

# 4 : Applications Client/Serveur et Web

## Cours 4 : Applications Client/Serveur et Web

Le Cocq Michel  
lecocq@ipgp.fr

Licence Pro SIL

17 Janvier 2017

# Rappel plan général

- Introduction au système UNIX
- Services, processus, signaux
- Scripting shell
- **Applications Client/Serveur et Web**
- Serveurs et configuration

1 Présentation Générale

2 Clients/Serveurs

3 Architecture 3 tiers

# Présentation Générale

## internet history

- **1962 ARPA** projet de recherche d'un réseau global d'ordinateurs.
- **1967 Arpanet** premier réseau à transfert de paquets.
- **1969** Connexion des ordinateurs entre 4 universités américaines.
- **1971** 23 ordinateurs sont reliés sur ARPANET.
- **1972** Naissance du **InterNetworking Working Group**
- **1973** Angleterre et Norvège rejoignent Internet avec 1 ordinateur
- **1979** Création des NewsGroups par des étudiants américains
- **1981** Apparition du Minitel en France
- **1982** Définition du protocole TCP/IP et du mot « Internet »
- **1983** Premier serveur de noms de sites
- **1984** 1 000 ordinateurs connectés
- **1987** 10 000 ordinateurs connectés
- **1989** 100 000 ordinateurs connectés
- **1989** Tim Berners-Lee lie le principe d'**hypertexte avec Internet.**
- **1990** Disparition d'ARPANET

# Présentation Générale

## internet history

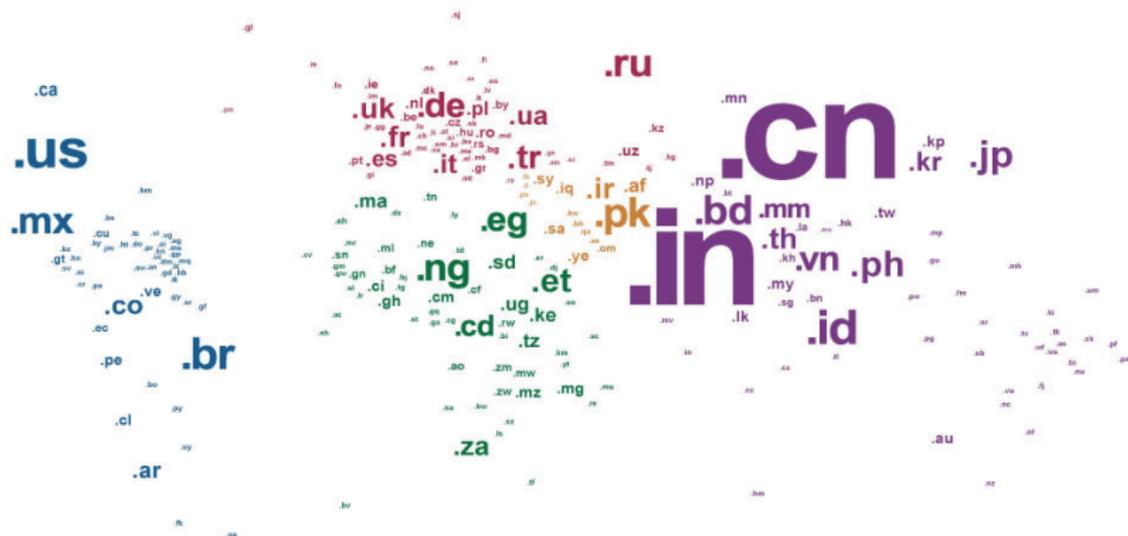
- **1991** Annonce publique du **World Wide Web**
- **1992** 1 000 000 d'ordinateurs connectés
- **1996** 10 000 000 ordinateurs connectés
- **1999** 200 000 000 utilisateurs dans le monde
- **2000** Explosion de la Bulle internet
- **2005** 1 000 000 000 utilisateurs dans le monde
- **2007** 2 320 000 000 utilisateurs dans le monde
- **2014** 2 900 000 000 utilisateurs dans le monde
  
- nous sommes 7,4 milliards sur terre : soit 40%

L'internaute est défini sur le site [internetworldstats.com](http://internetworldstats.com) comme étant une personne de plus de 2 ans, qui s'est connectée durant les 30 derniers jours.



