

Objectifs :

L'objectif de cet article est de réaliser une analyse sur les Bases de données orientées graphes Neo4j, les cas d'usages les plus adaptés où ce dernier est indispensable. Installation de neo4j et son déploiement sur docker et différents types de requêtes.

Propriétés des bases de données orientées graphes

- Une BD graphe utilise la structure de graphe, les nœuds, les arêtes et les propriétés pour représenter et stocker les données

Les nœuds :

Représentent les objets dans le graphe (tuples)

Peuvent comporter un ou des labels / types

Les relations

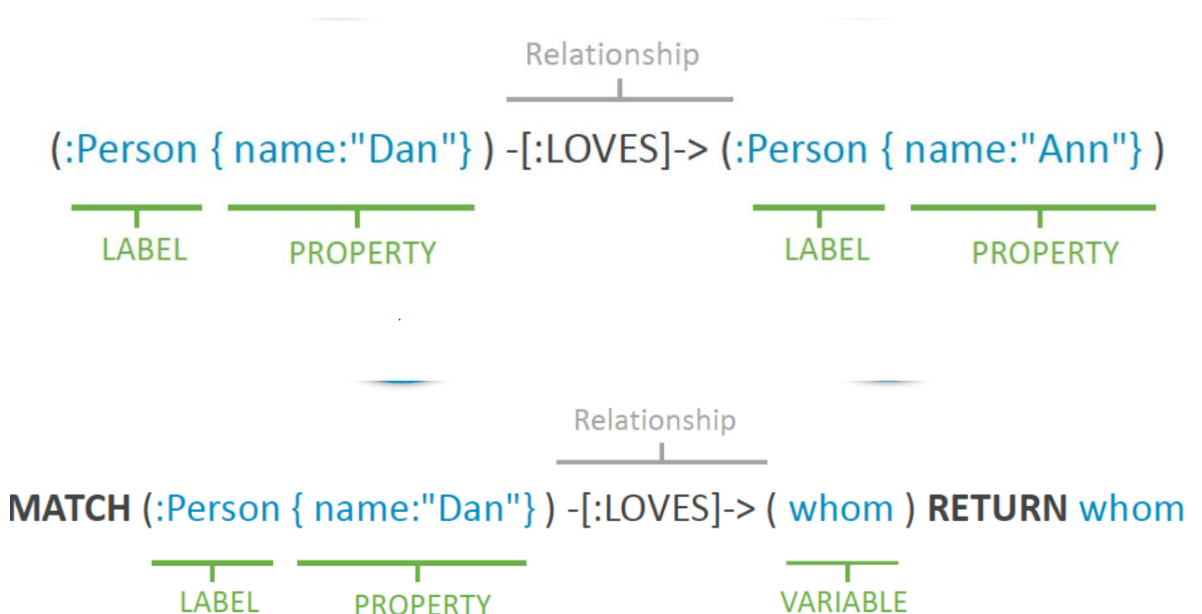
Relient les nœuds par type et direction

Les propriétés

Paires nom-valeur possibles sur les nœuds et relations

Les Labels

Regroupent les nœuds par rôle



- Un système de gestion de bases de données orienté graphe offre des méthodes Create, Read, Update et Delete (CRUD) pour accéder et manipuler les données












- Les BD graphe non pas besoin de schéma, les données peuvent être accumulées progressivement sans avoir besoin d'un schéma prédéfini
- Flexibilité des BD graphes : Il est très simple d'incorporer de nouvelles informations dynamiquement
- Les relations dans un graphe forment naturellement des chemins. Les interroger c'est - réellement - traverser le graphe, i.e., suivre ces chemins.
- Efficacité : aller d'un nœud à un autre est une opération en temps constant.

Base de données graphe neo4j

Neo4j est une base de données graphe développer par neo4j, il permet de représenter les données en tant que nœuds reliés par un ensemble d'arcs.

