



## Big Data & NoSQL

– Chapitre 2–

# Les Bases de Données NoSQL

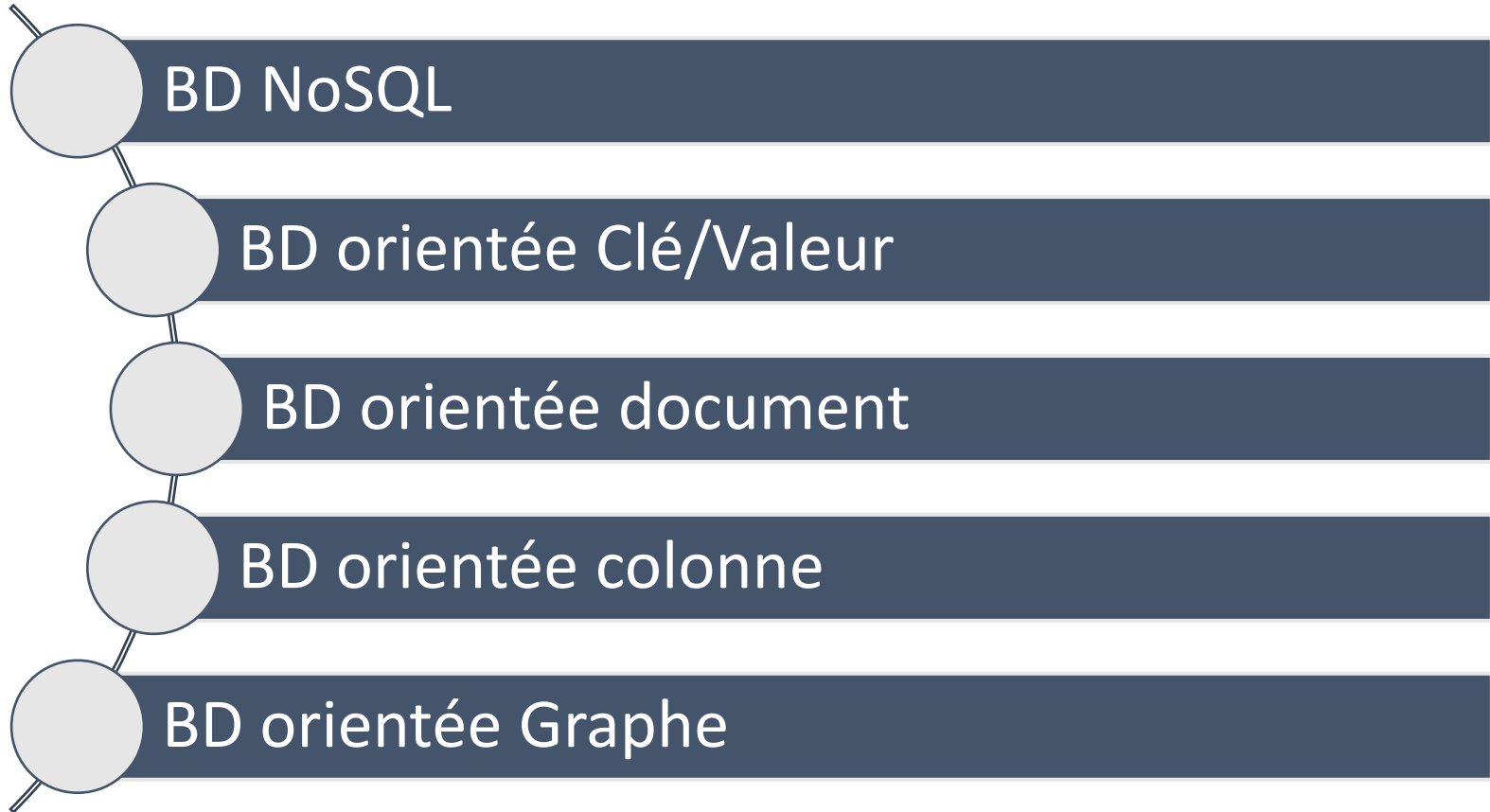
**Dr. GHEMMAZ W**

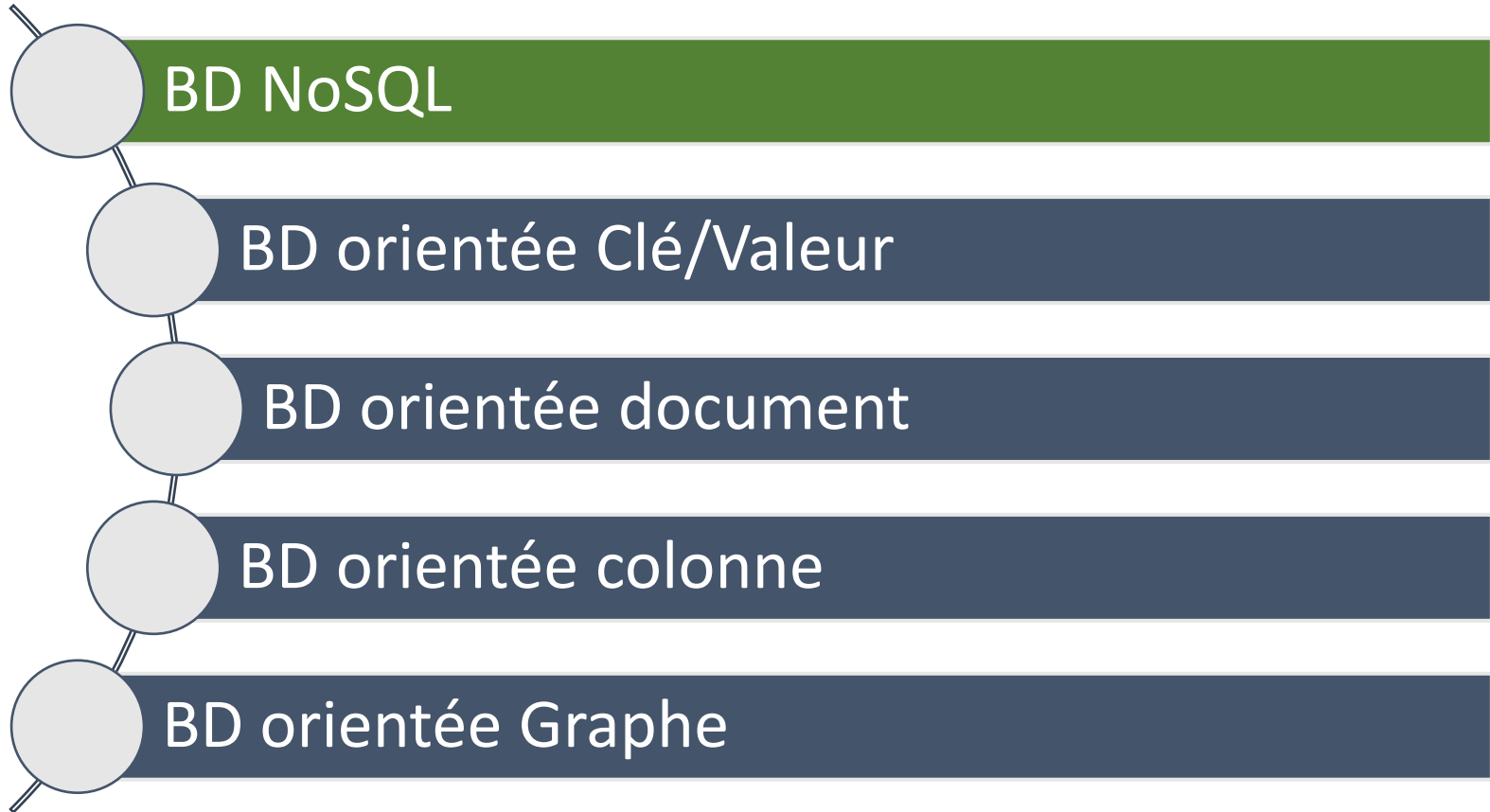
NTIC Faculty

Wafa.ghemmaz@univ-constantine2.dz

### Etudiants concernés

Faculté/Institut	Département	Niveau	Spécialité
NTIC	TLSI	Master 1	SDSI





# BDs NoSQL

## Objectif des BDs NoSQL

The logo for NoSQL, featuring the word "Not" in black, "only" in black, and "SQL" in black. The "N" and "O" are large, red, and stylized, with the "O" having a white outline. The "S" is also large and black.

L'objectif derrière chaque type de base de données NoSQL est de **faciliter le stockage et le traitement de volumes massifs de données.**

# BDs NoSQL

## Caractéristiques des BDs NoSQL

Les bases de données NoSQL sont par nature non relationnelles. Elles sont donc **plus évolutives** que les bases de données relationnelles

