Bases de données documentaires et distribuées, http://b3d.bdpedia.fr

Cloud et scalabilité

Infrastructures *cloud*

Les données sont massives quand elles dépassent les capacités d'une seule machine.

- en mémoire RAM : quelques dizaines de GOs ;
- en mémoire persistante : quelques TOs.

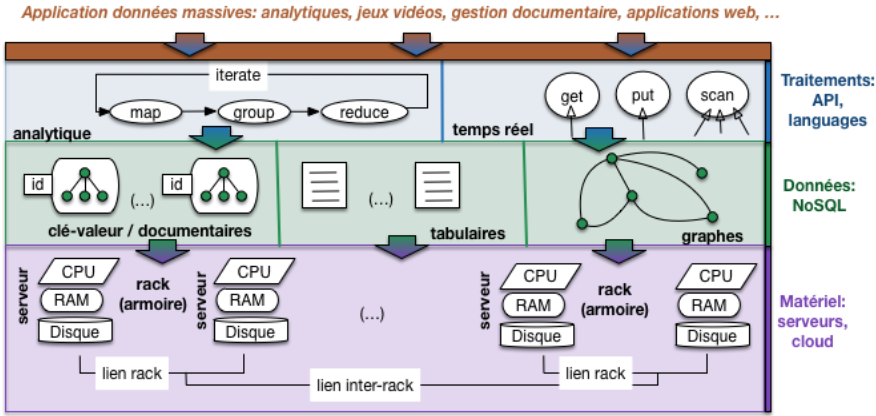
Qui dit **données massives** dit donc **système distribué**. Point communs (le seul ?) des systèmes dits NoSQL.

De fait : l'infrastructure est constituée de grappes de serveurs, avec

- des machines à bas coût (*commodity servers*),
- la possibilité d'en ajouter/retirer en quelques minutes (**élasticité**),
- des environnements logiciels spécialisés.

Nous appellerons l'ensemble un *cloud* (un peu réducteur).

Vision générale



Serveurs et baies

Dans un centre de données, les serveurs sont empilés dans des baies.

Un serveur

Une baie (*rack*) de serveurs

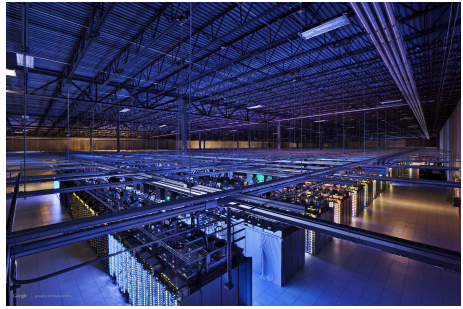
Infrastructures *cloud*

Une baie contient environ 40 serveurs.

Un centre de données contient quelques centaines de baies.

Ordre de grandeur : quelques milliers de serveurs par centre.

Combien des centres chez Google, Facebook, Amazon ... ? Des millions de serveurs.



La virtualisation

Prenez une machine avec 64GOs de mémoire, 12 disques de 3 TO chacun.

Découpez-là en sous-machines “virtuelles”.

- 18 serveurs virtuels avec 4 GO de RAM chacun (2 TO de disques) ;
- 8 serveurs virtuels avec 8 GO de RAM chacun (3 TO de disques) ;
- etc...

Permet d’attribuer des serveurs à la demande, optimise l’utilisation des machines.

Pour le client : 0 coût installation, maintenance, amortissement.

Cf. Amazon EC2, OVH, Gandi (français), etc.

Les pannes

Choix essentiel : **on augmente la puissance du système par ajout de machines à bas coût (scalabilité horizontale)**, et pas par l'augmentation des capacités d'une machine (scalabilité verticale).

Conséquence essentielle : des pannes tout le temps !

- panne de disque ;
- redémarrage serveur (panne logicielle/matérielle)
- coupure réseau.

Méthode générale : réplication des données ; redondance des services.

Les systèmes (NoSQL) disposent d'une reprise sur panne (*failover*) **automatique**.

Systèmes distribués

Système distribué = ensemble de composants logiciels, ou **nœuds**, répartis sur une **grappe de serveurs**, coopérant pour une tâche précise.

Ces nœuds communiquent par **envoi de messages** (aucun partage de mémoire).