# Présentation Générale

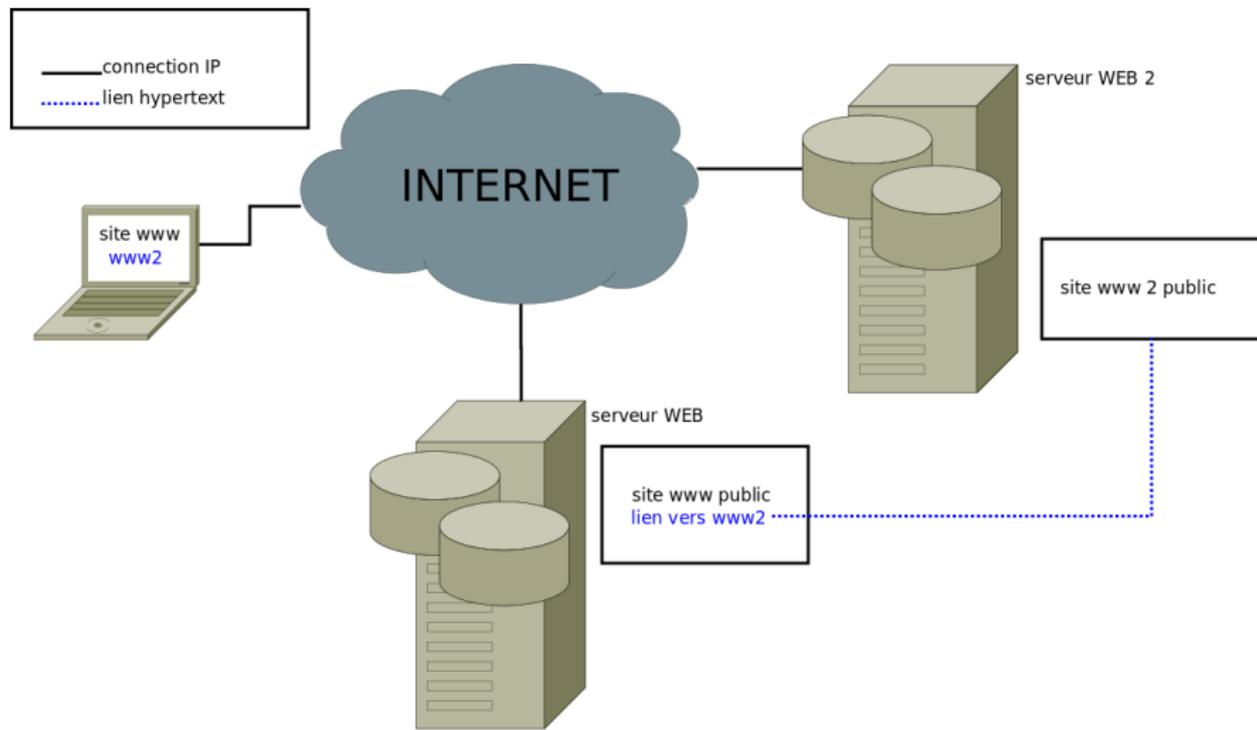
top-level domain Around the world



- Internet est le réseau informatique mondial qui rend accessibles au public des services comme le **courrier électronique** et le **World Wide Web**.
- Le réseau public mondial utilise le protocole de communication **IP** (Internet Protocol).
- Le World Wide Web est un système **hypertexte** public fonctionnant sur Internet et qui permet de consulter, avec un navigateur, des pages mises en ligne dans des sites. L'image de **la toile** vient des **hyperliens** qui lient les pages web entre elles.
- Le **Web** n'est qu'une des applications d'Internet. D'autres applications d'Internet sont le **courrier électronique**, la **messagerie** instantanée, **Usenet**, etc.

