David RIGAUX
Mehdi DALAA
Mandry MBUNDU
Maxime LUNDQUIST

Big Data

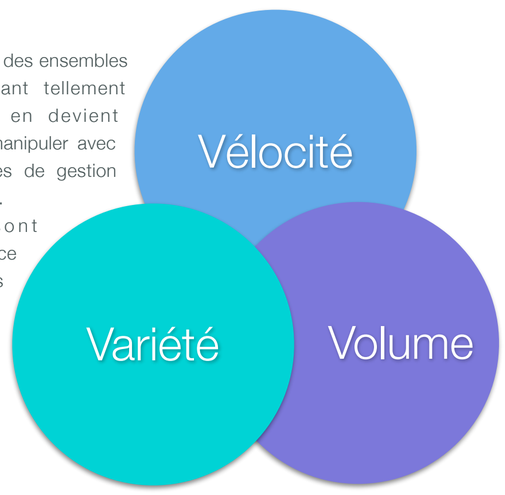
TIPE 2016-2017

Algorithmes PySpark pour traitement de données

Nous créons environ 2,5 trillions d'octets de données par jour. En 2 ans nous avons créé 90% des données présentes dans le monde. Ce nombre de données provient de l'avancée de toutes les technologies et aussi de nos habitudes usuelles telles que l'envoie de messages, mails ou vidéos. C'est pourquoi il en devient difficile de travailler sur ce volume de données avec les outils classiques de gestion de base de données.
De plus, le nombre de données à traiter ne cesse d'augmenter dû à nos habitudes de vie, c'est pourquoi la manipulation des données représente bel et bien un enjeu extrêmement important pour notre futur.

Le Big Data

Le Big Data désigne des ensembles de donnés devenant tellement volumineux qu'il en devient impossible de les manipuler avec des outils classiques de gestion de base de données.
Ces données sont caractérisées par ce qu'on appelle les "3V"



Qu'est ce que Spark ?

LE DÉVELOPPEMENT
Développé par AMPLab, de l'Université UC Berkeley et passé sous forme de projet Apache en 2010

HADOOP
Permet à des applications sur clusters Hadoop d'être exécutées jusqu'à 100 fois plus vite en mémoire et 10 fois plus vite sur disque

APACHE SPARK
Framework[1] de traitements Big Data open source[2] réalisé pour effectuer des analyses sophistiquées et conçu pour la rapidité et la facilité d'utilisation

LE FRAMEWORK
Spark propose un framework complet et unifié pour subvenir aux besoins de traitements Big Data pour divers jeux de données, divers par leur nature aussi bien que par le type de source

APPLICATIONS
Écrire rapidement des applications en Java, Scala ou Python et inclure plus de 80 opérateurs haut-niveau. Possibilité de l'utiliser de façon interactive pour requêter les données depuis un shell[3]. Spark supporte les requêtes SQL[4], le streaming de données et des fonctionnalités de machine learning et de traitements orientés graphe[5]

Les fonctionnalités de Spark

LANGAGE DE SPARK
Spark est écrit en Scala et s'exécute sur la Machine Virtuelle Java (JVM)[6]

AMÉLIORE MAPREDUCE
Apporte des améliorations à MapReduce[8] grâce à des étapes de shuffle moins coûteuses. Mais Spark n'en reste pas là et propose d'autres fonctions que Map et Reduce

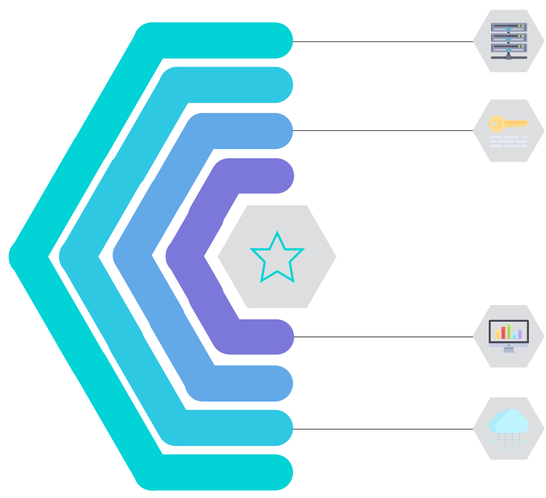
FLEXIBLE
Spark maintient les résultats intermédiaires en mémoire plutôt que sur disque. Mais le moteur d'exécution est conçu pour travailler aussi bien sur les deux.