- Neo4j est le SGBD graphe le plus utilisé, il domine le marché de bases de donnée

graphe : <https://dbengines.com/en/ranking/graph+dbms>

Rank			DBMS	Database Model	Score		
May 2022	Apr 2022	May 2021			May 2022	Apr 2022	May 2021
1.	1.	1.	Neo4j 	Graph	60.14	+0.62	+7.91
2.	2.	2.	Microsoft Azure Cosmos DB 	Multi-model 	40.22	-0.12	+5.51
3.	3.	 5.	Virtuoso 	Multi-model 	6.01	+0.34	+2.57
4.	4.	 3.	ArangoDB 	Multi-model 	5.55	-0.10	+1.17
5.	5.	 4.	OrientDB	Multi-model 	5.14	+0.06	+0.95

- Il utilise un langage de requête fait pour les graphes basé sur les patterns de graphe ie CYPHER

Forces et faiblesses

Les forces

- Performant sur des requêtes de type relation et sur les données complexes
- Sans obligatoirement de schéma
- Système de requêtage plus plaisant que le SQL
- Simplicité de mise en place

Les faiblesse

- Oblige à architecturer correctement les modèles de données
- Il sera facile de créer accidentellement un nœud avec un attribut "name", et un autre avec un attribut "Name", qui risque de passer alors qu'il s'agit juste d'une erreur.
- Pas très intéressants au niveau des données peu connectées

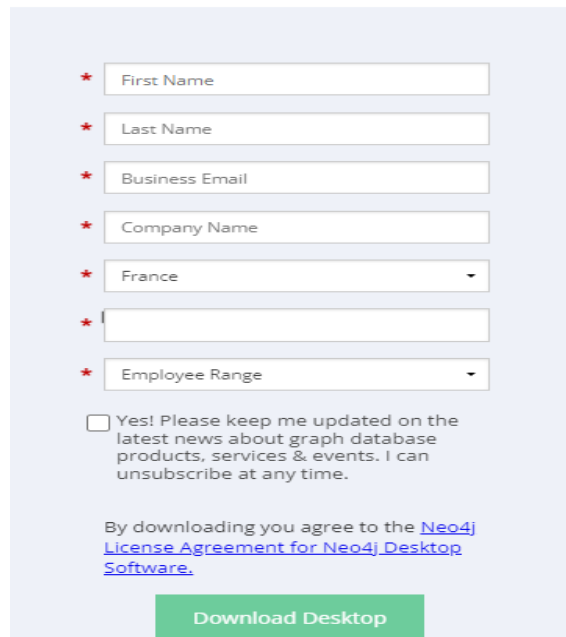
Cas d'usages les plus adaptés

Neo4j est très utilisée dans le traitement des données complexes avec des schémas de données évolutifs tels que les :

- Réseaux sociaux,
- Systèmes de recommandation : la recommandation d'articles scientifiques pour Servier, les recommandations du site WeLoveCinema de BNP Paribas, recommandation pour Konbini ont été réalisées avec Neo4J
- Applications géospatiales,
- Réseaux et data center,
- Systèmes de contrôles d'accès et d'autorisation,
- Réseaux de fraude,

Installation de neo4j sur Windows

1. Remplir les informations d'utilisateur et télécharger l'exécutable sur <https://neo4j.com/download-center/>



Registration form for Neo4j Desktop. The form includes the following fields:

- * First Name
- * Last Name
- * Business Email
- * Company Name
- * France (dropdown menu)
- * (empty text field)
- * Employee Range (dropdown menu)

Below the form, there is a checkbox: ☐ Yes! Please keep me updated on the latest news about graph database products, services & events. I can unsubscribe at any time.

