Chapitre 3 : Bases de données NOSQL

Section 1 : Présentation

Définition

Aiway
Your way to AI

- NOSQL → Not Only SQL (≠ No SQL)
- Bases de données non-relationnelles et largement distribuées
- Permet une analyse et une organisation rapide des données de très grands volumes et de types de données disparates
- Développées en réponse à l'augmentation exponentielle des données générées, enregistrées et analysées par les utilisateurs modernes et leur applications

Atouts

Aiway
Your way to AI

- Principaux atouts
  - Évolutivité
  - Disponibilité
  - Tolérance aux fautes
- Caractéristiques
  - Modèle de données sans schéma
  - Architecture distribuée
  - Utilisation de langages et interfaces qui ne sont pas uniquement du SQL

NOSQL et Big Data

Aiway
Your way to AI

- De point de vue métier, utiliser un environnement Big Data et NOSQL fournit un avantage compétitif certain

- Importance des données :
  - “Si vos données ne croissent pas, alors votre entreprise ne le fait pas non plus”
  - La prise de décisions stratégiques repose sur l’analyse des données

Types des BDD NOSQL

Aiway
Your way to AI

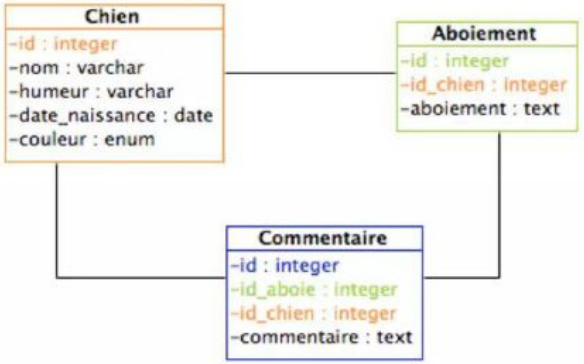
- Clé / valeur
- Orientées colonnes
- Orientées documents
- Orientées graphes

Clé / valeur

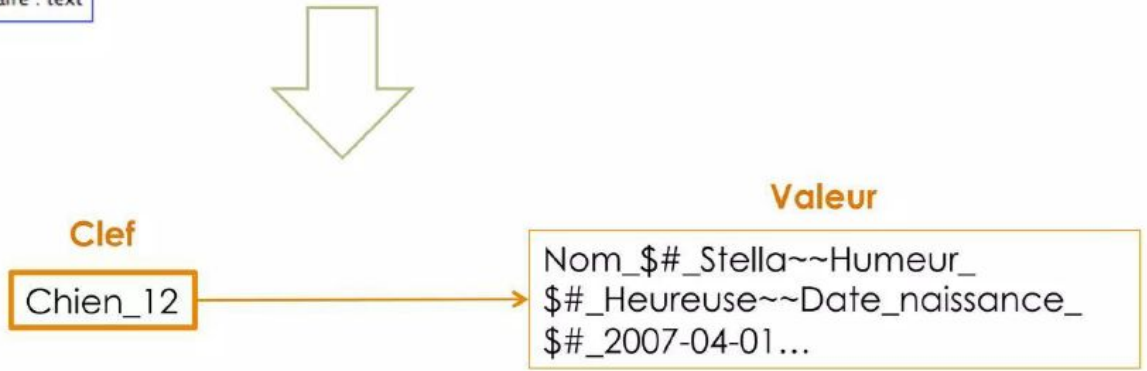
Aiway
Your way to AI

- L'un des types les plus simples, sorte de Hashmap distribuée
- Conçues pour sauvegarder les données sans définir de schéma
- Toutes les données sont sous forme de clé / valeur
  - La valeur peut être une string, un objet sérialisé etc
  - La donnée est opaque pour le système
    - Il n'est pas possible d'y accéder sans passer par la clé
- Absence de typage : toutes l'intelligence auparavant portée par les requêtes devra être portée par l'applicatif qui interroge la DB
- Objectif : fournir un accès rapide aux informations
- Exemples : DynamoDB, Redis, BerkeleyDB, Voldemort

