

Bases de données NoSQL

Mohamed-Lamine Messai

mohamed-lamine.messai@univ-lyon2.fr



**INSTITUT
de La
communication**



Université de Lyon, Lyon 2, ERIC UR 3083, Lyon, France

2022-2023

Sommaire

I. SQL vs NoSQL

II. Le big data

III. Modèles de données NoSQL

IV. Atouts et faiblesses du NoSQL

Sommaire

I. SQL vs NoSQL

II. Le big data

III. Modèles de données NoSQL

IV. Atouts et faiblesses du NoSQL

HOW TO WRITE A CV



Leverage the NoSQL boom

Les données en entreprise

- Les entreprises sauvegardent des données et utilisent des logiciels mais en générale les données perdurent dans le temps.
- Utilisation du modèle relationnel pour structurer les données.
- Les entreprises ont besoin d'optimiser le stockage et l'archivage de ces données.
- Les volumes de données dans les entreprises augmentent rapidement.

Émergence du NoSQL

Un peu d'historique

- 1950 : Développement du modèle hiérarchique
- 1970 : Apparition du modèle relationnel (Edgar F. T. Codd)
- 1980's : Domination du modèle relationnel
- 2000's Émergence du terme NoSQL
- 2011 Émergence du NewSQL

La force du modèle relationnel

- Organisation des données selon un modèle mathématique (théorie des ensemble & algèbre relationnelle).
- Résultat : Bases de données stables.
- Le SGBDR assure la cohérence et gère l'accès concurrent aux données.
- Abstraction de l'implémentation physique des données.

Propriétés ACID

ACID = *Atomicity, Consistency, Isolation* et *Durability*

- Tout ou rien
- Les données doivent toujours être cohérentes
- Pas d'interférences entre les transactions
- Les résultats d'une transaction une fois validée sont permanents

Inconvénients du modèle relationnel

- Un modèle rigide.
- Difficulté de passer à l'échelle (*scalability*).

NoSQL : définition

Des systèmes de gestion de base de données de nouvelle génération traitent principalement certains points : être non relationnel, distribués, open source et évolutifs horizontalement.

Quelques SGBD NoSQL



Liste complète : <http://nosql-database.org/>

SQL vs NoSQL

- La principale différence entre les SGBDR et le NoSQL réside dans l'absence de schéma.
- Dans une base NoSQL, des objets appartenant à une même table ne sont pas contraints d'avoir les mêmes colonnes.
- Cette absence de schéma permet d'intégrer des données non structurées.

Faut-il passer au NoSQL ?

SQL vs NoSQL

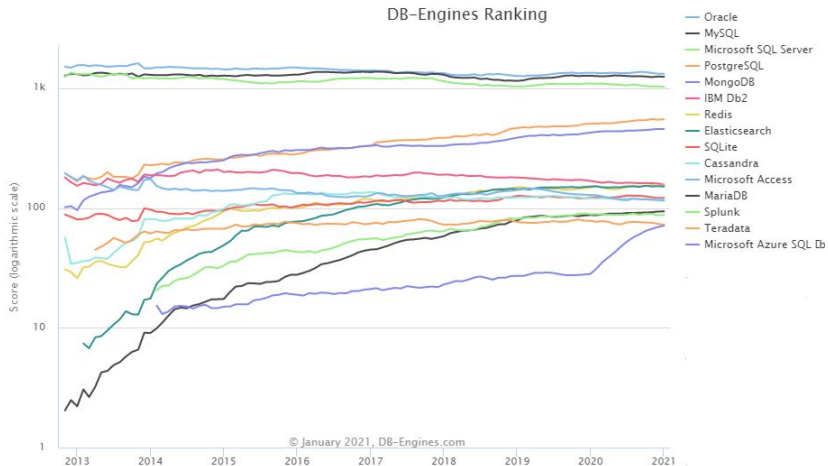
- La principale différence entre les SGBDR et le NoSQL réside dans l'absence de schéma.
- Dans une base NoSQL, des objets appartenant à une même table ne sont pas contraints d'avoir les mêmes colonnes.
- Cette absence de schéma permet d'intégrer des données non structurées.

