

# Big Data – Enjeux et Perspectives

# Sommaire

1. Introduction au Big Data
2. Enjeux Stratégiques et Métier
3. Architecture et Technologies du Big Data
4. Enjeux Organisationnels et Gouvernance
5. Big Data & Intelligence Artificielle
6. Perspectives et Tendances
7. Enjeux futurs et défis

# Introduction au Big Data

# Introduction au Big Data

## 1 Définition et les 5V du Big Data

Le Big Data se définit par un ensemble de caractéristiques qui dépassent les capacités traditionnelles des systèmes d'information classiques. Historiquement, on parle des **3V** (Volume, Vélocité, Variété), mais la définition a été enrichie à **5V** (et parfois même 7V, 10V dans certaines littératures).

# Introduction au Big Data

## 1 Définition et les 5V du Big Data

### ◆ Les 5V en détail

#### 1. Volume

- Explosion de la quantité de données produites : données transactionnelles, logs systèmes, réseaux sociaux, IoT, capteurs, images, vidéos, données médicales...
- On parle de **pétaoctets à exaoctets** (et même zettaoctets à l'échelle mondiale).
- Exemple : Facebook génère plusieurs pétaoctets de données chaque jour ; les voitures autonomes produisent jusqu'à **4 To de données/jour/voiture**.

# Introduction au Big Data

## 1 Définition et les 5V du Big Data

### ◆ Les 5V en détail

#### 2. Vélocité

- Capacité à traiter les données en **temps réel ou quasi temps réel**.
- Flux continu (streaming) issus de capteurs IoT, transactions bancaires, interactions clients.
- Exemple : détection de fraude bancaire doit se faire en **millisecondes**.

# Introduction au Big Data

## 1 Définition et les 5V du Big Data

### ◆ Les 5V en détail

#### 3. Variété

- Multiplicité des formats :
  - Structurées (SQL, tableaux relationnels)
  - Semi-structurées (JSON, XML, logs, CSV)
  - Non structurées (vidéos, images, audio, mails, réseaux sociaux).
- Plus de **80 % des données mondiales** sont **non structurées**.

# Introduction au Big Data

## 1 Définition et les 5V du Big Data

### ◆ Les 5V en détail

#### 4. Véracité

- Fiabilité et qualité des données : données bruitées, incomplètes, biaisées.
- Risques : fake news, données corrompues, erreurs de saisie.
- Importance des techniques de **Data Cleaning, Data Governance, Data Lineage**.



# Introduction au Big Data

## 1 Définition et les 5V du Big Data

### ◆ Les 5V en détail

#### 5. Valeur

- Le but du Big Data : **transformer la donnée en valeur ajoutée.**
- Business value : optimiser coûts, améliorer l'expérience client, créer de nouveaux services, innover.
- Exemple : Netflix utilise le Big Data pour personnaliser ses recommandations → valeur directe en fidélisation et revenus.

# Introduction au Big Data

## 2 Historique : de l'entrepôt de données au Lakehouse

### ◆ Années 1980 – 1990 : Data Warehouse (Entrepôt de données)

- Vision : centraliser et consolider les données de l'entreprise pour le reporting et l'analyse décisionnelle.
- Architecture : ETL (Extract – Transform – Load) → Data Warehouse (schéma rigide) → Outils BI (Tableau, Business Objects, Cognos).
- Limite : données uniquement **structurées**, intégration lente, pas adapté aux données massives et variées.

# Introduction au Big Data

## 2 Historique : de l'entrepôt de données au Lakehouse

### ◆ **Années 2000 : Explosion du Web et des données**

- Web 2.0, réseaux sociaux, vidéos, e-commerce → croissance exponentielle.
- Google, Yahoo et Amazon développent de nouvelles approches (MapReduce, GFS, DynamoDB).
- Hadoop (2005-2008) popularise le stockage distribué et le traitement parallèle.

# Introduction au Big Data

## 2 Historique : de l'entrepôt de données au Lakehouse

### ◆ **Années 2010 : Data Lake**

- Concept introduit par James Dixon (CTO Pentaho).
- Stockage brut et massif de données **structurées, semi-structurées et non structurées** dans un même réservoir.
- Principe : “**Schema on Read**” (le schéma est appliqué au moment de la lecture et non de l'écriture).
- Avantage : flexibilité, coût faible avec le stockage objet (S3, Azure Blob, HDFS).

# Introduction au Big Data

## 2 Historique : de l'entrepôt de données au Lakehouse

### ◆ Années 2020 : Lakehouse

- Fusion des avantages du **Data Warehouse** et du **Data Lake**.
- Supporte à la fois :
  - Données structurées (requêtes SQL, BI)
  - Données semi/non structurées (logs, JSON, multimédia)
- Exemple de technologies : **Delta Lake (Databricks)**, **Snowflake**, **Apache Iceberg**.
- Concept clé : **unifier BI + IA/ML** sur une même plateforme.

# Introduction au Big Data

## 3 Comparaison : Data Warehouse vs Data Lake vs Data Mesh

### ◆ Data Warehouse (Entrepôt de données)

- **Approche** : centralisée, structurée.
- **Schéma** : *Schema on Write* (schéma défini avant intégration).
- **Cas d'usage** : reporting, BI, analyses historiques.
- **Exemples** : Teradata, Oracle, SQL Server, SAP BW, Google BigQuery.
- **Limites** : coûteux, rigide, pas adapté aux données massives et non structurées.

# Introduction au Big Data

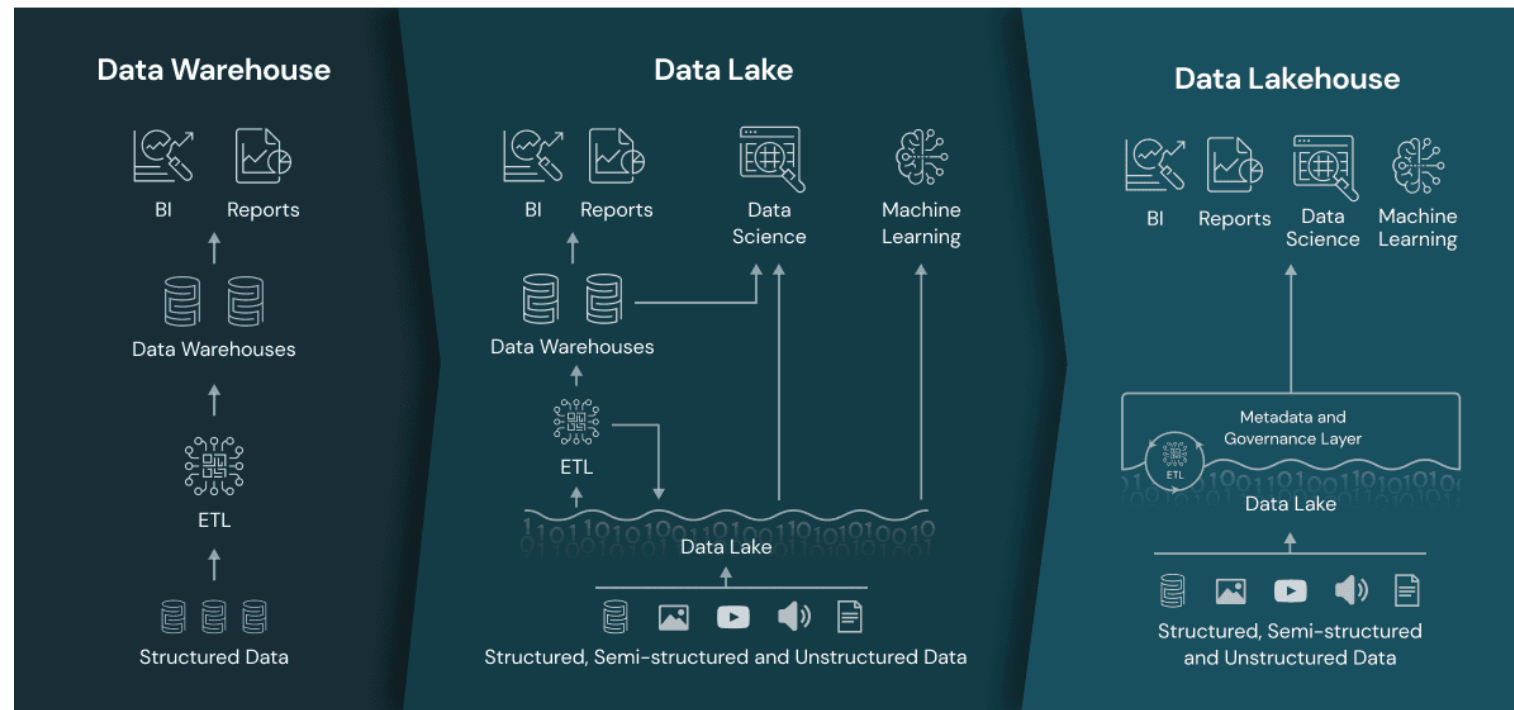
## 3 Comparaison : Data Warehouse vs Data Lake vs Data Mesh

### ◆ Data Lake

- **Approche** : stockage brut et massif.
- **Schéma** : *Schema on Read*.
- **Cas d'usage** : exploration, machine learning, IA, stockage universel.
- **Exemples** : Hadoop HDFS, AWS S3, Azure Data Lake Storage, GCP Cloud Storage.
- **Limites** : gouvernance difficile, risque de Data Swamp.

# Introduction au Big Data

## 3 Comparaison : Data Warehouse vs Data Lake vs Data Mesh





# Introduction au Big Data

## 3 Comparaison : Data Warehouse vs Data Lake vs Data Mesh

### ◆ Data Mesh (nouvelle génération, années 2020)

- **Approche** : décentralisée et orientée produit.
- Chaque domaine (marketing, finance, supply chain...) gère **sa propre donnée comme un produit**.
- S'appuie sur des **principes d'architecture distribuée** et des équipes multidisciplinaires.
- **Concepts clés** :
  - Domain-driven design appliqué à la donnée
  - Data as a Product
  - Self-serve data platform (plateformes en libre-service)
  - Gouvernance fédérée (politiques globales + autonomie locale).
- **Cas d'usage** : grandes organisations multi-domaines cherchant à éviter le "monolithe data".
- **Exemples d'implémentation** : sur Databricks, Snowflake, Azure Synapse + outils de gouvernance type Collibra, Alation.

# Introduction au Big Data

## 3 Comparaison : Data Warehouse vs Data Lake vs Data Mesh

Aspect	Data Warehouse	Data Lake	Data Mesh
Architecture	Centralisée	Centralisée (stockage brut)	Décentralisée (par domaine)
Schéma	Schema on Write	Schema on Read	Variable (selon domaine)
Types de données	Structurées	Tous types (structurées, semi, non)	Tous types
Cas d'usage	BI, reporting	IA, ML, analyse exploratoire	Gouvernance distribuée, scalabilité
Exemples	Oracle, BigQuery, Redshift	Hadoop, S3, Azure Data Lake	Snowflake + gouvernance distribuée

# Enjeux Stratégiques et Métier

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique (innovation, compétitivité, optimisation)

### 1) Trois leviers de valeur (cadre rapide à réutiliser)

- **Innovation (croissance)** : nouveaux produits & services data-driven, modèles économiques (data sharing/marketplaces), expériences augmentées (perso, temps réel), “platformisation”.
- **Compétitivité (différenciation)** : time-to-insight réduit, boucles d'apprentissage (A/B, bandits), effet réseau (plus d'utilisateurs = meilleure donnée = meilleur produit), barrières à l'entrée (coûts d'entraînement, données propriétaires).
- **Optimisation (efficience)** : automatisation analytique/IA, prévision & allocation d'actifs, qualité opérationnelle (OEE, SLA), réduction des risques & pertes (fraude, défauts), compliance by design.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 1) Trois leviers de valeur (cadre rapide à réutiliser)

#### Chaîne de valeur data (mémo)

**Ingestion → Stockage → Gouvernance → Traitements batch/stream → Feature store → Modèles/IA → Serving (API/BI) → Observabilité (DataOps/MLOps) → ROI/impact.**

#### Indicateurs d'impact (à suivre systématiquement)

- **Innovation** : % CA de nouveaux produits data, NPS/CES, conversion nouvelle offre, temps idée → MVP.
- **Compétitivité** : part de marché, coût d'acquisition (CAC), vitesse de déploiement (lead time), fréquence de release.
- **Optimisation** : coût unitaire, OEE, délai moyen (lead time), taux d'erreur/retour, MTBF/MTTR, pertes évitées.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### A. Santé (médecine personnalisée, génomique)

##### Objectifs stratégiques

- Passer du “moyen” au **personnalisé** (protocoles, dosage, suivi).
- **Prédire** les risques (sepsis, réadmission 30j), optimiser le **parcours patient**.
- Accélérer la **recherche clinique** (cohorte/éligibilité, RWE).

