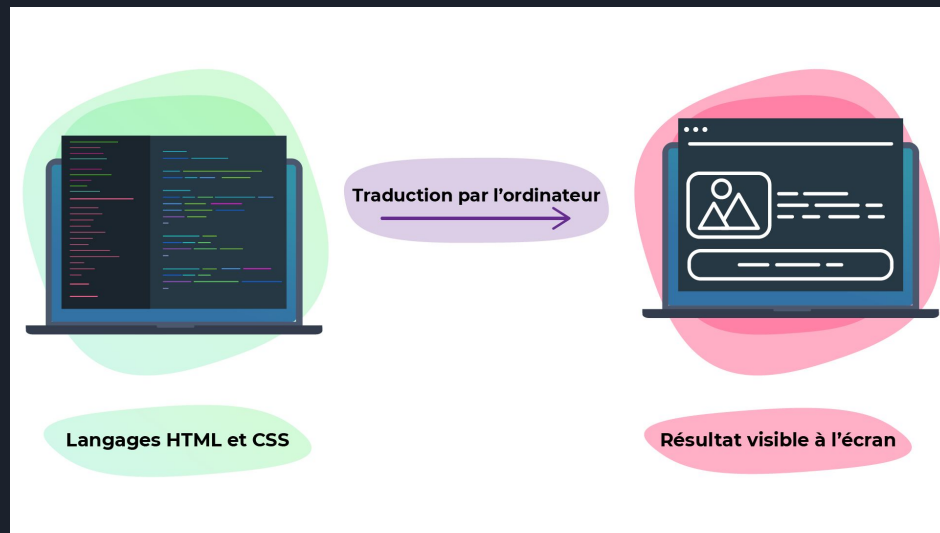
JavaScript (souvent abrégé en « JS ») est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web. Mais il est aussi utilisé dans de nombreux environnements extérieurs aux navigateurs web tels que Node.js, Apache CouchDB voire Adobe Acrobat. Le code JavaScript est interprété ou compilé à la volée (JIT). C'est un langage à objets utilisant le concept de prototype, disposant d'un typage faible et dynamique qui permet de programmer suivant plusieurs paradigmes de programmation : fonctionnelle, impérative et orientée objet.

Les navigateurs

Pour accéder aux sites web, on a besoin de navigateurs web. Il existe notamment :

- Google Chrome
- Mozilla Firefox
- Internet Explorer, et son successeur appelé Edge
- Opera
- Safari

Le rôle des navigateurs est de *traduire* les langages HTML, CSS et JavaScript sous la forme de sites web utilisables par tout le monde.



Les langages serveur





Différence langage client / langages serveur

Les langages serveur sont, comme leur nom l'indique, gérés par les serveurs. Les clients n'y touchent pas. Leur rôle est un peu différent :

- Les **langages client** décrivent comment le site web doit *s'afficher*.
- Les **langages serveur** décrivent comment le site web doit *se comporter*.

Avec un langage client, je peux dire "Un menu doit s'afficher à gauche de mon site web".

Avec un langage serveur, je peux dire "Le menu ne doit s'afficher que si cette personne a créé un compte sur mon site".



PHP, JAVA, PYTHON, JAVASCRIPT ...

- Les langages serveurs ou langages backend permettent d'adapter le comportement de votre page web en fonction de l'utilisateur
- Un langage serveur permet de communiquer avec une base de données ou de modifier le comportement d'une page web ou d'une application si un utilisateur possède un compte ou non, si celui-ci est connecté ou non

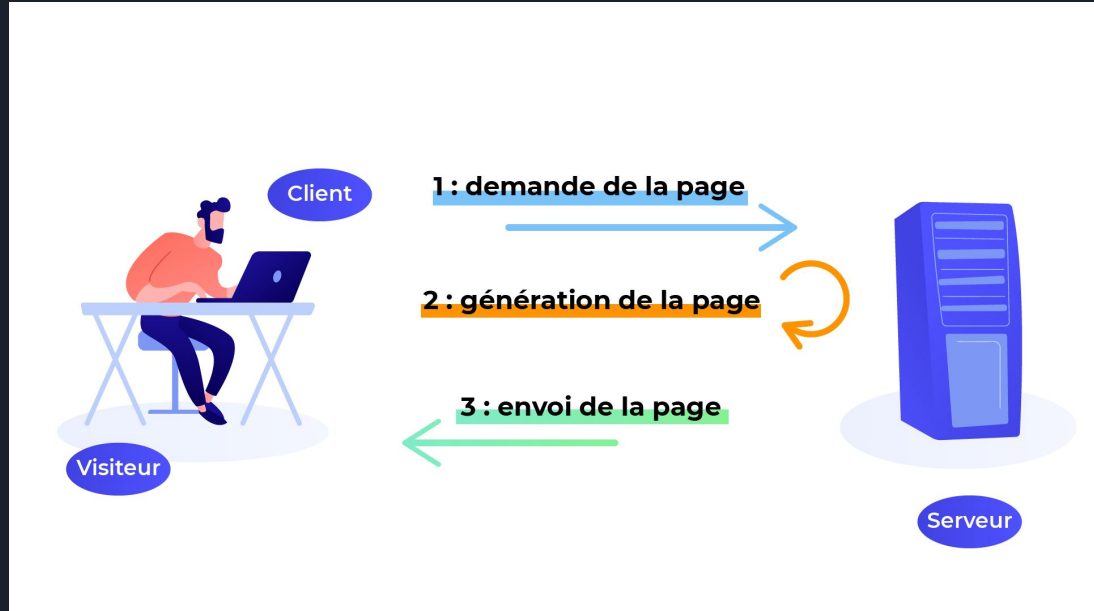
On va trouver ces langages serveurs parmi les sites les plus connus du monde :