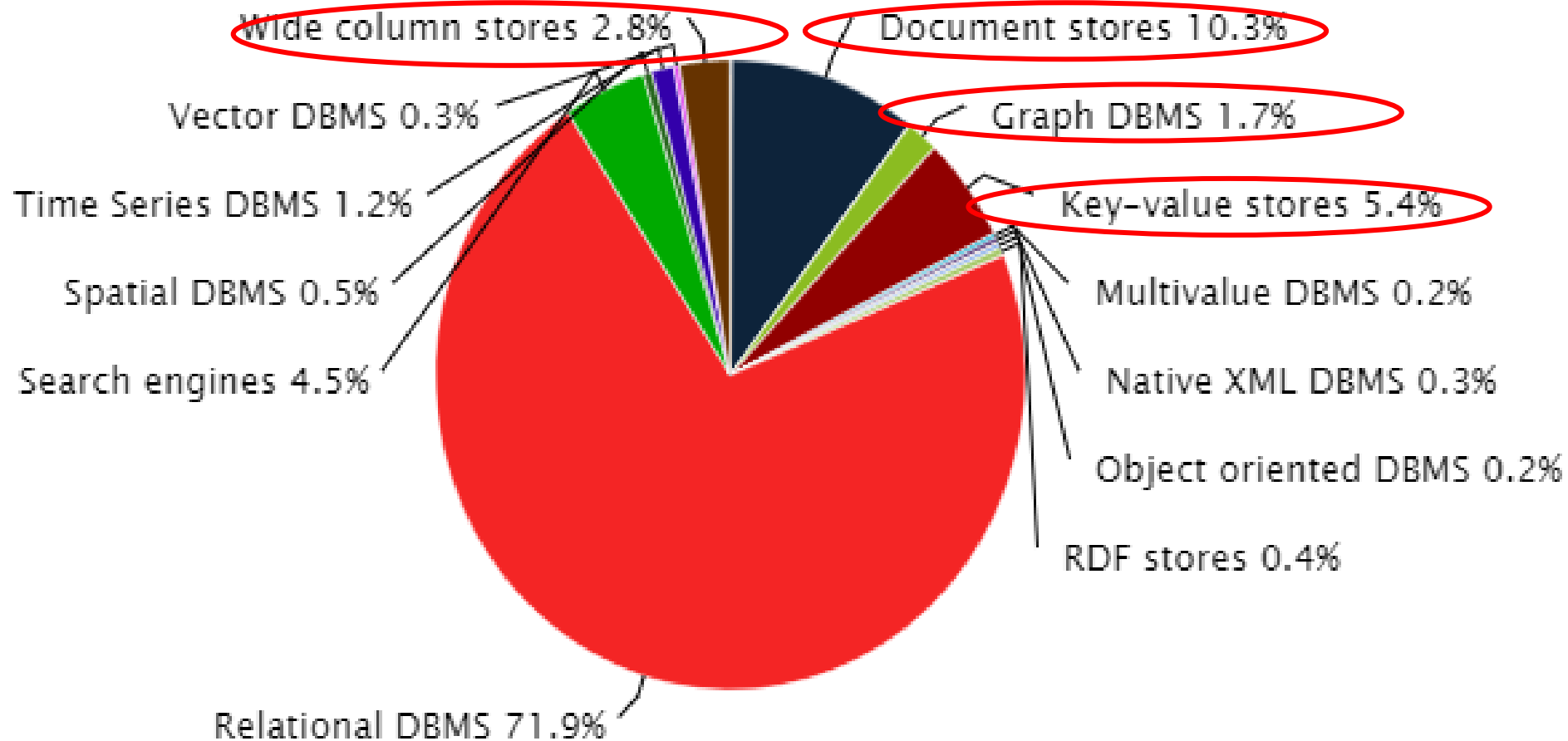
**N**ot **O**nly SQL

Les deux caractéristiques distinctives de tout type de base de donnée NoSQL sont :

- **Leur caractère non-relationnel** : les bases de données NoSQL ne suivent jamais le modèle relationnel.
- **Le “sans schéma”** : les bases de données NoSQL sont “sans schéma” (ou ont des schémas assouplis). Elles permettent ainsi d’obtenir des structures de données hétérogènes dans le même domaine.

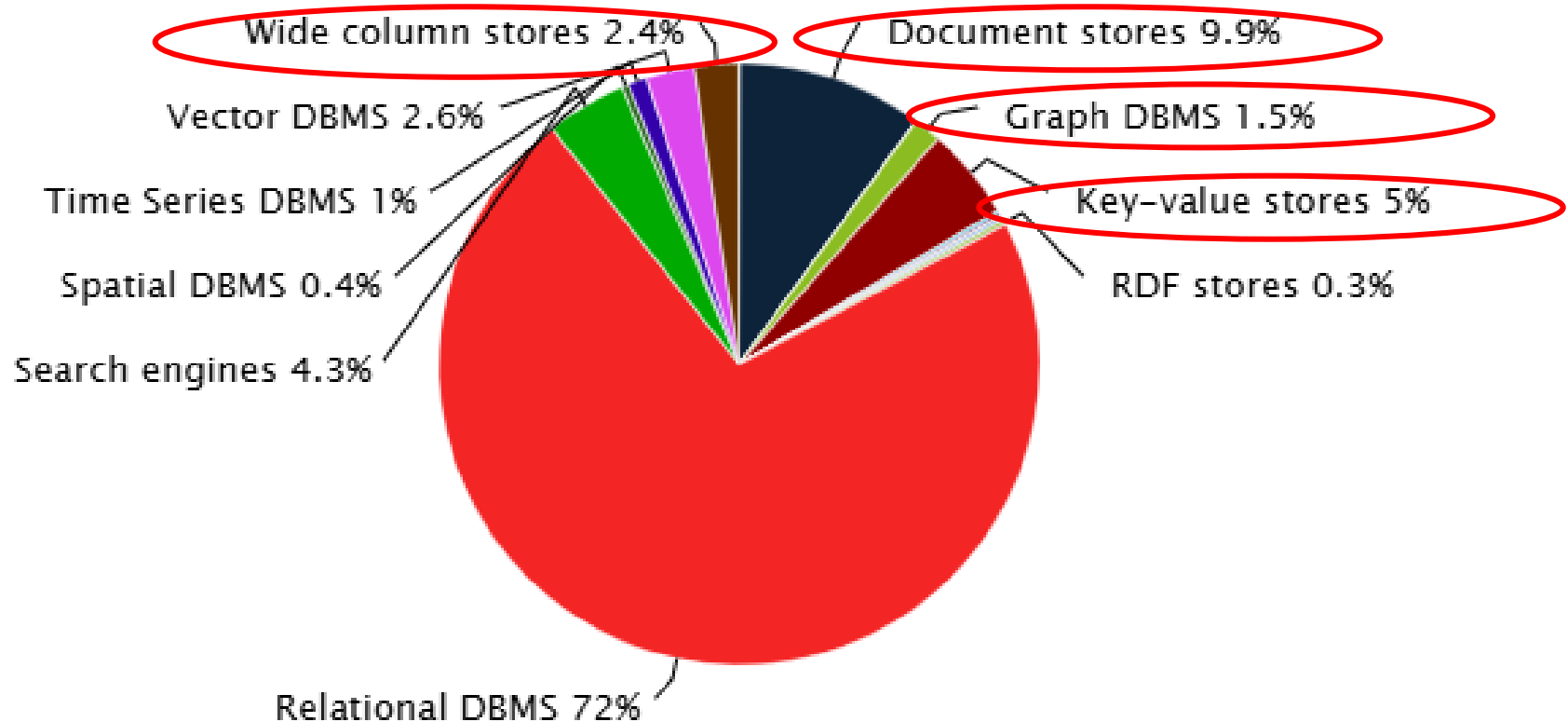
# Types des BDs NoSQL

**Ranking scores per category in percent, February 2024**



# Types des BDs NoSQL

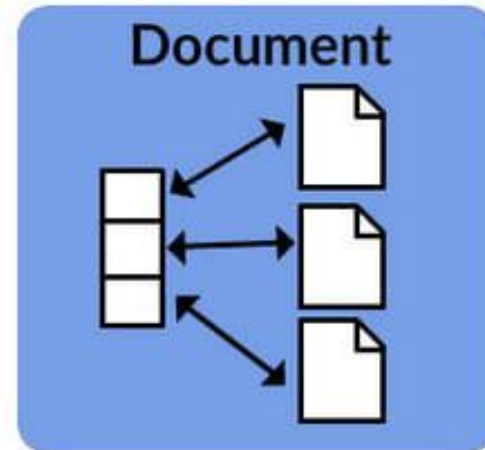
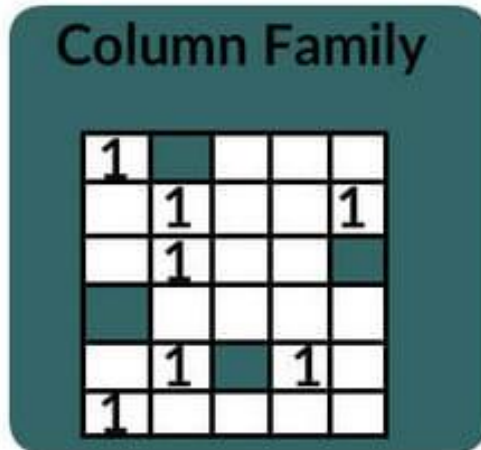
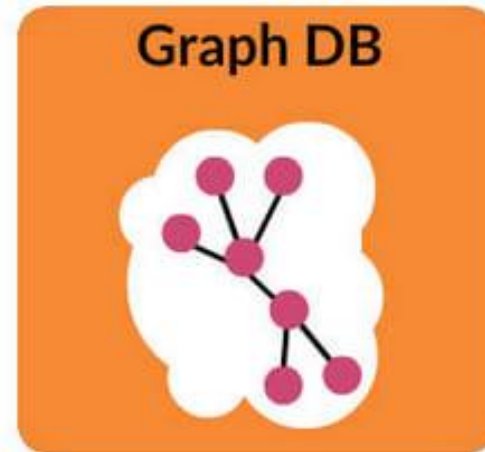
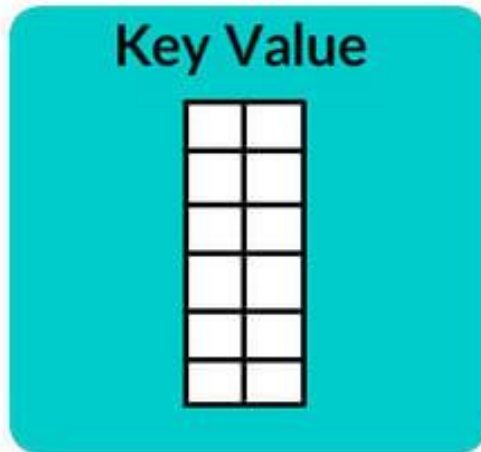
## Ranking scores per category in percent, February 2024



