

NOSQL

NOT ONLY SQL

No SQL ne veut PAS dire Non SQL mais Not Only SQL.

NOSQL LE BESOIN

Le NoSQL vient des nouveaux besoins des années 2000.

UNE FAMILLE DE SGBD

Le NoSQL est une famille de base de données.

Les SGBD NoSQL sont souvent mis en opposition aux SGBDR.

ILS PROPOSENT DE RETIRER LES CONTRAINTES DU SQL

- La structure
- Le langage d'interrogation
- Cohérence des données.

FAVORISE LA DISTRIBUTION

La diminution des contraintes a pour but de favoriser la distribution des données.

NOSQL ET BIGDATA

Les SGBD NoSQL ont souvent pour but de répondre aux 3 V du BigData.

Ils sont des outils importants des architectures du BigData.

VOLUME

Ils favorisent la distribution des données permettant de suivre l'accroissement du volume des données avec moins de modifications que les SGBDR.

VELOCITY

Ils proposent des performances supérieures que les SGBDR pour leurs besoins.

Ils sont plus spécialisés et donc optimisés pour certains cas d'utilisation.

VARIETY

Ils proposent souvent de prendre en charge des données semi-structurées, voire des données brutes.

BASE ET CAP

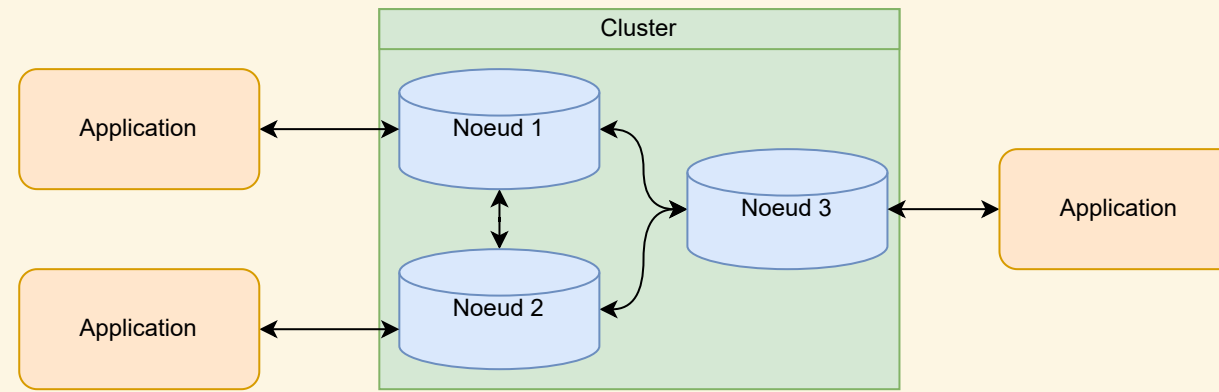
Ils ne respectent pas tous les principes ACID mais favorisent les principes BASE.

Ils sont pour la plupart CP ou AP dans le triangle de CAP.

LES CLUSTERS NOSQL

DÉFINITION CLUSTER

Un cluster est une architecture distribuée dans laquelle un ensemble de noeuds travaillent ensemble pour répondre aux besoins.



FORCES DE LA CAUTERISATION

- **scaling vertical** possibilité de déployer le système sur plusieurs servers pour répartir les charges.
- Facilement extensible : pour ajouter de la puissance ou du stockage, il est possible d'ajouter de nouvelles instances.
- Meilleure tolérances aux pannes.
- Haute disponibilité.
- Meilleurs temps de réponse.
- Haute capacité d'ingestion.



ADMINISTRATION AUTOMATIQUE DU CLUSTER

- Ils proposent souvent des systèmes automatiques pour la gestion automatique du cluster.
 - Répartition des charges automatiques.
 - Répartition du stockage.
 - Optimisation des lectures et écritures sur les nœuds.
 - Spécialisation des nœuds.
 - Répartition des charges géographiques.

TECHNOLOGIES DU CLOUD

Les SGBD NoSQL sont souvent utilisés dans des environnements Cloud.