- PHP chez Facebook
- JAVA sur le site de votre banque
- Du Python et du JAVA chez google

Génération de page avec un langage serveur

Les langages serveurs sont lus et interprétés par des serveurs.

Ces serveurs interprètent le code et envoi le résultat aux clients afin que celui-ci puisse le traiter et afficher la page





Les Frameworks

Un **framework** (ou infrastructure logicielle en français) désigne en programmation informatique un ensemble d'outils et de composants logiciels à la base d'un logiciel ou d'une application. ... L'objectif du **framework est** de simplifier et d'uniformiser le travail des développeurs.

PHP : Symfony, Laravel, Zend

JAVA : Jakarta ou Java EE

PYTHON : Django

JAVASCRIPT : VueJS

C# : ASP.NET



Library

En programmation, une librairie porte bien son nom. C'est, en apparence, comme une vraie bibliothèque pour le développeur : il peut y puiser tel ou tel ouvrage, selon ses besoins. D'ailleurs, librairie est la traduction quelque peu erronée car semblable au mot original en anglais : *library*. On devrait, officiellement, parler, justement, de bibliothèques.

C'est un ensemble de classes et de fonctionnalités déjà développé et dont on a juste à se servir pour ne pas réinventer la roue

Trop souvent, ce terme librairie est confondu avec celui de framework.

Javascript : JQuery

CSS : Bootstrap, Material



Les bases de données



Qu'est ce qu'une base de données ?

Une **base de données** est un ensemble structuré et organisé de **données** qui représente un système d'informations sélectionnées de telle sorte qu'elles puissent être consultées par des utilisateurs ou par des programmes.

Les bases de **données** permettent aux utilisateurs de centraliser et partager leurs informations à tout moment. ... L'information **est** également obtenue plus rapidement et plus facilement, ce qui simplifie grandement le travail et vous permet d'avoir un meilleur service à la clientèle.

