Université des Sciences et de la Technologie Houari Boumediene
Laboratoire des Systèmes Informatiques

JOURNÉES INRAE

SYSTÈMES D'INFORMATION POUR LES DONNÉES AGRO-ENVIRONNEMENTALES

BASES DE DONNÉES NOSQL

PR. KAMEL BOUKHALFA
kboukhalfa@usthb.dz

PLAN

- Contexte
- BD Clé-Valeur
- BD Orientées Colonnes
- BD Documents
- BD Graphes
- Ouvertures de recherche sur les BD NOSQL

BD RELATIONNELLES

- Introduites par Edgar F. Codd en 1970 : « A Relational Model for Large Shared Data Banks »

HOTEL (NUMHOTEL, NOM, VILLE, ETOILES)
CHAMBRE (NUMCHAMBRE, NUMHOTEL*, ETAGE, TYPECHAMBRE, PRIXNUIT)
CLIENT (NUMCLIENT, NOM, PRENOM)
RESERVATION (NUMCLIENT*, NUMHOTEL*, DATEARRIVEE, DATEDEPART, NUMCHAMBRE*)

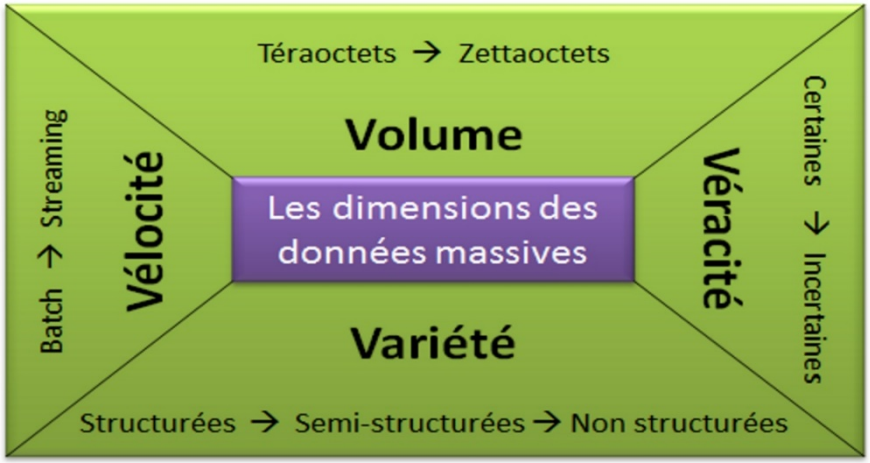
Caractéristiques

- Tout est relation (tuples + attributs)
- Basées sur l'algèbre relationnel
- Un langage de requête standard (SQL)
- Basées sur les propriétés ACID
- Normalisation
- Très populaires : MySQL, Oracle, SQL Server, etc.

NUMHOTEL	NOM	VILLE	ETOILES
1	Renaissance	Tlemcen	5
2	Seybouse	Annaba	3
3	Hôtel Novotel	Constantine	4
4	Saint George d'Alger	Alger	5
5	Ibis Alger Aéroport	Alger	2
6	El Mountazah Annaba	Annaba	3
7	Hôtel Albert 1er	Alger	3
8	Chems	Oran	2
9	Colombe	Oran	3
10	Mercure	Alger	4
11	Le Méridien	Oran	5
12	Hôtel Sofitel	Alger	5

BIG DATA

- Nous manipulons de plus en plus de données volumineuses



Les quatre V (« 4V »)