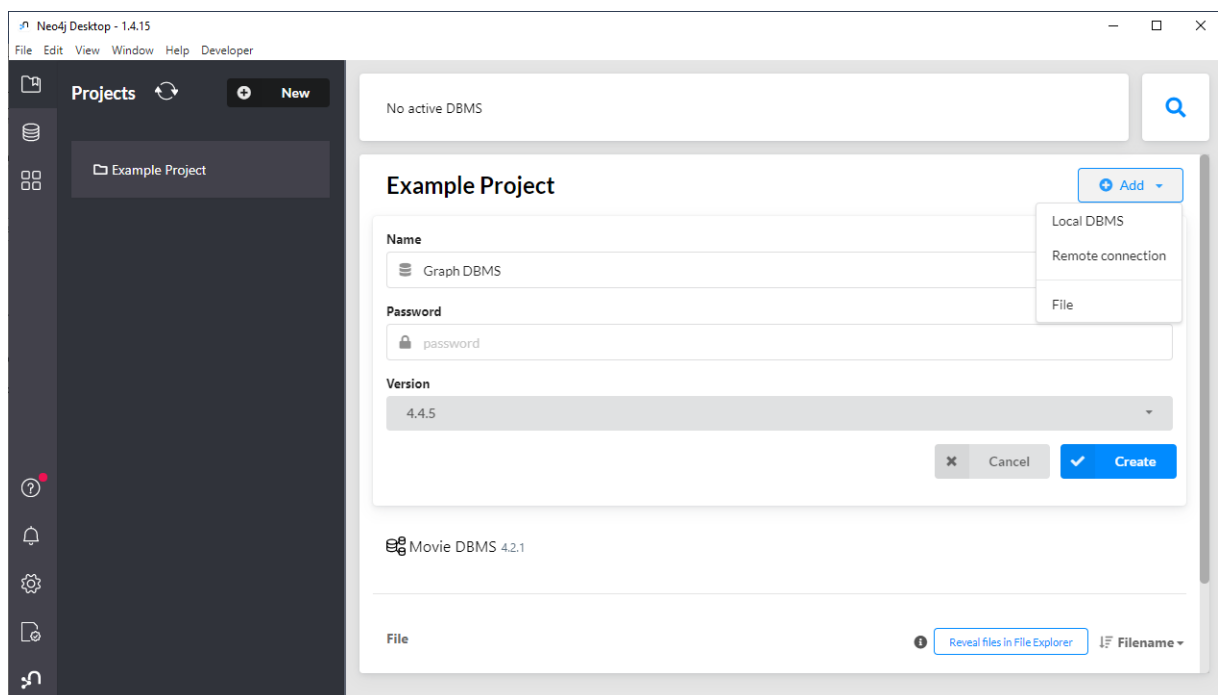
By downloading you agree to the [Neo4j License Agreement for Neo4j Desktop Software.](#)

Download Desktop

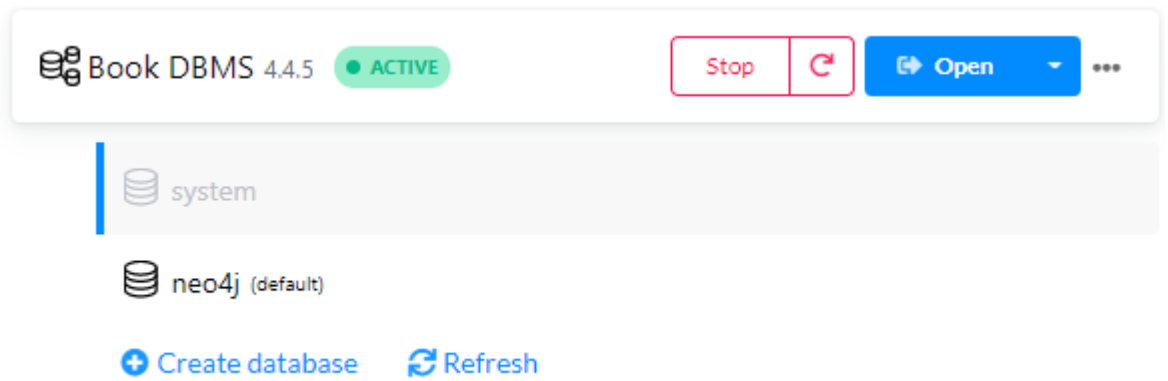
2. Lancer l'exécutable

Création et utilisation de base de données sur neo4j Desktop

1.1 Créer un « data bases management système » dans lequel on peut créer plusieurs bases de données, lui donner un nom et un password