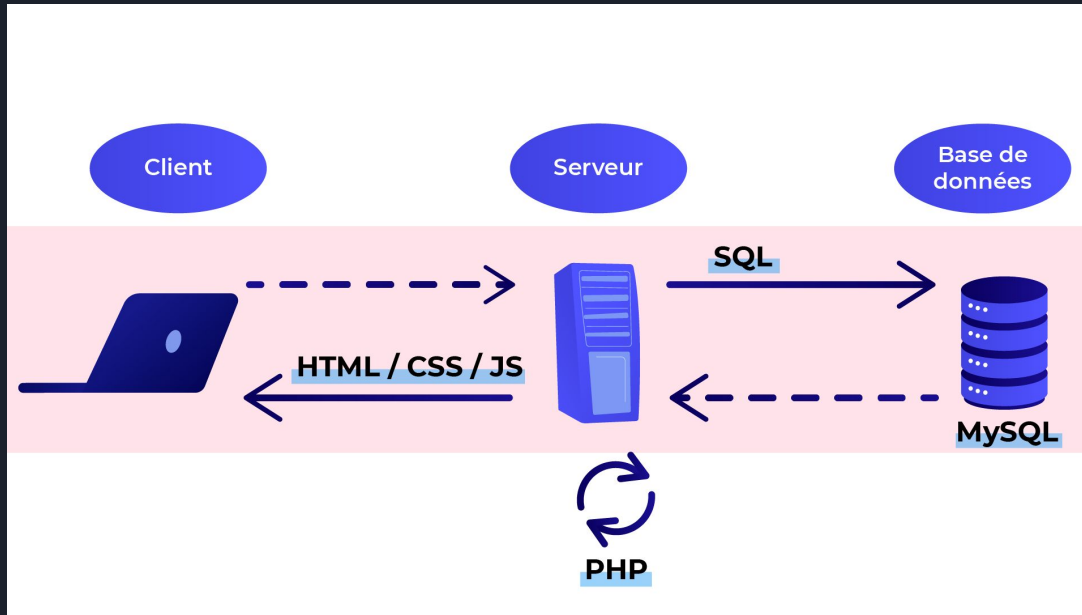


Quelques gestionnaires de bases de données

- MySQL
- PostgreSQL
- SQL Server
- Oracle
- SQLite
- Firebase
- MangoDB

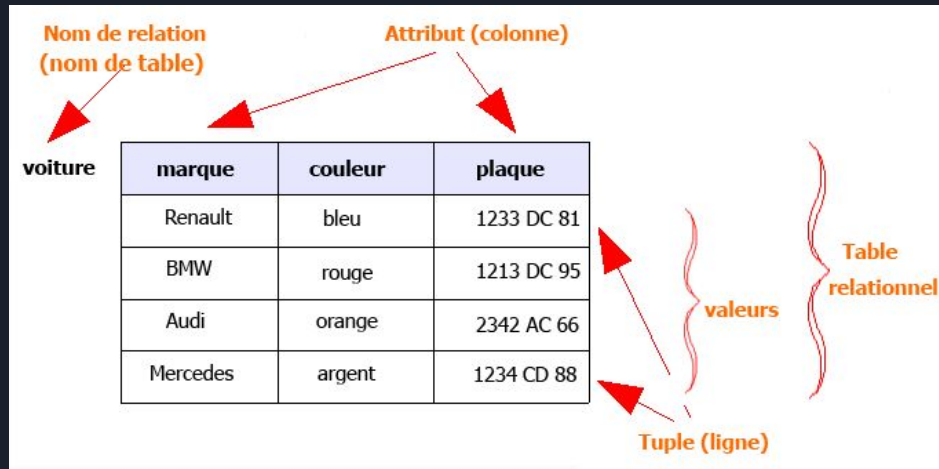
Pour communiquer avec ces bases de données on utilise le langage SQL

Fonctionnement d'une base de données



Comment est constitué une base de données ?

Une **base de données** est composée d'une collection de fichiers ; on y accède par le SGBD qui reçoit des demandes de manipulation du contenu et effectue les opérations nécessaires sur les fichiers. Il cache la complexité des opérations et offre une vue synthétique sur le contenu.





Les serveurs



La relation client-serveur

Nous pouvons communiquer entre tous ces ordinateurs, mais cela ne se fait pas tout seul.

Parmi les millions d'ordinateurs reliés ensemble par cette immense toile qu'est Internet, certains sont appelés des « serveurs », d'autres sont appelés des « clients ».

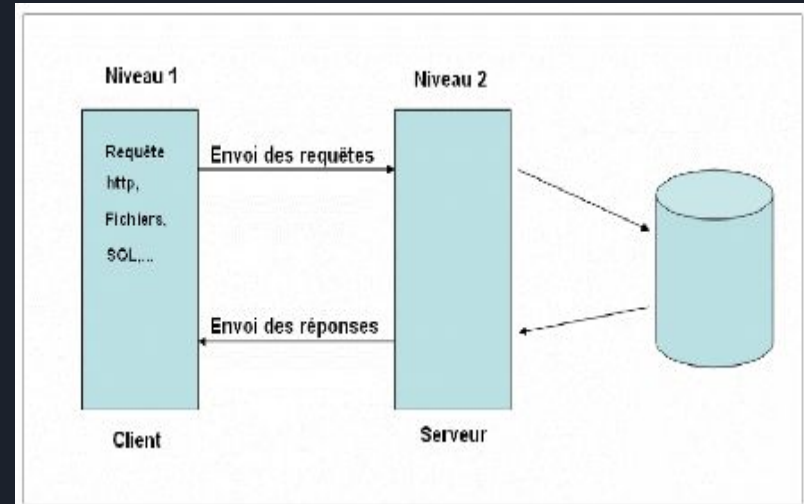
Pour simplifier, le client (vous) va effectuer une requête, puis le serveur distant va vous livrer cette information.

Architecture à 2 niveaux

L'architecture de base pour un site Web est une architecture à 2 niveaux entre un client (Google Chrome, Firefox, etc ...) et un serveur distant :

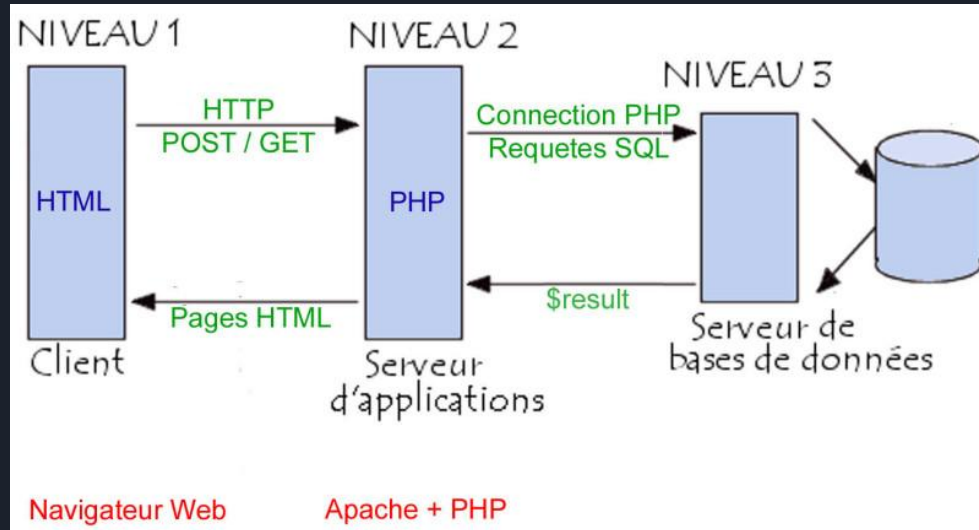
L'utilisateur émet une requête HTTP vers le serveur.