LES FAMILLES NOSQL

UN SEUL TYPE DE SGBD?

Non. Chaque SGBD NoSQL propose ses fonctionnalités.
Il existe tout de même 4 grandes familles de SGBD NoSQL.

LES CLÉ-VALEUR

LES CLÉ-VALEUR

- Ne possèdent pas de schéma.
- Simple.
- Utilisent une clé pour identifier des valeurs.
- Proche d'un annuaire ou d'un dictionnaire.

EXEMPLE

Clé	Valeur
nombre-connection	12124
config	{"host":"mon-site","description":"une description"}
un-text	"un super text"

OPÉRATIONS CRUD

Les commandes sont souvent simple.

```
SET ma-cle "une valeur"
```

```
GET ma-cle
```

```
REMOVE ma-cle
```

FORCES DES CLÉ-VALEURS

- Facile d'utilisation
- Facile d'administration
- Haute performance
- Proche du temps réel

FAIBLESSES DES CLÉ-VALEURS

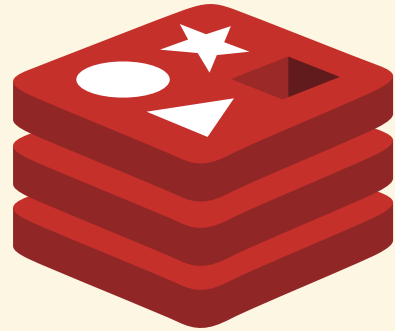
- Aucune structure
- Les rechecks se font uniquement sur les clés
- Souvent limité dans le volume
- Ne sont pas fait pour la persistance à long terme.
- Haute disponibilité
- Pas de relations entre les données.

CAS D'UTILISATION

- Mise en cache
- Historique
- Buffer
- Gestion de panier d'achat
- Collecte d'événement
- PubSub

