

Big Data, Hadoop, MapReduce, ...

J. Bigot - A. Richou

21 septembre 2017

1 Big Data

2 Hadoop

3 TP R

Table des matières

1 Big Data

2 Hadoop

3 TP R

Qu'est-ce que le Big Data ?

« Le big data, littéralement « grosses données », ou mégadonnées, parfois appelées données massives, désignent des ensembles de données qui deviennent tellement volumineux qu'ils en deviennent difficiles à travailler avec des outils classiques de gestion de base de données ou de gestion de l'information. » (Wikipédia)

Les Données massives se caractérisent par « les trois V » : Volume, Variété, Vélocité.

Volume :

- ce n'est pas juste un fichier « trop gros pour être ouvert sous excel » (stockage mémoire vive/stockage disque dur).
- données numériques créées dans le monde : 540 Millions SMS/jours (540Go/j -> 200To/a), 500 Millions de tweets/jours, 4,5 milliards de recherches Google/jours, 145 milliards de courriels échangés/jours.
- En 2013 : Twitter génère 7 téraoctets de données par jour, Facebook 10 téraoctets.
- Radiotélescope « Square Kilometre Array » (Australie) prévu pour 2020/2024 : 50 téraoctets de données analysées par jour, 7000 téraoctets de données brutes par secondes.
- Le Volume est une caractéristique relative (ce qui était trop gros hier peut être considéré comme acceptable aujourd'hui).

Vélocité :

- Fréquence à laquelle sont générées, capturées, partagées et mise à jour les données.
- Flux croissants de données doivent être analysés en temps réel. ex : bourse/information économique, comportement de l'utilisateur d'un site de commerce électronique.
- Importance de la scalabilité (passage à l'échelle, capacité à monter en charge) des installations informatiques et des méthodes de traitement informatique/statistique.

Variété :

- Les données sont brutes, semi-structurées voire non structurées. ex : données provenant du web, composées de texte, d'images, de vidéos, de métadonnées, de code informatique,...
- Besoin de nettoyage et de structuration pour le traitement des données.

Se pose le problème de la distribution des ressources informatiques.

- Partage des ressources de calcul (calcul distribué ou parallèle). Le but est de répartir un calcul sur plusieurs entités (coeurs de processeurs ou processeurs). Ces entités peuvent être sur un même ordinateur (ex : supercalculateur) ou sur plusieurs ordinateurs (cluster de calcul).
- Partage des ressources de stockage des données. Mémoire partagée/mémoire distribuée.

Dans le cas des Megadonnées, la solution retenue est très souvent un cluster de calcul avec une mémoire distribuée. Problématique très différente des codes de simulations numériques (météo par exemple).

Intérêt :

- Faire coopérer des machines simples peut être aussi puissant que développer un gros serveur,
- réduction des coûts, meilleure tolérance aux pannes (résilience), scalabilité (passage à l'échelle),
- réduire les temps de lecture/écriture des données

Problématique de stockage

Coût et vitesse d'écriture des disques d'un Tera en 2016 :

- HDD 50€ 1 To - 80 Mo/s
- SSD 250€ 1 To - 500 Mo/s

Coût d'un Po :

- HDD 50000 € - 144 jours pour écrire un disque
- SSD 250000 € - 23,1 jours pour écrire un disque

1 DD vs plusieurs DD :

- 1To - 3,5h d'écriture
- 10 * 100Go - 20 minutes d'écriture. Nb : le transfert réseau n'est pas limitant (10Gb/s).

Problématique de calcul

Temps de transfert d'un Po vers un HPC (High Performance Computer) :

- Débit réseau/HDD 10Gb/s
- Transfert d'un Po du DD au HPC : 27h

Idée pour réduire le temps de calcul :

- On découpe les données sur 10 DD de 100Go, sur 10 machines
- On traite les données directement sur les machines où sont stockées les données

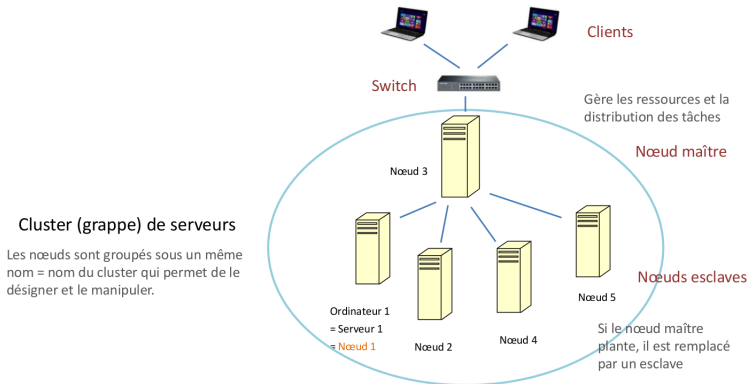


Figure – Source : programmation MapReduce sous R (R. Rakotomalala)

Table des matières

① Big Data

② Hadoop

③ TP R

Hadoop est un logiciel « open source » de la fondation Apache. C'est un environnement logiciel dédié au stockage et au traitement de larges volumes de données.