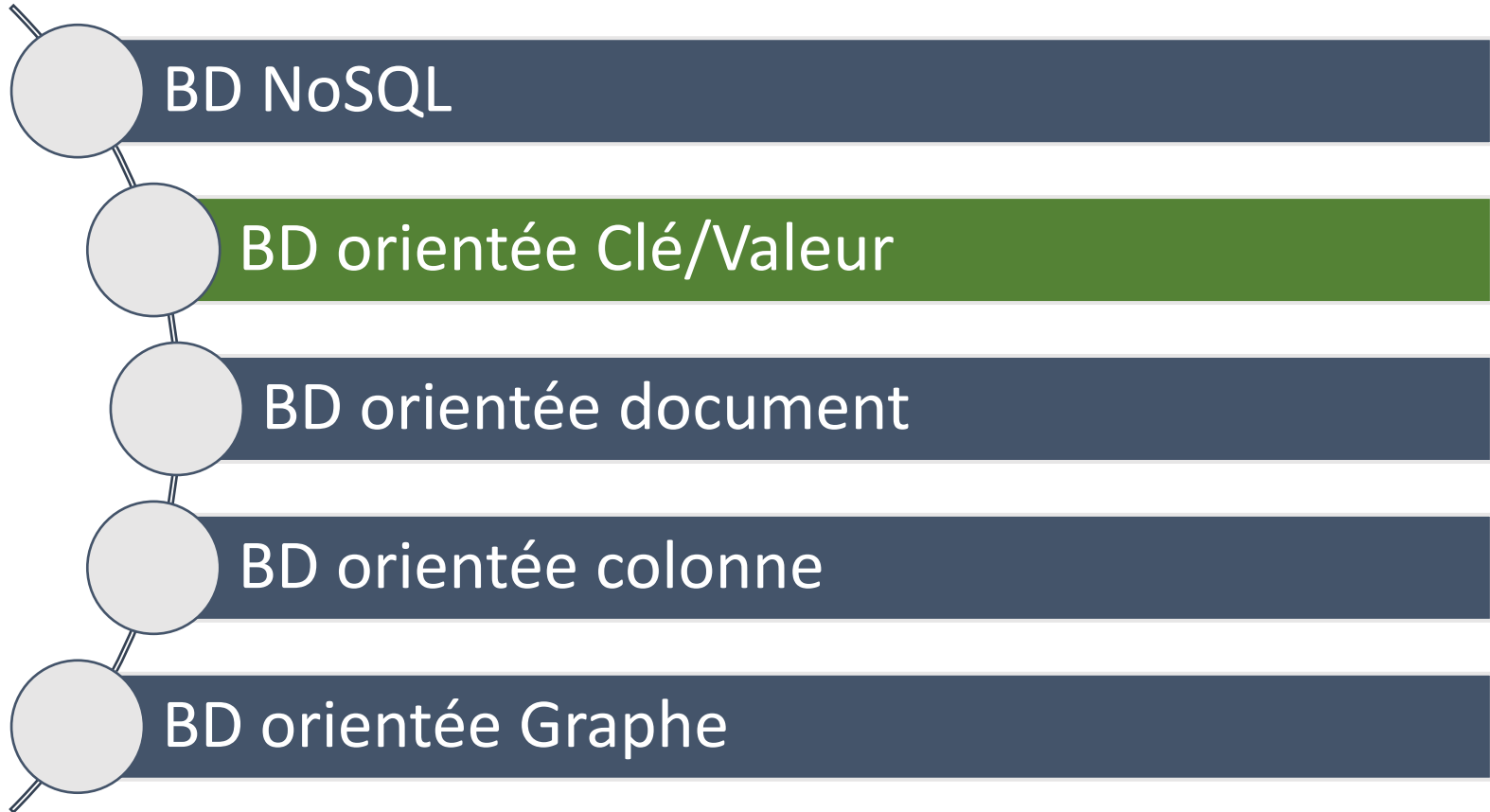
© 2025, DB-Engines.com

# Principaux Types des BDs NoSQL

Il existe **quatre types principaux** de BDs NoSQL







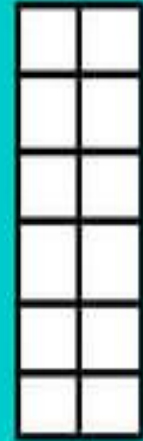
# BDs orientées Clé/Valeur

## Clé/Valeur (Key-value stores)

Il s'agit de la représentation la plus simple. Cette structure est très adaptée à la gestion de cache pour fournir un accès rapide aux informations.

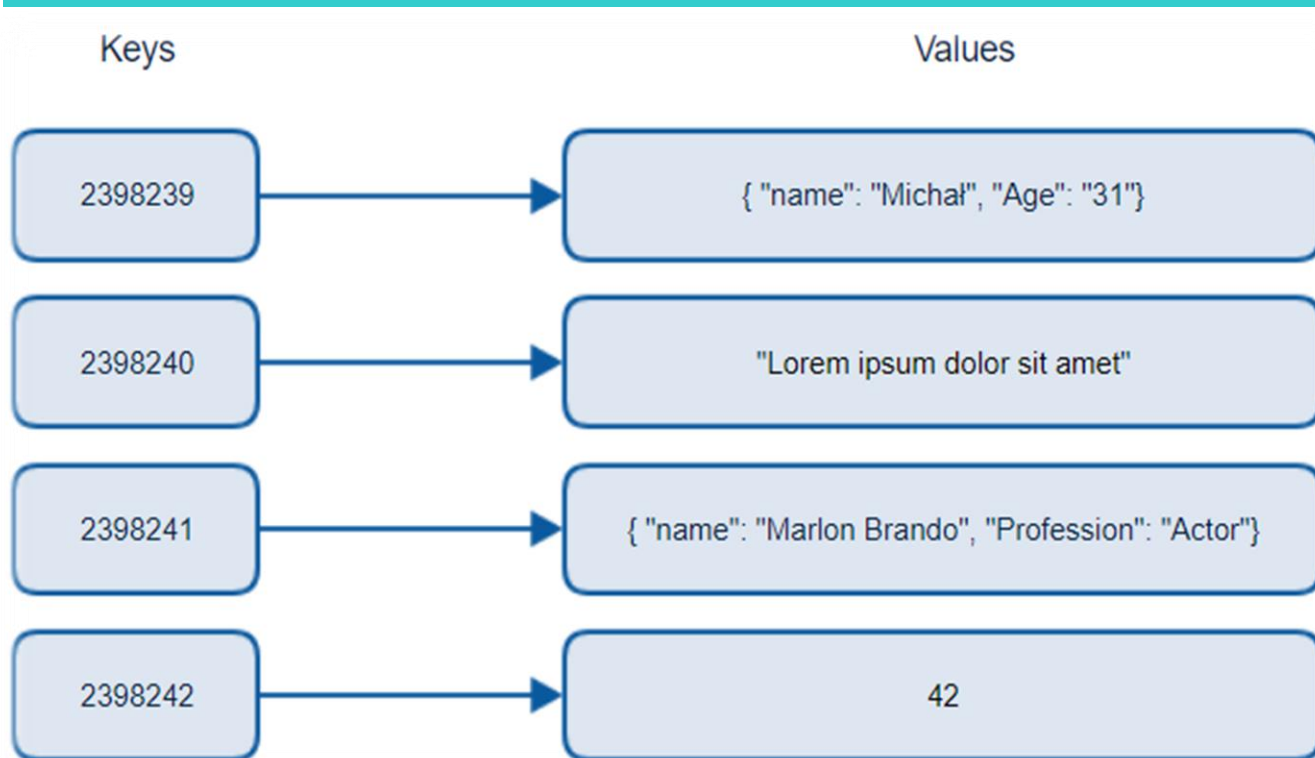
Cette représentation fonctionne comme un grand tableau associatif et retourne une valeur ( de nature éventuellement complexe) à partir d'une clé.

Key Value




# BDs orientées Clé/Valeur

## Clé/Valeur

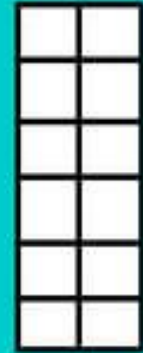


## Key Value


# BDs orientées Clé/Valeur

## Clé/Valeur

Key Value



### Les avantages :

- Sa simplicité ;
- Son caractère évolutif ;
- Son performance;
- Pas de contraintes -> pas de contrôle ;

### Les inconvénients:

- Recherche uniquement par Clé
- Absence complète de structure

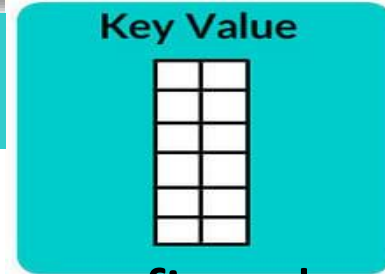
Id	Nom	Prénom	Age
Etd112	AAA	Ahmed	22
Etd113	BBB	Ali	23
Etd114	CCC	Sara	19
Etd115	DDD	meriem	30

Etd112

Nom\$\$AAA~~Prenom\$\$Ahmed~~Age\$\$22

# BDs orientées Clé/Valeur

## Clé/Valeur



### Principaux cas d'utilisation:

- Le stockage des informations de session, afin de sauvegarder et de restaurer des sessions d'utilisateur

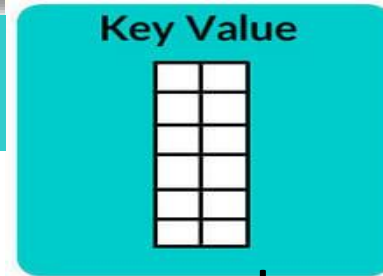
**Exemple : Un site web utilise Redis pour stocker les sessions des utilisateurs connectés.**

**Clé : session:12345**

**Valeur: { "user\_id": 789, "last\_login": "2025-02-24 10:30:00" }**

# BDs orientées Clé/Valeur

## Clé/Valeur



### Principaux cas d'utilisation:

- **Les préférences ou Profils des utilisateurs**, pour stocker les données spécifiques à un utilisateur particulier ;

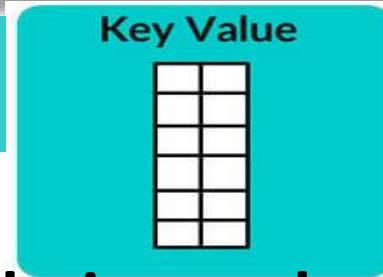
**Exemple : Une application de streaming pour stocker les préférences des utilisateurs.** L'application charge immédiatement le bon thème et les recommandations personnalisées au moment de la connexion.