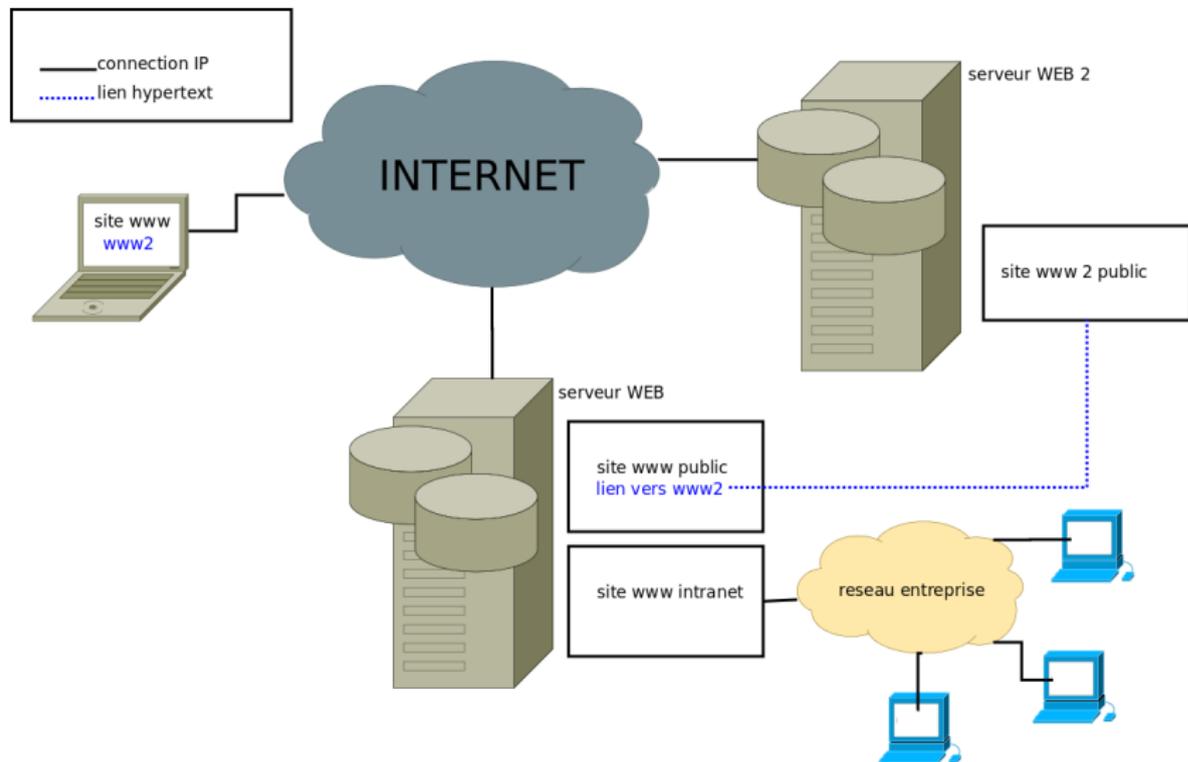
# Présentation Générale

## internet connection client serveur



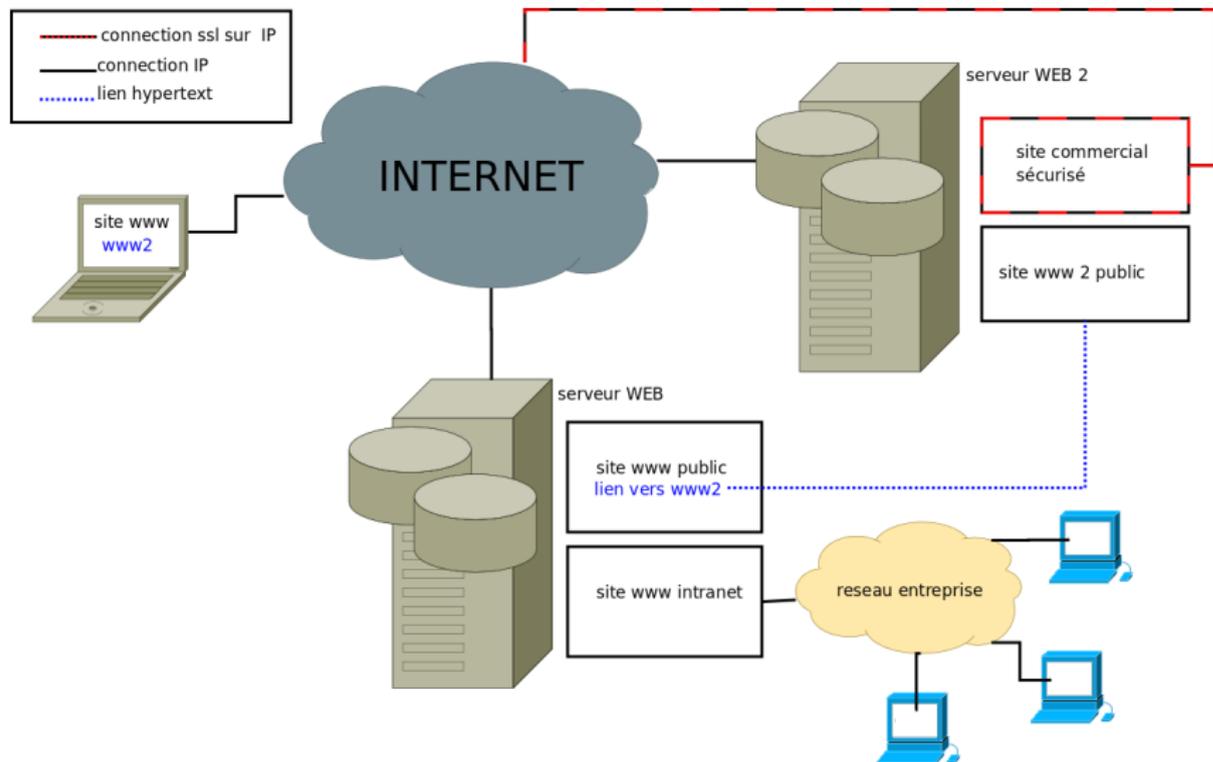
# Présentation Générale

## intranet connection client serveur



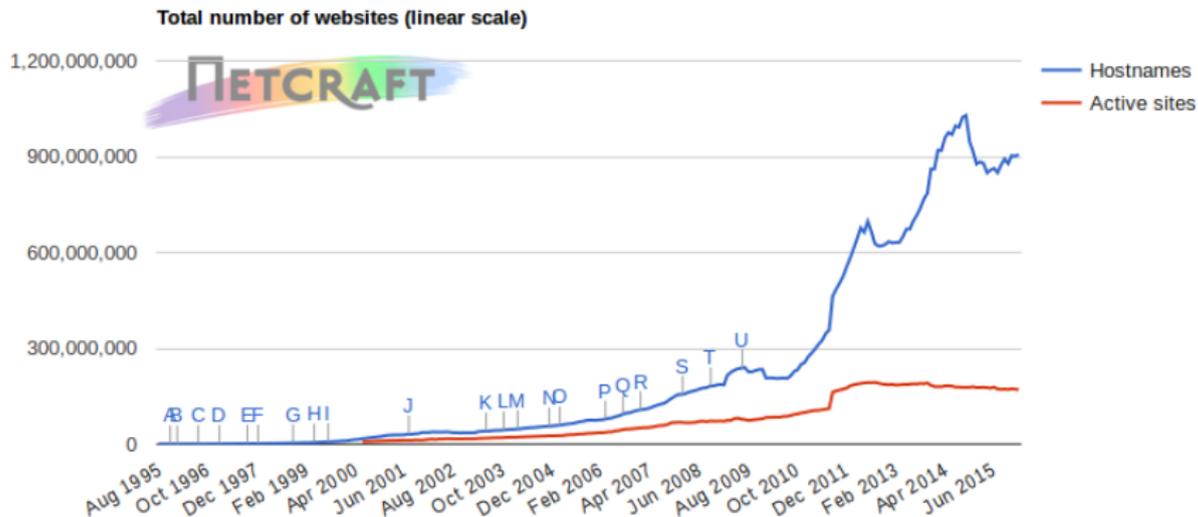
# Présentation Générale

internet **ssl** connection client serveur



# Présentation Générale

Total Sites Across All Domains August 1995 - April 2009



news.netcraft.com

# Clients/Serveurs

Qu'est-ce qu'un serveur ?

- Un serveur est un programme qui offre un service sur le réseau.
- Le serveur accepte des requêtes, les traite et renvoie le résultat au demandeur.
  - ▶ Une requête est un appel de fonction, la réponse éventuelle pouvant être synchrone ou asynchrone (le client peut émettre d'autres requêtes sans attendre)
  - ▶ Les arguments et les réponses sont énoncés dans un protocole
- Le terme serveur s'applique à la machine sur lequel s'exécute le logiciel serveur.
- Pour pouvoir offrir ces services en permanence, le serveur doit être sur un site avec accès permanent
- il doit s'exécuter en permanence (daemon - suffixe d pour le nom du logiciel ex. ftpd).