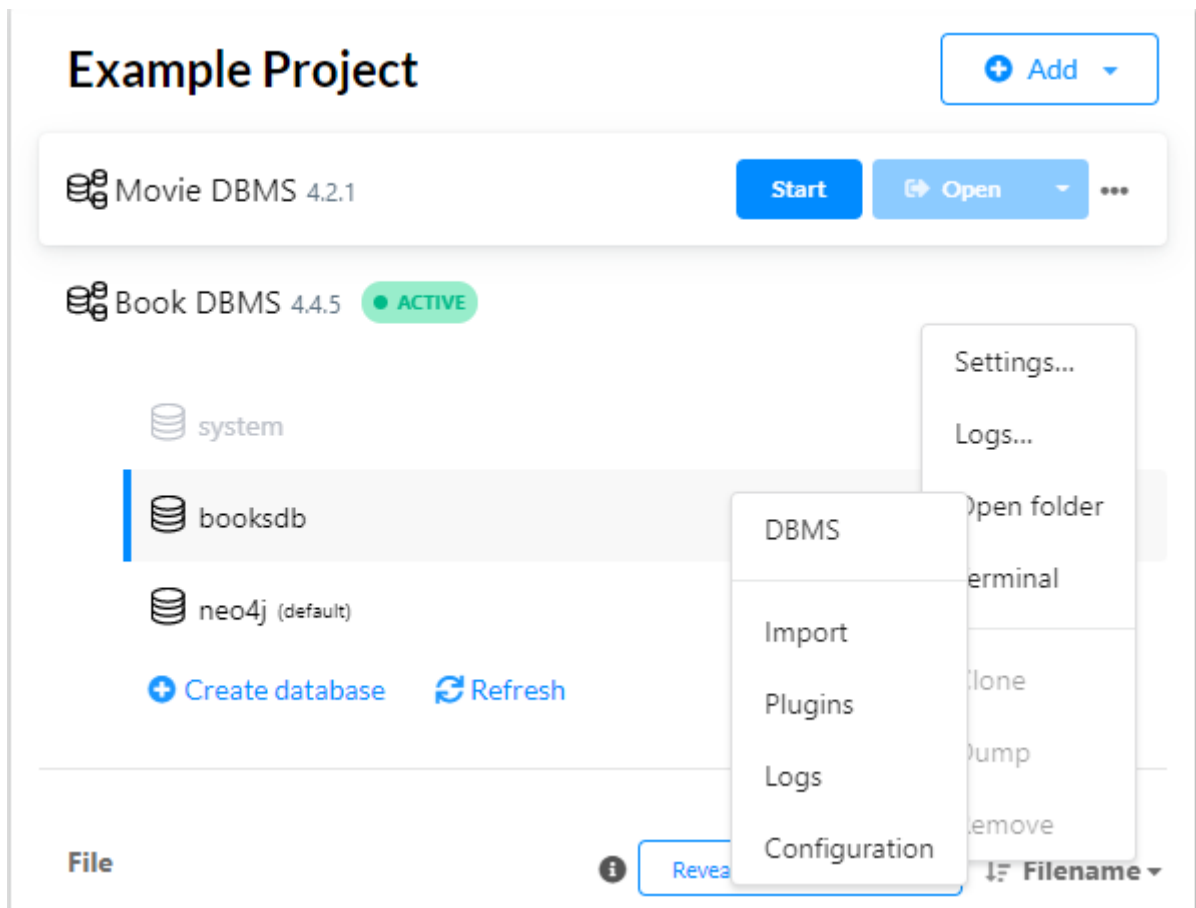


1.2 Créer une base de données



1.3 Importer les fichiers qu'on veut charger et manipuler

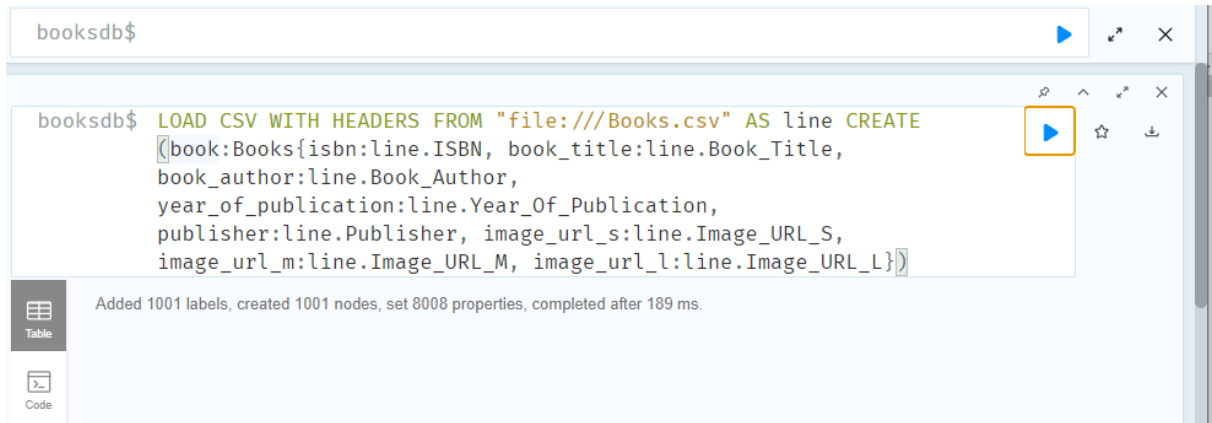
- Books.csv
- Ratings.csv
- Users.csv



1.4 Après l'importation des fichiers on exécute la commande 'LOAD CSV HEADERS FROM « file:/// notrefichier.csv » ...' pour l'importation des données et la création des nœuds

Exemple de requêtes cypher :

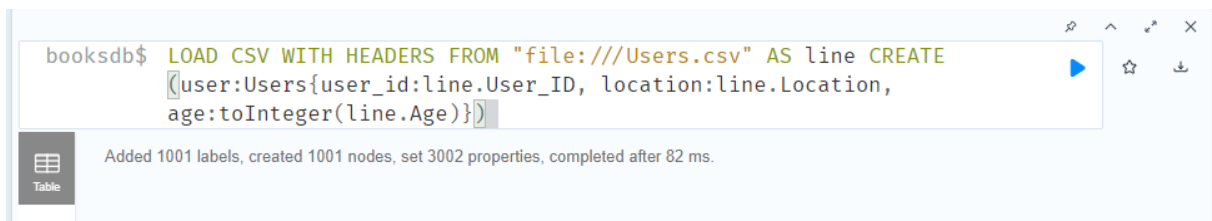
Load et création : Importation des données et création des tables correspondantes



The screenshot shows a Cypher query execution window titled 'booksdb\$'. The query is: `LOAD CSV WITH HEADERS FROM "file:///Books.csv" AS line CREATE (book:Books{isbn:line.ISBN, book_title:line.Book_Title, book_author:line.Book_Author, year_of_publication:line.Year_Of_Publication, publisher:line.Publisher, image_url_s:line.Image_URL_S, image_url_m:line.Image_URL_M, image_url_l:line.Image_URL_L})`. A blue play button is highlighted. Below the query, a status message reads: 'Added 1001 labels, created 1001 nodes, set 8008 properties, completed after 189 ms.' On the left, there are icons for 'Table' and 'Code'.

```
booksdb$ LOAD CSV WITH HEADERS FROM "file:///Books.csv" AS line CREATE
  (book:Books{isbn:line.ISBN, book_title:line.Book_Title,
  book_author:line.Book_Author,
  year_of_publication:line.Year_Of_Publication,
  publisher:line.Publisher, image_url_s:line.Image_URL_S,
  image_url_m:line.Image_URL_M, image_url_l:line.Image_URL_L})
```

Added 1001 labels, created 1001 nodes, set 8008 properties, completed after 189 ms.