- Le serveur accède alors à ce fichier et le retourne au navigateur.
- Le navigateur lit ce fichier HTML et affiche la page.



Architecture à 3 niveaux

Le développement de sites interactifs, nécessitant de conserver des données (sur les utilisateurs par exemple), d'accéder à de grandes masses d'information, ou bien de modifier régulièrement du contenu, repose aujourd'hui sur une architecture à 3 niveaux.





Architecture à 3 niveaux

Ce type d'architecture, plus complexe que le client-serveur, permet l'accès aux bases de données stockées elles-mêmes sur un serveur. Les 3 niveaux s'articulent de la manière suivante :

- le côté client, l'interface utilisateur
- le serveur web, qui est complété par un serveur d'application qui va récupérer les données demandées. Le serveur HTTP, aussi appelé middleware, est donc à la fois serveur et client. Il envoie une requête au serveur d'application et dont il attend en retour le résultat. Une fois reçu, le serveur HTTP envoie ce résultat à l'utilisateur.
- le troisième niveau assure la gestion des données au sein d'un SGBD (Système de Gestion de Bases de Données) et répond aux requêtes du serveur HTTP.

Qu'est ce qu'un serveur ?

Le terme de « serveur » possède deux significations en informatique. On qualifie par serveur non seulement l'ordinateur qui fournit les ressources d'un réseau informatique, mais aussi le programme fonctionnant sur cet ordinateur.





Définition Serveur (Hardware)

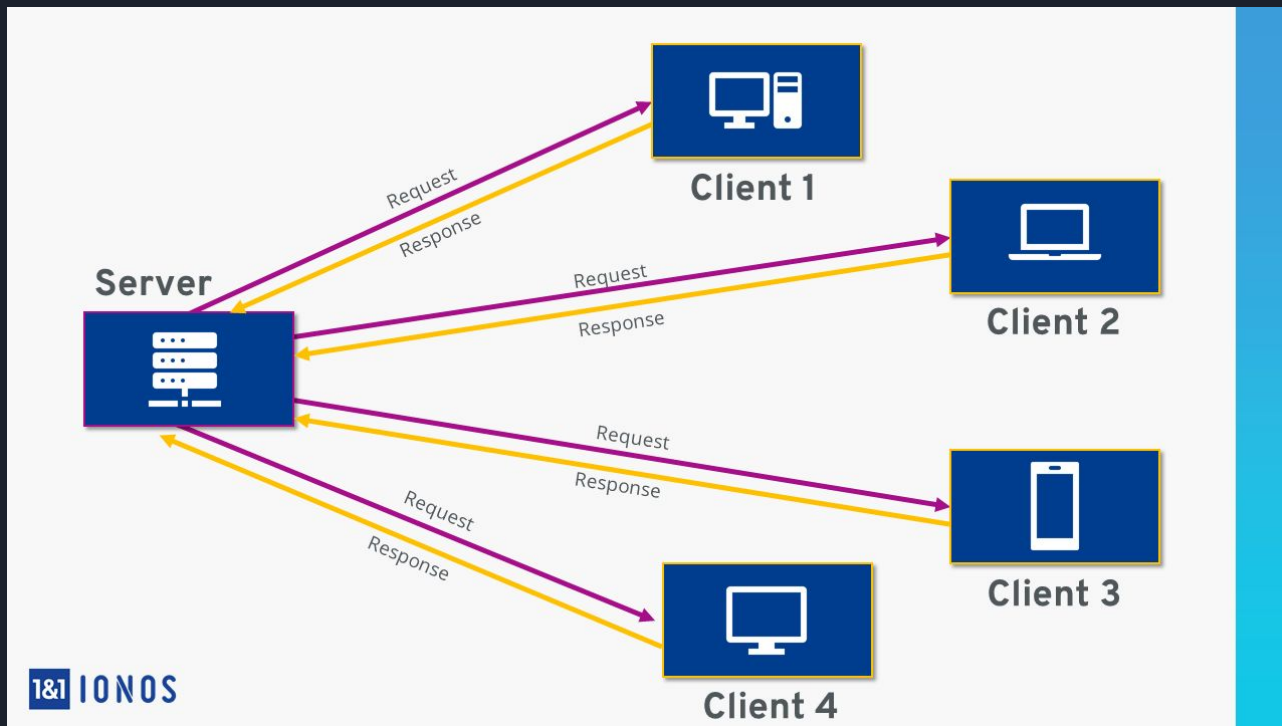
Un serveur matériel (hardware) est un réseau d'ordinateurs reliés par une machine physique et sur lequel fonctionnent un ou plusieurs serveurs logiciels (software). Une alternative au terme de serveur (Hardware) est « hébergeur » (Host en anglais). En principe chaque ordinateur est utilisé avec un serveur logiciel.



Définition Serveur (Software)

Un logiciel serveur est un programme effectuant des interactions en réseau avec d'autres programmes appelés logiciels clients. Le service apporté dépend du type de logiciel serveur. La base de la communication en réseau est cette relation Client-serveur. Lors de l'échange de données, différents protocoles de transmission entrent en jeu.