QUELQUES SGBD CLÉ-VALEURS

Redis



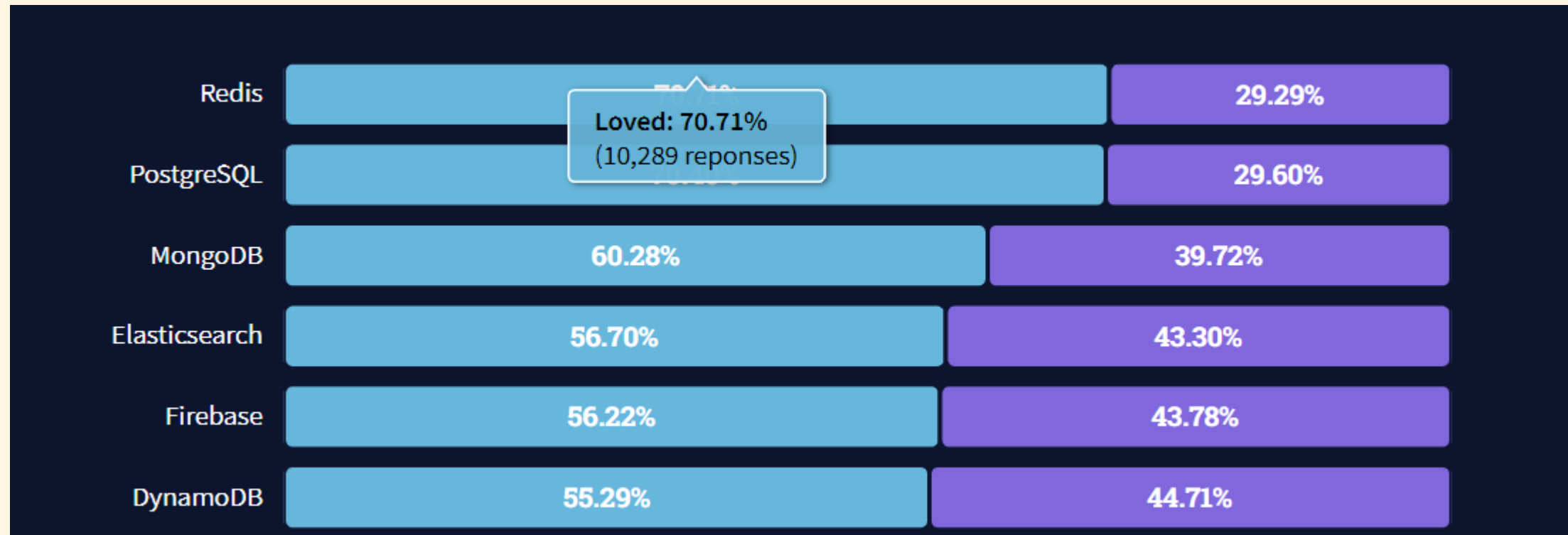
Amazon DynamoDB



Azure CosmosDB



REDIS ET DYNAMODB



ORIENTÉ COLONNES

ORIENTÉ COLONNES

Les données sont proches dans la représentation des SGBDR

id	Age	Taille	Poids
Truc	45	184	75
bidule	32	172	71
Machin	28	179	85

ORIENTÉ COLONNES

Les SGBD se focalisent plus sur les attributs (colonnes) que les entités (lignes) pour s'orienter sur des calculs analytiques.

PEUVENT UTILISER DES LANGAGES PROCHES DU SQL

Exemple de requêtes CQL de Cassandra.

```
SELECT id, age, taille, poids  
FROM ma_table  
WHERE  
    age > 18;
```


LES FORCES DES SGBD ORIENTÉ COLONNES

- Flexibilité
- Temps de traitement
- Non stockage des valeurs null
- Historisation des valeurs
- Forte injection
- Gros volume de données

LES FAIBLESSES DES SGBD ORIENTÉES COLONNES

- Non-adaptées aux relations entre les données.
- Non-adaptées aux données complexes.

CAS D'UTILISATIONS

- Stockage de gros volume.
- gestion d'événement.
- DataLake
- Moteur de recherche.
- Calcule statistique.

QUELQUES SGBD ORIENTÉES COLONNES

Cassandra



GCP BigTable



Apache HBase



Elasticsearch



ORIENTÉ DOCUMENT

ORIENTÉ DOCUMENT

Stocke des documents souvent au format JSON.

Repose sur le principe du clé/valeur, mais avec une extension sur les champs qui composent ce document.

Les données ne sont pas limitées au format des lignes dans une table.

ORIENTÉ DOCUMENT

Exemple de document.

```
{  
  "_id": "507f1f77bcf86cd799439011",  
  "nom": "Raoux",  
  "prenom": "Killian",  
}
```



```
"adresse":{
  "ville":"Lille",
  "rue":"Avenue de la paix",
  "code postal":59000
},
"langages":[
  "Java", "Python", "C#", "Typescript", "SQL"
]
}
```

ORIENTÉ DOCUMENT

Equivalent SGBDR

Utilisateurs

id	nom	prenom	adresse_id
1	Raoux	Killian	1

ORIENTÉ DOCUMENT

Adresses

id	ville	rue	code_postal
1	Lille	Avenue de la paix	59000

ORIENTÉ DOCUMENT

Langages

id	utilisateur_id	nom
1	1	Java
2	1	Python
3	1	C#
4	1	Typescript

EXEMPLE DE REQUÊTE

```
db.utilisateurs.find({"nom":"Martin", "prenom":"Dupont"})
```

Cette requête retourne la liste des utilisateurs ayant pour nom "Martin" et prénom "Dupont" avec mongoDB.

Equivalent SQL:

```
SELECT *  
FROM  
  utilisateur AS u,  
  JOIN address AS a ON u.adresse_id == a.id  
  JOIN langages AS l ON u.id == utilisateur_id  
WHERE  
  u.nom == "Martin"  
  AND u.prenom == "Dupont"
```

EXEMPLE DE REQUÊTE

- Les SGBDR doivent effectuer des jointures ou plusieurs requêtes pour récupérer l'ensemble des données et reconstruire une requête. Ce qui prend plus temps.
- Les SGBD Document n'ont besoin que d'une unique requête. Ce qui est beaucoup plus rapide.