**Clé : user:789**

**Valeur : { "name": "Alice", "theme": "dark", "language": "fr", "recently\_watched": ["movie\_1", "movie\_2"] }**

# BDs orientées Clé/Valeur

## Clé/Valeur



### Principaux cas d'utilisation:

- **Les paniers e-commerce et recommandations de produits.** Ce type de BD NoSQL peut proposer des recommandations produit basées sur les données comportementales d'un internaute.

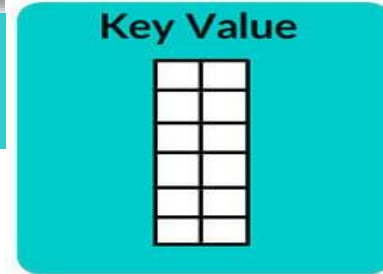
**Exemple: Un site e-commerce comme Amazon utilise Cassandra pour stocker temporairement les paniers d'achats. Le panier est sauvegardé même si l'utilisateur quitte le site et revient plus tard.**

**Clé : cart:456**

**Valeur: { "user\_id": 789, "items": ["product\_123", "product\_456"], "total\_price": 49.99 }**

# BDs orientées Clé/Valeur

## Clé/Valeur



### Principaux cas d'utilisation:

- Les données des capteurs

### Exemple:

Une entreprise gère un réseau de capteurs de température et stocke les relevés dans Redis afin d'accéder rapidement aux dernières valeurs enregistrées pour la surveillance en temps réel.

Clé : sensor:temperature:001

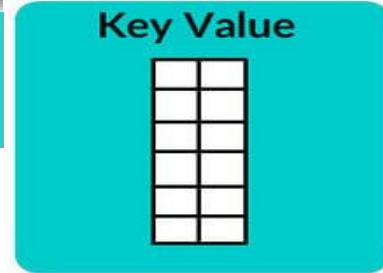
Valeur : { "timestamp": "2025-02-24 10:45:00", "value": 22.5 }



# BDs orientées Clé/Valeur

## Clé/Valeur

### Les produits sur marché

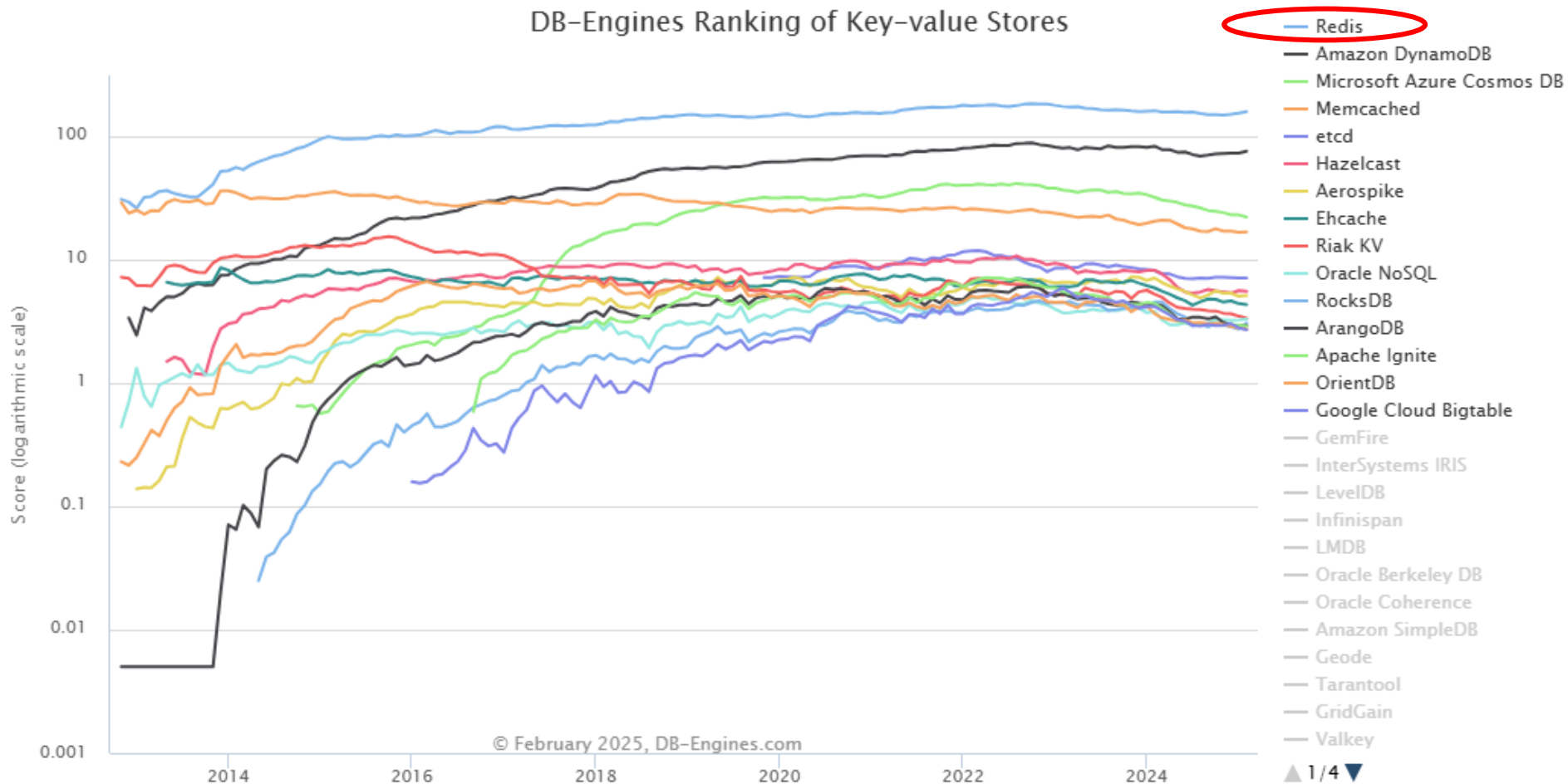


Amazon DynamoDB



# Principaux Types des BDs NoSQL

## BDs orientées Clé/Valeur



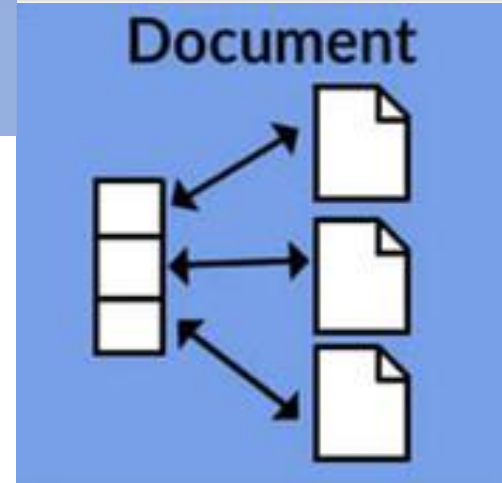
# Principaux Types des BDs NoSQL

## Document (Document database)

Une représentation qui ajoute au modèle Clé-valeur, l'association d'une valeur complexe.

Ce type de BD NoSQL orientée document stocke, comme son nom l'indique, les données dans des **documents semi-structurés** JSON, BSON ou XML.

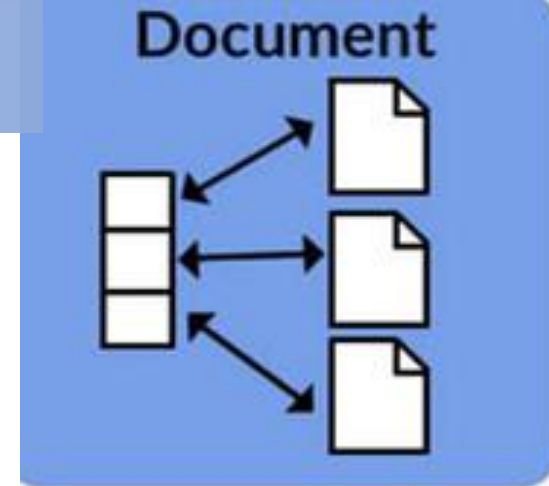
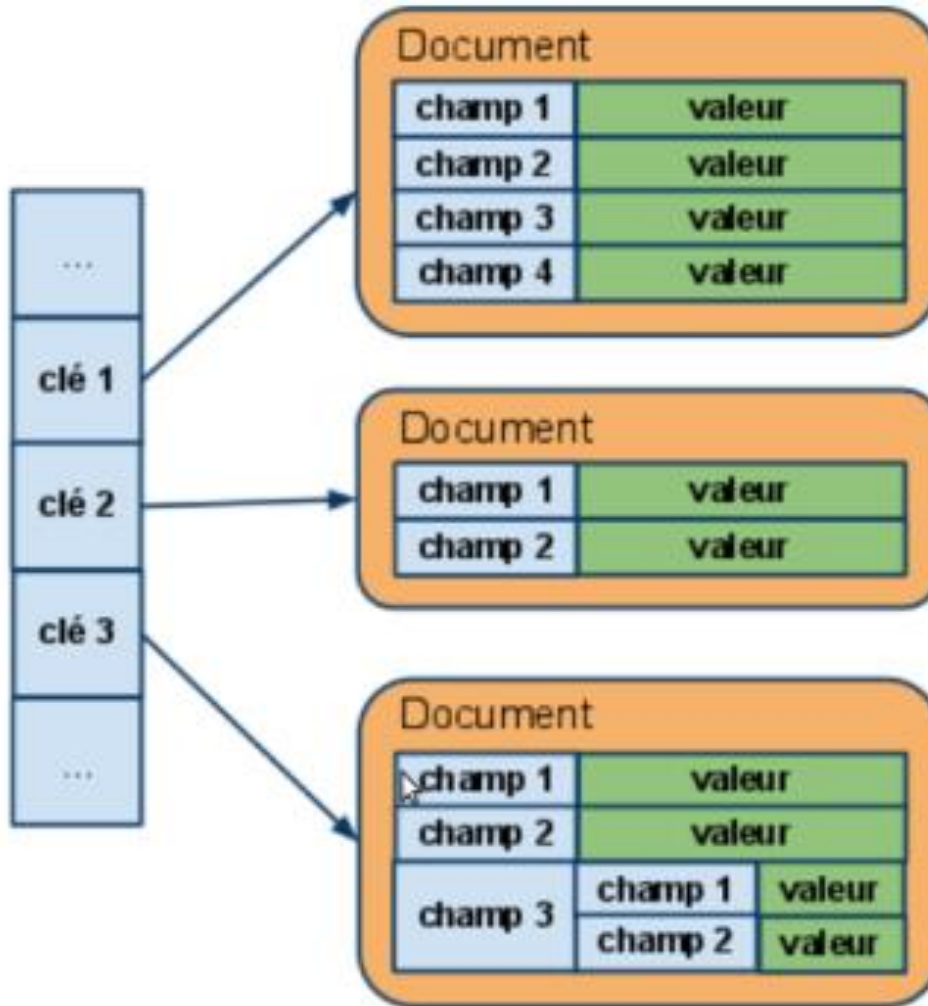
Les documents peuvent être imbriqués, et des éléments particuliers indexés pour une interrogation plus rapide.