##### Données & flux

- Dossiers patients (EHR/EMR), HL7/FHIR, imagerie (DICOM), notes cliniques (NLP), **omiques** (génomique, transcriptomique), dispositifs connectés (IoMT), pharmacovigilance.
- Contraintes : **RGPD**, consentement, pseudonymisation, minimisation, traçabilité d'accès.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### A. Santé (médecine personnalisée, génomique)

##### Cas d'usage phares

- **Score de risque** (réadmission, sepsis) : modèles survie, gradient boosting, deep tabular.
- **NLP clinique** (extraction concepts SNOMED/ICD), **résumés** automatiques.
- **Imagerie** (détection lésions) : CNN/transformers.
- **Médecine personnalisée** : variants génétiques → protocole (règles + ML).
- **Planification hôpital** : prévisions lit/OR, optimisation bloc.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### A. Santé (médecine personnalisée, génomique)

##### KPIs & ROI

- -X% réadmissions, -X h délai diagnostic, +X% inclusion essais, conformité audit FHIR/consent.
- Architecture de référence : **ingestion FHIR + data lake chiffré + Delta/Iceberg + NLP/vision + MLOps (drift clinique).**

##### Risques & contrôles

- Biais (population non représentée), explicabilité (SHAP/LIME), **fédéré** (FL) pour données sensibles, gouvernance d'accès clinicien/chercheur.



# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### B. Banque / Finance (fraude, scoring)

##### Objectifs

- **Réduire pertes** (fraude, défaut), **accélérer** on-boarding (KYC), **personnaliser** l'offre (next-best-action), respecter AML/PSD2/Bâle.

##### Données

- Transactions cartes & virement, KYC/CRS, logs canaux, open banking (API PSD2), graphes relations (comptes, bénéficiaires), données externes (bureaux de crédit).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### B. Banque / Finance (fraude, scoring)

##### Cas d'usage

- **Fraude temps réel** (streaming Kafka/Flink, features glissantes) + **graph analytics** (communautés/chemins).
- **Scoring crédit** (GBMs/autoML, fairness), **limites dynamiques**.
- **AML** (détection schémas, réseau), **priorisation alertes**.
- **Segmentation valeur/CLV**, **prix dynamique**.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### B. Banque / Finance (fraude, scoring)

##### KPIs

- Capture de fraude (recall) ↑, **false positives** ↓, délai décision carte < 50 ms, coût conformité ↓, temps KYC ↓.

##### Architecture

- **Event streaming** + feature store **on-line/off-line**, modèles servés en **low latency**, auditabilité (versioning jeux/poids), **explanations**.

##### Risques

- Biais discriminatoires (genre, origine), détournements adversariaux, gouvernance modèle (validation indépendante), régulation (EBA/ACPR).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### C. Industrie 4.0 (IoT, maintenance prédictive)

##### Objectifs

- **OEE** ↑, arrêts non planifiés ↓, **rendement matière** ↑, énergie ↓, fiabilité & sécurité.

##### Données

- Capteurs temps réel (vibration, température), SCADA/MES/ERP, **OPC-UA**, MQTT, logs qualité, vision (défauts).

##### Cas d'usage

- **Maintenance prédictive** (prognostics & health management) : séries temporelles, survival analysis, remaining useful life.
- **Détection défauts** (vision), **process mining**, **jumeau numérique** (digital twin).
- **Optimisation énergie** (prévision charge, arbitrage).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### C. Industrie 4.0 (IoT, maintenance prédictive)

##### KPIs

- MTBF↑, MTTR↓, arrêts ↓, scrap ↓, consommation kWh/u ↓, TRS/OEE ↑.

##### Architecture

- **Edge computing** (prétraitement), bus temps réel, **data lake/lakehouse** industriel, MLOps avec surveillance **drift sensoriel**, boucle vers **MES/PLC**.

##### Risques

- Sécurité OT (segmentation réseau), intégration legacy, latence, cybersécurité (IEC 62443).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### D. Smart Cities (trafic, énergie, urbanisme)

##### Objectifs

- Fluidité mobilité, **sobriété énergétique**, qualité de l'air, sécurité, services proactifs.

##### Données

- Boucles inductives, caméras (vision/edge), IoT environnement, smart meters, **open data**, mobile traces, météo.

##### Cas d'usage

- **Contrôle trafic** dynamique (coordination feux, prédiction flux).
- **Gestion énergie** (prévision consommation/production renouvelable, effacement).
- **Sécurité urbaine** (anomalies vidéo respect privacy).
- **Urbanisme** data-driven (jumeau numérique territoire).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### D. Smart Cities (trafic, énergie, urbanisme)

##### KPIs

- Temps trajet ↓, émissions NO<sub>x</sub>/CO<sub>2</sub> ↓, pics énergie aplatis, satisfaction citoyens ↑.

##### Architecture

- **Edge + stream** (latence), data mesh inter-directions (mobilité/énergie/propreté), gouvernance éthique (privacy zone-based, floutage on-device).

##### Risques

- Vie privée/surveillance, dépendance fournisseurs, interopérabilité (standards, API publiques).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### E. Retail & e-commerce (reco, personnalisation)

##### Objectifs

- **Conversion** ↑, panier moyen ↑, **stock** optimisé, churn ↓, marges ↑.

##### Données

- Clickstream, paniers, catalogue/attributs, CRM, campagnes, avis, logistique, prix concurrents.



# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### E. Retail & e-commerce (reco, personnalisation)

##### Cas d'usage

- **Recommandations** (collaboratif, content-based, session-based, graph).
- **Personnalisation** omnicanale (ranking, bandits contextuels).
- **Forecast** demande (saisonnalité, promos), **prix dynamique**, **churn**.
- **Attribution** multicanale, **optimisation promo**.

##### KPIs

- CTR/CR ↑, AOV ↑, rupture ↓, démarque inconnue ↓, LTV/CAC ↑, uplift promo ↑.

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 2) Secteurs impactés – modules “prêts à déployer”

#### E. Retail & e-commerce (reco, personnalisation)

##### Architecture

- Event streaming, **feature store**, moteurs de **ranking** online, A/B/n, MLOps (shadow/canary), BI temps réel.

##### Risques

- “Filter bubble”, fairness (exposition vendeurs), conformité cookies/consent, robustesse aux attaques (scraping, fraude avis).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 3) Modèles d'architecture réutilisables (templates)

#### Pipeline générique “temps réel + batch”

- **Ingestion** : Kafka/Flink (events), CDC/ELT (batch).
- **Stockage** : Data Lake(Lakehouse) + Warehouse (serving BI).
- **Traitement** : Spark/Flink + orchestrateur (Airflow/DBX).
- **Gouvernance** : catalogue (lineage, PII), politiques d'accès (ABAC).
- **ML** : feature store (off-line/on-line), registry modèles, MLOps (CI/CD, drift, canary).
- **Serving** : APIs faibles latences, dashboards, reverse ETL.
- **Observabilité** : test données (DQ), SLO pipelines, coût FinOps (tagging).

# Enjeux Stratégiques et Métier

## Big Data est un levier stratégique

### 4) Méthode d'identification & priorisation des cas d'usage (prête à l'emploi)

- **Backlog:** pour chaque idée → *Impact business* (€, KPI), *Faisabilité* (données, technique, change), *Dépendances*, *Risques* (juridiques/éthiques).
- **Scoring** (1-5) et matrice **Impact** × **Faisabilité** → quick wins (T0-T3 mois), core bets (T3-T12), explorations (>12).
- **Business case:** Bénéfices annuels – (Data/Cloud/Opex + Change) → **ROI & Payback**; définir **North Star Metric** et **leading indicators**.

# Enjeux Stratégiques et Métier

## Impact sur le métier du Chief Data Officer (CDO)

### 1) Mandat & périmètre

- **Vision & stratégie data** alignée aux OKR d'entreprise (offensive vs défensive).
- **Gouvernance** (comités, politiques PII/RGPD, qualité, access management).
- **Plateformes & outillage** (data platform, lakehouse, catalogue, MLOps, sécurité).
- **Data as a Product** : instaurer le **data mesh** (si pertinent), définir standards cross-domain.
- **Monétisation & partenariats** : data sharing, marketplaces, API produits.
- **Éthique & conformité** : privacy by design, fairness, auditabilité modèles.

# Enjeux Stratégiques et Métier

## Impact sur le métier du Chief Data Officer (CDO)

### 2) Operating model (organisation)

- **Hub & Spoke** : une équipe centrale (standards, plateforme, sécurité, MLOps) + équipes **domaines** (produits data).
- Rôles clés : **Data Product Owner**, Data Engineer, ML Engineer, Data Scientist, Analytics Engineer, Steward, Architecte Data/Sécu.
- **Compétences** : cloud/data, ML, produit, change management, finance/FinOps, juridique (RGPD/contrats).

# Enjeux Stratégiques et Métier

## Impact sur le métier du Chief Data Officer (CDO)

### 3) Cadre de gouvernance

- **Data Contracts** entre domaines, SLA/SLO (fraîcheur, complétude, disponibilité).
- **Catalogue & lineage** obligatoires (découvrabilité).
- **Qualité** : tests auto (Great Expectations/DBT tests), seuils d'alerte.
- **Sécurité** : classification PII/PHI, chiffrement, masquage dynamique, **least privilege**.
- **Modèles** : registre, validation indépendante, **model risk management**, traçabilité des features/versions.

# Enjeux Stratégiques et Métier

## Impact sur le métier du Chief Data Officer (CDO)

### 4) Feuille de route “100 jours” (exécutable)

1. **Semaine 1-2** : audit de maturité (données, équipes, outils, use-cases), cartographie sources/PII, risques.
2. **Sem. 3-4** : définir **North Star** + 5 KPIs d'entreprise; charte data; comité de gouvernance.
3. **Sem. 5-8** : cadrer 3 **quick wins** (1 par levier : innovation/compétitivité/optimisation) avec business case, data contract, plan MLOps.
4. **Sem. 9-12** : lancer plateforme minimale (ingestion, lakehouse, catalogue, CI/CD data), déployer **1 modèle** en production (pilot), mettre en place observabilité & coûts (FinOps).
5. **Livrables** : standard de nommage, modèles d'ABAC, modèle de data product, RACI & processus d'accès.



# Enjeux Stratégiques et Métier

## Impact sur le métier du Chief Data Officer (CDO)

### 5) KPIs du CDO (tableau de bord)

- **Adoption** : nb utilisateurs actifs (BI/DS), % domaines publiant des data products.
- **Time-to-Value** : délai idée → MVP, cycles A/B.
- **Qualité** : incidents DQ/mois, % datasets avec SLA respecté.
- **Économie** : coût/Go, coût par requête, waste cloud ↓, ROI use-cases.
- **Conformité** : audits réussis, temps de réponse aux demandes RGPD (AIPD, droit d'accès).

# Architecture Big Data

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 1 Principes fondamentaux

- **Scalabilité horizontale** : ajout de nœuds/serveurs standards plutôt que des machines très coûteuses.
- **Tolérance aux pannes** : réplication des données → disponibilité garantie.
- **Traitement distribué** : les calculs sont envoyés là où se trouvent les données.
- **Polyglot persistence** : utiliser différents types de bases selon les cas (SQL, NoSQL, graph, colonnes...).
- **Temps réel + batch** : combiner analyse historique et analyse instantanée.

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 1. Sources de données

- **Internes** : ERP, CRM, logs serveurs, transactions bancaires, données capteurs (IoT).
- **Externes** : open data, réseaux sociaux, données partenaires, APIs, web scraping.
- **Caractéristiques** : hétérogènes, massives, flux continu + historiques.

#### ◆ 2. Ingestion des données

- Objectif : **collecter et transporter** les données vers le système central.
- **Technologies** :
  - **Batch** : ingestion périodique → Sqoop, Talend, Informatica.
  - **Streaming (temps réel)** : Kafka, Flume, NiFi, Pulsar.
- **Exemple** : Kafka collecte en temps réel des millions de clics d'utilisateurs sur un site e-commerce.