Fonctionnement d'un serveur





Fonctionnement d'un serveur

La mise à disposition de services de serveurs sur un réseau s'effectue sur le **modèle client-serveur**. Ce concept permet de répartir les tâches sur différents ordinateurs et les rendre accessibles à plusieurs utilisateurs finaux, de manière indépendante. Chaque service disponible sur un réseau est administré par un logiciel serveur. Il s'agit de la seule manière d'assurer un accès permanent au serveur et selon les besoins pour des clients tels que des navigateurs Web ou des programmes de messagerie électronique.



Type de serveur : Le Serveur Web

La tâche primaire d'un serveur Web est de stocker les pages Web, de les traiter et les livrer à des clients comme un navigateur ou à un robot d'indexation (Crawler) d'un moteur de recherche. La communication entre le serveur (software) et le client s'établit grâce au protocole HTTP (Hypertext Transfer Protocol) ou sa variante sécurisée HTTPS. En général les images, feuilles de style ou script sont transmis via des documents HTML. Les serveurs Web courants sont des serveurs http Apache, Microsoft Internet Information Services (IIS) ou Nginx.



Type de serveur : Le Serveur de fichiers (File-Server)

un serveur de fichiers sert à l'enregistrement central de fichiers accessibles depuis différents clients sur un réseau. Les entreprises misent sur de tels systèmes de gestion de fichiers dans le but que plus de groupes de travail puissent accéder aux mêmes fichiers. Un serveur de fichiers sert à résoudre les conflits que peut entraîner la présence de différentes versions de données locales. Il permet de stocker l'ensemble des versions de fichiers et permet une sauvegarde centrale des données de l'entreprise. Si l'accès au serveur de fichiers se fait par Internet, les protocoles de transmission tels que FTP (File Transfer protocol), SFTP (Secure File Transfer Protocol), FTPS (FTP via SSL) ou encore SCP (Secure Copy) entrent en jeu. Les réseaux locaux (LAN) recourent quant à eux aux protocoles SMB (Server Message Block) et NFS (Network File System).



Type de Serveur : Le Serveur Email

un serveur mail est composé de plusieurs modules logiciels dont l'assemblage permet la réception et l'expédition de courriers électroniques. En général, le protocole utilisé est le Simple Mail Transfer Protocol (SMTP). Les utilisateurs qui voudraient accéder à un serveur email ont besoin d'un client email dont le but est de récupérer les messages du serveur et de les expédier dans la boîte de réception électronique. Ce chargement passe par le protocole IMAP (Internet Message Access Protocol) ou POP (Post Office Protocol).



Type de serveur : Le Serveur de bases de données

on définit un serveur de bases de données comme un programme informatique permettant à d'autres programmes l'accès en réseau à un ou plusieurs systèmes de bases de données. Les logiciels de ce type ayant les plus grandes parts de marché sont Oracle, MySQL, Microsoft SQL Server, PostgreSQL et DB2. Les serveurs de bases de données sont en général compatibles avec les serveurs Web pour l'enregistrement et l'expédition de données.



Type de Serveur : Le GameServer

un serveur de gaming (jeu vidéo) est un logiciel serveur conçu spécialement pour les jeux multi-joueurs en ligne. Ces serveurs gèrent les données du jeu en ligne et permettent une interaction synchrone avec le monde virtuel correspondant. La base matérielle d'un serveur de gaming est obtenue grâce aux centres de données locaux de fournisseurs spécialisés.



Type de serveur : Serveur Proxy

le serveur proxy sert d'interface de communication dans des réseaux d'ordinateurs. En tant qu'instance de transmission, le serveur Proxy accepte les demandes du réseau et les transmet sur sa propre adresse IP. Un serveur Proxy permet de filtrer les communications, de contrôler la bande passante, la mémoire cache ou encore d'augmenter la disponibilité de vos ressources en répartissant les charges de données. De plus, un serveur Proxy permet un anonymat plus avancé, car l'adresse IP du client derrière le proxy n'est pas dévoilée.



Type de Serveur : Le serveur DNS

DNS (Domain Name System) ou système de nom de domaine est la dénomination d'un réseau informatique. Pour le World Wide Web, les serveurs DNS sont d'une importance capitale car ils traduisent des noms de domaines tels que `www.exemple.com` en adresses IP correspondantes.



Principe de DNS

Les machines présentes sur le réseau échangent des messages en spécifiant l'adresse IP du destinataire.

Lorsque l'utilisateur demande à son navigateur d'afficher une page d'un site Web (par exemple www.philiance.com), le navigateur envoie une requête au serveur qui détient le site.

Le navigateur doit donc au préalable transformer le nom du site www.philiance.com en adresse IP 217.70.180.135.

Le Domain Name System est une base de données qui contient la correspondance entre des noms symboliques et des adresses IP.

Ceci permet de faire référence aux machines par leur nom plutôt que par leur adresse IP.

On peut ainsi transférer un site d'un serveur à un autre (dont l'adresse IP est différente) tout en conservant le même nom.