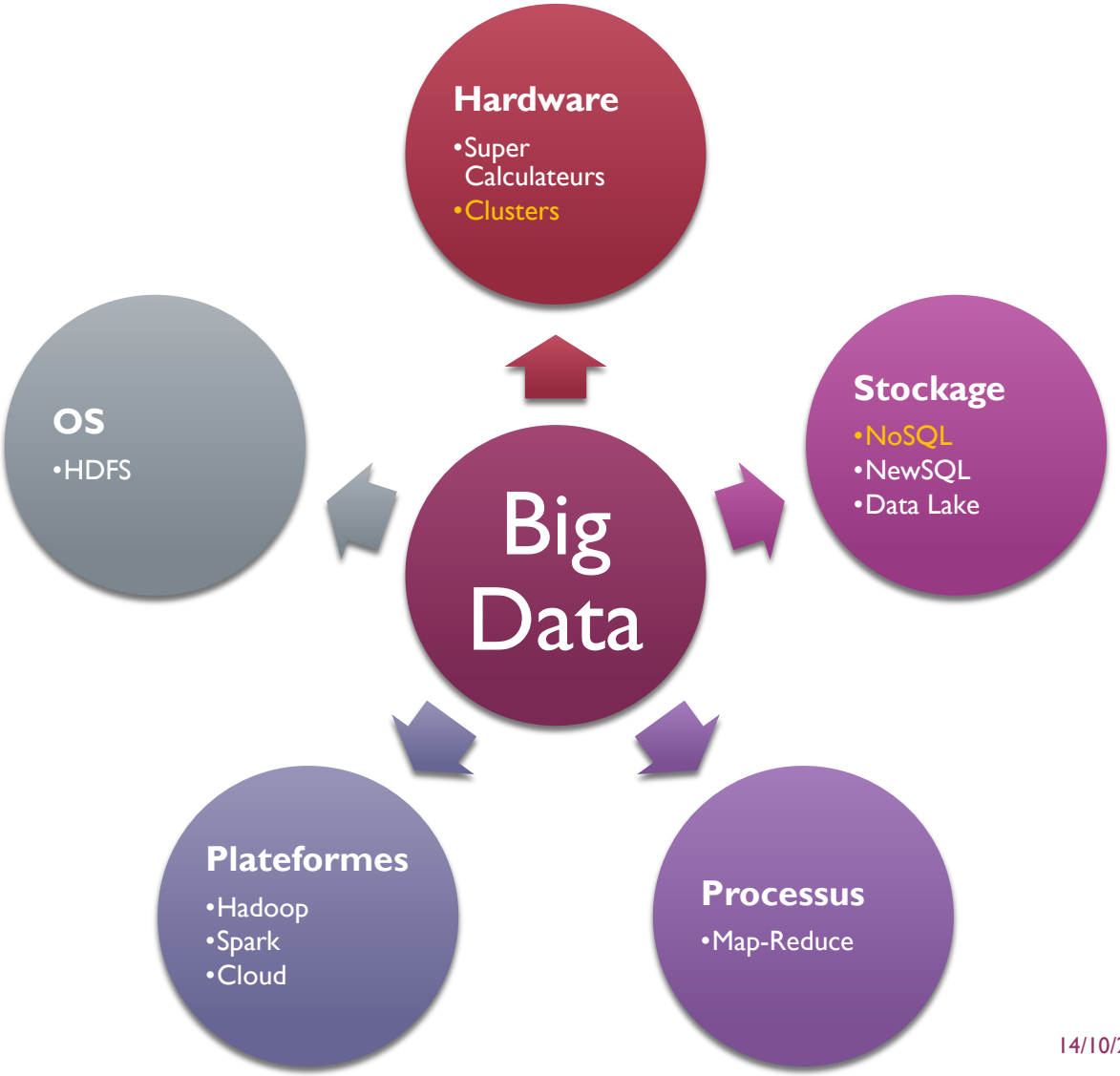
Volume *: taille excessive des données*

Vélocité *: vitesse avec laquelle ces données sont générées/traitées*

Variété *: diversité des formats/structures des données*

Véracité *: problème de fiabilité/précision des données massives*

CONSÉQUENCES DU BIG DATA



SCALABILITÉ

- **Scalabilité Verticale**
  - Effectuée en faisant une évolution hardware (CPU plus rapide, plus de RAM, Disques volumineux, etc).
  - Limitée par le nombre de CPU, la RAM et les capacités maximum des disques configurées sur une seule machine
  - Supercalculateurs chers non accessibles
- **Scalabilité Horizontale**
  - Utilisation de machines Hardaware Commodity
  - Distribution des données et traitement (Clusters, Data Center, etc.)

BDR – FORCES ET LIMITES

- **Forces**
  - La technologie est mature, le SQL est un langage standard et normalisé
  - Le transactionnel est garanti via les propriétés ACID
  - La possibilité de mettre en œuvre des requêtes complexes
  - Un large support est disponible et il existe également de fortes communautés.
- **Limites**
  - La modification du modèle établi peut être couteuse
  - L'évolutivité des performances est privilégiée de manière verticale
  - Sur un très grand volume de données distribuées le modèle peut atteindre des limites en terme de performance

→ Nécessité d'une nouvelle génération de BD non relationnelles, distribuées, open source et scalable horizontalement

→ BD NOSQL : Clé-Valeur, Document, Colonne, Graphe

→ MongoDB, Hbase, Redis, CouchDB, Cassandra, DynamoDB, Neo4j, etc.