LES FORCES DES SGBD ORIENTÉ DOCUMENT

- Permettent de stocker des données complexes.
- Plus performantes pour la récupération de données hiérarchiques.
- Ne nécessitent pas toujours de structure prédéfinie.
- Permettent le développement d'applications Code-first.

LES FAIBLESSES DES SGBD ORIENTÉ DOCUMENT

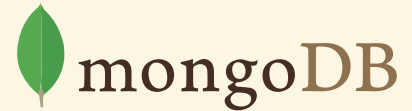
- Peu adaptés aux relations entre les données.
- Langages de requêtes spécifiques

CAS D'UTILISATION

- Données clients
- Gestion de données complexe.
- Le web Analytics.

QUELQUES EXEMPLES

MongoDB



GCP BigTable



ORIENTÉS GRAPHERS

ORIENTÉS GRAPHES

Les SGBD orientés graphes se focalisent sur les relations entre les données et pas les données elles-mêmes.

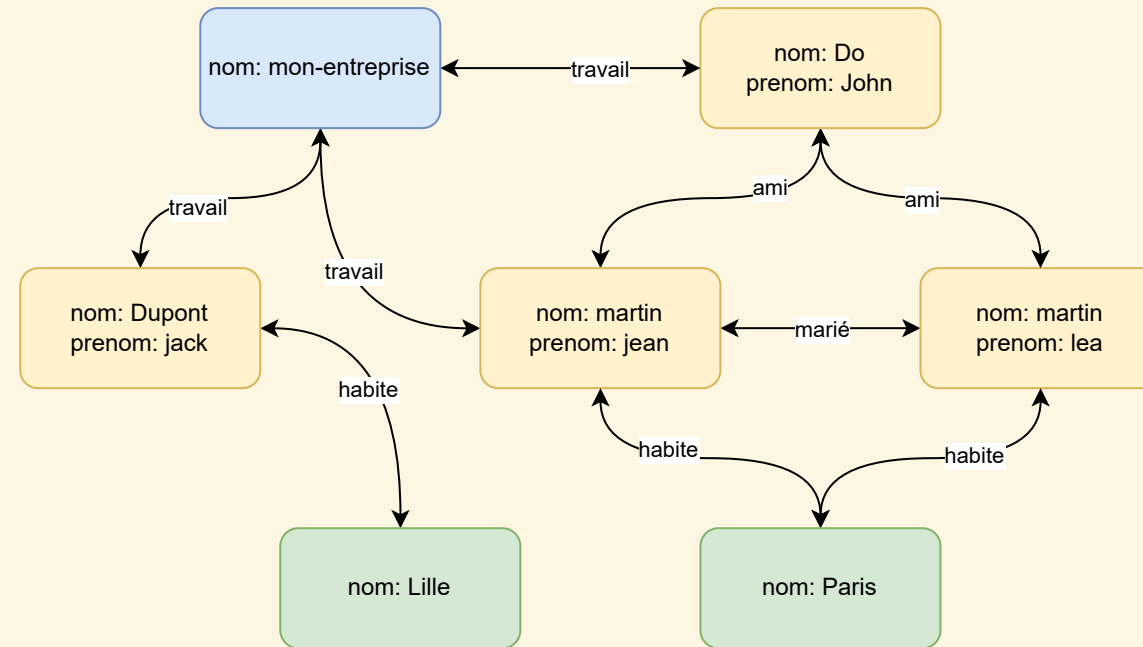
Ils se basent sur la théorie des graphes.

ORIENTÉS GRAPHES

Les données et leurs relations sont représentées sous forme de noeuds (les données) et des liaisons (leurs relations).

Les requêtes se font sur les liaisons.

ORIENTÉ GRAPHES



LES LANGAGES DES SGBD ORIENTÉS GRAPHES

Ils utilisent des langages spécifiques à leurs besoins.

Contrairement au SQL, les liaisons entre les données portent un nom.