Illustration



Id	Nom	Humeur	Date_naissance	Couleur
12	Stella	Heureuse	2007-04-01	NULL
13	Wimma	Faim	NULL	Noire
9	Ninja	NULL	NULL	NULL

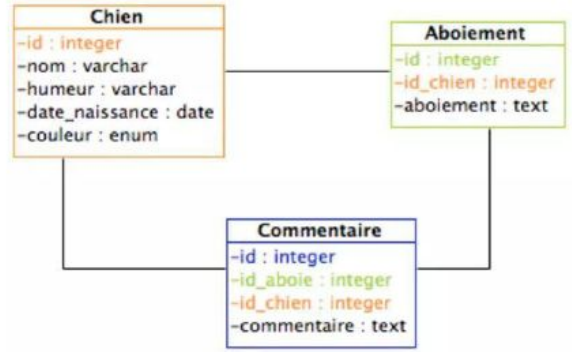


Orientées Documents

Aiway
Your way to AI

- Étendent le paradigme clé / valeur, avec des “documents” plus complexes à la place des données simples
- Une clé unique par document
- Chaque document est un objet (JSON ou XML) contenant un ou plusieurs champs et chaque champ contient une valeur
- Permettent de stocker, extraire et gérer les données semi-structurées
- Avantages : pouvoir récupérer via une seule clé un ensemble d’informations structurées de manière hiérarchique
- Exemples : MongoDB, CouchDB, RavenDB

Illustration



Id	Nom	Humeur	Date_naissance	Couleur
12	Stella	Heureuse	2007-04-01	NULL
13	Wimma	Faim	NULL	Noire
9	Ninja	NULL	NULL	NULL

Clef

Chien_12

Document (V1)

```
{
    type :    « Chien »,
    nom :     « Stella »,
    humeur :      « Heureuse »,
    date_naissance : 2007-04-01
}
```

Document (V2)

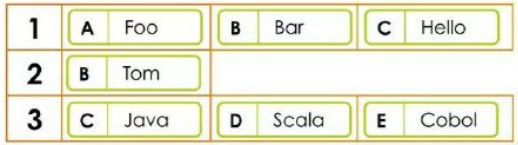
```
{
    type :    « Chien »,
    nom :     « Stella »,
    humeur :  « Heureuse »,
    date_naissance : 2007-04-01
    aboiement : [
        { texte : « j'ai mangé de la pâtée »
          commentaires : [
              { id_chien : « chien_4 »,
                texte : « on s'en fout! »
              }
          ]
        }
    ]
}
```