DNS est lui même un protocole Internet, où les clients interrogent les serveurs pour obtenir les informations souhaitées.

Le programme nslookup permet d'interroger manuellement DNS

The background is a dark navy blue. On the left side, there are two overlapping geometric shapes: a blue parallelogram and a light green parallelogram, both tilted at an angle. The text 'TCP / IP' is centered in the right half of the image in a white, sans-serif font.

TCP / IP



Que signifie TCP/IP ?

Le protocole TCP/IP (Transmission Control Protocol/Internet Protocol) réunit les deux protocoles TCP et IP. Il s'agit donc d'une suite de protocoles associée au domaine d'Internet pour lequel elle facilite le transfert de données.

Présenté simplement, le protocole TCP/IP est un standard de communication entre deux processus. Il détermine et fixe les règles inhérentes à l'émission et à la réception de données sur un réseau. L'association des deux protocoles permet d'apporter des garanties de fiabilité dans le transfert des données. Avec le TCP/IP, vous êtes certain(e) que les informations envoyées arriveront bel et bien au bon destinataire.



Fonctionnement de la suite TCP/IP

Côté fonctionnement, la suite TCP/IP se base en partie sur l'adressage IP. Elle est capable de fractionner les informations échangées en paquets pour que les grosses données puissent être acceptées par les IP, et de contrôler les éventuelles erreurs de transmissions des données.



Les CDN




Qu'est ce qu'un réseau CDN ?

Le terme « réseau de distribution de contenu » (Content Delivery Network, CDN) désigne un groupe de serveurs géographiquement distribués travaillant de concert afin de diffuser rapidement un contenu Internet.

Un réseau CDN permet le transfert rapide des ressources nécessaires au chargement d'un contenu Internet, notamment les pages HTML, les fichiers JavaScript, les feuilles de style, les images et les vidéos. La popularité des services CDN continue de croître et, aujourd'hui, la majorité du trafic web est acheminée via ce type de réseau, y compris le trafic provenant de sites importants, tels que Facebook, Netflix et Amazon.

Correctement configuré, un réseau CDN peut également contribuer à protéger les sites web contre certaines menaces courantes, comme les attaques par déni de service distribué (DDoS).




Quelle est la différence entre un réseau CDN et un hébergeur web ?

Un réseau CDN n'héberge pas de contenu et ne peut pas remplacer la nécessité de disposer d'un hébergement web approprié, mais il peut faciliter la mise en cache du contenu à la périphérie du réseau, afin d'augmenter la rapidité d'un site web. De nombreux sites se tournent vers les réseaux CDN, car les services d'hébergement traditionnels ont du mal à répondre à leurs besoins en matière de performances.

En utilisant la mise en cache pour réduire la bande passante d'hébergement, afin d'empêcher les interruptions de service et d'améliorer la sécurité, les réseaux CDN constituent un choix populaire pour contourner certains des principaux problèmes associés à l'hébergement web traditionnel.

Les avantages d'un réseau CDN





1. Amélioration des temps de chargement du site web

en rapprochant le contenu des visiteurs du site à l'aide d'un serveur CDN situé à proximité (entre autres optimisations), ces derniers peuvent bénéficier de temps de chargement plus courts au niveau des pages. Les visiteurs étant plus enclins à éviter les sites lents, l'utilisation d'un réseau CDN peut permettre de réduire le taux de rebond et d'augmenter le temps qu'ils passent sur le site. En d'autres termes, un site web plus rapide entraîne une augmentation du nombre de visiteurs et de la durée de visite.



2. Réduction des coûts de bande passante

la consommation de bande passante pour l'hébergement constitue un poste de dépense principal pour les sites web. Grâce à la mise en cache et à d'autres optimisations, les réseaux CDN parviennent à diminuer la quantité de données qu'un serveur d'origine doit fournir, réduisant ainsi les coûts d'hébergement pour les propriétaires de sites web.



3. Augmentation de la disponibilité et de la redondance du contenu

les forts volumes de trafic ou les pannes matérielles peuvent interrompre le fonctionnement normal d'un site web. Grâce à sa nature distribuée, un réseau CDN peut traiter davantage de trafic et résiste mieux aux défaillances des équipements que de nombreux serveurs d'origine.



Amélioration de la sécurité du site web

un réseau CDN peut améliorer la sécurité grâce à la mise en œuvre de diverses solutions d'optimisation, comme l'atténuation DDoS et l'amélioration du traitement des certificats de sécurité.

Comment fonctionne un réseau CDN ?



Content Delivery Network (CDN)



Comment fonctionne un réseau CDN ?

Fondamentalement, un réseau CDN désigne un ensemble de serveurs reliés entre eux dans le but de diffuser du contenu aussi rapidement que possible, à moindre coût, mais aussi de manière fiable et sécurisée. Afin d'améliorer la vitesse et la connectivité, l'approche CDN placera les serveurs aux points d'échange entre les différents réseaux.