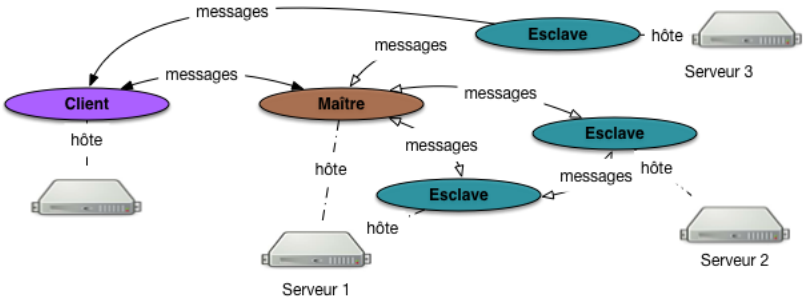
Chaque nœud est en écoute sur un port réseau (ex., le 80 pour le Web, 27017 pour Mongodb).

On peut avoir un système distribué (mais pas scalable) sur une seule machine !

En général, un nœud par machine.

Le client, les maîtres et les esclaves

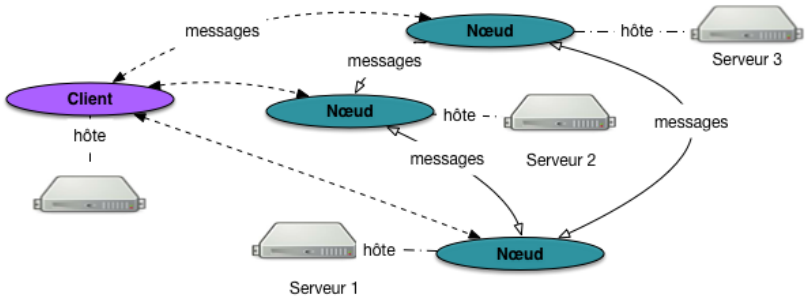
Deux organisations principales. La première (plus courante) est dite **maître-esclave**.



Rôle essentiel du maître (mais aussi *single point of failure*).

Autre topologie

C'est la démocratie ! Il n'y a que des maîtres (*master-master*, ou aussi multi-nœuds).



Plus robuste en théorie, mais soulève des problèmes compliqués (donc, moins courant).

Gestion des pannes, ou *failover*

Un système distribué peut s'appuyer des centaines, des milliers de serveurs : **il y a des pannes tout le temps**.

Caractéristique d'un bon système : tolérant aux pannes, assure une disponibilité 24/7.

- surveillance automatisée de tous les nœuds participants (*heartbeat*, messages toutes les n secondes).
- **méthode de reprise sur panne** (*failover*) basée sur la redondance/réplication.

Un cas particulièrement épineux : le **partitionnement réseau**.

Le NoSQL, qu'est-ce que c'est

Systèmes NoSQL (Ma définition)

Un système NoSQL est un système distribué dont la tâche est la gestion de données persistantes dans un environnement *cloud*. Il fournit les fonctionnalités suivantes :

- Recherche et mise à jour de données
- Adaptation automatique aux ressources matérielles : équilibrage, élasticité.
- Performances "scalables" (voir plus loin).
- Reprise sur panne automatisée.

Beaucoup de modèles différents ("documents", clé-valeur, graphes ...). Deux types principaux :

- Systèmes **temps réel** : accès en millisecondes aux unités d'information.
- Systèmes **analytiques** : traitements appliqués à tout ou partie d'une collection.

Petit historique : systèmes analytiques

Quelques publications de Google : *Google File System* (2003), MapReduce (2004), BigTable (2006).

Clonés par des systèmes Open Source dans l'environnement Hadoop : HDFS, MapReduce, HBase.

Autres successeurs : Cassandra, maintenant Spark, Flink, etc.

Applications : construction d'index, analyses de données massives.

Petit historique : systèmes temps réels

Publication majeure d'Amazon : le système Dynamo (2007).

Application : gestion robuste, distribuée, scalable de centaines de millions de comptes client, produits, transactions.

Cloné par le système Voldemort ; principes repris par Mongodb, CouchDB, Riak, beaucoup d'autres

NB : précédé par beaucoup de recherche et de systèmes distribués bien sûr.

Ce qu'il faut retenir

Environnement dit *cloud* = grappes de serveurs à bas coût, stockés dans des centres de données.

Très "élastique" car facile d'ajouter/supprimer une ressource = **scalabilité horizontale**.

Très sujet aux **pannes**, de disque, de réseau, de logiciels.

Système distribué (NoSQL) : système exploitant au mieux les ressources du *cloud*, tolérant aux pannes.