

Soutenance de mémoire

Méthodes d'analyse des processus métier pour le choix
d'une architecture Big Data adaptée

Ludwig SIMON

3 Juillet 2019

Université Paris Nanterre
Tuteur : Mcf. Emmanuel HYON



1. Introduction
2. Architectures Big Data
3. Sélection
4. Perspectives d'évolution
5. Conclusion

Introduction

Objectif : Proposer une architecture Big Data correspondant à des besoins spécifiques.

Problématique : Domaine très vaste.

Contexte : Explosion du nombre de données à traiter/stocker

Questions :

- De quoi est constitué une architecture Big Data ?
- Quelles sont les architectures existantes ?
- Quels outils permettent de constituer une architecture Big Data ?

Architectures Big Data

Architecture générale

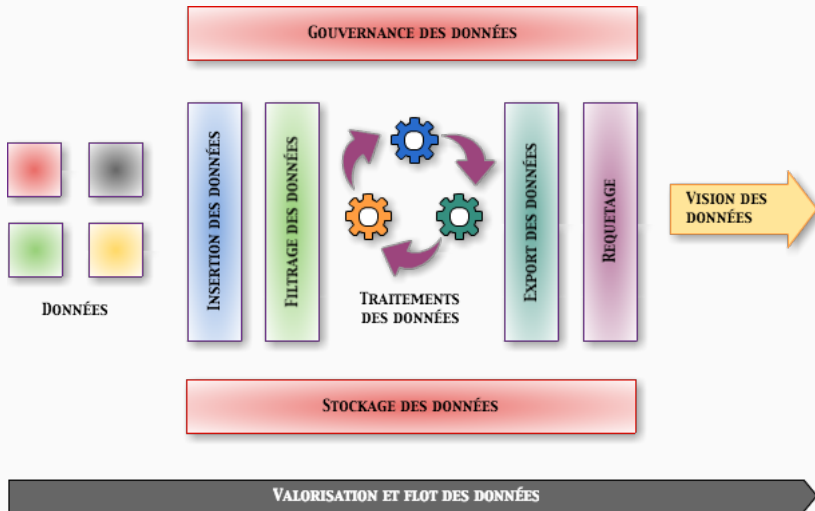


FIGURE 1 – Composants d’une architecture Big Data

Architecture Lambda

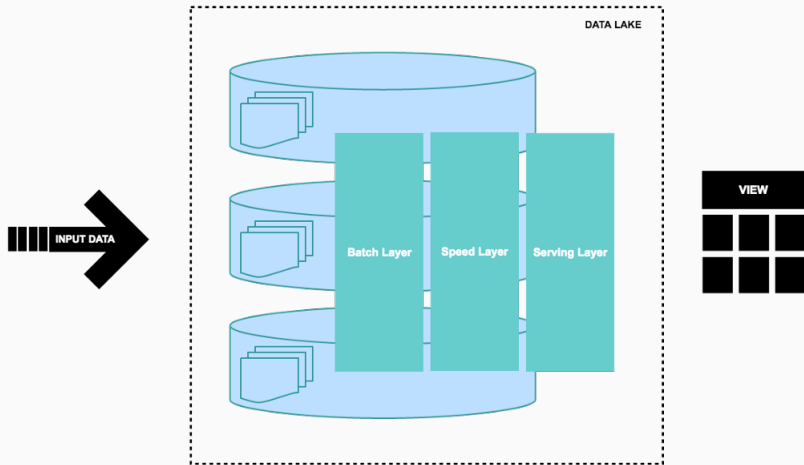


FIGURE 2 – Schéma de l'architecture Lambda

Architecture Kappa

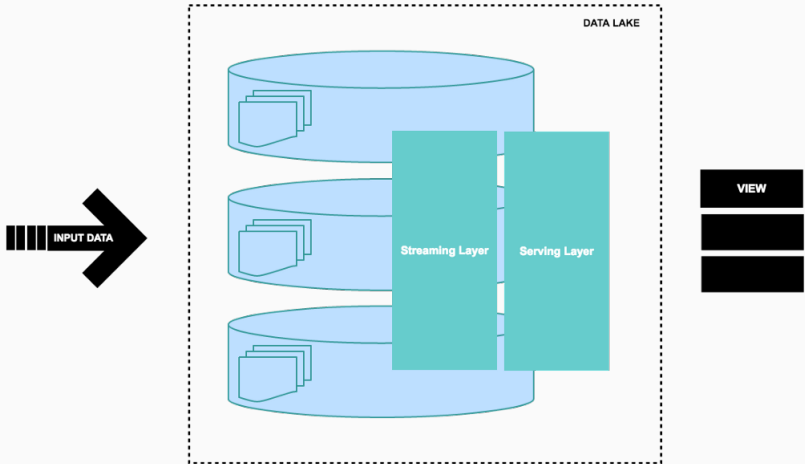


FIGURE 3 – Schéma de l'architecture Kappa

Sélection

Comment choisir ?

- Analyse des cas d'utilisation.
- Définition de critères de choix.
- L'approche de sélection en entreprise.

Critère	Architecture
Prédiction d'évènement entrant à l'aide de modèle d'apprentissage automatique	Lambda
Traitement des données en temps réel et par lots radicalement différents	lambda
Traitement des données par lots complexe	lambda
Très faible latence entre récupération et affichage des données	Kappa
Traitement des données par lots et en temps réel similaires	Kappa
Stockage permanent des données batch avant le traitement	Lambda/Kappa

TABLE 1 – Table des critères pour le choix de l'architecture

Critère	Solution
Garantir la consommation d'un message par un seul consommateur	RabbitMQ
Nécessité d'ingérer rapidement une grande quantité de messages	Kafka
Ordre des messages primordial	Kafka
Nécessité de conserver les messages à plus au moins long terme	Kafka
Utilisation de protocoles spécifiques (MQTT, AMQP, ...)	RabbitMQ / ActiveMQ