Orientées Colonnes

Aiway
Your way to AI

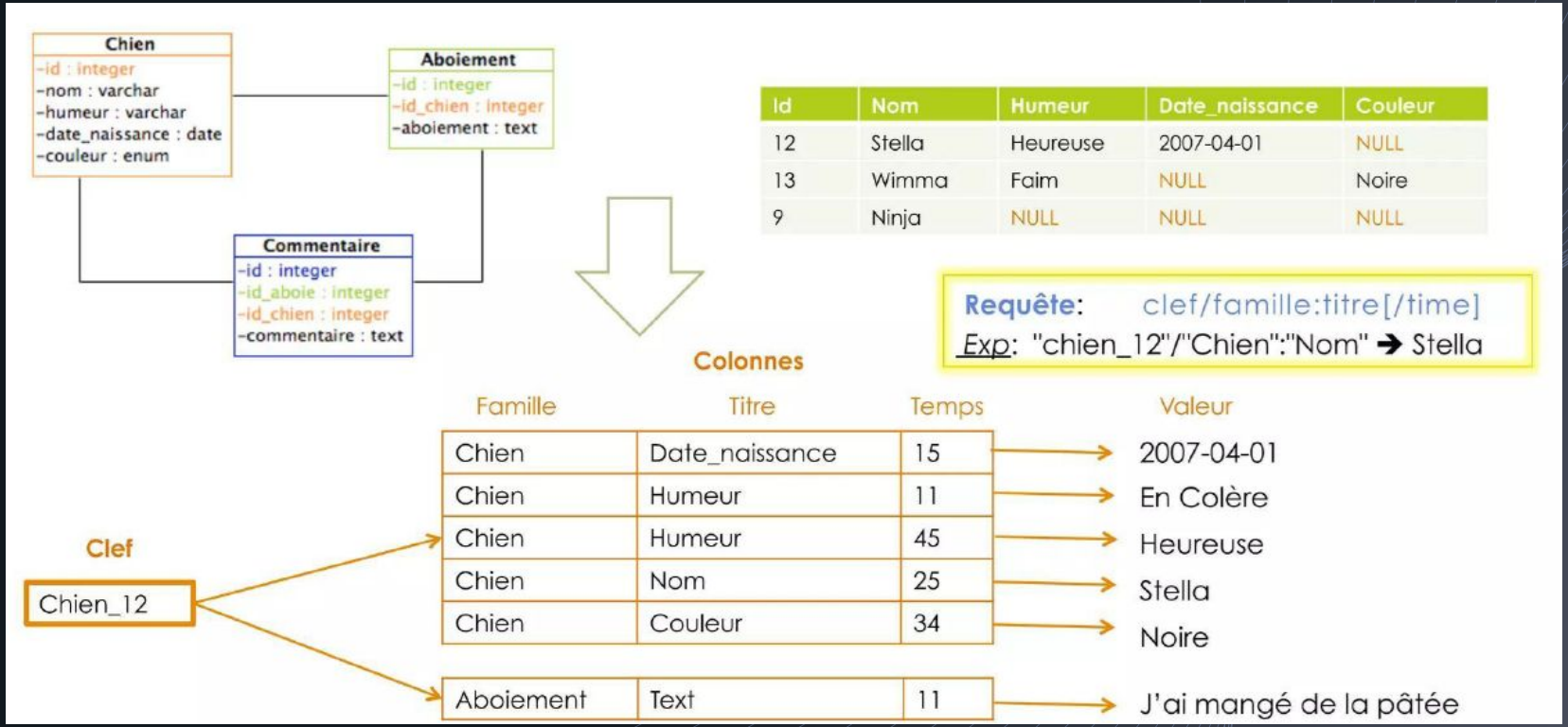
- Évolution de la BDD clé / valeur
- Ressemble aux SGBDR, mais avec un nombre de colonnes dynamique, différent d'un enregistrement à un autre

	A	B	C	D	E
1	Foo	Bar	Hello		
2		Tom			
3			Java	Scala	Cobol



- Offrent de très hautes performances et une architecture hautement évolutive
- Exemples : Hbase (Hadoop), Cassandra, Big Table