Ces points d'échange Internet (IXP) constituent les principaux points auxquels les différents fournisseurs d'accès Internet se connectent afin de se donner mutuellement accès au trafic provenant de leurs différents réseaux. Une connexion à ces emplacements à haute vitesse et hautement interconnectés permet à un fournisseur CDN de réduire les coûts et les temps de transit en matière de diffusion de données à haute vitesse.

En plus de placer les serveurs dans les IXP, les réseaux CDN apportent également un certain nombre d'optimisations aux transferts de données client/serveur standard. Ils positionnent ainsi les datacenters à des endroits stratégiques à travers le monde et améliorent la sécurité. En outre, ils sont conçus pour résister à divers types de défaillances et de congestions du réseau Internet.



Les Ports



A quoi servent les ports ?

Une adresse IP possède 65535 ports dédiés à recevoir les informations venant d'autres adresses IP.

Certaines applications utilisent des ports particuliers (en général situés entre 0 et 1023 et référencés dans une documentation normalisée pour les plus connus : RFC-1700) :

Serveur web (HTTP) : 80

Serveur ftp (FTP) : 20-21

Serveur web sécurisé (HTTPS) : 443

Serveur de messagerie entrant (POP) : 110

Serveur de messagerie sortant (SMTP) : 25



A quoi servent les ports ?

Ces ports sont les ports d'écoute des serveurs. Comme une radio sur une fréquence FM, chaque station émet sur une fréquence particulière que votre radio sait écouter. Les serveurs font la même chose que votre radio. Ils écoutent sur un port ("fréquence") particulier.

Entre l'émetteur et le récepteur, les ports utilisés ne sont pas les mêmes. Traditionnellement, les émetteurs utilisent des ports dynamiques (jamais les mêmes), entre les ports 49152 et 65535.



Les ports les plus connus

Il existe plusieurs protocoles de communication, mais les deux plus connus sont:

- TCP: la communication entre l'émetteur et le récepteur se fait de telle manière que le récepteur est prévenu de l'arrivée des données et l'émetteur reçoit un accusé de réception des données.
- UDP: le récepteur n'est pas averti de l'arrivée des données et l'émetteur ne reçoit pas d'accusé de réception des données.

Les protocoles





HTTP

Le protocole HTTP (HyperText Transfer Protocol) est le protocole le plus utilisé sur Internet depuis 1990. La version 1.0 du protocole (la plus utilisée) permet désormais de transférer des messages avec des en-têtes décrivant le contenu du message en utilisant un codage de type MIME.

Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisés grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web.

Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur. Elle comprend :

- Une ligne de requête (pour la méthode et le type de document)
- Les champs d'en-tête (optionnels)
- Le corps de la requête (si besoin)



HTTPS

Le protocole HTTP peut représenter un grand danger pour les internautes car il ne crypte pas les échanges de données entre le serveur et le navigateur. Les informations sont donc visible de tous, ce qui permet de les intercepter et de les lire sans difficulté.

C'est pour corriger cela qu'un protocole plus sécurisé est né : le HTTPS. Contrairement au HTTP, le HTTPS (HyperText Transfer Protocol Secure) va permettre de crypter les échanges avec une couche de chiffrement SSL ou TLS, rendant les données illisibles si elles venaient à être interceptées par la mauvaise personne.

Un certificat SSL est un fichier de données qui lie une clé cryptographique aux informations. Installé sur un serveur, le certificat active le cadenas et le protocole « https » (via le port 443) dans les navigateurs, afin d'assurer une connexion sécurisée entre le serveur web et le navigateur.

Source: medialibs



Type de requête HTTP

Il existe plusieurs requêtes HTML, mais nous allons voir sommairement les plus utilisées :

- GET : Cette méthode de requête existe depuis le début du Web. Elle est utilisée pour demander une ressource, par exemple un fichier HTML, au serveur Web. Par exemple:

`www.exemple.com/test.html` sera formulé ainsi:

`GET /test.html`

- POST : pour envoyer des données, nous n'utilisons pas la méthode GET car celle-ci transmet toutes les informations dans la barre URL de votre navigateur. Dans ces cas, la méthode POST est la plus adaptée car elle ajoute les données dans l'en-tête HTTP.



Type de requête HTTP

- PUT : La requête PUT n'est pas toujours utilisée, et peut être remplacée par une requête de type POST. Son but est de mettre à jour des données.
- DELETE: Comme son nom l'indique, cette requête sert surtout à supprimer des éléments, comme dans informations à l'intérieur d'une base de données par exemple.



Le protocole FTP

Le protocole FTP (File Transfer Protocol) est, comme son nom l'indique, un protocole de transfert de fichier.

Le protocole FTP a pour objectifs de :

- permettre un partage de fichiers entre machines distantes
- permettre de transférer des données de manière efficace



Le protocole SFTP

Comme nous l'avons vu précédemment, nous utilisons le protocole FTP pour transférer nos fichiers d'un client à un serveur. Le problème est qu'avec le FTP, la sécurité est très faible, sans aucun chiffrement.

Le SFTP (SSH File Transfer Protocol) a été développé pour offrir une alternative clairement améliorée sur le plan de la sécurité et se protéger d'éventuelles attaques.

Quelle est la différence entre SFTP et FTP ?

