

Introduction aux Graphes et leur importance en Bases de Données

Contexte Apache Hop

Cours Data Engineering

Pourquoi parler de Graphe ?

- Explosion des données relationnelles complexes
- Limites des bases SQL classiques
- Besoin de modéliser des relations dynamiques
- Cas réels : fraude, réseaux sociaux, supply chain

Message clé

Quand les relations deviennent centrales, le modèle tabulaire devient insuffisant.

Rappel : Modèle Relationnel

- Tables
- Lignes
- Colonnes
- Clés étrangères

Limite

Les relations sont implicites (via JOIN).

Complexité croissante :

$JOIN \rightarrow JOIN \rightarrow JOIN$

Qu'est-ce qu'un Graphe ?

Un graphe est composé de :

- Nodes (nœuds)
- Relationships (relations)
- Properties (propriétés)

Node —[RELATION]— Node

Différence majeure

La relation devient un objet de première classe.

Exemple Taxi (Comparaison SQL vs Graph)

SQL :

- Table Driver
- Table Trip
- Table Zone
- Jointures multiples

Graph :

- (:Driver)-[:PERFORMED]->(:Trip)
- (:Trip)-[:TO_ZONE]->(:Zone)

Avantage

Lecture naturelle des relations.

Cypher : Langage déclaratif

Exemple :

```
MATCH (d:Driver)-[:PERFORMED]->(t:Trip)  
RETURN d.name, SUM(t.amount)
```

Déclaratif

On décrit le pattern recherché, pas l'algorithme de parcours.

Pourquoi les Graph sont puissants ?

- Traversée multi-niveaux rapide
- Détection de patterns
- Analyse de centralité
- Réseaux complexes

Cas typiques

Fraude bancaire, logistique, réseaux sociaux, recommandations.

Graph et Apache Hop

Apache Hop est un moteur ETL.

- Ingestion données relationnelles
- Transformation
- Injection dans Neo4j

Pipeline :

$SQL \rightarrow HOP \rightarrow Graph$

Rôle de Hop

Hop devient le pont entre monde tabulaire et monde graphe.

Sources → Hop → Multi-storage

- SQL (structure stable)
- MongoDB (flexible)
- Neo4j (relations dynamiques)

Vision

Chaque technologie répond à un besoin spécifique.

Quand choisir un Graph ?

Choisir un graphe si :

- Relations nombreuses et évolutives
- Jointures complexes
- Recherche de chemins
- Analyse réseau

Ne pas utiliser un Graph

Pour simple stockage transactionnel tabulaire.

- SQL : données structurées
- MongoDB : documents flexibles
- Graph : relations explicites
- Hop : orchestrateur intelligent

Conclusion

Comprendre les graphes, c'est comprendre la structure cachée des données.