Illustration

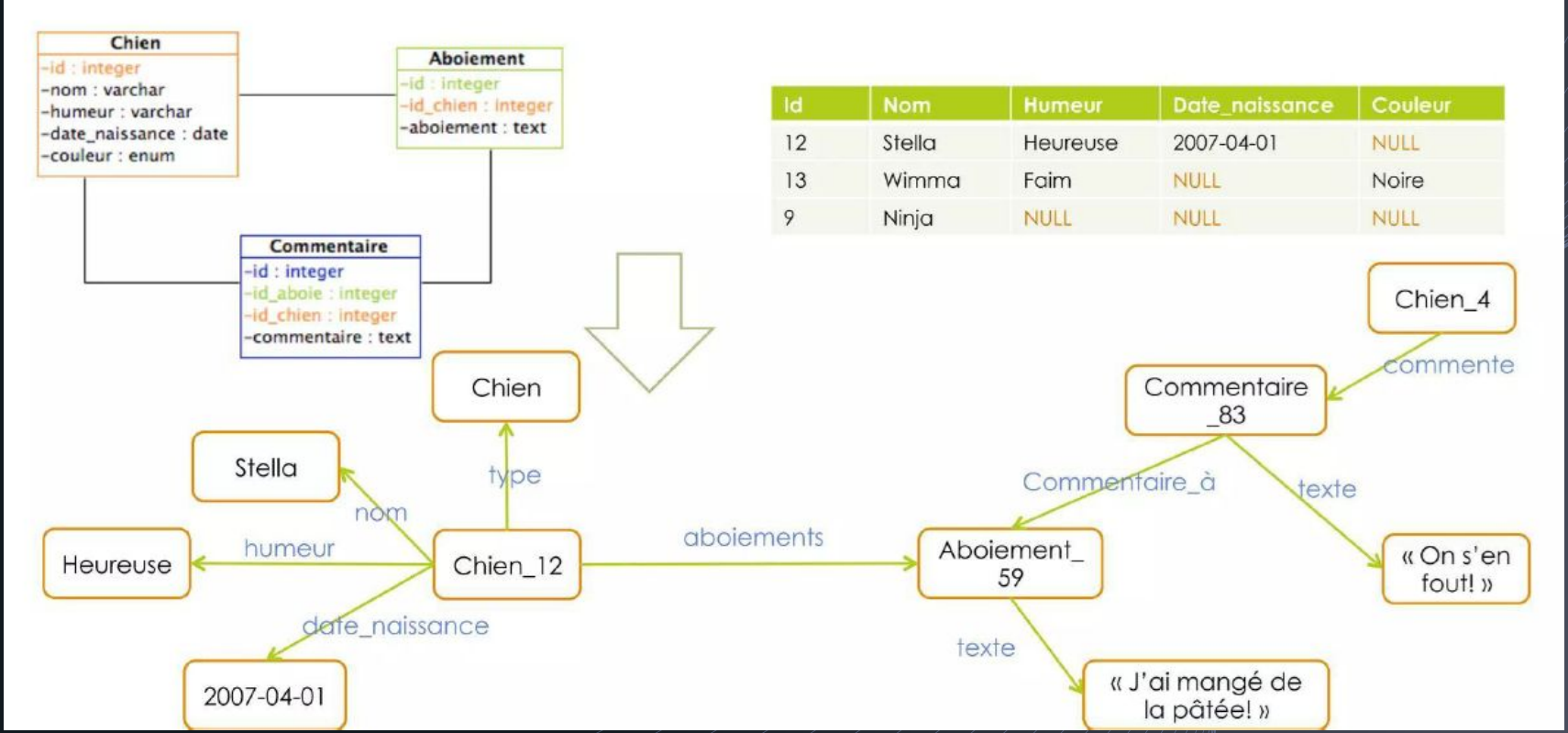


Orientées Graphes

Aiway
Your way to AI

- Basées sur les théories des graphes
- S’appuie sur les notions de noeuds, de relations et des propriétés qui leur sont rattachées
- Conçues pour les données dont les relations sont représentées comme graphes, et ayant des éléments interconnectés, avec un nombre indéterminé de relations entre elles
- Adaptées aux traitements des données des réseaux sociaux
- Exemples : Neo4j et InfiniteGraph, OrientDB

Illustration



Chapitre 3 : Bases de données NOSQL

Section 2 : NOSQL vs BDR

Aiway

Your way to AI

NOSQL et BDR (1/3)

Aiway
Your way to AI

- Le choix de NOSQL en opposition aux bases de données relationnelles doit être induit par les contraintes du marché et les besoins techniques
- Big Data
  - Adaptation des BDD NOSQL au Big Data
    - Vitesse, Variété, Volume, Complexité
- Disponibilité continue des données
  - Les BDD NOSQL utilisent une architecture hautement distribuée
    - Pas de SPOF
  - Redondance des données et traitements : tolérance aux fautes
  - Toute mise à jour est faite sans déconnecter la base

NOSQL et BDR (2/3)