Hadoop repose sur deux composantes essentielles :

- Un système de fichiers distribué (**HDFS** : hadoop distributed file system) : permet de manipuler les données comme si elles étaient dans un seul fichier.
- Une implémentation efficace de l'algorithme **Mapreduce** : permet de traiter les données en parallélisant les calculs.

L'utilisateur gère le calcul et la mémoire comme s'il n'avait affaire qu'à un seul ordinateur.

Créé par Doug Cutting en 2004 en reprenant le concept MapReduce inventé par Google. utilisé par Yahoo, Facebook, Microsoft,...

Le format HDFS :

- Les données ne sont pas structurées et sont stockées sous forme de paires (clé,valeur) : un morceau de données correspond à une valeur et à chaque valeur est associée une clé.
- Des surcouches existent pour traiter des données structurées : Hbase (base de donnée orientée colonne), Hive/Pig langage pour interroger une base Hadoop avec une syntaxe de type SQL
- Les données sont découpées en blocs par le noeud maitre et sont réparties sur le cluster. La taille d'un bloc est de 64Mb ou 128Mb.
- HDFS est tolérant aux pannes : les blocs sont répliqués trois fois. Si un des éléments du cluster tombe en panne, le cluster s'adapte.
- La gestion de l'emplacement des différents blocs et de leurs répliques est géré par le NameNode. Tout cette gestion est automatisée et transparente pour l'utilisateur. C'est le NameNode qui répartie les taches de calcul en fonction des charges de travail de chaque noeuds de calcul (i.e. chaque partie du cluster).

Ecosystème Hadoop

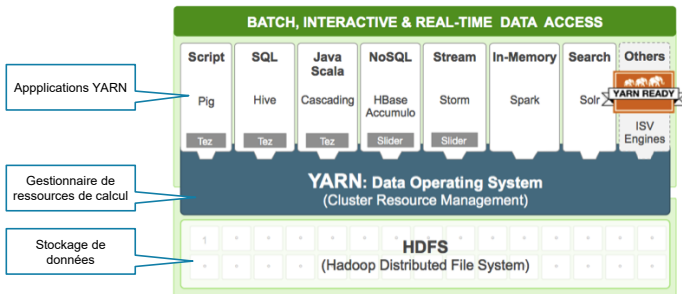


Figure – Source : Cours doctoral de D. Auber

Ecosystème Hadoop

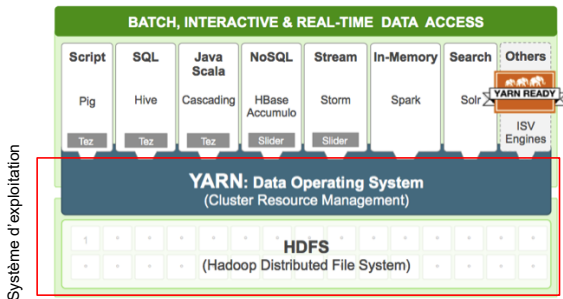


Figure – Source : Cours doctoral de D. Auber

Le paradigme MapReduce : modèle de programmation permettant de programmer simplement des calculs distribués dans une architecture Hadoop.

`mapreduce(input,map,reduce)`

- Les données (`input`) sont au format HDFS, elles sont donc découpées en blocs, dispatchées sur plusieurs machines.
- Sur chaque bloc est appliquée la fonction `map` qui prend en entrée des paires (`clé,valeur`) et renvoie des paires (`clé,valeur`).
- Les résultats sont regroupés par clés (`shuffle`) et répartis sur les différentes machines.
- Pour chaque groupe de données possédant la même clé est appelé la fonction `reduce()` qui prend en entrée des paires (`clé,valeur`) et renvoie des paires (`clé,valeur`).

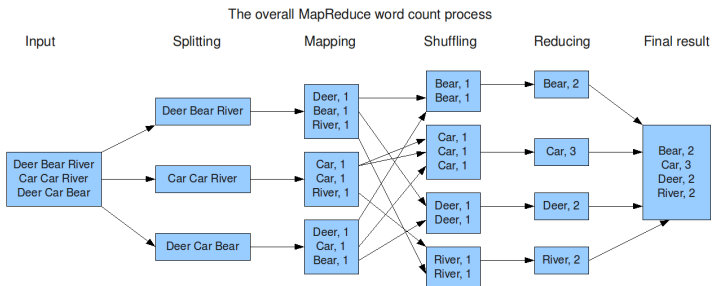


Figure – Exemple du comptage de mot (Source : <http://www.milanor.net/blog>)

Pour optimiser les calculs on peut ajouter une étape « Combine » qui permet d'appliquer des traitement du type `reduce()` à des sous parties triées des données. Attention, l'étape Compose n'est pas forcément utilisée par Hadoop (dépend de la situation du cluster, bande passante,...).

- Limitation du MapReduce : écriture entre chaque Map/Reduce.
- Avantage de Spark : utilisation de la mémoire-vive. Bibliothèques statistiques disponibles.

Table des matières

1 Big Data

2 Hadoop

3 TP R

TP R