Faut-il passer au NoSQL ?

les bases de données NoSQL conviennent mieux quand vos données ne rentrent pas bien dans les tables.

Si les données sont difficilement interrogeables en langage SQL.

NoSQL populaires



Sommaire

I. SQL vs NoSQL

II. Le big data

III. Modèles de données NoSQL

IV. Atouts et faiblesses du NoSQL

Contexte

- Les entreprises font face à l'augmentation du volume de données.
- Gérer des pétaoctets!

Un défi qui n'est pas à la portée du modèle relationnel

Les 4 Vs

- Volume
- Vitesse taux de croissance des données et vitesse traitement
- Variété avec plusieurs sources et types (image, vidéo, son...)
- Véracité des données, obsolescence, intégrité et sécurité

Pour le faire

- Distribuer le traitement
- Distribuer les données

Sommaire

I. SQL vs NoSQL

II. Le big data

III. Modèles de données NoSQL

IV. Atouts et faiblesses du NoSQL

Théorème de Brewer

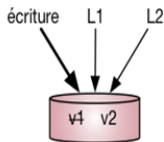
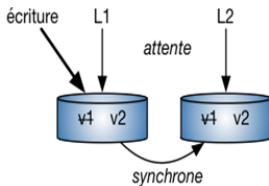
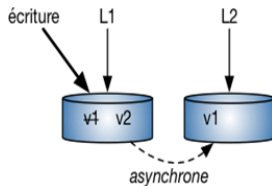
Théorème de Brewer dit théorème de CAP.

En 2000, Eric A. Brewer a formalisé un théorème très intéressant reposant sur 3 propriétés fondamentales pour caractériser les bases de données (relationnelles, NoSQL et autres)

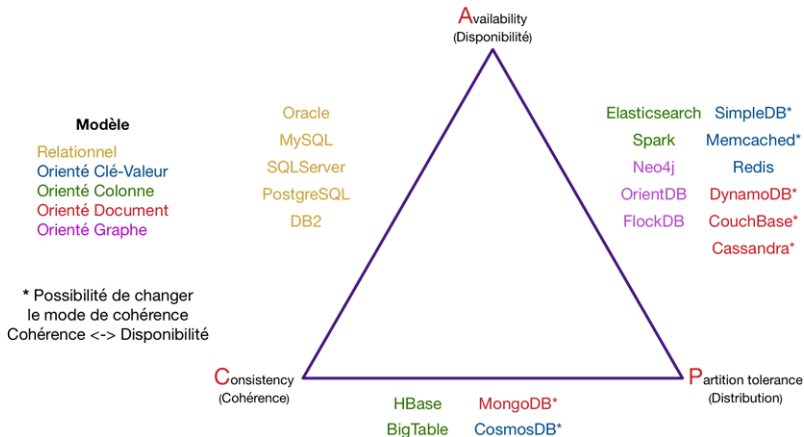
- *Consistency* (Cohérence) : Une donnée n'a qu'un seul état visible quel que soit le nombre de répliquas.
- *Availability* (Disponibilité) : Tant que le système tourne (distribué ou non), la donnée doit être disponible.
- *Partition Tolerance* (Distribution) : Quel que soit le nombre de serveurs, toute requête doit fournir un résultat correct.