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 3. Stockage distribué

- Objectif : stocker de manière fiable et scalable.
- **Data Lake** : stockage brut (S3, ADLS, HDFS, GCS).
- **Bases NoSQL** : MongoDB (documents), Cassandra (colonnes), Neo4j (graphes).
- **Formats optimisés** : Parquet, ORC, Avro (compacts, colonnes → rapides pour analyse).
- **Exemple** : un Data Lake S3 hébergeant vidéos, logs JSON, fichiers CSV et données IoT.

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 4. Traitement des données

Deux modes complémentaires :

##### 1. Batch (traitement différé)

- Outils : Hadoop MapReduce, Spark.
- Cas : analyses historiques, calculs lourds (fraude sur 1 an de transactions).

##### 2. Streaming (temps réel)

- Outils : Spark Streaming, Flink, Storm.
- Cas : détection de fraude en direct, suivi trafic urbain, monitoring IoT.

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 5. Gouvernance, qualité et sécurité

- **Data Catalog** : Alation, Collibra (inventaire des données).
- **Qualité** : Data Cleaning, lineage (traçabilité).
- **Sécurité** : chiffrement, contrôle d'accès (Kerberos, Ranger), anonymisation (RGPD).
- **Exemple** : masque des données sensibles avant de les partager avec les équipes marketing.

# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 6. Stockage analytique & requêtage

- **Entrepôts de données (Data Warehouse)** : Redshift, BigQuery, Synapse.
- **Data Lakehouse** : Delta Lake (Databricks), Snowflake, Iceberg.
- **Requêtage SQL massivement parallèle** : Presto, Trino, Hive.
- **Exemple** : analyste exécutant une requête SQL sur 10 ans de données clients (en quelques secondes via BigQuery).



# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 7. Machine Learning & Intelligence Artificielle

- **Moteurs ML** : MLlib (Spark), TensorFlow, PyTorch, Scikit-Learn.
- **MLOps** : Kubeflow, MLflow pour industrialiser les modèles (monitoring, versionning).
- **Cas** : recommandation Netflix, maintenance prédictive, NLP pour assistants virtuels.

# Architecture et Technologies du Big Data

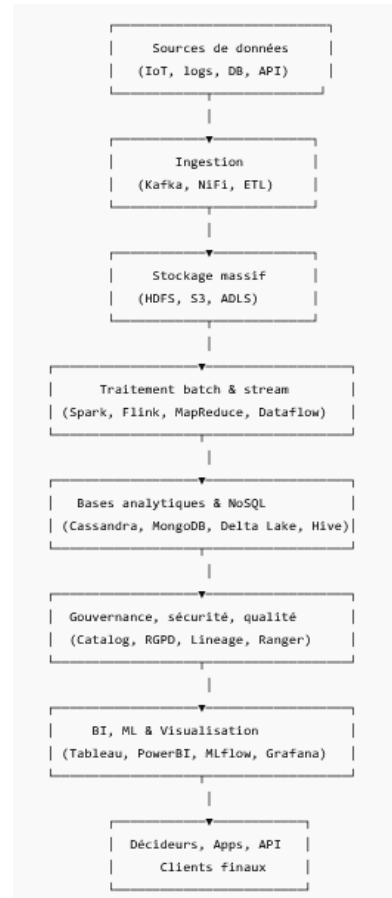
## L'Architecture du Big Data

### 2 Les différentes couches de l'architecture Big Data

#### ◆ 8. Restitution & visualisation

- **BI / Analytics** : Tableau, Power BI, Qlik Sense.
- **Dashboards temps réel** : Grafana, Kibana.
- **APIs** : exposer les données aux applications (REST/GraphQL).
- **Exemple** : tableau de bord affichant en temps réel le volume de commandes par région.

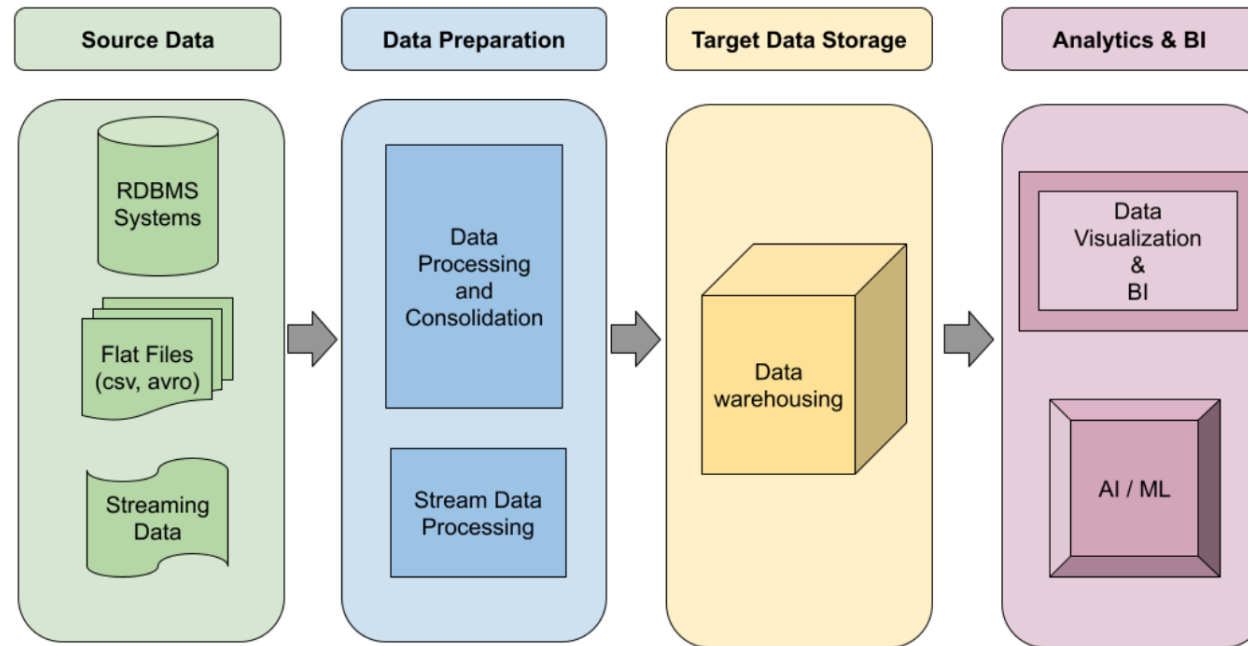
### 3 Schéma global d'une architecture Big Data moderne



# Architecture et Technologies du Big Data

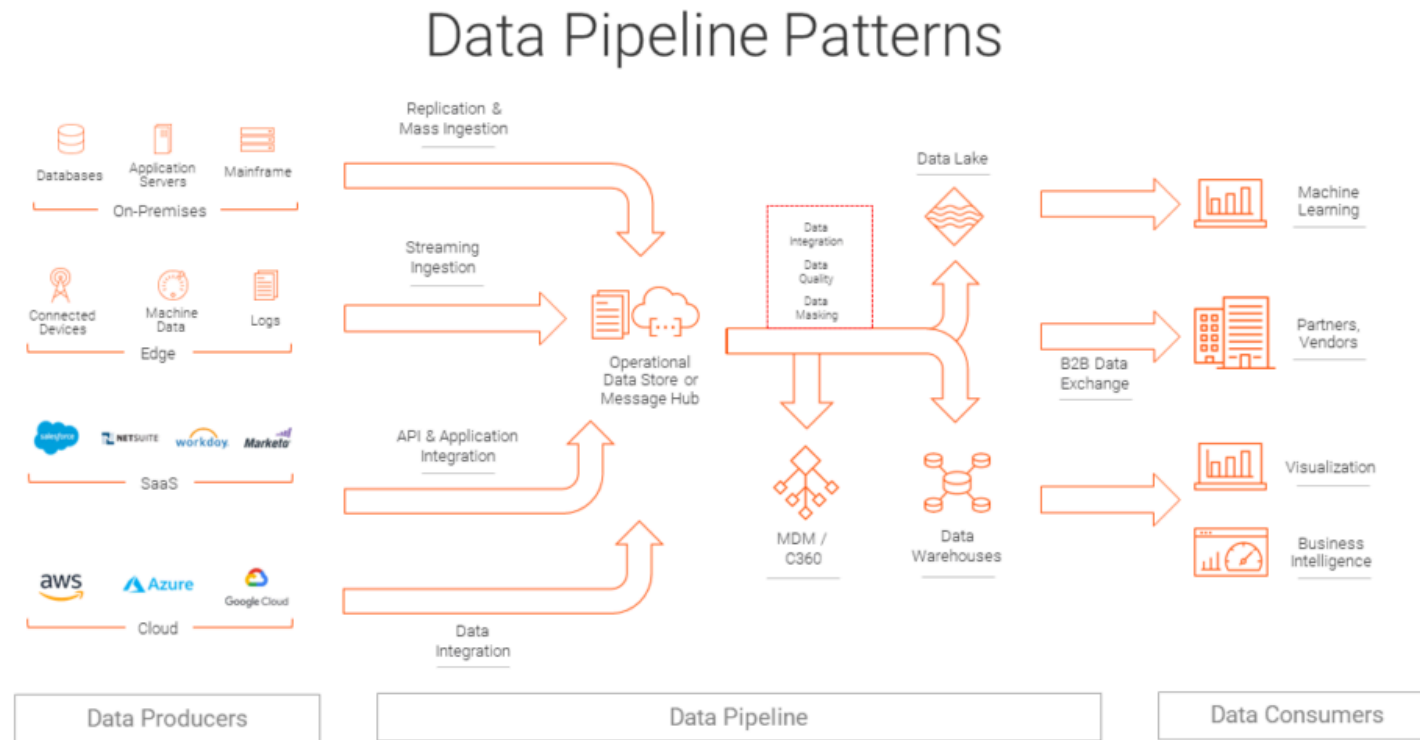
## L'Architecture du Big Data

### 3 Schéma global d'une architecture Big Data moderne



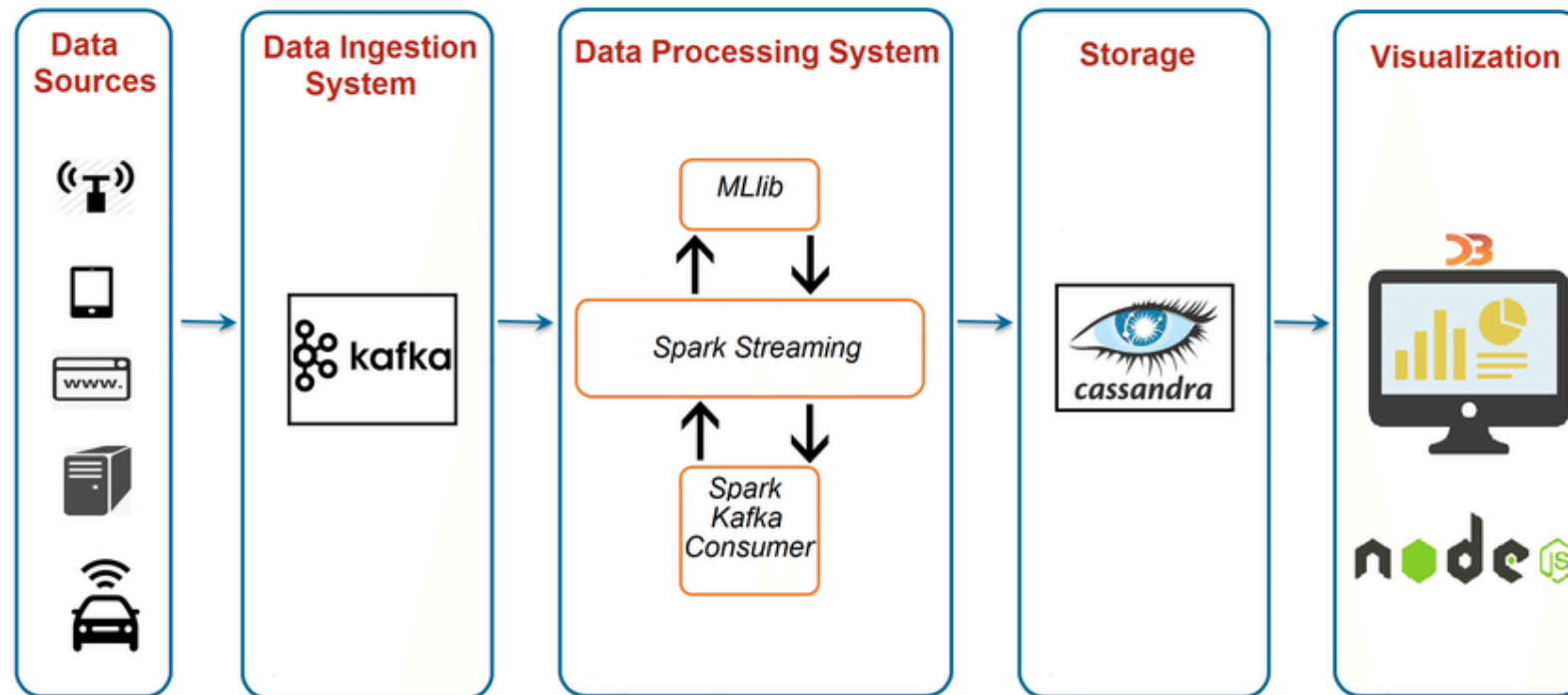
# Architecture et Technologies du Big Data