La différence principale réside dans le fait que, avec le transfert SFTP, l'authentification et l'ensemble du transit des données entre le client et le serveur sont chiffrés. Même si un pirate parvenait à mettre la main sur des données, inutilisables pour lui. En cas d'identifiants manipulés ou de tentative d'accès, le SSH file transfer protocol réagit en interrompant la connexion.




Le SSH

Le SSH permet à deux ordinateurs d'établir une connexion directe, de manière sécurisée.

Le serveur envoie un certificat au client pour vérifier qu'il s'agit bien du bon serveur. Puis le client doit également s'identifier au serveur en tant qu'utilisateur autorisé.

Après une authentification mutuelle, les deux participants à la communication établissent une connexion cryptée.



Les différents types de développeurs



Le développeur Front-End

Le développeur front-end s'occupe de toute la partie visible du site : il doit développer des interfaces au code optimisé, qui facilitent l'accès des utilisateurs au contenu. Pour cela, il maîtrise le CSS et le HTML5 afin de respecter les normes W3C. Grâce à une bonne connaissance des langages de programmation, le développeur front-end optimise l'interface du site, tout en respectant les impératifs liés à l'accessibilité, la compatibilité des navigateurs et les terminaux (desktop, mobile, tablette).



Les missions du développeur Front-End

- Assurer l'ergonomie visuelle et fonctionnelle du site
- Assurer la portabilité et interopérabilité du site par la prise en compte des différentes plateformes et navigateurs
- Maîtrise du responsive design, de l'UX et des interfaces mobiles
- Utilisation des outils collaboratifs
- Garantir l'accessibilité du site grâce au respect des normes W3C
- Participer à l'optimisation du SEO (référencement naturel) par l'intégration de la sémantique du web lors du développement (balises, mots clefs, optimisation d'URL)



Le développeur Back-End

Le développeur back-end gère toute la partie « non visible » du développement : il conçoit les éléments techniques nécessaires au fonctionnement d'un site, il se charge de la gestion des bases de données... Il a donc une connaissance approfondie des langages de programmation tels que PHP, Ruby, Python, Java, .Net et des bases de données SQL Serveur, MySQL. Il maîtrise les frameworks, les plus connus sur le marché étant Symfony ou Zend. Il maîtrise également les notions d'API pour dialoguer avec des sites partenaires.

Doté d'un bagage informatique complet et passionné de code, il participe à tout le développement du site web, en étroite collaboration avec le développeur front-end.



Les missions du développeur Back-End

- Conception et évolution de l'infrastructure d'un site web
- Architecture et développement de la plateforme de gestion des contenus
- Développement de solutions et d'applications web
- Evolution dans un environnement agile (scrum)
- Recettage, contrôles qualités (montée en charge, tests unitaires) et veille à la performance du site web
- Interprétation et conception des briefs techniques
- Programmation et paramétrage
- Test et contrôle qualité
- Documentation technique



Le développeur FullStack

Le saviez-vous ? La notion de développeur Full stack a été créée par Randy Schmidt pour cerner le « développeur à tout faire », qui sait empiler un grand nombre de compétences (stack = empilement en anglais) : back, front, architecture, expérience utilisateur...

Le développeur full stack, un métier confirmé aujourd'hui, est un profil très recherché des start-ups et des jeunes projets innovants du fait de son large spectre de compétences.

Sa polyvalence et sa parfaite autonomie lui permettent de s'impliquer sur des missions variées. Il sait à lui tout seul créer, développer, coder et maintenir d'une application ou d'un site web de A à Z. Il peut aussi compter sur l'aide d'un chef de projet ou d'un pôle marketing.

Vous allez ici vite comprendre pourquoi le développeur full-stack a les compétences les plus convoitées par les employeurs et pourquoi ce profil plus qu'un autre est devenu vital pour les entreprises ayant besoin d'internaliser l'intégralité de n'importe quel projet web ou applicatif.

En véritable couteau suisse qu'il est, l'ingénieur développeur full stack est ainsi capable d'intervenir à différents niveaux d'un projet informatique : du développement à la production, du CSS du front-end au React en back-end en passant par les bases de données et les serveurs.



Les missions du développeur fullstack

Le développeur full stack possède des compétences dans plusieurs langages de programmation back-end et front-end et Framework d'applications web: HTML, CSS, PHP, JavaScript, Python, Node.JS, React, AngularJS... Ces compétences techniques lui permettent de comprendre et d'avoir la main sur toutes les tâches techniques des sites web et apps.

- Développement de webservices pour faire communiquer des applis entre elles
- Conception d'API pour permettre à d'autres applis d'utiliser les données et fonctionnalités d'une appli principale
- Conception de plug-in pour permettre à un logiciel hôte d'avoir de nouvelles fonctionnalités intéressantes
- Gestion du côté client/front-end. Il s'occupe de tout ce que l'utilisateur voit : codage de l'interface, des éléments interactifs autres fonctionnalités du site ou de l'appli, intégration de la maquette html...
- Configuration de serveur et gestion de la base de données contenant les informations des utilisateurs (back-end)
- Rédaction de documentations techniques