Théorème de Brewer

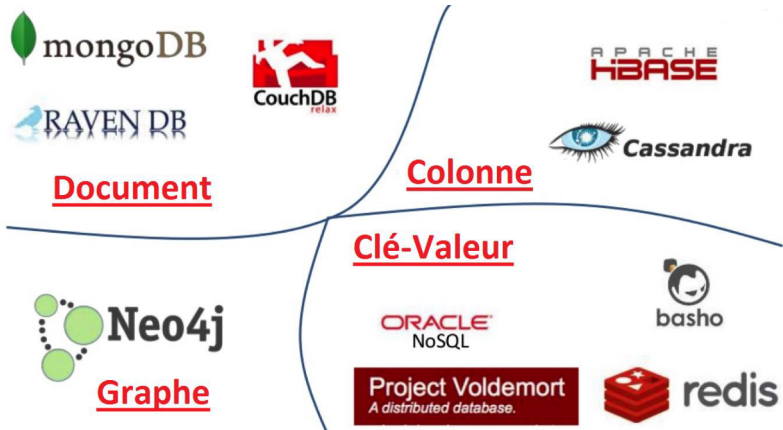
- Dans toute base de données, vous ne pouvez respecter au plus que 2 propriétés parmi la cohérence, la disponibilité et la distribution.

CA*Cohérence + Disponibilité***CP***Cohérence + Distribution***AP***Disponibilité + Distribution*

Le triangle de CAP



Les différents familles NoSQL



Clé-Valeur

- Stocker les données sous la forme d'un couple clé-valeur.
- La clé identifie la donnée de manière unique et permet de la gérer.
- La valeur contient n'importe quel type de données.

Exemples :

- Redis : Samsung, Nokia, ...
- SimpleDB : Amazon

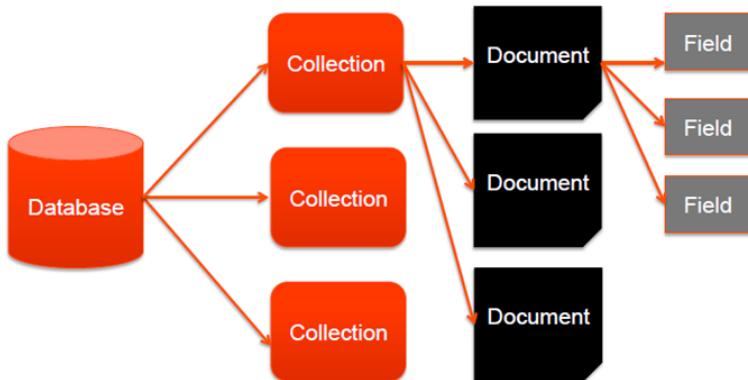
Document

- Données stocké sous la forme de document
- Chaque document est identifié de manière unique par un ID
- Une base de données = collection de documents

Exemples :

- MongoDB : eBay, Cisco
- DynamoDB : Amazon

Document



Colonne

- Stockage sur disque des colonnes au lieu des lignes.
- Modèle proche d'une table dans un SGBDR mais ici le nombre de colonnes est dynamique.
- Le nombre de colonnes peut varier d'un enregistrement à un autre ce qui évite de retrouver des colonnes ayant des valeurs NULL => données éparses

Exemples :

- BigTable : Google
- Hbase : Hadoop

Graphe

- Représente un réseau, relations entre les données.
- Un ensemble de nœuds ou sommets reliés par des liens pouvant posséder une direction.

Exemples :

- Neo4j : HP
- FlockDB : Twitter

Sommaire

I. SQL vs NoSQL

II. Le big data

III. Modèles de données NoSQL

IV. Atouts et faiblesses du NoSQL

Atouts

- Modèles de données flexibles

Atouts

- Modèles de données flexibles
- Adapté au big data et aux architectures distribuées

Atouts

- Modèles de données flexibles
- Adapté au big data et aux architectures distribuées
- Disponibilité continue des données

Faiblesses

- 10-12 ans d'existence vs +50 pour les SGBDR => moins d'outils, moins mature ?

Faiblesses

- 10-12 ans d'existence vs +50 pour les SGBDR => moins d'outils, moins mature ?
- Pas de langage de requêtage unifié (tel que SQL)

Fin

Merci de votre



Attention