## L'Architecture du Big Data



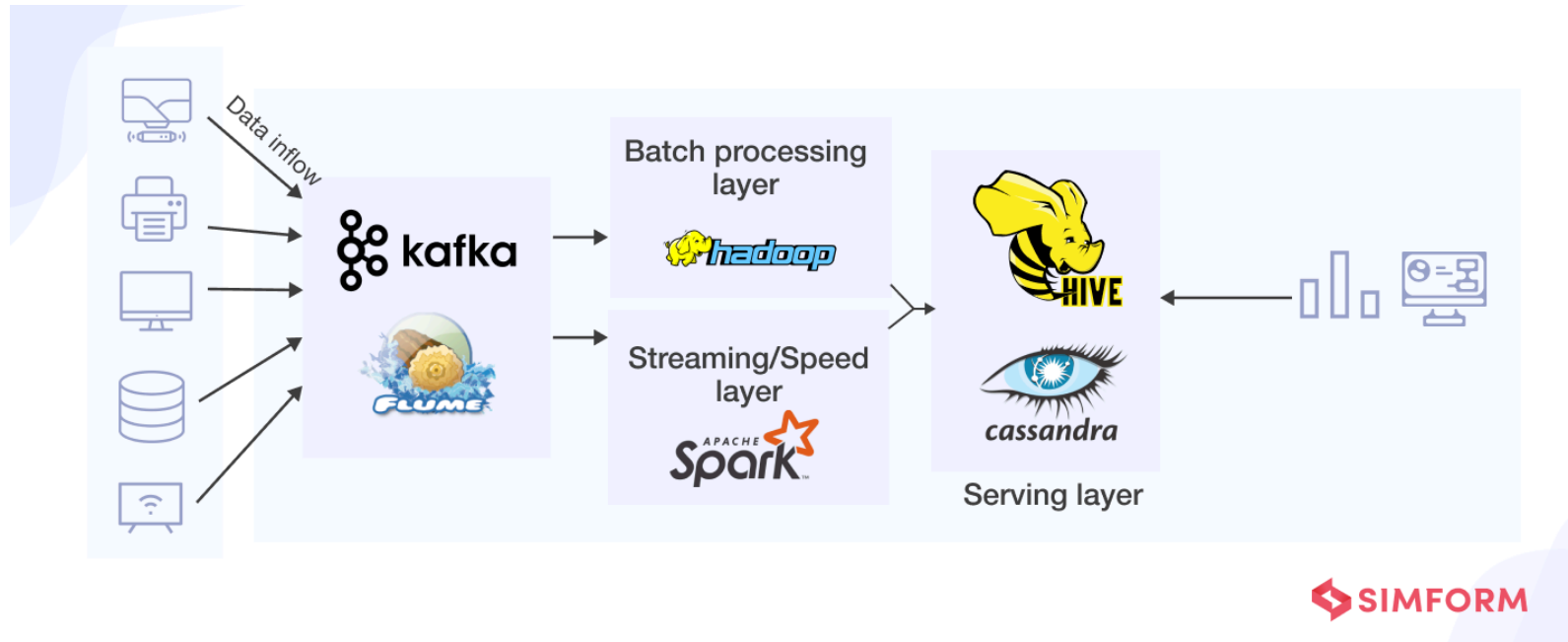
# Architecture et Technologies du Big Data

## L'Architecture du Big Data



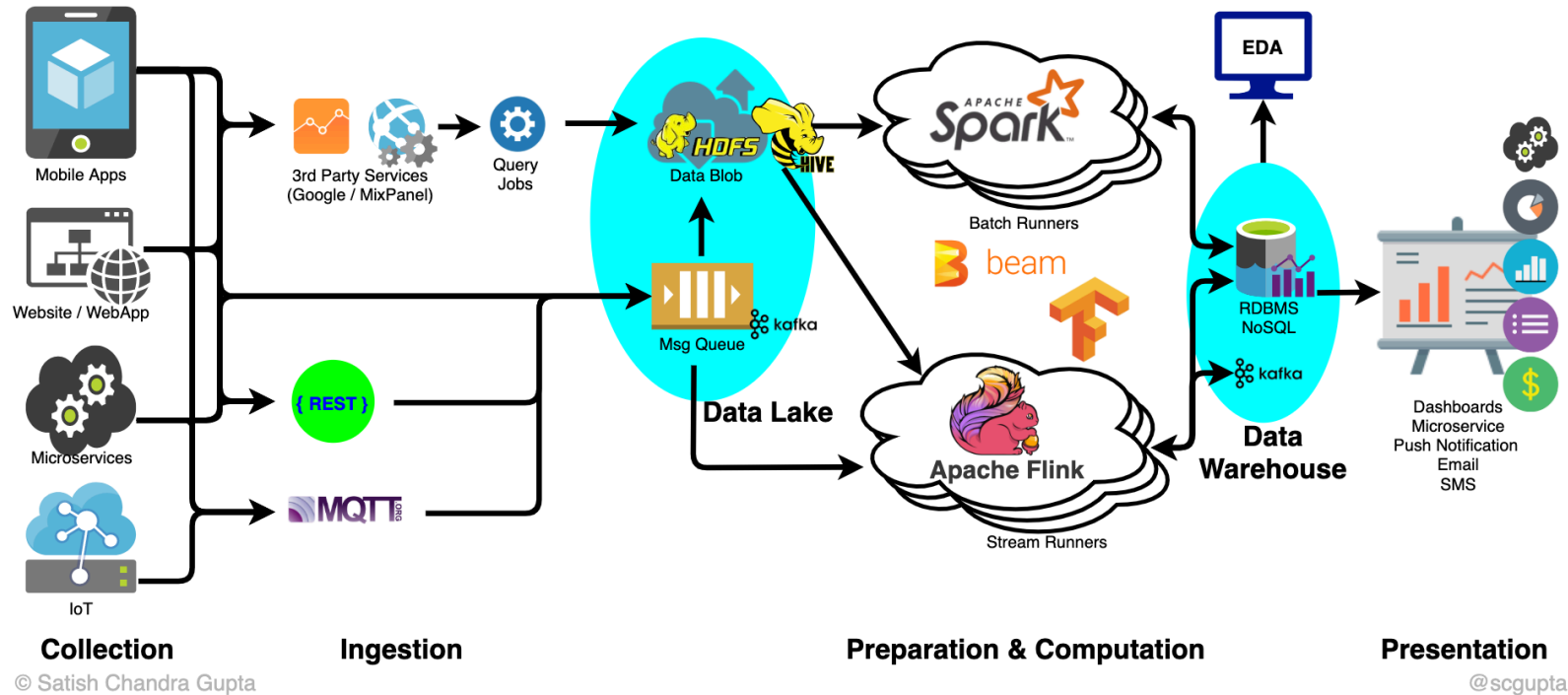
# Architecture et Technologies du Big Data

## L'Architecture du Big Data



# Architecture et Technologies du Big Data

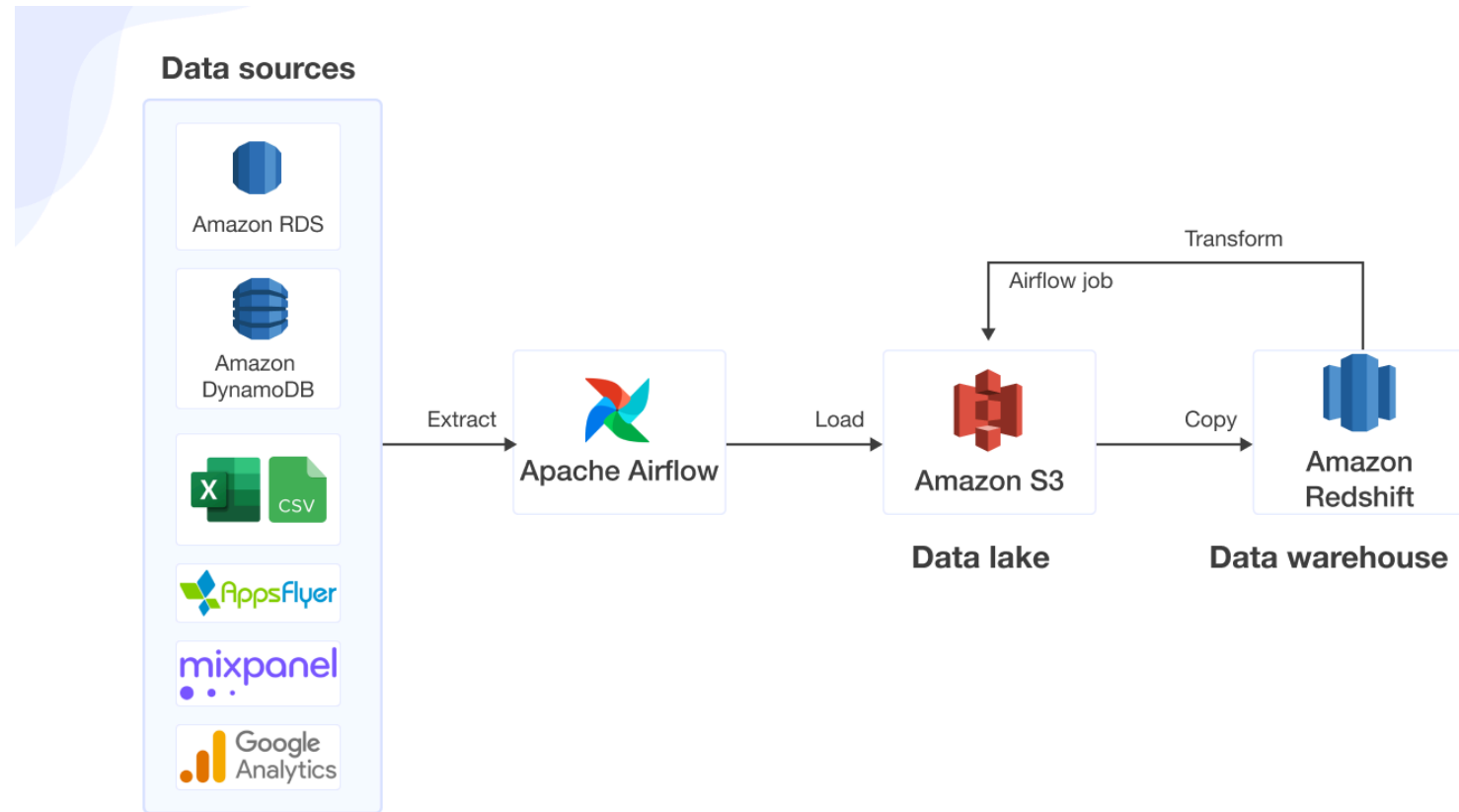
## L'Architecture du Big Data





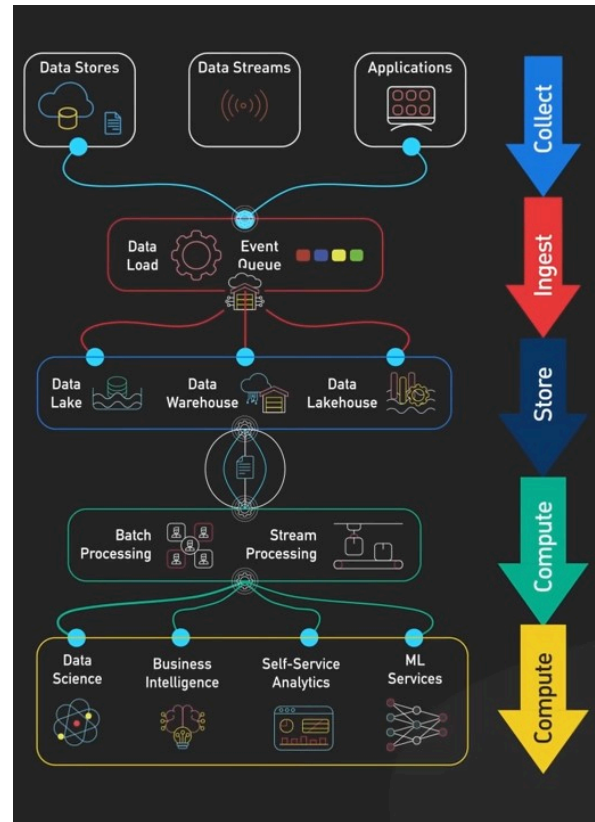
# Architecture et Technologies du Big Data

## L'Architecture du Big Data



# Architecture et Technologies du Big Data

## L'Architecture du Big Data



# Architecture et Technologies du Big Data

## L'Architecture du Big Data

### 4 Évolutions de l'architecture Big Data

- **Hier** : Hadoop (stockage HDFS + batch MapReduce).
- **Aujourd'hui** : Spark/Flink + Kafka + Cloud Data Lakehouse.
- **Demain** : Data Mesh (données gérées comme des produits par domaines), Edge + 5G (traitement proche des sources), Quantum Big Data.