EXEMPLE AVEC NEO4J

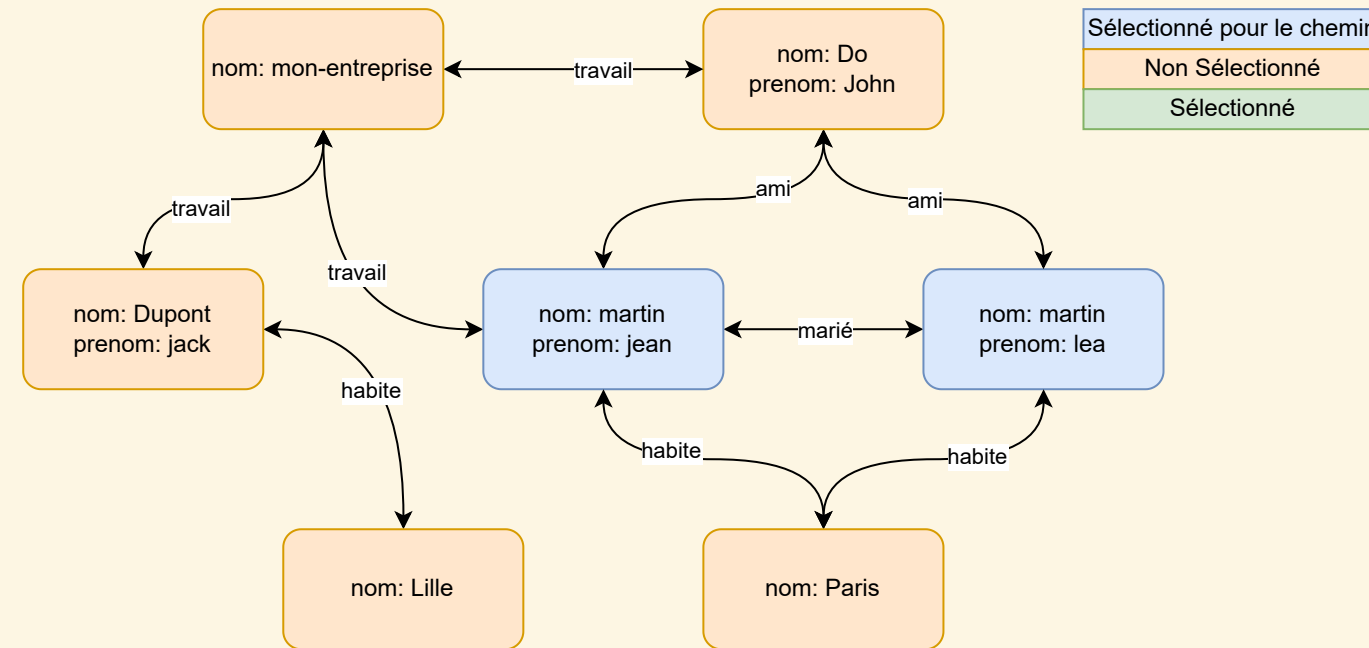
Exemple d'une requête Cypher de Neo4j:

```
MATCH (:Personne {nom:"Martin"})-[:Travail]->(:Entreprise)
      -[:Travail]->(collaborateur:Personne)
RETURN collaborateur.nom, collaborateur.prenom
```

Retourne les personnes qui travaillent dans la même entreprise, que les personnes ayant comme nom "Martin".