La structure de JSON est peu verbeuse par rapport à XML

# Principaux Types des BDs NoSQL

## Document



# Principaux Types des BDs NoSQL

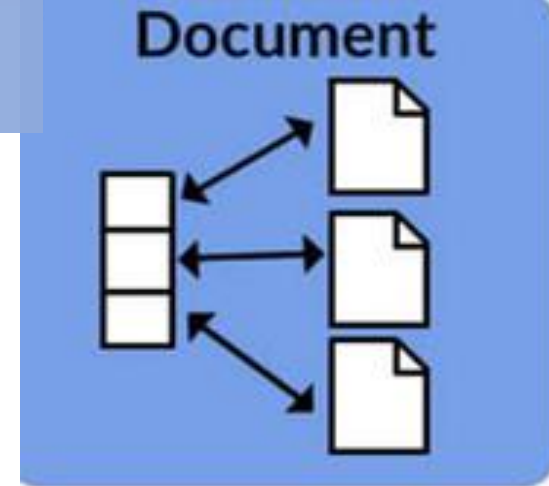
## Document

### Les avantages :

- Son simplicité;
- Modèle Puissant;
- Modèle évolutif;

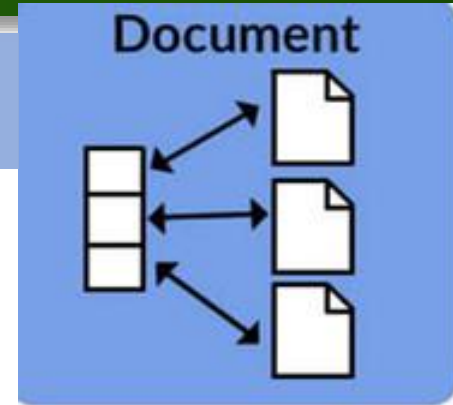
### L'inconvénient:

Cette base ne convient pas aux données relationnelles, autrement dit il n'est pas possible de lier des données ensemble.



# Principaux Types des BDs NoSQL

## Document

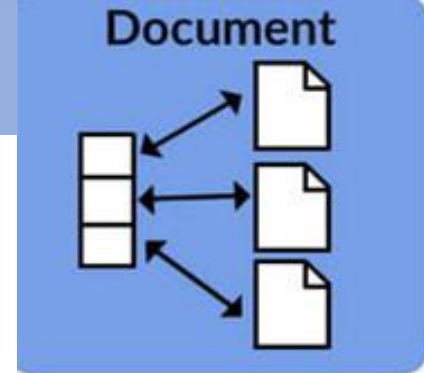


### Principaux cas d'utilisation:

- **La gestion des profils d'utilisateurs :**  
leur caractère flexible permet aux utilisateurs de stocker différents types d'informations ;
- **La gestion du contenu :** la collecte et le stockage de toutes les données sont facilités par ce type de base de donnée NoSQL. Elle permet donc la création de nouveaux types de contenus, notamment des images, des vidéos, des commentaires, etc. On les utilise ainsi souvent pour les plateformes de blog.

# Principaux Types des BDs NoSQL

## Document



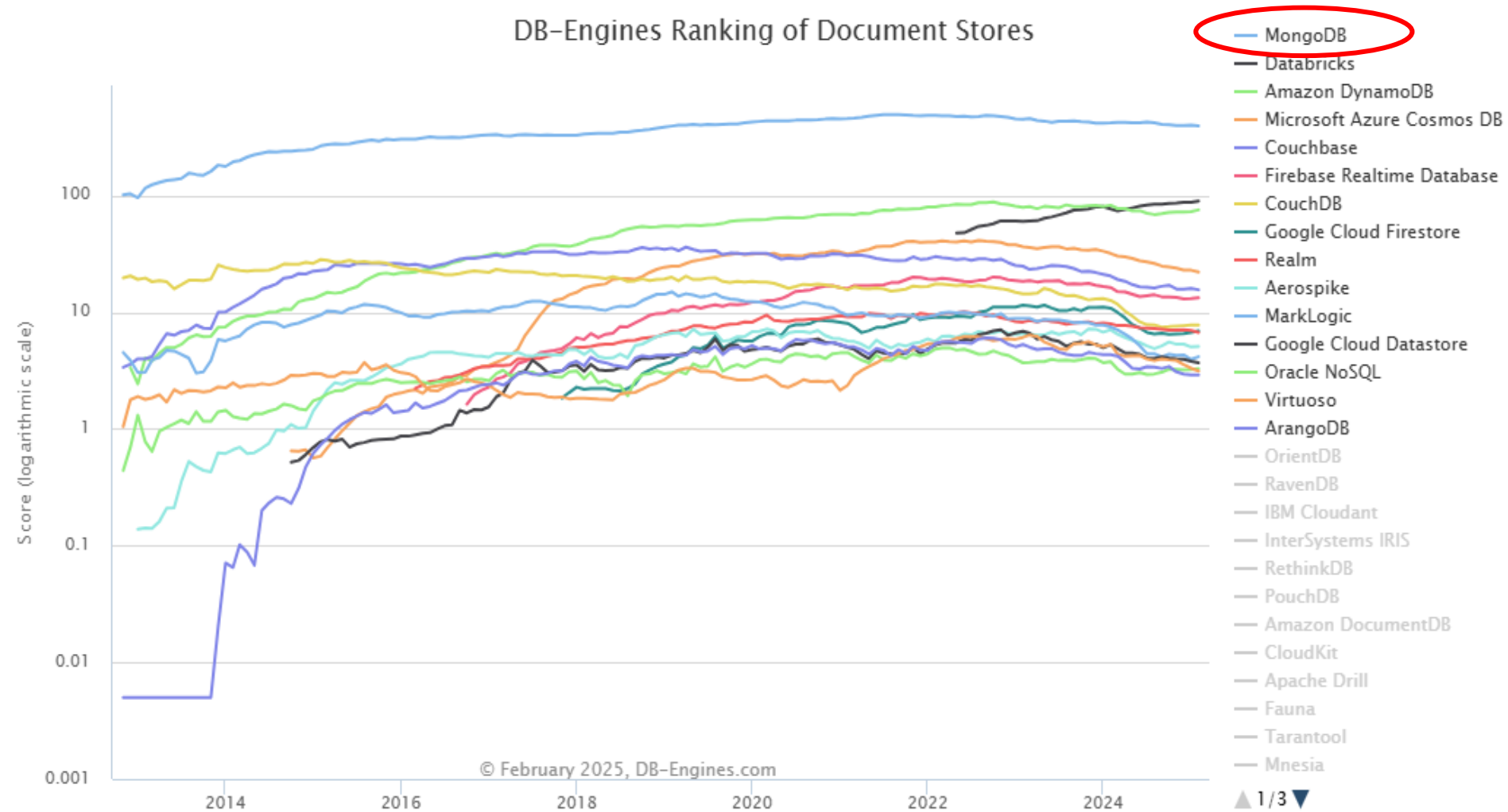
## Les produits sur marché



Lotus Notes  
Storage Facility

# BDs orientées Document

DB-Engines Ranking of Document Stores





# Principaux Types des BDs NoSQL

## Colonne

Une évolution du modèle Clé/Valeur qui permet de disposer de plusieurs colonnes, permettant ainsi de stocker les relations de type **one-to-many** (1-n)

Column Family

1				
	1			1
	1			
	1		1	
1				

# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Id	Nom	Prénom	Age
Etd112	Null	Ahmed	22
Etd113	BBB	Ali	Null
Etd114	CCC	Null	19
Etd115	DDD	meriem	Null

### Column Family

1				
	1			1
	1			
	1		1	
1				

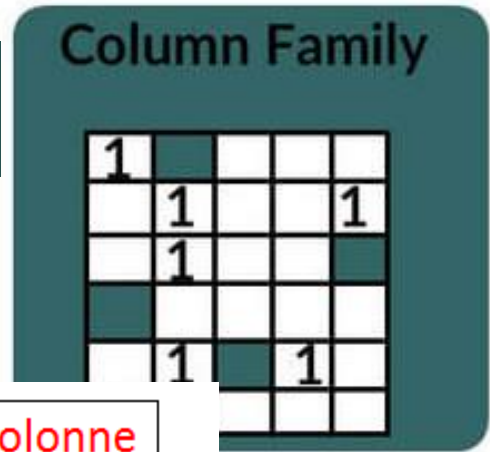
**Insertion par  
ligne  
(BDR: Row-  
Based System)**

```
Etd112 ; Null ; Ahmed ; 22 //  
Etd113 ; BBB ; Ali ; Null //  
Etd114 ; CCC ; Null ; 19//  
Etd115 ; DDD ; Meriem ; Null//  
.....
```

On insère par ligne lorsque la structure est prédéfinie au départ.  
La structure change assez rarement.

# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)



Id	Nom	Prénom	Age	Nouvelle colonne
Etd112	Null	Ahmed	22	Null
Etd113	BBB	Ali	Null	Null
Etd114	CCC	Null	19	Null
Etd115	DDD	meriem	Null	Null

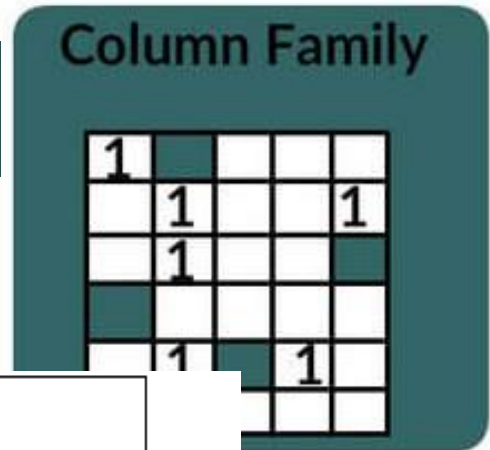
```
Etd112 ; Null ; Ahmed ; 22 , Null //
Etd113 ; BBB ; Ali ; Null , Null //
Etd114 ; CCC ; Null ; 19 , Null //
Etd115 ; DDD ; Meriem ; Null , Null //
.....
```

(BDR: Row-  
Based System)

# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Les BDs orientées colonne sont utilisées afin d'éviter le stockage des Null



Id	C1	C2	C3	C4
1	Null	6	7	Null
2	10	4	Null	2
3	1	Null	19	Null
4	Null	3	Null	Null

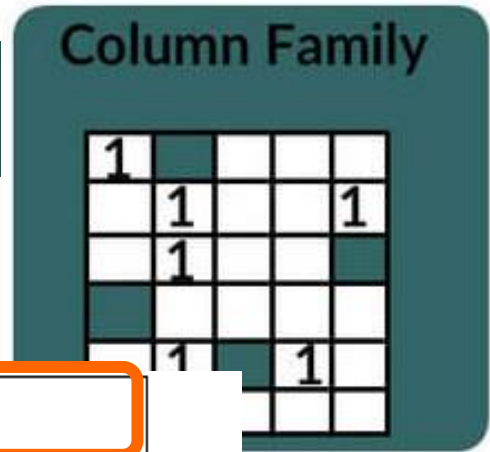
2:C1:10 ; 3:C1:1; 1:C2:6 ; 2:C2:4 ; 4:C2:3 ;  
1:C3:7 ; 3:C3:19 ; 2:C4:2 ;

(BDNoSQL:  
Column-Based  
System)

# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Les BDs orientées colonne sont utilisées afin d'éviter le stockage des Null



Id	C1	C2	C3	C4
1	Null	6	7	Null
2	10	4	Null	2
3	1	Null	19	Null
4	Null	3	Null	Null

Famille

Fichier1

( Column-Based System)  
Exp: Hbase

2:C1:10 ; 3:C1:1; 1:C2:6 ; 2:C2:4 ; 4:C2:3 ;

1:C3:7 ; 3:C3:19 ; 2:C4:2 ;

Fichier2

# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Column Family

1				
	1			1
	1			
	1		1	

Id	C1	C2	C3	C4
1	Null	6	7	Null
2	10	4	Null	2
3	1	Null	19	Null
4	Null	3	Null	Null

Famille

( Select id  
Where C1>5)

2:C1:10 ; 3:C1:1; 1:C2:6 ; 2:C2:4 ; 4:C2:3 ;  
1:C3:7 ; 3:C3:19 ; 2:C4:2 ;

On parcourt que les données de la colonne C1

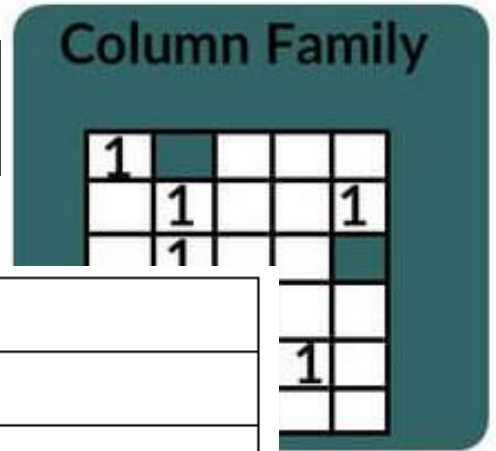
# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Id	C1	C2	C3	C4
1	Null	6	7	Null
2	CNE	4	Null	2
3	CNE	Null	19	Null
4	CNE	3	Null	Null

# Compression

2,3,4:C1:CNE ; 1:C2:6 ; 2:C2:4 ; 4:C2:3 ;  
1:C3:7 ; 3:C3:19 ; 2:C4:2 ;



# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

### Column-family

	Personal			Professional		
	Date de naissance	Nom	Prénom	Grade	Poste	Salaire
1						
2						
3						
4						
5						
6						

Column Family

1				
	1			1
	1			
1			1	

Familles

Les BDs orientées colonne sont utilisées quand on a beaucoup de colonnes. Ces dernières doivent être triées.

La recherche séquentielle est évitée. On utilise plutôt la recherche dichotomique

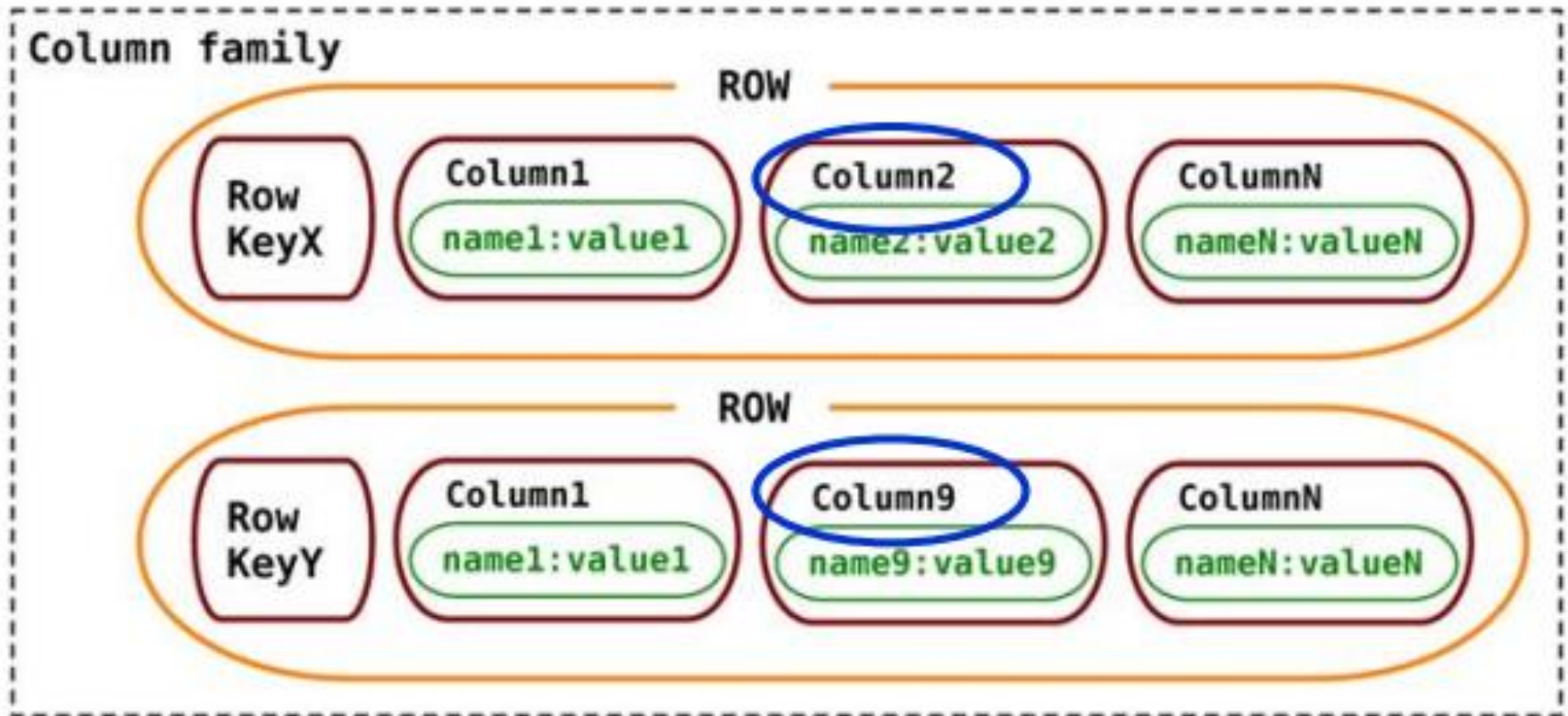


# Principaux Types des BDs NoSQL

## Colonne

Column Family

1			
	1		1
	1		



# Principaux Types des BDs NoSQL

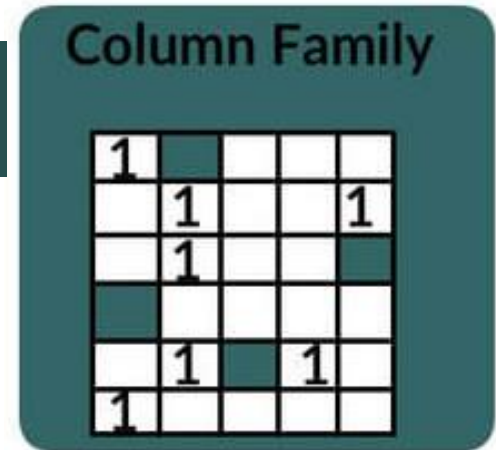
## Colonne (Column-oriented databases)

### Les avantages :

- Elles permettent de prendre en charge des données semi-structurées ;
- Ce modèle de base est évolutif.

### L'inconvénient:

Moins adaptées aux opérations transactionnelles complexes.

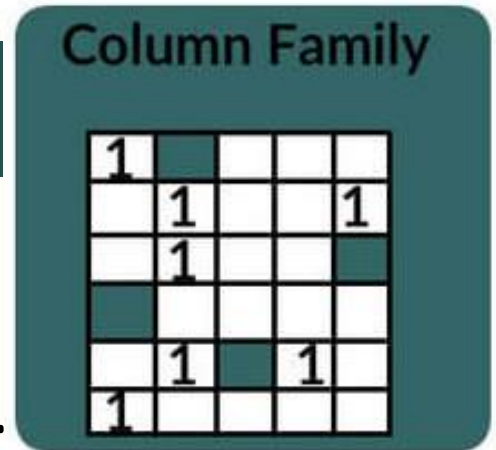


# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

### Principaux cas d'utilisation:

- **Le suivi de colis**, ou tout événement dont le statut est amené à changer régulièrement;
- **La récupération et l'analyse de données en temps réel.** Et notamment les données issues de capteurs ou utilisées pour l'Internet des Objets.