# Technologies du Big Data

# Technologies du Big Data

## 1 L'écosystème Hadoop : fondations historiques

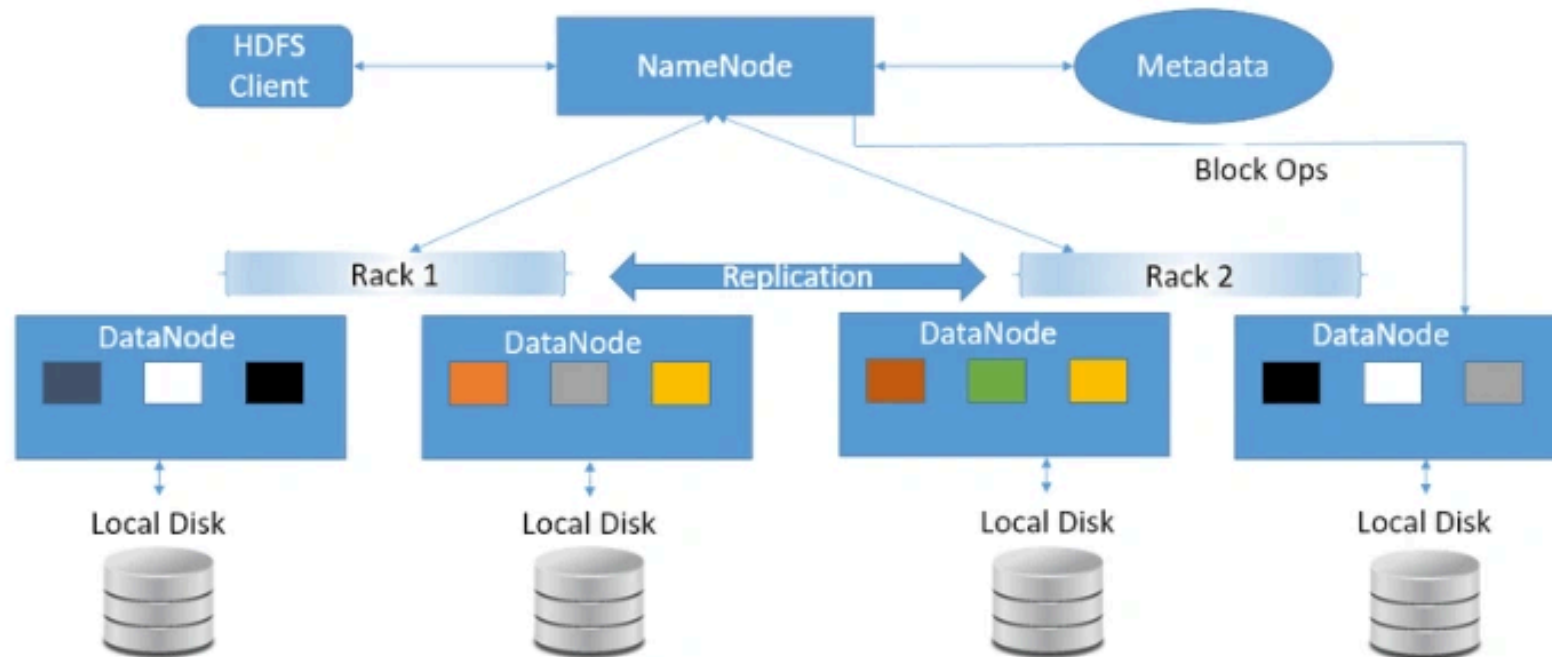
Lancé au milieu des années 2000, **Hadoop** a révolutionné la manière de stocker et traiter les données massives distribuées. Il repose sur trois piliers principaux

### ◆ HDFS (Hadoop Distributed File System)

- **But** : stocker de très grands volumes de données sur des clusters de serveurs standards.
- **Concept** : chaque fichier est découpé en blocs (par défaut 128 Mo) et répliqué sur plusieurs nœuds (3 copies typiques).
- **Avantages** :
  - Tolérance aux pannes (un nœud qui tombe → données accessibles ailleurs).
  - Scalabilité horizontale (ajout de serveurs = plus de stockage).
- **Exemple** : stocker des pétaoctets de logs web répartis sur 100 serveurs.

# Technologies du Big Data

## HDFS Architecture



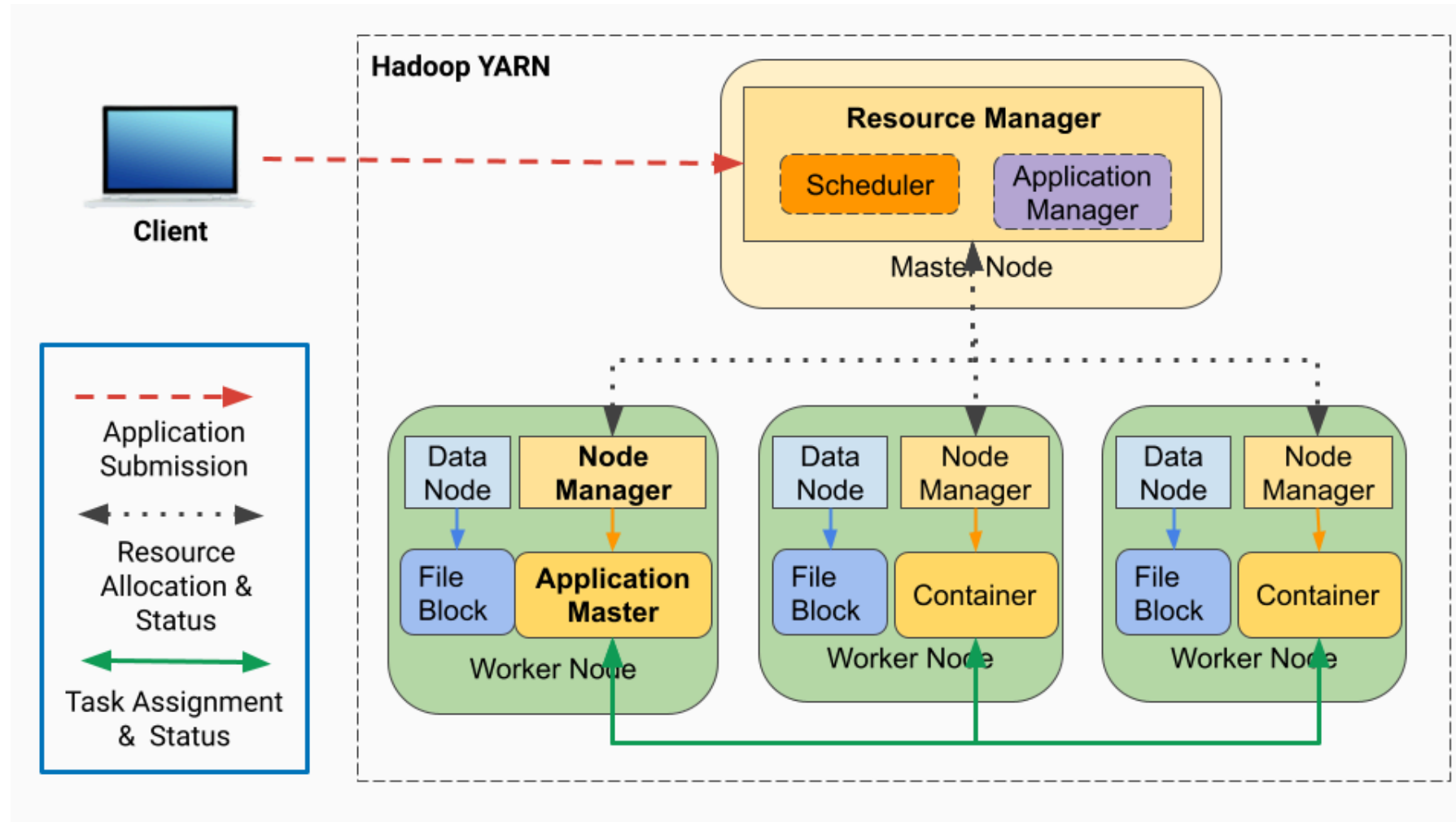
# Technologies du Big Data

## 1 L'écosystème Hadoop : fondations historiques

### ◆ YARN (Yet Another Resource Negotiator)

- **But** : gestionnaire de ressources du cluster Hadoop.
- **Fonction** : attribue CPU/RAM aux jobs soumis au cluster.
- **Analogie** : un “chef d'orchestre” qui décide quelle application tourne où et quand.
- **Exemple** : si plusieurs applications (Spark, MapReduce, Hive) tournent sur le même cluster, YARN alloue équitablement les ressources.