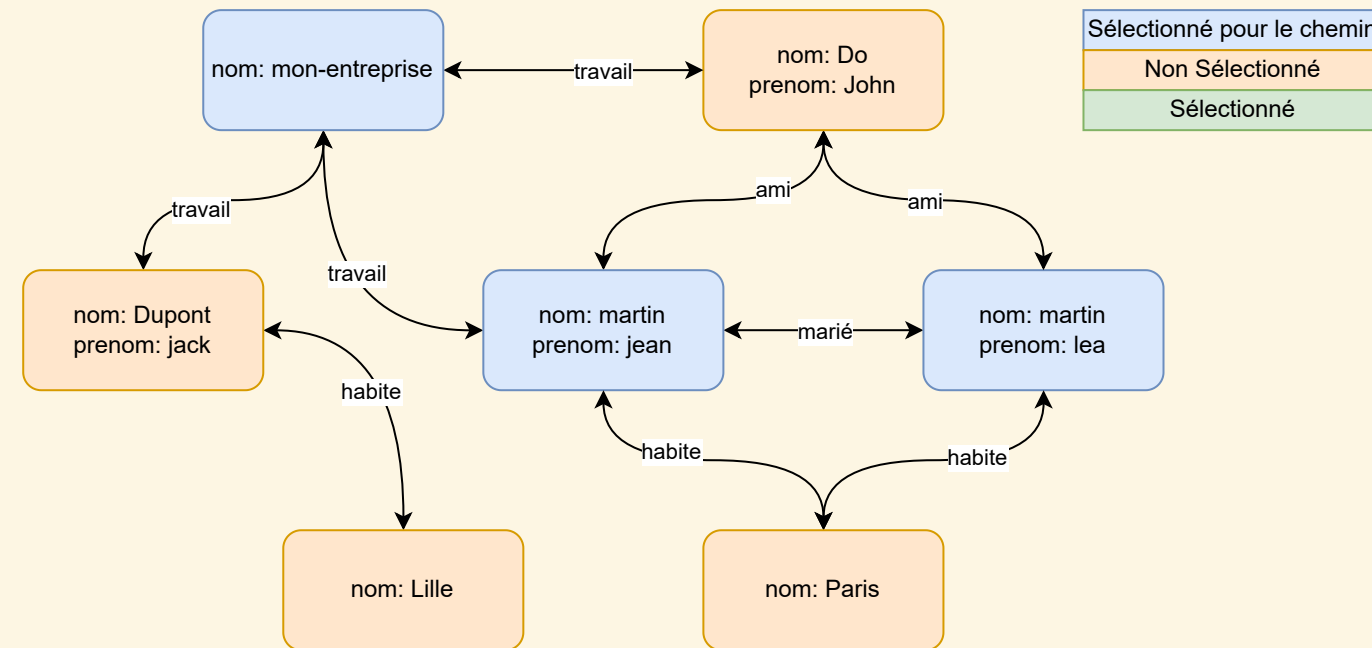
EXEMPLE AVEC NEO4J

- **MATCH** est equivalent au **WHERE**.
- **(:Personne {nom:"Martin"})** filtre les noeuds de depart de type *Personne* et ayant l'attribut *nom* valant *Martin*.