# Clients/Serveurs

Qu'est-ce qu'un client ?

- Un logiciel client est un programme qui utilise le service offert par un serveur.
- Le client envoie une requête et reçoit la réponse.
- Il peut-être raccordé par une liaison temporaire.

# Clients/Serveurs

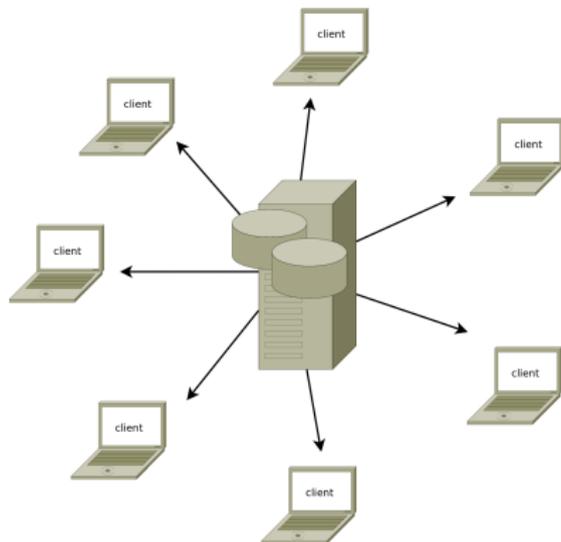
Qu'appelle-t-on architecture client/serveur ?

- C'est la description du fonctionnement coopératif entre le serveur et le client.
- Les services internet sont conçus selon cette architecture.
- Chaque application est composée de logiciel serveur et logiciel client.
- A un logiciel serveur, peut correspondre plusieurs logiciels clients développés dans différents environnements: Unix, Mac, PC...
- la seule obligation est le respect du protocole entre les deux processus communicants.
- Ce protocole étant décrit dans un RFC (Request For Comment).

# Clients/Serveurs

## Architecture générale

- L'architecture client serveur s'appuie sur un **poste central**, le serveur, qui **envoie des données** aux machines clientes.
- Des programmes qui accèdent au serveur sont appelés **programmes clients** (client FTP, client mail, navigateur).



# Clients/Serveurs

## Caractéristiques

### ● **Serveur**

- ▶ initialement **passif** en attente d'une requête
- ▶ **à l'écoute**, prêt à répondre aux **requêtes clients**
- ▶ quand une requête lui parvient il la **traite** et envoie la réponse

### ● **Client**

- ▶ **actif** en premier
- ▶ envoie des **requête au serveur**
- ▶ attend et **reçoit les réponse** du serveur

- le clients et le serveur doivent utiliser le **même protocole**
- un serveur peut répondre à **plusieurs clients en simultané**

# Clients/Serveurs

## Avantages/Inconvénients

### ● Avantages

- ▶ **Unicité de l'information** : toutes les données sont stockées sur un même serveur
- ▶ **Meilleure sécurité** : simplification des contrôles de sécurité item **mise à jours** : mise à jour centralisé aussi bien des données et logiciels
- ▶ **Meilleure fiabilité** : En cas de panne, seul le serveur fait l'objet d'une réparation
- ▶ architecture plus **mature** que les autres
- ▶ **Facilité d'évolution** : architecture évolutive, il est très facile de rajouter ou d'enlever des clients ou des serveurs.

### ● Inconvénients

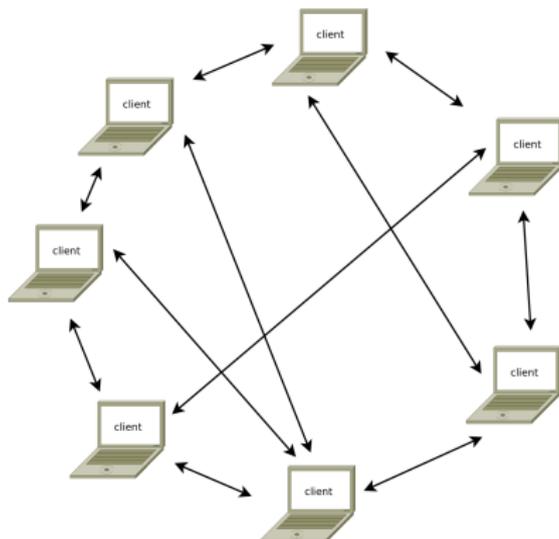
- ▶ Un **coût d'exploitation élevé** (bande passante, câbles, ordinateurs surpuissants)
- ▶ En **cas de panne** du serveur, plus aucun client n'a accès aux informations
- ▶ si trop de clients veulent communiquer avec le serveur ce dernier risque de ne pas supporter la **charge**