# Technologies du Big Data





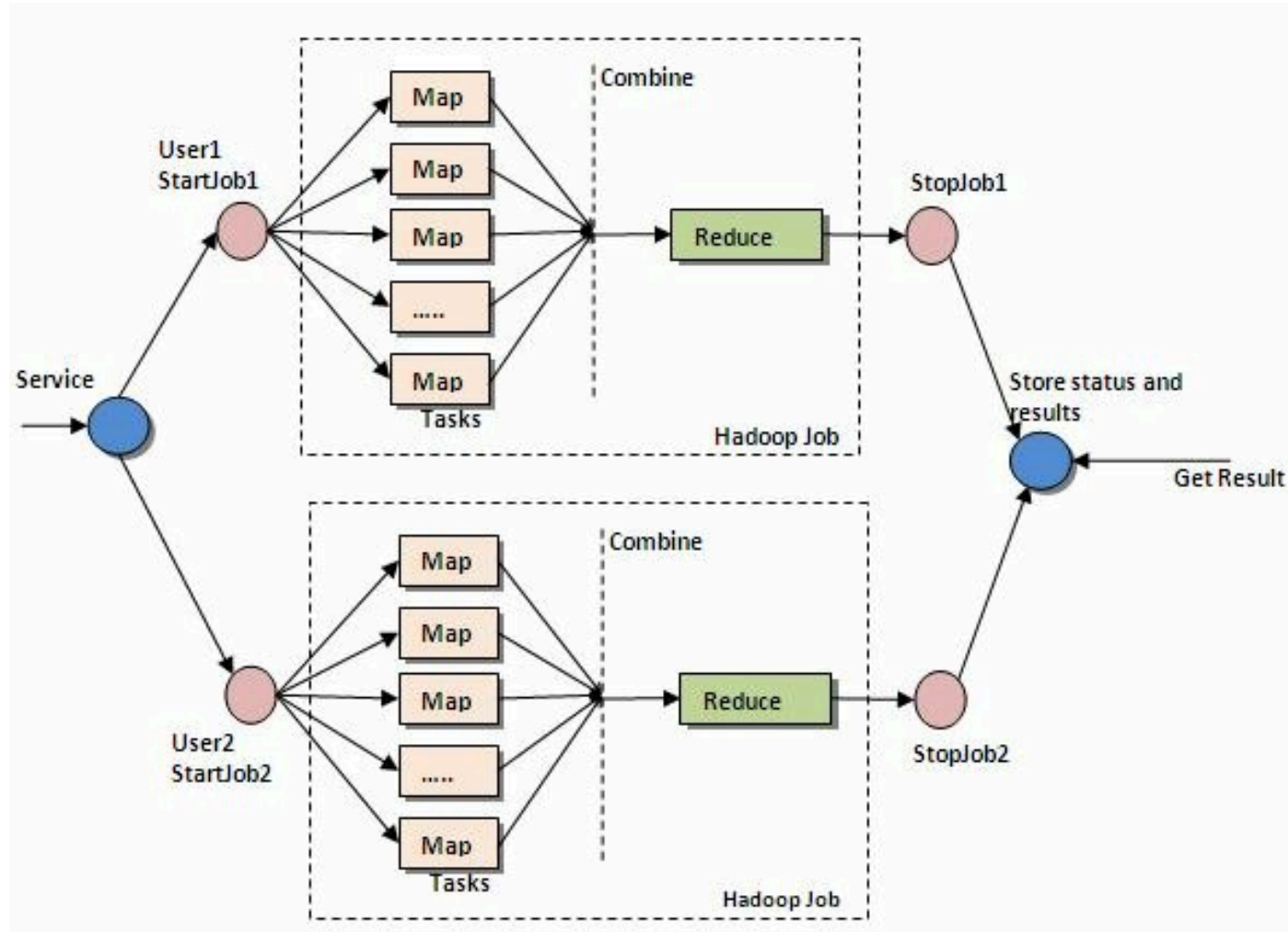
# Technologies du Big Data

## 1 L'écosystème Hadoop : fondations historiques

### ◆ MapReduce

- **Paradigme de calcul distribué** développé initialement par Google.
- **Deux étapes principales :**
  - **Map** : transformation des données (ex : extraire tous les mots d'un document).
  - **Reduce** : agrégation des résultats (ex : compter la fréquence de chaque mot).
- **Avantages** : traitement parallèle massif.
- **Limites :**
  - Lourd et lent (chaque étape nécessite d'écrire sur disque).
  - Moins adapté au **temps réel** → d'où l'émergence de Spark.

# Technologies du Big Data



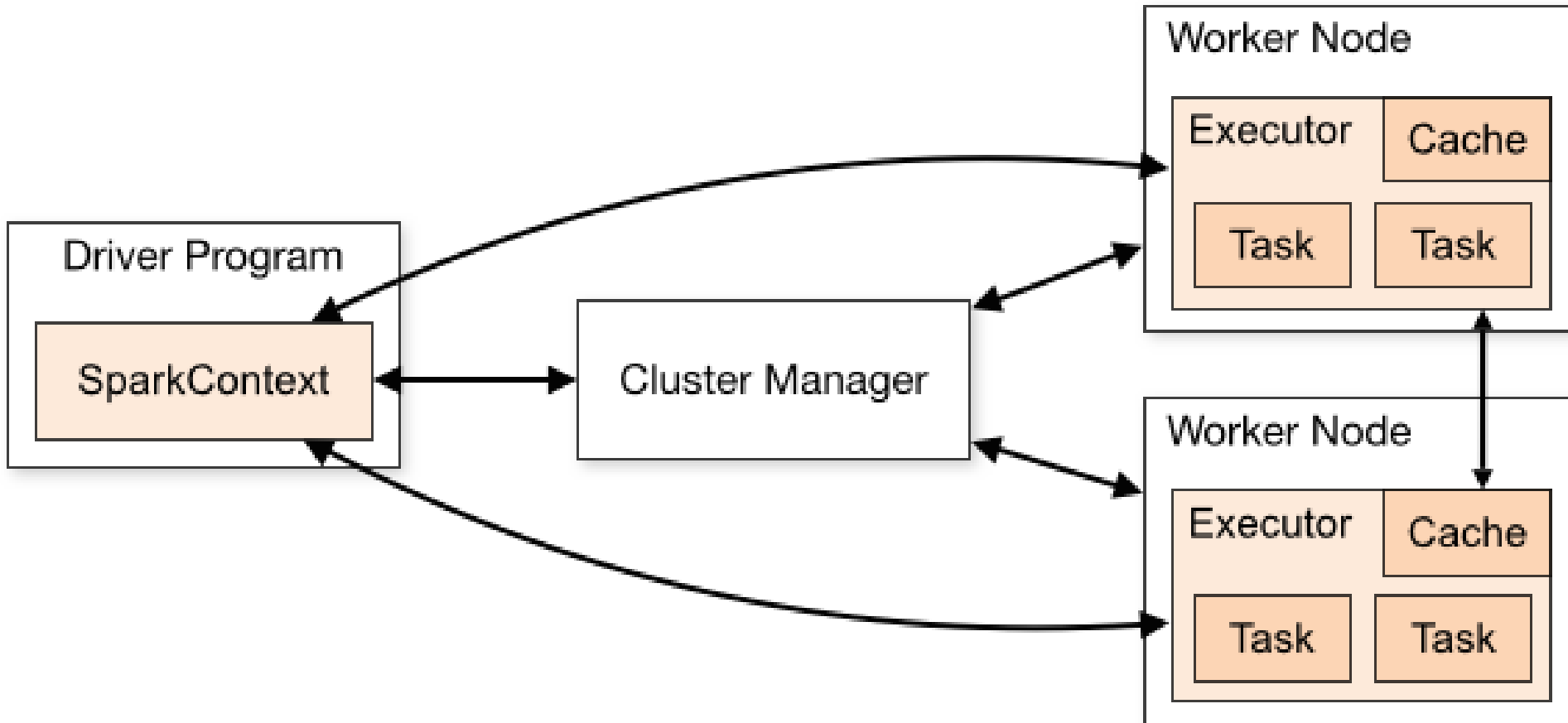
# Technologies du Big Data

## 2 Traitement batch & streaming : Spark, Kafka, Flink

### ◆ Apache Spark

- **Successeur de MapReduce** : beaucoup plus rapide (traitement en mémoire).
- **Modules** :
  - Spark SQL (requêtes SQL sur données massives).
  - MLlib (Machine Learning distribué).
  - GraphX (analyse de graphes).
  - Spark Streaming (flux temps réel “micro-batch”).
- **Exemple** : analyser en quelques secondes les logs de millions de clics pour comprendre le comportement utilisateur.

# Technologies du Big Data



# Technologies du Big Data

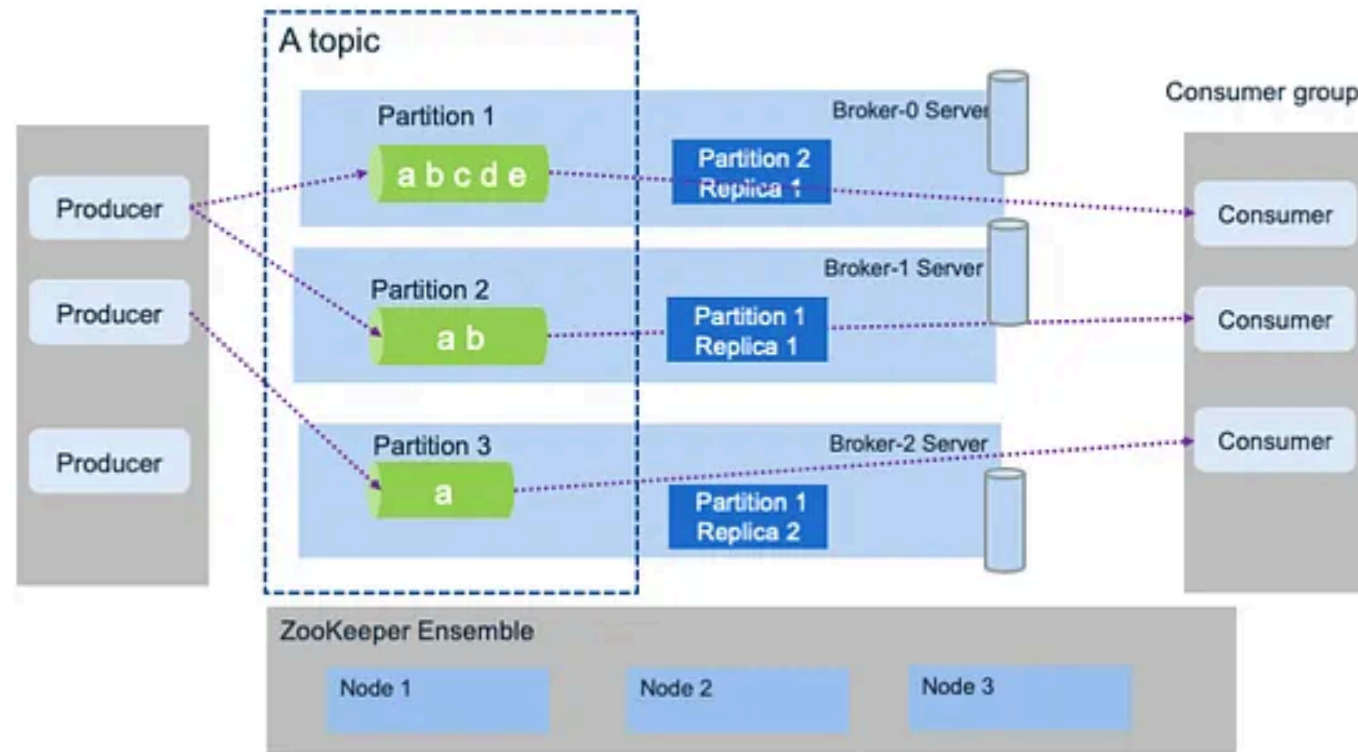
## 2 Traitement batch & streaming : Spark, Kafka, Flink

### ◆ Apache Kafka

- **Plateforme de streaming distribué** (pub/sub).
- **Concept** : données → topics → consommateurs.
- **Cas d'usage** :
  - Collecter logs d'applications.
  - Streaming de transactions bancaires.
  - Systèmes d'alertes en temps réel.
- **Exemple** : Uber → Kafka collecte en temps réel la géolocalisation des chauffeurs et passagers.