BD CLÉ-VALEUR

- Base de données **non relationnelle** qui utilise une structure **clé-valeur** simple pour **stocker des données**.

Clé	**Valeur**
20202145	'Type: étudiant; Année Bac: 2020; Age : 18'
Matricule 20202145	Mod1:Note1, …Mod n:Note n BD : 12,50, Arch:10, …..

- La clé sert d'identifiant unique.
- Les clés et les valeurs peuvent se présenter sous toutes les formes, des **objets simples** aux **objets composés complexes**.
- **Hautement divisibles**
- Pas de schéma
- La valeur stockée est gérée au niveau applicatif
- Généralement utilisées en tant que **cache** ou pour des données temporaires.
- DynamoDB (Amazon), Azure Table Storage (Microsoft,), Riak, Redis

OPÉRATIONS

Quatre opérations principales

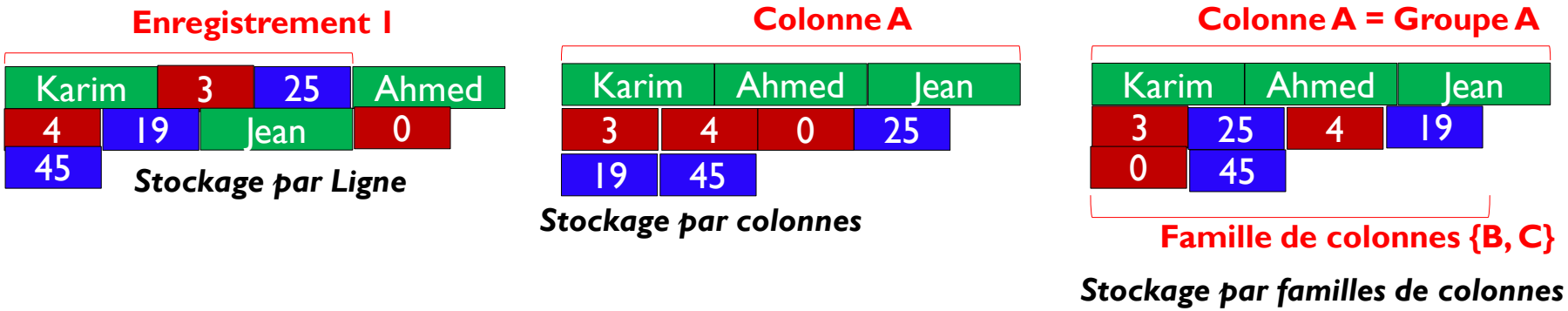
- **Création**
  - Créer un nouveau couple (clé, valeur).
- **Lecture**
  - Lire un objet en connaissant sa clé
- **Modification**
  - Mettre à jour l'objet associé à une clé
- **Suppression**
  - supprimer un objet connaissant sa clé

FORCES-LIMITES

- **Forces**
  - Leur simplicité, scalabilité, disponibilité
  - Très bonnes performances dans la mesure où les lectures et écritures sont réduites à un accès disque simple
- **Limites**
  - Pas de requêtes sur le contenu des objets stockés (seulement sur la clé)
  - Non-conservation des relations entre les objets (elles ne sont pas faites pour les contextes où la modélisation métier est complexe)
  - La couche applicative doit gérer toute la complexité des systèmes.

BD ORIENTÉES COLONNES

- Les bases de données colonnes sont hybrides entre BD relationnels et clés-valeurs
- Le stockage est organisé en colonnes
- Les valeurs sont stockées dans des groupes de plusieurs colonnes



- Les colonnes sont dynamiques (nombre de colonnes différents pour la même entité)
- Pas de stockage des valeurs nulles
- Cassandra, Google Big Table, Hbase

HYPERTABLE INC cassandra APACHE HBASE Google BigTable

LES BDS ORIENTÉES COLONNE

Concepts de base

- **Colonne**
  - Entité de base représentant un champ de donnée
  - Chaque colonne est définie par un couple ***clé / valeur***
- **Super colonne**
  - Elle présente une colonne dont les valeurs sont d'autres colonnes
- **Famille de colonnes**
  - Composée d'un ensemble de colonnes
  - Un conteneur permettant de regrouper plusieurs ***colonnes*** ou ***super colonnes.***