CINQ LANGAGES
Spark propose une interface de programmation de haut-niveau pour une meilleure productivité (API[7] en Java, Scala et Python) et un shell interactif pour Scala et Python. Spark supporte également le Clojure et le R

EVALUATION PARESSEUSE
Spark supporte les évaluations paresseuses[9] ("lazy evaluation") des requêtes, ce qui aide à l'optimisation des étapes de traitement.

OPTIMISE LES GRAPHES
L'optimisation de graphes d'opérateurs arbitraires

L'écosystème de Spark



SPARK STREAMING
Peut être utilisé pour traitement temps-réel des données en flux.

SPARK SQL
Permet d'exposer les jeux de données Spark via API JDBC[10] et d'exécuter des requêtes de type SQL en utilisant les outils BI et de visualisation traditionnels.

SPARK MLLIB
Librairie de machine learning[11] contenant tous les algorithmes et utilitaires d'apprentissage classiques, comme la classification, la régression, le clustering[12], le filtrage collaboratif, la réduction de dimensions, en plus des primitives d'optimisation sous-jacentes.

SPARK GRAPHX
La nouvelle API pour les traitements de graphes et de parallélisation de graphes.

La méthode

PYSPARK
Pour réaliser ce projet nous allons utiliser l'API de Spark Python appelée PySpark. Cet API nous permet d'utiliser toute la puissance de Spark avec le langage de programmation Python.

ALGORITHMES
Le but est en fait d'utiliser PySpark pour pouvoir faire des algorithmes de traitement de données simple.

Conclusion

1. *Framework* : Ensemble d'outils constituant les fondations d'un logiciel informatique ou d'applications web, et destiné autant à faciliter le travail qu'à augmenter la productivité du programmateur qui l'utilisera.
2. *Open Source* : Permet de distribuer et d'utiliser gratuitement un logiciel, ainsi que de le modifier et de l'améliorer en donnant accès à son code source.
3. *Shell* : interface utilisateur d'un système d'exploitation destinée à lancer d'autres programmes et gérer leurs interactions.
4. *SQL* : Langage informatique normalisé servant à exploiter des bases de données relationnelles.
5. *Graphe* : ensemble de points nommes noeuds, sommets ou cellules reliés par un segment fléché ou non nommé arrête.
6. *JVM* : La machine virtuelle Java est un appareil informatique virtuel qui exécute des programmes compilés sous forme de bytecode Java.
7. *API* : (Application Programming Interface) interface de programmation qui permet de se " brancher " sur une application pour échanger des données.
8. *MapReduce* : Modèle de programmation massivement parallèle adapté au traitement de très grandes quantités de données.
9. *Évaluation paresseuse* : Technique où l'évaluation d'un paramètre de fonction ne se fait pas avant que les résultats de cette évaluation ne soient réellement nécessaires.
10. *JDBC* : (Java Database Connectivity) intergiciel qui permet à une application Java de manipuler plusieurs bases de données.
10.1 *Intergiciel* : logiciel servant d'intermédiaire de communication entre plusieurs applications, généralement complexes ou distribuées sur UN réseau informatique.
11. *Machine Learning* : Mise en place d'algorithmes en vue d'obtenir une analyse prédictive à partir de données dans un but précis.
12. *Cluster* : concentration géographique d'entreprises reliées ensemble, de fournisseur, et d'institutions associés dans un domaine particulier.

EISTI