# Principaux Types des BDs NoSQL

## Colonne (Column-oriented databases)

Column Family

1				
	1			1
	1			
	1		1	
1				

## Les produits sur marché

Google's  
BigTable



HYPERTABLE

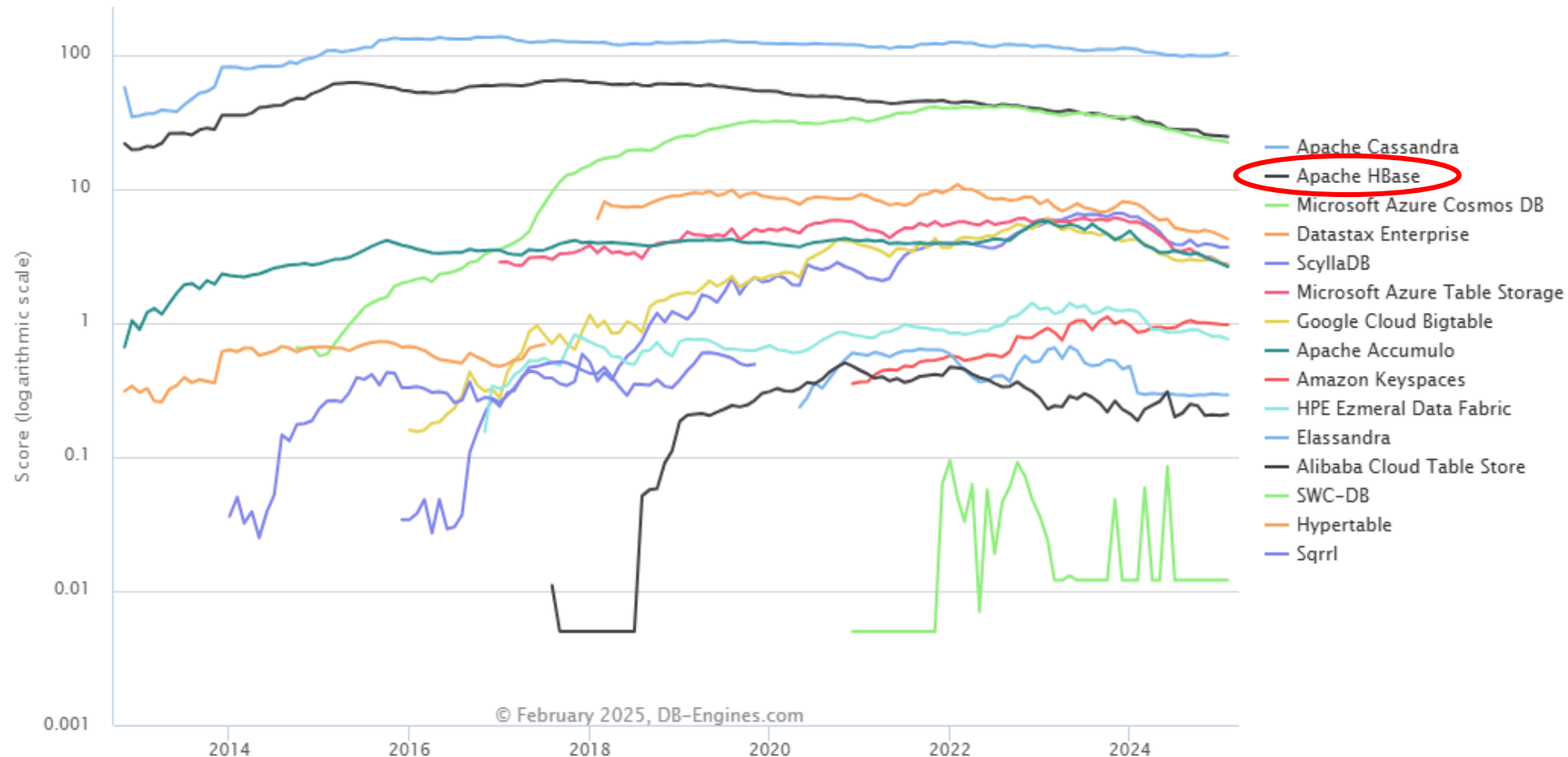


SimpleDB

# Principaux Types des BDs NoSQL

## BDs orientées Colonne

DB-Engines Ranking of Wide Column Stores



# Principaux Types des BDs NoSQL

## Graphe (Graph Databases).

Une représentation qui permet la modélisation, le stockage et la manipulation de données très complexes liées par des relations variées.

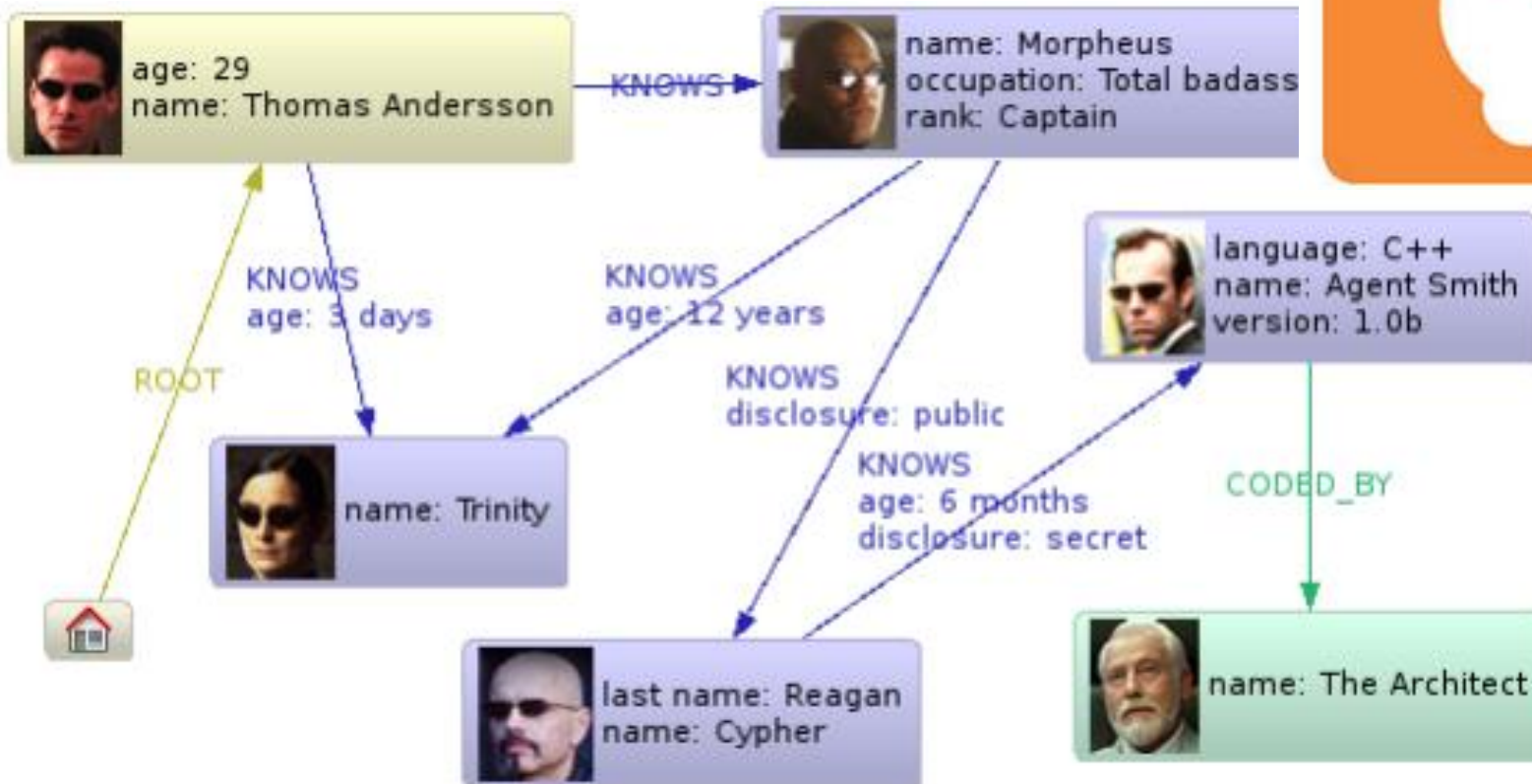
Graph DB



# Principaux Types des BDs NoSQL

## Graphe

### Graph DB





# Principaux Types des BDs NoSQL

## Graphe

### Graph DB



### Principaux cas d'utilisation:

- **Les réseaux sociaux** : par exemple, pour représenter les relations entre abonnés sur Twitter ou Instagram ;
- **Les moteurs de recommandation**
- **Les bases de données logistiques**, pour suivre tous les processus, de la conception à la vente ;
- **L'évaluation des risques** : notamment la détection des fraudes et la recherche de pannes.



# Principaux Types des BDs NoSQL

## Graphe

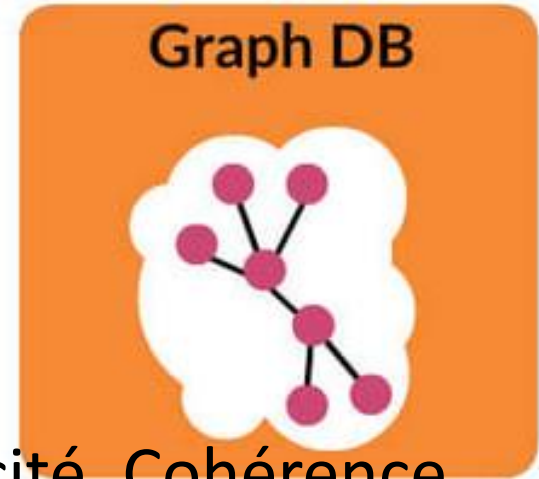
### Principaux cas d'utilisation:

#### Les avantages :

- La puissance et l'agilité ;
- Leur capacité à fournir de l'ACID (Atomicité, Cohérence, Isolation, Durabilité) ;
- Des résultats en temps réel.