- Fonctionnement d'un réseau client /serveur :
  - ▶ Le client pour recevoir des informations du serveur lui **émet une requête** passant par un port du PC (exemple : port 25 pour les mails, port 80 pour le web et 21 pour le FTP).
  - ▶ Le serveur lui envoie ensuite les informations grâce à l'**adresse IP** de la machine cliente.
  - ▶ Le client **traite** et affiche les informations en provenance du serveur

# Clients/Serveurs

## Autre type d'architecture

- Réseau pair à pair ou (peer-to-peer - P2P). Chaque ordinateur est à la fois client et serveur



- sous Unix `/etc/services` indique :
  - ▶ un nom de service
  - ▶ un numéro de port (canal) où il est rendu
  - ▶ le protocole de communication utilisé par ce port.
- Un même service peut-être rendu sous différents protocoles, par exemple chiffré et non chiffré (voir POP et POPS)
- différents ports peuvent rendre le même service, pour améliorer sa disponibilité
- `/etc/services` n'est qu'une information standard consultable sur la machine cliente, chaque serveur pouvant adopter d'autres configurations

# Clients/Serveurs

## Exemples

- La consultation de pages sur un site web : Un **internaute** connecté via son navigateur web est le **client**, le serveur est composé du ou des ordinateurs contenant les applications qui servent les pages demandées. Le protocole de communication HTTP est utilisé.
- Les courriels sont envoyés et reçus par des clients et gérés par un serveur de messagerie. Les protocoles utilisés sont : SMTP POP IMAP.
- une base de données centralisée sur un serveur peut se faire à partir de plusieurs postes clients qui permettent de visualiser et saisir des données.
- X Window fonctionne sur une architecture client/serveur. En général le client tourne sur la même machine que le serveur mais peut être lancé sur un autre ordinateur faisant partie du réseau.
- Un Client léger est un ordinateur s'appuyant sur un serveur central qui héberge et exécute toutes les applications.

# Architecture 3 tiers

## Définition et concepts

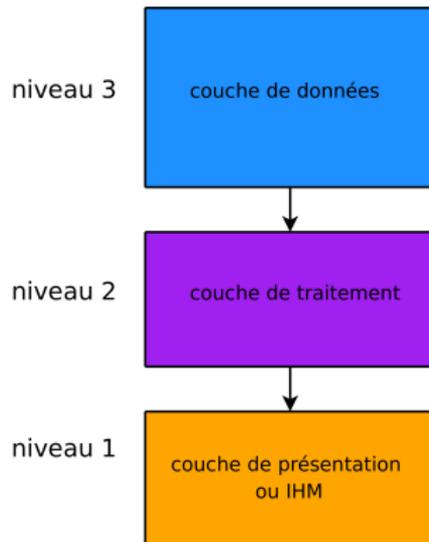
- C'est une extension du modèle client/serveur.

L'architecture 3-tier ou architecture à trois niveaux est l'application du modèle plus général qu'est le multi-tiers. L'architecture logique du système est divisée en trois niveaux ou couches :

- couche présentation
- couche métier ou traitement
- couche accès aux données
- affichage, la restitution sur le poste de travail, dialogue avec l'utilisateur
- mise en œuvre de l'ensemble des règles de gestion applicative
- données destinées à être conservées sur la durée