Aiway
Your way to AI

- Indépendance de l'emplacement
  - Possibilité de consulter et modifier une BDD sans savoir où les opérations ont réellement lieu
  - Toute opération d'écriture est propagée pour être disponible aux utilisateurs à partir d'autres sites
    - Difficile à appliquer aux BDR, surtout pour l'écriture
- Modèles de données flexibles
  - Dans les modèles relationnels, un schéma strict défini les relations
    - Problème d'évolutivité et de performance avec de grands volumes
  - Les BDD NOSQL peuvent accepter tout type de données
  - Dans les BDR, les performances posent problème, surtout avec des lignes "larges" et des modifications nombreuses

NOSQL et BDR (3/3)

Aiway
Your way to AI

- Business Intelligence et analyse
  - Extraction d'informations décisionnelles à partir d'un grand volume de données, difficile avec des BDR
  - Bases NOSQL permettent le stockage et la gestion des données des applications métier, et fournissent une possibilité de comprendre les données complexes et de prendre des décisions
- Capacité transactionnelles modernes
  - Nouvelle définition du principe de transaction
  - Utilisation des propriétés BASE au lieu des propriétés ACID

Propriétés ACID

Aiway
Your way to AI

- Atomicité, Cohérence, Isolation, Durabilité
  - Atomicité : Soit toutes les instructions sont exécutées, soit aucune
  - Consistance : toute transaction amène d'un état valide à un autre
    - Renforcée par les contraintes d'intégrité et les clés étrangères
  - Isolation : Même si plusieurs transactions peuvent être exécutées par un ou plusieurs utilisateurs simultanément, une transaction ne devrait pas voir les effets des autres transactions concurrentes
  - Durabilité : Une fois la transaction enregistrée dans la base, les changements sont persistants
- Toutes les BDR supportent les transactions ACID
- Mais :
  - Besoin d'évolutivité et de BDD distribuées en réseau

Théorème CAP (1/3)

Aiway
Your way to AI

- Destiné à évaluer les systèmes de stockage distribués
- “Il est impossible de satisfaire les trois propriétés CAP en même temps”
- **Propriétés CAP** : *Consistency, Availability, Partition tolerance*
  - Consistance : Si j’écris une donnée dans un noeud et que je la lis à partir d’un autre noeud dans le système, je retrouve la donnée
  - Haute disponibilité : à tout moment, pour chaque requête, la réponse est garantie, même en cas de panne
  - Tolérance au partitionnement : les données peuvent être partitionnées sur différents supports sans souci de localisation. Les activités continuent sans interruption lors des la modification du système et en cas de chute de réseau

Théorème CAP (2/3)

Aiway
Your way to AI

- Pourquoi est-il impossible de satisfaire les 3 propriétés ?
- Prenons un système distribué. On est en train de modifier une donnée sur le noeud N1 et d'essayer de la lire à partir du noeud N2
  - N2 peut retourner la dernière bonne valeur dont il dispose, ce qui viole la consistance
  - N2 attend que la nouvelle valeur lui parvienne, ce qui provoque une atteinte au principe de disponibilité
  - Si on veut satisfaire à la fois la consistance et la disponibilité, le système de stockage ne doit pas être partitionné.
    D'où la violation de la tolérance au partitionnement

Théorème CAP (3/3)

Aiway
Your way to AI

- Pour les BDD NOSQL, il n'y a pas de jointures
  - La propriété de consistance n'est plus assuré de la même manière

- Consistance dans NOSQL :
  - Consistance immédiate
  - Consistance éventuelle
  - Variable à travers les données sur les différents noeuds de la base

Propriété BASE

Aiway
Your way to AI

- Basically Available, Soft-state, Eventual consistency
  - *Basically Available*
    - Le système garantit la disponibilité, défini dans le théorème CAP
  - *Soft-state*
    - L'état du système peut changer dans le temps, même sans nouvelles entrées, et ce à cause du principe de consistance éventuelle
  - *Eventual consistency*
    - Les modifications arriveront éventuellement à tous les serveurs, si on leur donne suffisamment de temps
- BASE est plus flexible que ACID, accepte certaines erreurs, dont l'occurence est assez rare