# Technologies du Big Data

## Kafka Architecture



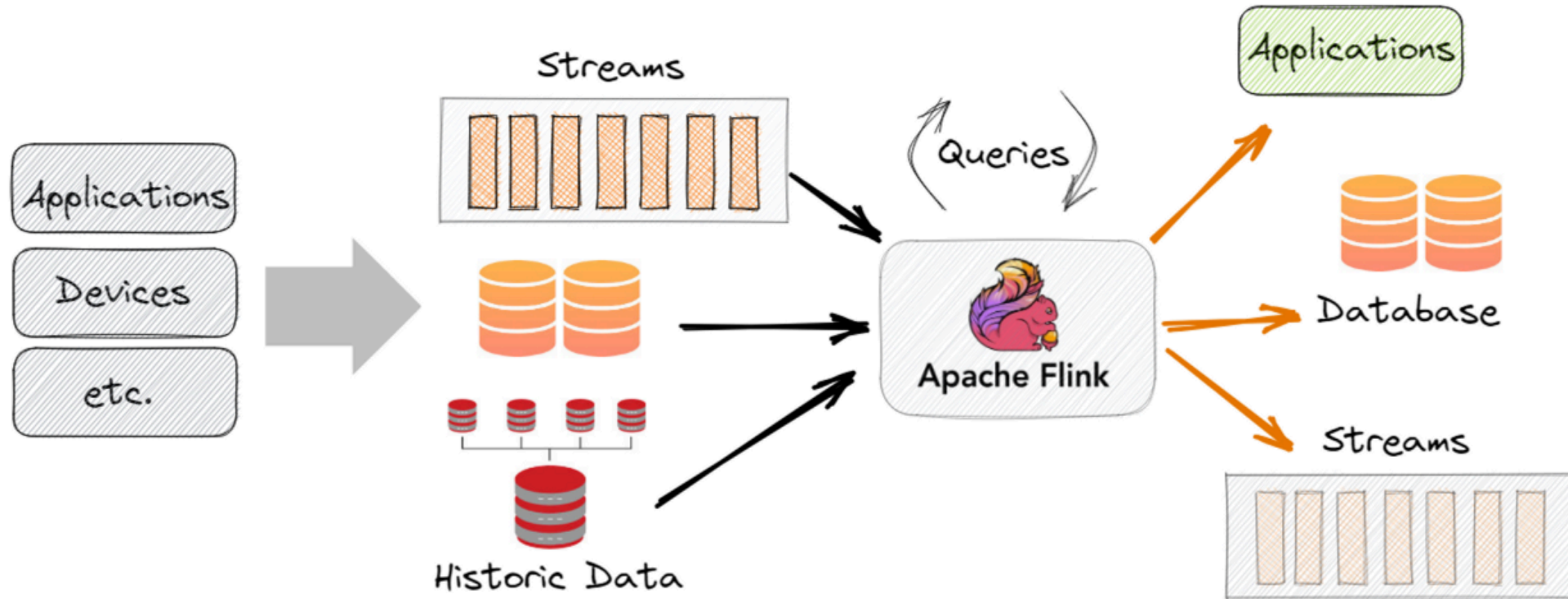
# Technologies du Big Data

## 2 Traitement batch & streaming : Spark, Kafka, Flink

### ◆ Apache Flink

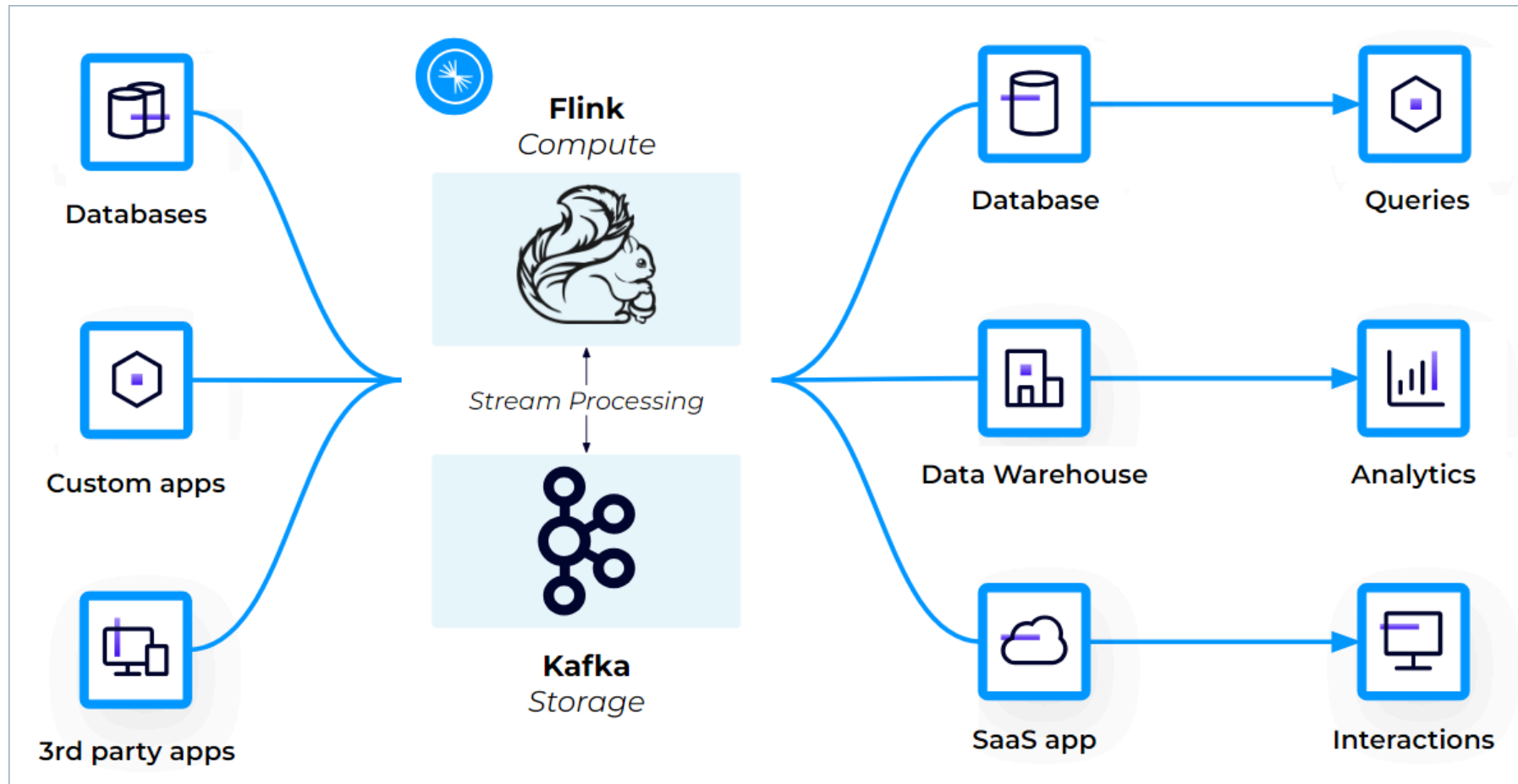
- Moteur de traitement temps réel natif (vs Spark micro-batch).
- Points forts :
  - Latence très faible (< 100 ms).
  - Fenêtres temporelles puissantes (sliding, tumbling).
  - Gestion avancée des états distribués.
- **Exemple** : analyse en temps réel des flux IoT (capteurs industriels, smart meters).

# Technologies du Big Data

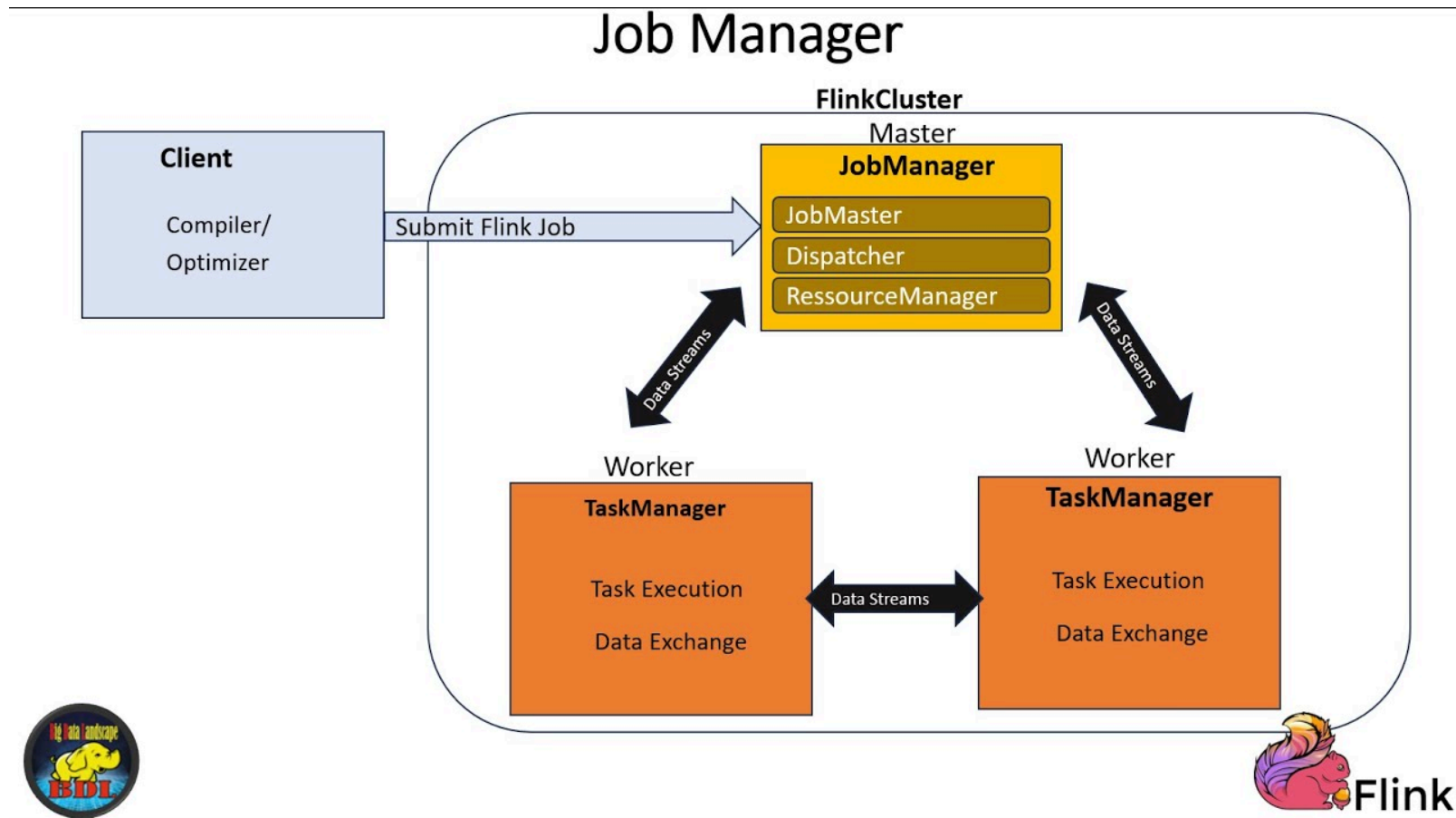




# Technologies du Big Data



# Technologies du Big Data



# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ Cassandra

- **Base orientée colonnes distribuée** (inspirée de BigTable de Google).
- **Forces** : haute disponibilité, tolérance aux pannes, scalabilité linéaire.
- **Exemple** : Netflix l'utilise pour gérer les données de visionnage de millions d'utilisateurs en temps réel.

# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ Cassandra

Cassandra suit une **architecture distribuée sans maître** :

Tous les nœuds d'un cluster sont **égaux** (architecture **peer-to-peer**).

Cela garantit :

- Pas de **single point of failure**.
- Une **réplication automatique**.
- Une **scalabilité horizontale** : on peut ajouter des nœuds facilement.

### Composants clés :

- **Cluster** → Ensemble de nœuds.
- **Nœud** → Serveur qui stocke une partie des données.
- **Keyspace** → "Base de données logique" qui définit la stratégie de réplication.
- **Table** → Stocke les lignes organisées par clés.
- **Partition** → Ensemble des lignes partageant une clé commune.

# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ Cassandra

Cassandra suit une **architecture distribuée sans maître** :

Tous les nœuds d'un cluster sont **égaux** (architecture **peer-to-peer**).

Cela garantit :

- Pas de **single point of failure**.
- Une **réplication automatique**.
- Une **scalabilité horizontale** : on peut ajouter des nœuds facilement.

### Composants clés :

- **Cluster** → Ensemble de nœuds.
- **Nœud** → Serveur qui stocke une partie des données.
- **Keyspace** → "Base de données logique" qui définit la stratégie de réplication.
- **Table** → Stocke les lignes organisées par clés.
- **Partition** → Ensemble des lignes partageant une clé commune.

# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ Cassandra

Cassandra est une **base NoSQL orientée colonnes distribuées**.

Elle a été conçue pour des besoins spécifiques :

- **Haute disponibilité**
- **Écriture massive**
- **Scalabilité horizontale**
- **Résilience multi-datacenter**

Contrairement à d'autres modèles, Cassandra **sacrifie la cohérence stricte** au profit de la **disponibilité** et de la **tolérance aux pannes** (principe **BASE** → *Basically Available, Soft state, Eventual consistency*).

# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ MongoDB

- **Base orientée documents (JSON/BSON).**
- **Forces :**
  - Flexibilité du schéma.
  - Facile à utiliser pour applications modernes (APIs REST).
- **Exemple :** e-commerce (catalogue produits aux attributs variables).

# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ HBase

- **Base distribuée clé-colonne sur HDFS.**
- **Points forts :**
  - Intégrée à l'écosystème Hadoop.
  - Accès aléatoire rapide à de gros volumes.
- **Exemple :** Facebook l'a utilisée pour sa messagerie.



# Technologies du Big Data

## 3 Bases de données NoSQL : flexibilité et scalabilité

### ◆ HBase

- **Base distribuée clé-colonne sur HDFS.**
- **Points forts :**
  - Intégrée à l'écosystème Hadoop.
  - Accès aléatoire rapide à de gros volumes.
- **Exemple :** Facebook l'a utilisée pour sa messagerie.