FORCES-LIMITES

- **Forces**
  - Flexibilité
  - Temps de traitement rapide
  - Non-stockage des valeurs null
- **Faiblesses**
  - Non-adaptée aux données interconnectées
  - Non adaptées aux données non structurées
  - Requêtes impliquant tous les attributs sont très couteuses

BD ORIENTÉE DOCUMENT

- Stocke des collections de documents composés d'un ensemble de couples propriété/valeur
- Cas particulier d'une BD clé-valeur où la valeur est un document dont la structure est libre (XML, JSON).
- Absence de schéma à priori.
- Une arborescence de champs
- Les documents peuvent être très **hétérogènes** au sein de la BD
- Exemple : MangoDB, Couche DB, etc.

```
{
  "titre": BD NoSQL",
  "datePublication" : Date("20/05/2018"),
  "auteur": " K. D.",
  "tags": [ "bigdata", "nosql" ],
  "commentaires": [ {
      "auteur": " Said",
      "commentaire": " Excellent!"
    }, {
      "auteur": " Karim",
      "commentaire": " Introuvebale"
    }
  ]
}
```

MONGODB

- S'appuie sur un modèle de données semi-structuré (encodage JSON)
- Pas de schéma (complète flexibilité)
- Un langage d'interrogation spécifique
- Supporte les index
- Replication et haute disponibilité
- Partitionnement automatique des données
- Pas de support transactionnel.

```
> db.user.findOne({age:39})
{
        "_id" :
ObjectId("5114e0bd42…"),
        "first" : "John",
        "last" : "Doe",
        "age" : 39,
        "interests" : [
                "Reading",
                "Mountain Biking ]
        "favorites": {
                "color": "Blue",
                "sport": "Soccer"}
}
```

- **Concepts** : Collection, document, champs, référence, etc.

FORCES ET LIMITES »

- **Forces**
  - Modèle de données simple mais puissant (expression de structures imbriquées)
  - Bonne mise à l'échelle avec le partitionnement (sharding)
  - Forte expressivité de requêtage (requêtes assez complexes sur des structures imbriquées)
- **Limites**
  - Peu adaptée pour les données interconnectées

BD ORIENTÉES GRAPHE

- Base de données spécifiquement dédiée au stockage de structures de données de type **graphe**.
- Tout système de stockage fournissant une adjacence entre éléments voisins sans indexation
  - Tout voisin d'une entité est accessible directement par un pointeur physique
- **Concepts de base** : Nœud, Arc, Propriété, Label
- **Types de graphes modélisés**
  - Homogène, hétérogène
  - Orienté, non-orienté
  - Simple, Hypergraphe, etc.

BD NEO4J

- Neo4j est une des premières base de données **orientée graphe**
- Libre et écrite en **Java**
- Développée par **Neo Technology**
- L'une des plus évoluées et robustes.
- Respecte les propriétés ACID
- Peut stocker et requêter des milliards de nœuds et de relations

FORCES ET LIMITES

- **Forces**
  - Modèle de données simple mais puissant pour les données **fortement connectées**
  - **Bonne mise à l'échelle** : les nouvelles relations ou les nœuds peuvent être ajoutés sans interférence avec les requêtes et les applications existantes
  - **Gains en performance** : les jointures sont stockées physiquement sur disque,
- **Limites**
  - Problèmes de densités et de partitionnement de big graphes sur un cluster
  - Non performantes pour calculer de grandes agrégations de données.

OUVERTURES DE RECHERCHE SUR LES BD NOSQL

- Tout n'est pas encore maitrisé
- **Sécurité** : intrusion, injection, etc.
- **Performance** : Optimisation, Indexation, Jointure
- **Intégration de données complexes :** données spatiales (Hbase, Terrastore)
- **Stockage** : disque, mémoire, etc.
- **Transactionnel :** protocoles de vérification des propriétés ACID
- **Requêtage** : réécriture des requêtes sémantiques

CONCLUSION

- **Forces BD NOSQL**
  - L'évolutivité se fait de manière horizontale
  - La représentation des données est flexible par l'absence de schéma
  - La majorité des solutions est Open Source
- **Limites**
  - Absence de langage d'interrogation standardisé
  - La mise en œuvre d'un environnement fortement transactionnel reste complexe
  - L'écriture de requêtes complexes est difficile à mettre en œuvre
  - L'offre NoSQL est segmentée en plusieurs familles où chacune répond à un besoin précis.

MERCI POUR VOTRE ATTENTION