Récapitulatif

Aiway
Your way to AI

BRD	NOSQL
- Consistance forte	- Consistance éventuelle
- Grandes quantité de données	- Énorme quantité de données
- Évolutivité possible	- Évolutivité facile
- SQL	- MapReduce
- Bonne disponibilité	- Très haute disponibilité

Synthèse

Aiway
Your way to AI

- Bases de données NOSQL
  - Performances sur de gros volumes de données
  - Performances sur des données non structurées
  - Évolutivité très importante, même pour de faibles volumes
- Cependant :
  - Technologie encore en évolution, pas de standards
  - Pas de langage de requêtage commun comme le SQL, mais divers
    - Requêtes spécifiques au langage (Java, Python…)
    - Requêtes spéciales pour la base (Cassandra Query Language)
    - API basée sur Map Reduce ou sur des graphes d'objets
  - On doit faire plus de travail au niveau du code, ce qui peut influer sur la performance

Chapitre 3 : Bases de données NOSQL

Section 3 : NOSQL et Hadoop

Aiway

Your way to AI

Remplacer HDFS par NOSQL

Aiway
Your way to AI

- HDFS représente l'un des atouts majeurs de Hadoop car
  - Distribué en cluster
  - Facilement extensible
  - Offre une haute disponibilité
- Mais il présente certains désavantages
  - Utilise un système de stockage direct (DAS et non SAN)
  - Problème de disponibilité pour les utilisateurs des anciennes versions de Hadoop où le NameNode n'est pas dupliqué
  - Si les utilisateurs utilisent déjà une base de données distribuée et ne veulent pas perdre du temps à copier les données
- Plusieurs options sont proposées pour remplacer HDFS, dont l'utilisation de bases NOSQL

NOSQL et MapReduce

Aiway
Your way to AI

- C’est l’approche la plus utilisée
- NOSQL offre des données diversifiées, en grand nombre et de divers types, regroupées dans un endroit unique
- MapReduce pourra parcourir ces données, les filtrer, les traiter et afficher les résultats
  - Profiter des capacités de stockage des bases NOSQL
  - Profiter de la tolérance aux pannes pour éviter la perte de données
  - Extraction facile des données, plus facile que pour un fichier
  - Moins de risque de données erronées ou non conformes
- Les résultats obtenus pourront être stockés dans un fichier texte, une BDD NOSQL (stockage++) ou une BDR (reporting++)

HBase - Définition

Aiway
Your way to AI

- HBase est un sous-projet d'Hadoop, un framework d'architecture distribuée. La base de données HBase s'installe généralement sur le système de fichiers HDFS d'Hadoop pour faciliter la distribution, même si ce n'est pas obligatoire
- Basées sur une architecture maître/esclave, les bases de données de ce type sont capables de gérer d'énormes quantités d'informations
- Allie la force du système de fichier distribué et la puissance des BDD NOSQL

HBase - Caractéristiques

Aiway
Your way to AI

- Base de données clé / valeur montée sur le système HDFS
- Stockage de données non structurées ou semi-structurées

- Deux types de noeuds
  - Master : noeud unique (un seul fonctionne à la fois) qui supervise les opérations du cluster
  - RegionServer : noeuds servant à héberger les données et à réaliser les opérations de lecture et d'écriture
- Pas de réplication des RegionServers mais utilisation de la réplication implémenté dans HDFS

HBase - HFiles

Aiway
Your way to AI

- Stockage des entrées sous forme de fichiers HFiles
- Les clés et les valeurs sont des objets ByteArray
- Les fichiers HFiles sont immuables, car HDFS ne supporte pas la mise à jour d'un fichier existant
- Equilibrage récurrents du nombre de fichiers stockés afin d'assurer une répartition équitable à travers le cluster

Synthèse

Aiway
Your way to AI

- Les bases de données NOSQL sont une création en réaction à la croissance exponentielle des données produites et traitées
- Flexibilité d'évolution et accessibilité accrue au prix de la consistance qui devient éventuelle
- Interfaçage possible avec les techniques de traitement parallèle des données tel que Hadoop