Règles de routage des messages complexe	RabbitMQ / ActiveMQ

TABLE 2 – Table des critères pour le choix du logiciel d'agent de messages

Critère	Solution
Manque de connaissances pour la réalisation de programmes personnalisés	ETL/ELT
Nécessité d'extraire de nombreuses sources de données	ETL/ELT
Peu de sources de données	Programme personnalisé
Nécessité d'avoir des performances élevées	Programme personnalisé

TABLE 3 – Table des critères pour le choix d'une solution complète ou d'un programme personnalisé pour l'ingestion des données.

Critère	Solution
Nécessité d'utiliser des librairies autres que l'apprentissage automatique	Spark
Nécessité d'avoir des performances accrues	Spark
Nécessité d'avoir une tolérance à la panne exemplaire	MapReduce
Les performances ne sont pas la priorité	MapReduce

TABLE 4 – Table des critères pour le choix de la solution de traitement par lots.

Critère	Solution
Source de données en micro batch	Spark Streaming
Source de données en streaming	apaches Storm

TABLE 5 – Table des critères pour le choix de la solution de traitement en temps réel.

- Clés/Valeurs.
- Grandes colonnes.
- Séries temporelles.
- Orientée graphe.
- Orientée documents.
- Moteur d'indexation.

Critère	Solution
Monitoring	Grafana
Visualisation complexe	D3.js
Visualisation simple	Kibana/Solr

TABLE 6 – Table des critères pour le choix de la solution de visualisation et analyse des données.

Critère	Solution
Nécessité d'effectuer une action spécifique en cas d'erreur	Apache Oozie
Nécessité de lancer une succession de tâche	Apache Oozie
Exécution de tâche simple	Cron

TABLE 7 – Table des critères pour le choix de la d'orchestration.

Perspectives d'évolution

- Benchmarks des solutions logicielles
- Augmenter le nombre de critères/solutions logicielles
- Évaluer le résultat

Conclusion

- Domaine très vaste.
- Enrichir les critères de choix.
- Enrichir le nombre de solution logicielle traitée.
- Évaluer le résultat obtenu.