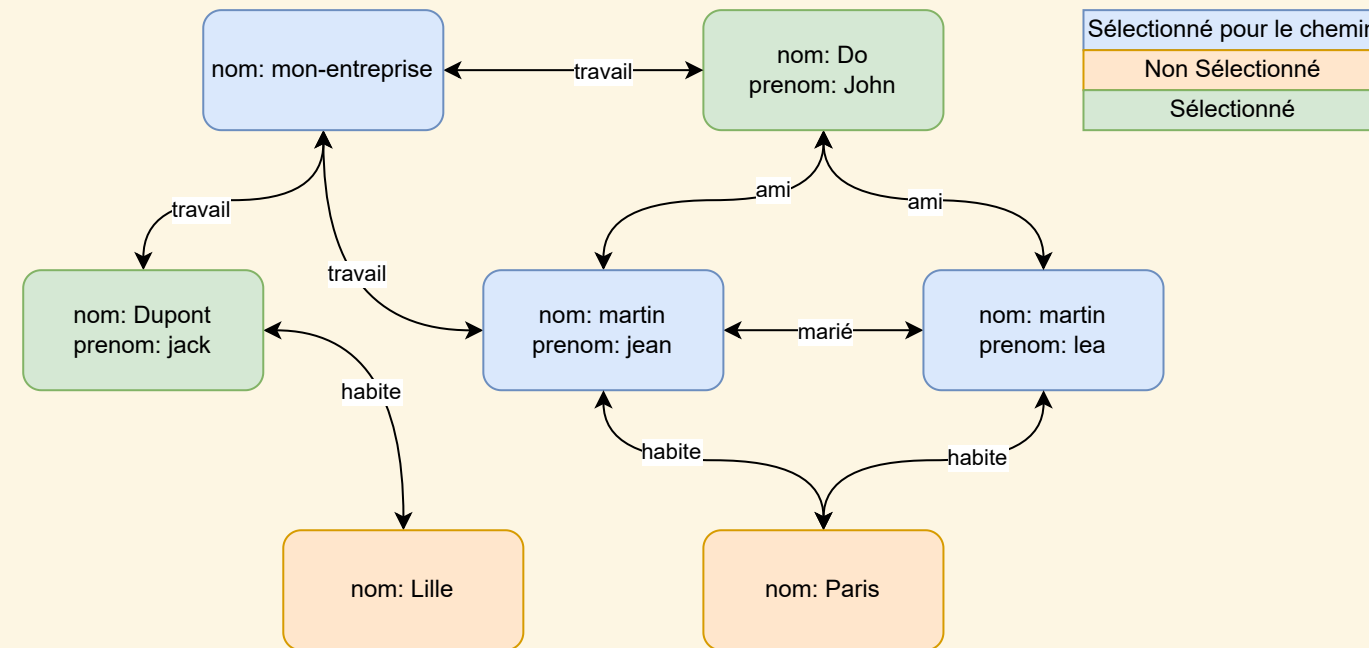
EXEMPLE AVEC NEO4J

- -[:Travail]-> définit une liaison entre deux nœuds. Cette liaison doit être de type Travail



EXEMPLE AVEC NEO4J

- (**collaborateur: personne**) recherche les nœuds du type *personne respectant la liaison précédente.
- Il stocke les valeurs du noeud sous le label collaborateur.



EXEMPLE AVEC NEO4J

Résultats:

nom	prenom
Do	John
Dupont	Jack

QUELLES DIFFÉRENCES AVEC LES JOINTURES SQL?

- Les jointures SQL sont accélérées avec l'indexation des clés primaires.
- Neo4j indexe les jointures en elle-même.
- Neo4j est beaucoup plus rapide que les SGBDR pour le chaînage des jointures.

QUELLES DIFFÉRENCES AVEC LES JOINTURES SQL?

Benchmark pour une requête de jointure sur une base de 1 000 000 utilisateurs :

Profondeur	Temps d'execution MySQL	Temps d'exécution Neo4j
2	0.016	0.010
3	30.267	0.168
4	1 543.505	1.359
5	Non fini en 1h	2.132

Benchmark MySQL et Neo4j

CAS D'UTILISATIONS

- Recherche sur des réseaux.
- Données avec beaucoup de liaisons
- Données de types graphes
- Analyse BigData pour les liaisons entre utilisateurs ou données.

POINTS FORTS

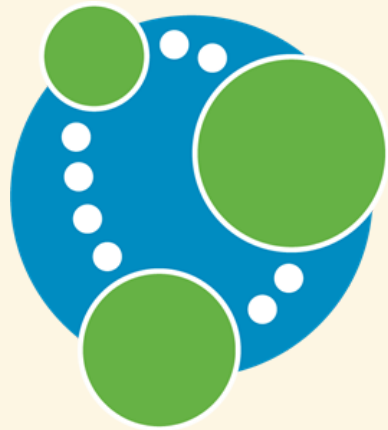
- Rapide pour les résolutions de connexions.
- Affichage sous forme de graphes.

POINTS FAIBLES

- Spécifiques pour certaines utilisations.
- Grande consommation de mémoire.
- Lent en écriture.
- Outils d'analyse et de recherche et non de stockage.

QUELQUES EXEMPLES

MongoDB



Azure CosmosDB



QUELQUES CHIFFRES

POPULARITÉ

Les SGBD NoSQL gagnent de plus en plus de popularité avec le temps.

[stackoverflow survey](#)

[db-engine](#)

NOUVEAUX LANGAGES

Les SGBD NoSQL se veulent plus proches des besoins des développeurs.

En général, elles proposent des langages proches des langages de développement moderne comme le Json ou la norme REST.