# 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

# Sommaire

# Introduction

# 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 constitué une architecture Big Data ?

# Architectures Big Data

# Architecture générale

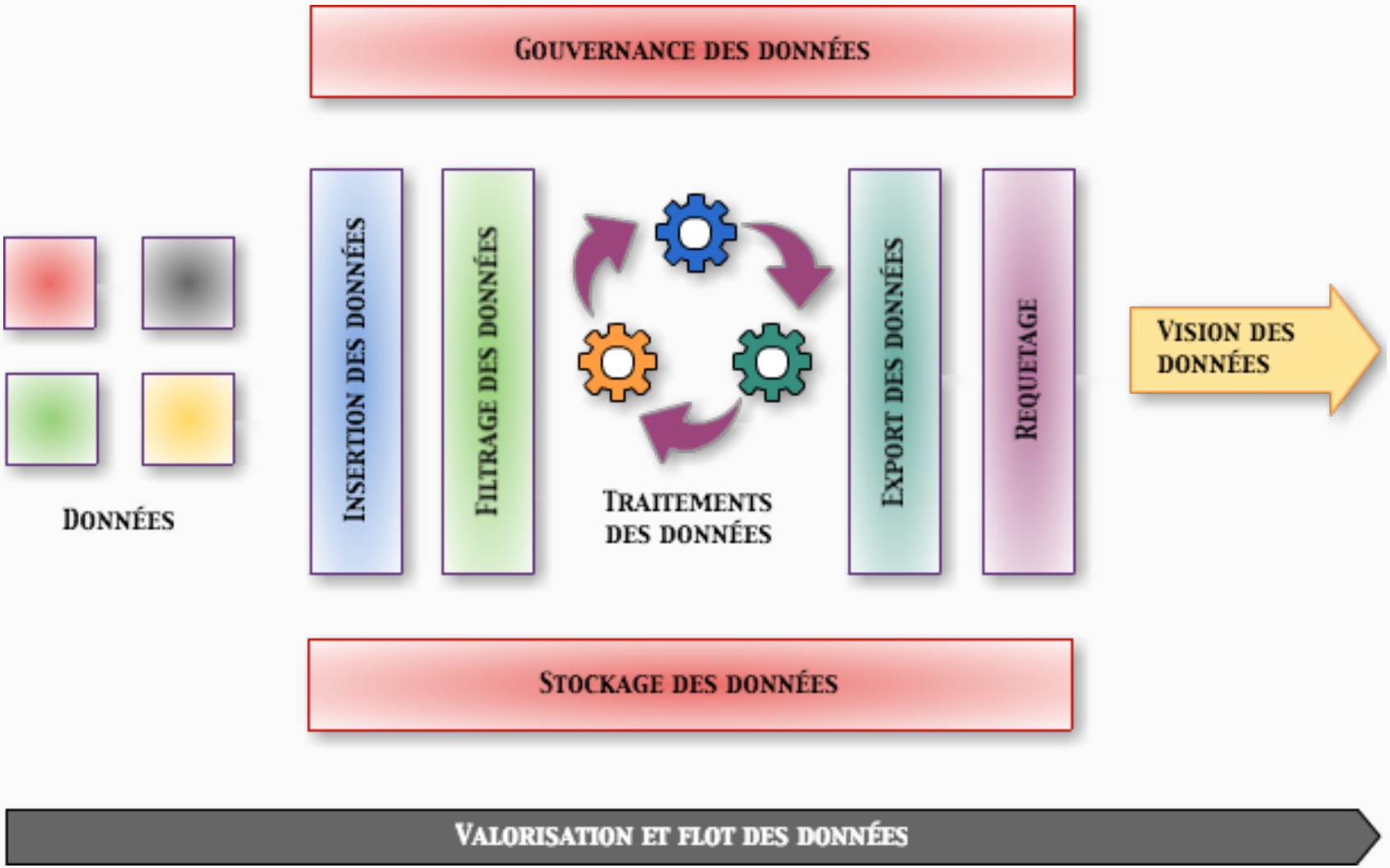


FIGURE 1 – Composants d'une architecture Big Data

# Architecture Lambda

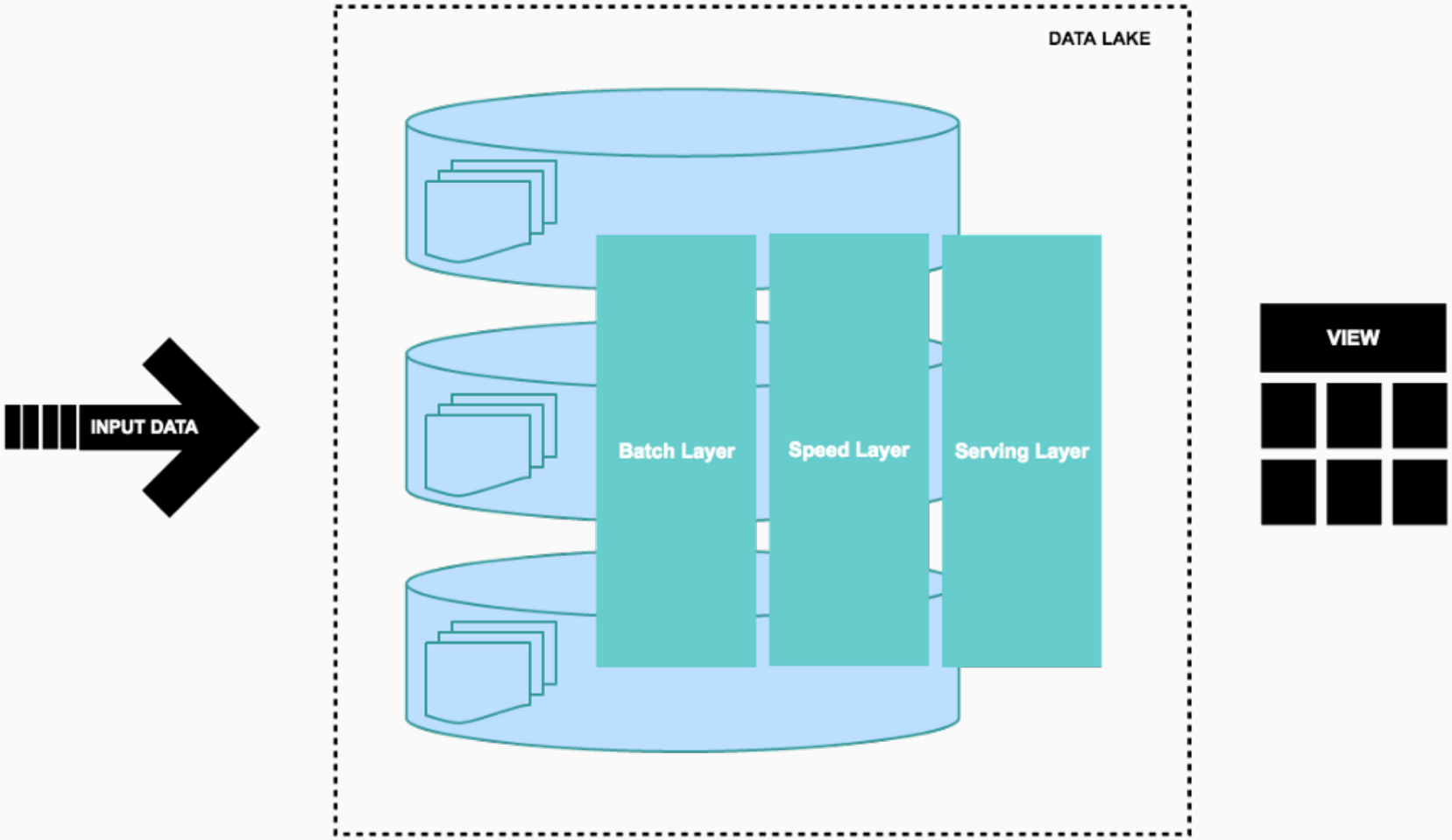


FIGURE 2 – Schéma de l'architecture Lambda

# Architecture Kappa

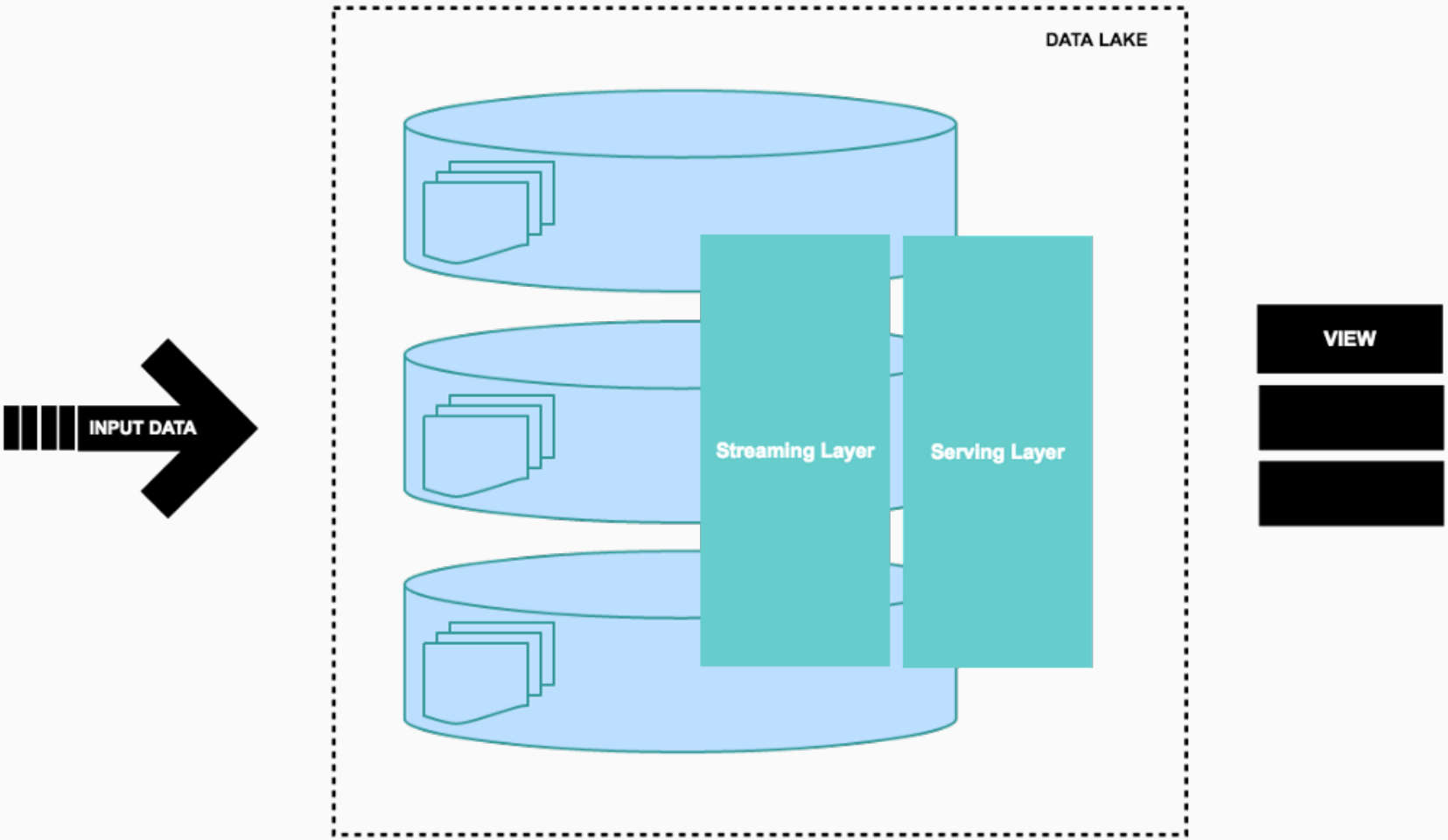


FIGURE 3 – Schéma de l'architecture Kappa

# Sélection

# Comment choisir ?

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

# Architecture

| 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

# Message Broker

| 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

# Ingestion des données

| 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.

# Traitement des données : Batch

| 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.

# Traitement des données : Temps réel

| Critère | Solution |
|---|---|
| Source de données en micro batch | Spark Streaming |
| Source de données en streaming | Apache Storm |

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

# Stockage des données

# Visualisation et analyse des données

| Critère | Solution |
|---|---|
| Monitoring | Grafana |
| Visualisation complexes | D3.js |
| Visualisation simples | Kibana/Solr |

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

# Orchestration 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 évolution

# Perspectives évolution

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

# Conclusion

# Conclusion

- Domaine très vaste.
- Enrichir les critères de choix.
- Enrichir le nombres de solution traitées.
- Évaluer le résultat obtenu.