Big Data – Cheat Sheet M2

Résumé opérationnel du cours – à garder pour réviser

1. Définition essentielle

Le Big Data n'est pas un outil. C'est une **architecture de gestion de données à grande échelle** conçue pour répondre à des contraintes que les systèmes classiques ne savent pas gérer.

2. Les 5V du Big Data (fondamentaux)

1. **Volume** : très grande quantité de données (To, Po)
2. **Vélocité** : données générées et traitées rapidement (streams, événements)
3. **Variété** : données hétérogènes (structurées, semi-structurées, non structurées)
4. **Véracité** : qualité, fiabilité, bruit dans les données
5. **Valeur** : capacité à extraire une information utile métier

Important : Les 5V décrivent le **problème**, pas la solution technique.

3. Pourquoi le Big Data existe ?

1. Les bases SQL ne scalent pas horizontalement
2. Les données arrivent trop vite pour être traitées immédiatement
3. Besoin de découpler producteurs et consommateurs
4. Besoin de conserver des données même sans usage immédiat

4. Architecture data classique vs Big Data

Classique : Application → Base SQL (schéma strict, transactions)
Big Data : Sources → Ingestion → Stockage → Traitement → Consommation

5. Pipeline Big Data (à connaître par cœur)

1. **Sources** : applications, capteurs, logs, APIs
2. **Ingestion** : Kafka, files, streams
3. **Stockage** : Data Lake (données brutes)
4. **Traitement** : batch ou streaming
5. **Consommation** : BI, API, ML, reporting

6. Batch vs Streaming

1. **Batch** : traitement différé, gros volumes, coût réduit
2. **Streaming** : traitement temps réel, faible latence, plus complexe

7. Kafka (conceptuellement)

1. Bus d'événements distribué
2. Découple producteurs et consommateurs
3. Permet la scalabilité horizontale

8. Data Lake & schémas

1. **Schema-on-write** : schéma avant écriture (SQL)
2. **Schema-on-read** : schéma à la lecture (Big Data)
3. Le Data Lake stocke des données brutes

9. Compromis à toujours expliquer

1. Latence vs coût
2. Simplicité vs flexibilité
3. Temps réel vs batch

10. Erreurs classiques

1. Choisir les outils avant le besoin
2. Copier une architecture sans comprendre
3. Sous-estimer la complexité opérationnelle

Phrase clé à retenir : Les 5V décrivent le problème, le Big Data est la réponse architecturale.