#### Les inconvénients:

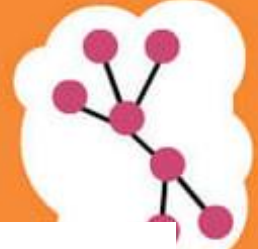
- Le Changement d'une propriété sur tous les nœuds n'est pas une opération simple
- Certaines BD orientées graphe peuvent ne pas être en mesure de gérer un grand nombre de données. De plus, la distribution d'un graphe est difficile.



# Principaux Types des BDs NoSQL

## Graphe

Graph DB



Les produits sur marché

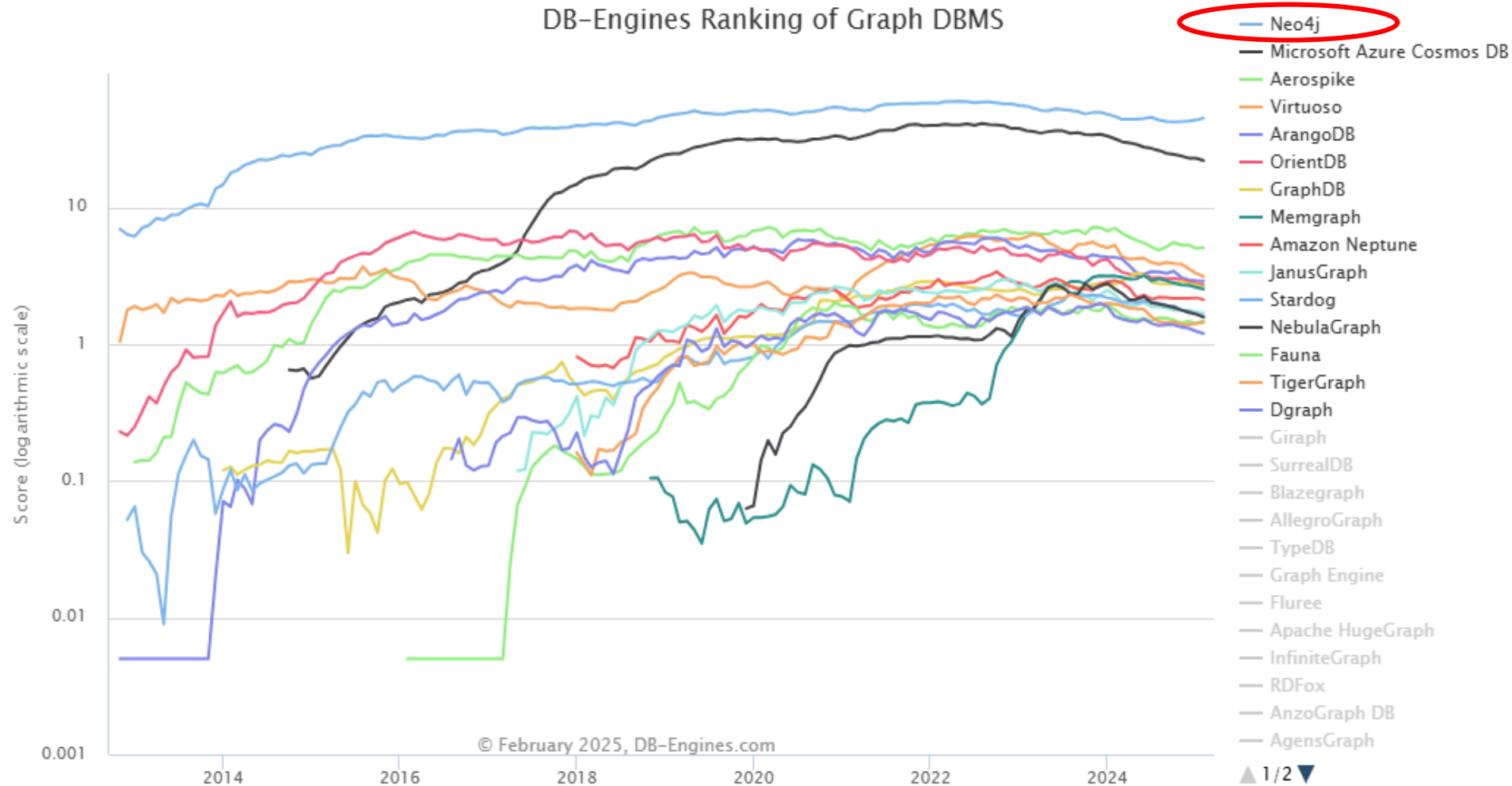


**FlockDB**

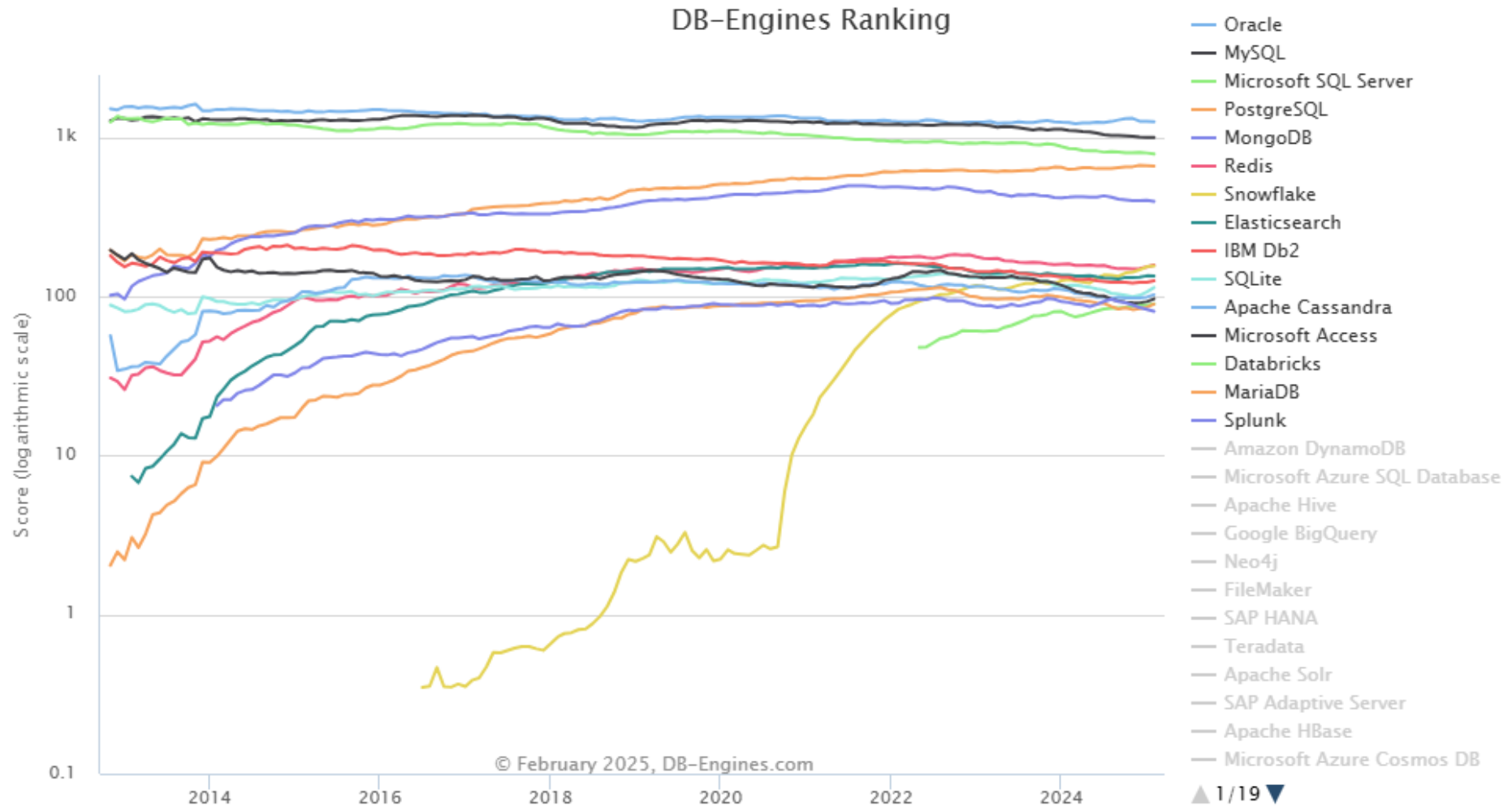
# Principaux Types des BDs NoSQL

## BDs orientées graphe

DB-Engines Ranking of Graph DBMS



# Principaux Types des BDs NoSQL



# Principaux Types des BDs NoSQL

Nom	Langage	Date création	Langage d'interrogation natif	Web Services	Bibliothèque (API)
Hbase	Java	2007	Non	/	Java
Hypertable	C++	2007	HQL	/	/
Cloudata	Java	2011	CQL	REST	JAVA
Membase	C et C++	2009	/	/	/
Kyoto	C++	/	/	/	C, C++, Java, C#, Python, Ruby, Perl...
Redis	C	2009	/	/	C, C++, Java, Python, Ruby, ...
Oracle	/	2012	/	/	Java, C
Cassandra	Java	2008	/	/	Java, Python, PHP, Ruby...
Voldemort	Java	2008	/	/	/
Riak	Erlang, C, Javascript	2008	/	/	Java, Ruby, Python, PHP, Javascript
CouchDB	C, Javascript	2005	/	REST	/
MongoDB	C++	/	OUI	/	C, C++, Java, C#, Ruby, Javascript...
Neo4j	Java	2003	SPARQL	REST	Java, Python, Ruby, PHP
FlockDB	Scala	2010	/	/	/

- Rudi Bruchez. (2015), « Les bases de données NoSQL et le Big Data », Editeur : Eyrolles, ISBN : 978-2-212-14155-9
- Rudi Bruchez (2021), « Les bases de données NoSQL », Editeur : Eyrolles ISBN : 978-2-212-67866-6
- Juvénal Chokogoue. (2017). « Hadoop - Devenez opérationnel dans le monde du Big Data », Editeur: ENI, ISBN: 978-2409007613