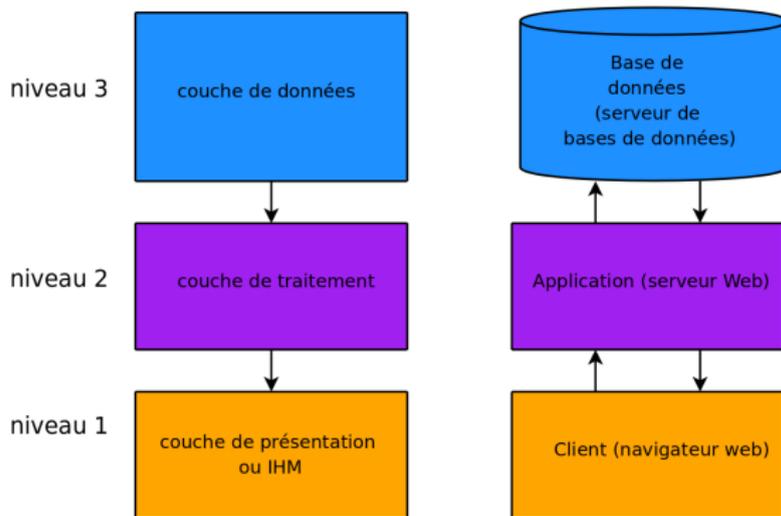
# Architecture 3 tiers

## Définition et concepts



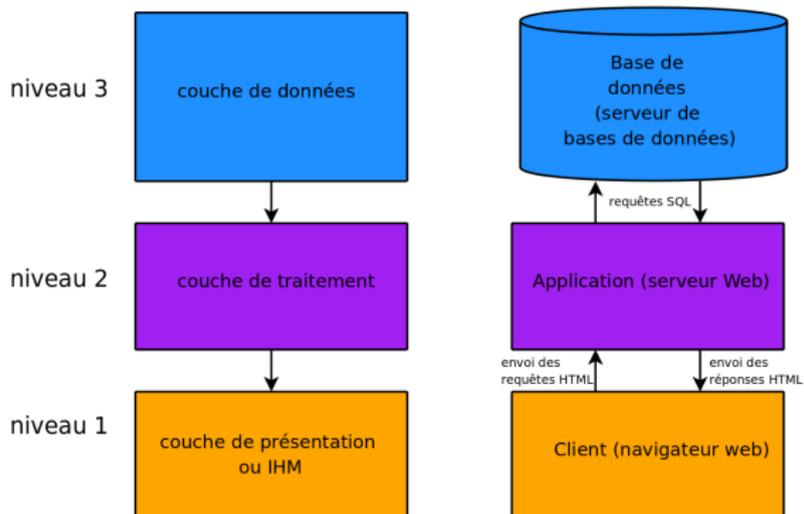
# Architecture 3 tiers

## Définition et concepts



# Architecture 3 tiers

## Définition et concepts



# Architecture 3 tiers

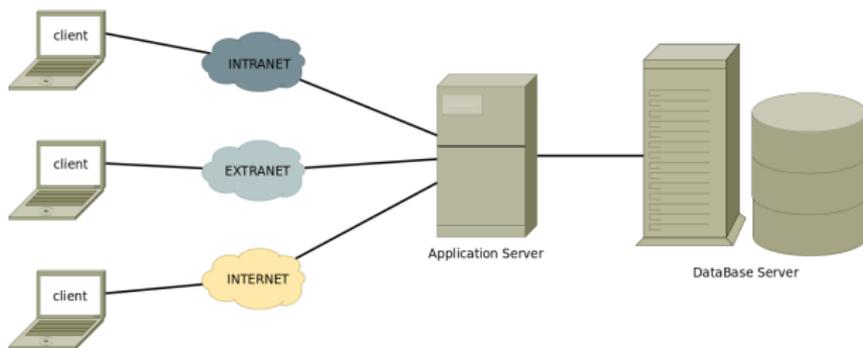
## Définition et concepts

- Les couches communiquent entre elles au travers d'un « modèle d'échange », et chacune propose un **ensemble de services**.
- Les services d'une couche sont **mis à disposition de la couche supérieure**.
- Il est interdit qu'une couche invoque les services d'une couche adjacente.
- les fonctionnalités de chacune des couches peuvent évoluer sans induire de changement dans les autres couches.
- Une nouvelle fonctionnalité de l'application peut avoir des répercussions dans plusieurs couches.

Il est essentiel de définir un modèle d'échange assez souple, pour permettre une maintenance aisée de l'application.