# Technologies du Big Data

Caractéristique	Cassandra (NoSQL Colonnes)	SQL classique (MySQL, PostgreSQL)	MongoDB (Document NoSQL)	Redis (Clé- Valeur)	Neo4j (Graphes)
Modèle	Colonnes distribuées	Relationnel	Documents JSON	Paires clé-valeur	Graphes orientés
Schéma	Flexible, colonnes dynamiques	Rigide, nécessite migrations	Flexible	Aucun	Flexible
Scalabilité	<b>Horizontale native</b>	Verticale (scaling limité)	Horizontale	Horizontale	Limitée
Performance écriture	<b>Très élevée</b>	Bonne mais dépend des verrous	Haute	<b>Excellente</b>	Moyenne

Caractéristique	Cassandra (NoSQL Colonnes)	SQL classique (MySQL, PostgreSQL)	MongoDB (Document NoSQL)	Redis (Clé- Valeur)	Neo4j (Graphes)
Performance lecture	Très bonne avec clé connue	Très bonne	Excellente	Excellente	Excellente
Réplication multi-DC	Native et simple	Complexe	Possible mais lourde	Limitée	Faible
Cohérence	Éventuelle configurable	Forte (ACID)	Forte ou éventuelle	Forte	Forte
Cas d'usage	IoT, logs, transactions massives	ERP, CRM, BI	API Web, catalogues	Cache, files d'attente	Réseaux sociaux, relations
Exemples d'entreprises	Netflix, Uber, Instagram	Banque, e- commerce, ERP	eBay, Expedia, Lyft	Twitter, GitHub	LinkedIn, Facebook

# Technologies du Big Data

## 4 Cloud & Big Data : démocratisation à grande échelle

### ◆ AWS

- **EMR (Elastic MapReduce)** : Hadoop/Spark/Kafka managés.
- **Redshift** : entrepôt de données massives.
- **S3** : stockage objet massivement scalable (souvent utilisé comme Data Lake).

### ◆ Azure

- **Azure Synapse Analytics** : Data Warehouse cloud.
- **Azure Databricks** : traitement big data & IA (Spark managé).
- **Data Lake Storage** : stockage distribué basé sur Blob.

# Technologies du Big Data

## 4 Cloud & Big Data : démocratisation à grande échelle

### ◆ GCP

- **BigQuery** : entrepôt de données serverless, très rapide (SQL sur pétaoctets).
  - **Dataflow** : traitement batch & stream (Apache Beam managé).
  - **Cloud Storage** : stockage distribué.
- 👉 **Avantage Cloud** : services managés → plus besoin d'administrer le cluster, pay-as-you-go, élasticité automatique.

# Technologies du Big Data

## 5 Modern Data Stack : vers le Data Lakehouse

### ◆ Data Lakehouse

- **Fusion du Data Lake (souplesse) + Data Warehouse (performance et gouvernance).**
- Permet à la fois :
  - Stockage de tous types de données (structurées + non structurées).
  - Requêtes SQL rapides pour la BI.
  - Support natif pour ML/IA.

# Technologies du Big Data

## 5 Modern Data Stack : vers le Data Lakehouse

### ◆ Delta Lake

- Développé par **Databricks**.
- Ajoute une **couche transactionnelle ACID** au Data Lake.
- **Fonctionnalités :**
  - Historisation (time travel).
  - Gestion de la qualité des données (schema enforcement).
  - Meilleure intégration Spark.
- **Exemple :** banques utilisant Delta Lake pour auditer toutes les transactions avec garanties ACID.

# Technologies du Big Data

## 5 Modern Data Stack : vers le Data Lakehouse

### ◆ Snowflake

- **Entrepôt de données cloud-native** (multi-cloud : AWS, Azure, GCP).
- **Points forts :**
  - Décorrélation stockage / calcul (scalabilité indépendante).
  - Concurrency massive (plusieurs équipes requêtent en parallèle sans ralentissement).
  - Partage sécurisé de données (“Data Sharing”).
- **Exemple :** une entreprise SaaS qui partage ses données analytiques avec ses clients en temps réel.



# Enjeux Organisationnels et Gouvernance du Big Data

# Enjeux Organisationnels et Gouvernance du Big Data

## 1 Organisation autour de la donnée

### ◆ Le Data Office

- **Mission** : piloter la stratégie data de l'entreprise, définir les règles de gouvernance, assurer la conformité et maximiser la valeur métier.
- **Composition typique** :
  - **Chief Data Officer (CDO)** : définit la stratégie globale et reporte souvent au DG ou DSI.
  - **Data Governance Lead** : met en place politiques & processus de gouvernance.
  - **Data Steward** : responsable qualité & documentation des données dans un domaine spécifique.
  - **Data Architect** : conçoit les architectures data (lakes, warehouses, pipelines).

# Enjeux Organisationnels et Gouvernance du Big Data

## 1 Organisation autour de la donnée

### ◆ Les rôles clés opérationnels

#### 1. Data Engineer

- Conçoit et maintient les pipelines de données (ETL/ELT, ingestion Kafka, jobs Spark).
- Gère la performance, la scalabilité, la disponibilité.
- Exemple : mettre en place un pipeline Kafka → S3 → Delta Lake.

#### 2. Data Scientist

- Exploite les données pour créer des modèles prédictifs et d'IA (ML, deep learning).
- Exemple : développer un modèle de prédiction de churn clients.

#### 3. Data Analyst

- Transforme les données en insights via SQL, visualisation (Tableau, PowerBI).
- Exemple : créer un tableau de bord des ventes par région et canal.

# Enjeux Organisationnels et Gouvernance du Big Data

## 2 Gouvernance et qualité de la donnée

### ◆ Gouvernance des données

- **Objectifs :**
  - Garantir **fiabilité, accessibilité et sécurité** des données.
  - Définir **qui peut faire quoi** avec les données.
  - Harmoniser les définitions métiers (ex. “client actif”).
- **Cadres et référentiels :** DAMA-DMBOK, DCAM, ISO/IEC 38505.

# Enjeux Organisationnels et Gouvernance du Big Data

## 2 Gouvernance et qualité de la donnée

### ◆ Qualité de la donnée

- Dimensions clés :
  - **Exactitude** : données correctes (ex. code postal valide).
  - **Complétude** : absence de valeurs manquantes critiques.
  - **Cohérence** : alignement entre systèmes (même client = même ID partout).
  - **Fraîcheur (timeliness)** : actualité des données (données de stock < 1h).
- Outils : **Great Expectations, Deequ, Talend DQ.**

# Enjeux Organisationnels et Gouvernance du Big Data

## 2 Gouvernance et qualité de la donnée

### ◆ Data Catalog

- **But** : inventaire des données disponibles dans l'entreprise.
- Fonctionnalités : recherche, métadonnées, classification (PII/PHI), accès.
- Outils : Collibra, Alation, AWS Glue Data Catalog.

# Enjeux Organisationnels et Gouvernance du Big Data

## 2 Gouvernance et qualité de la donnée

### ◆ Data Lineage

- **Traçabilité des données** : savoir d'où elles viennent, quelles transformations elles ont subies, où elles sont utilisées.
- Exemple : un indicateur financier calculé à partir de données clients doit être traçable jusqu'aux systèmes sources.
- Outils : Apache Atlas, OpenLineage.

# Enjeux Organisationnels et Gouvernance du Big Data

## 2 Gouvernance et qualité de la donnée

### ◆ MDM (Master Data Management)

- Vise à créer une “**golden record**” unique pour chaque entité clé (client, produit, fournisseur).
- Évite les doublons et incohérences (un même client = 5 entrées différentes dans différents systèmes).
- Exemple : banque qui doit avoir une vision unique d'un client pour gérer son risque global.



# Enjeux Organisationnels et Gouvernance du Big Data

## 3 Sécurité & conformité

### ◆ RGPD (Règlement Général sur la Protection des Données)

- Principes :
  - Consentement explicite.
  - Minimisation des données.
  - Droit d'accès, rectification, effacement ("droit à l'oubli").
  - Privacy by design & by default.
- Exemple : un site e-commerce doit permettre à un client de demander l'effacement complet de ses données personnelles.

# Enjeux Organisationnels et Gouvernance du Big Data

## 3 Sécurité & conformité

### ◆ Data Sovereignty (souveraineté des données)

- Les données doivent rester **dans la juridiction du pays** où elles sont produites.
- Exemple : certaines données de santé en France doivent être hébergées dans un HDS (Hébergeur de Données de Santé) agréé.
- Contrainte forte pour les multinationales utilisant le Cloud (AWS, Azure, GCP).

# Enjeux Organisationnels et Gouvernance du Big Data

## 3 Sécurité & conformité

### ◆ Techniques d'anonymisation et pseudonymisation

- **Anonymisation** : transformation irréversible (ex. suppression d'ID, agrégation).
- **Pseudonymisation** : remplacement par un identifiant artificiel, mais ré-identifiable via clé sécurisée.
- Exemple :
  - Études médicales → anonymisation totale.
  - Bases clients → pseudonymisation pour permettre analyses marketing.

# Enjeux Organisationnels et Gouvernance du Big Data

## 4 Dilemmes éthiques

### ◆ Biais algorithmiques

- Problème : modèles d'IA reproduisent (ou amplifient) les biais présents dans les données d'entraînement.
- Exemples :
  - Algorithmes de recrutement discriminant les femmes car entraînés sur des CV majoritairement masculins.
  - Modèle de scoring de crédit défavorisant certaines zones géographiques.
- Solutions :
  - Audits réguliers.
  - Métriques de fairness (disparate impact, equalized odds).
  - Jeux de données diversifiés.

# Enjeux Organisationnels et Gouvernance du Big Data

## 4 Dilemmes éthiques

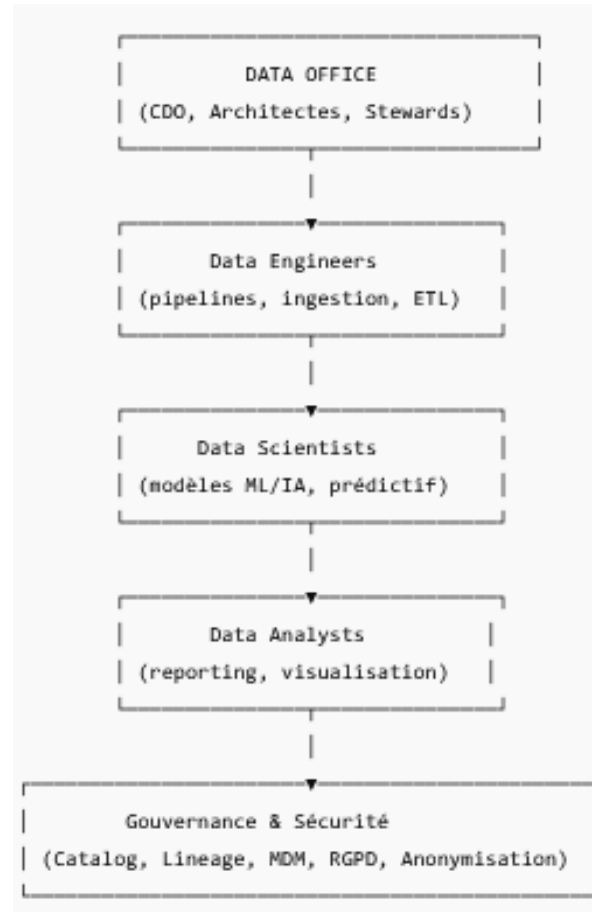
### ◆ Surveillance et vie privée

- Collecte massive de données = risque de dérive vers la surveillance intrusive.
- Exemple : caméras de reconnaissance faciale dans l'espace public.
- Question éthique : sécurité vs libertés individuelles.

### ◆ Confiance numérique

- Si les utilisateurs n'ont pas confiance, ils ne partagent pas leurs données.
- Exemples :
  - Scandale Cambridge Analytica → perte de confiance envers Facebook.
  - Banques mettant en avant la transparence pour rassurer clients (open banking).
- **Objectif clé** : créer un cadre de **transparence, explicabilité et contrôle utilisateur**.

# Enjeux Organisationnels et Gouvernance du Big Data



# Enjeux Organisationnels et Gouvernance du Big Data

## 4 Dilemmes éthiques

### ◆ Surveillance et vie privée

- Collecte massive de données = risque de dérive vers la surveillance intrusive.
- Exemple : caméras de reconnaissance faciale dans l'espace public.
- Question éthique : sécurité vs libertés individuelles.

### ◆ Confiance numérique

- Si les utilisateurs n'ont pas confiance, ils ne partagent pas leurs données.
- Exemples :
  - Scandale Cambridge Analytica → perte de confiance envers Facebook.
  - Banques mettant en avant la transparence pour rassurer clients (open banking).
- **Objectif clé** : créer un cadre de **transparence, explicabilité et contrôle utilisateur**.