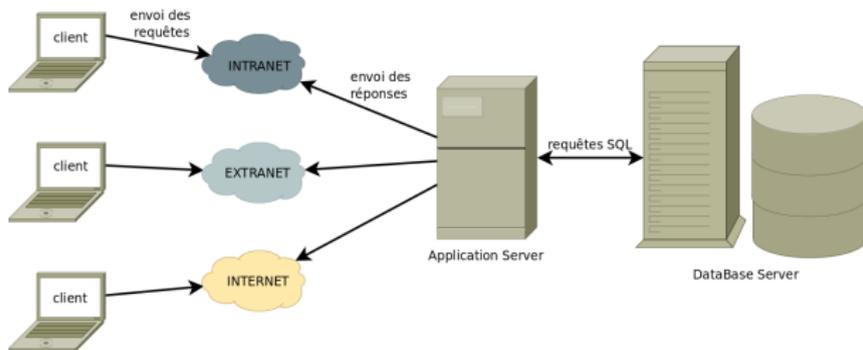
# Architecture 3 tiers

## Définition et concepts



# Architecture 3 tiers

## Définition et concepts



# Architecture 3 tiers

## Les trois couches

- **Couche Présentation (premier niveau)**
  - ▶ Correspond à la partie de l'application visible et interactive. On parle d'**Interface Homme Machine (IHM)**. Peut être réalisée par une application graphique ou textuelle.
  - ▶ Elle peut prendre de multiples facettes sans changer la finalité de l'application. Dans le cas d'un système de distributeurs de billets, l'automate peut être différent d'une banque à l'autre, mais les fonctionnalités offertes sont similaires et les services identiques (fournir des billets, donner un extrait de compte, etc.).
  - ▶ Une même fonctionnalité métier pourra prendre **différentes formes de présentation** selon qu'elle se déroule sur Internet, sur un distributeur automatique de billets ou sur l'écran d'un chargé de clientèle en agence...
  - ▶ Cette couche relaie les requêtes de l'utilisateur à destination de la couche métier, en retour elle lui **présente les informations** renvoyées par les traitements de cette couche. Il s'agit donc ici d'un assemblage de services métiers et applicatifs offerts par la couche inférieure.

- **Couche Métier / Business (second niveau)**
  - ▶ partie fonctionnelle de l'application : implémente la « logique », et décrit les opérations que l'application **opère sur les données** en fonction des requêtes des utilisateurs, effectuées au travers de la couche présentation.
  - ▶ mises en œuvre des différentes **règles de gestion** et de **contrôle** du système
  - ▶ La couche métier offre des **services applicatifs et métier** à la couche présentation. Pour fournir ces services, elle s'appuie, le cas échéant, sur les données du système, accessibles au travers des services de la couche inférieure. En retour, elle **renvoie** à la couche présentation les **résultats** qu'elle a calculés.

# Architecture 3 tiers

## Les trois couches

partie gérant l'accès aux gisements de données du système. Ces données peuvent être propres au système, ou gérées par un autre système. La couche métier n'a pas à s'adapter à ces deux cas, ils sont transparents pour elle, et elle accède aux données de manière uniforme

- Couche Accès aux données (troisième niveau)
  - ▶ Données propres au système
    - Données destinées à **durer dans le temps**. Elles peuvent être stockées indifféremment dans de simples fichiers texte, ou eXtensible Markup Language (XML), ou encore dans une **base de données**.
    - Les services sont **mis à disposition de la couche métier**. Les données renvoyées sont issues du/des gisements de données du système.
  - ▶ Données gérées par un autre système
    - Les données peuvent aussi être gérées de manière externe. Elles ne sont pas stockées par le système considéré,

# Architecture 3 tiers

## Buts et Objectifs

- Allègement du poste de travail client
- Prise en compte de l'hétérogénéité des plates-formes (serveurs, clients, langages, etc.)
- Introduction de clients dits 'légers' (plus liée aux technologies Intranet/HTML qu'au 3-tier proprement dit)
- Amélioration de la sécurité des données, en supprimant le lien entre le client et les données. Le serveur a pour tâche, en plus des traitements purement métiers, de vérifier l'intégrité et la validité des données avant de les envoyer dans la couche de données
- Meilleure répartition de la charge entre différents serveurs d'application.