The screenshot shows a Cypher query execution window titled 'booksdb\$'. The query is: `LOAD CSV WITH HEADERS FROM "file:///Users.csv" AS line CREATE (user:Users{user_id:line.User_ID, location:line.Location, age:toInteger(line.Age)})`. A blue play button is highlighted. Below the query, a status message reads: 'Added 1001 labels, created 1001 nodes, set 3002 properties, completed after 82 ms.' On the left, there are icons for 'Table' and 'Code'.

```
booksdb$ LOAD CSV WITH HEADERS FROM "file:///Users.csv" AS line CREATE
  (user:Users{user_id:line.User_ID, location:line.Location,
  age:toInteger(line.Age)})
```

Added 1001 labels, created 1001 nodes, set 3002 properties, completed after 82 ms.

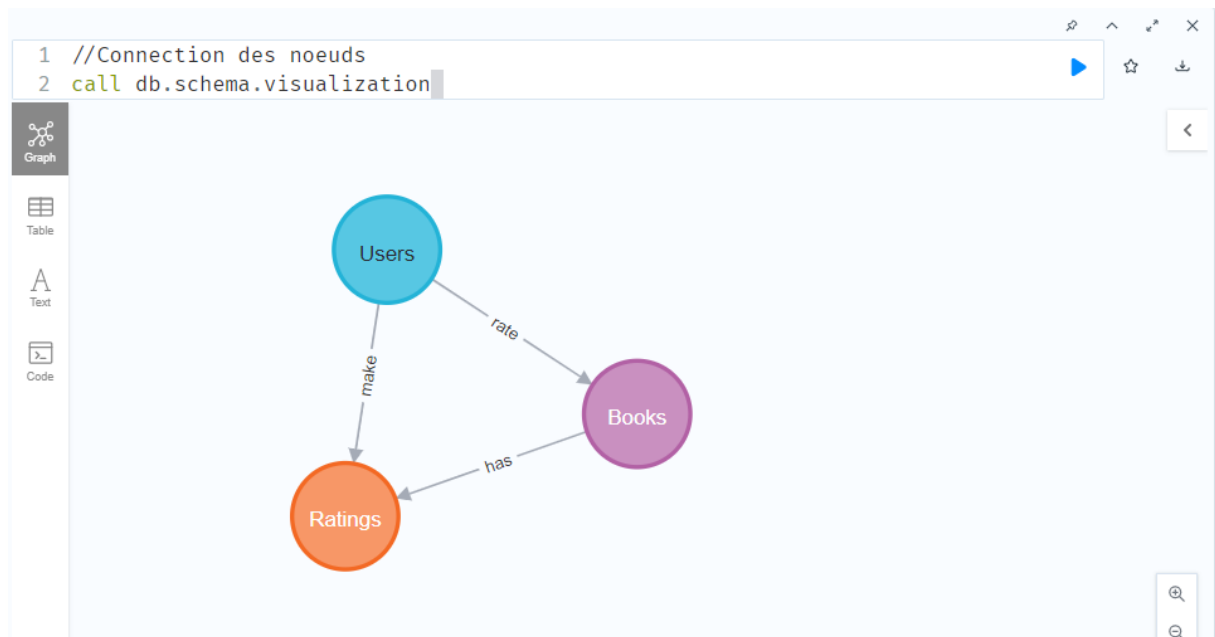


The screenshot shows a Cypher query execution window titled 'booksdb\$'. The query is: `LOAD CSV WITH HEADERS FROM "file:///Ratings.csv" AS line CREATE (rating:Ratings{user_id:line.User_ID, isbn:line.ISBN, book_rating:toInteger(line.Book_Rating)})`. A blue play button is highlighted. Below the query, a status message reads: 'Added 1001 labels, created 1001 nodes, set 3003 properties, completed after 186 ms.' On the left, there are icons for 'Table' and 'Code'.

```
booksdb$ LOAD CSV WITH HEADERS FROM "file:///Ratings.csv" AS line CREATE
  (rating:Ratings{user_id:line.User_ID, isbn:line.ISBN,
  book_rating:toInteger(line.Book_Rating)})
```

Added 1001 labels, created 1001 nodes, set 3003 properties, completed after 186 ms.

Call `db.schema.visualization` : récupération de schéma de relation des tables



Match : sélection des données

Requête sur le nœud book

Récupérer les livre d'un auteur précis

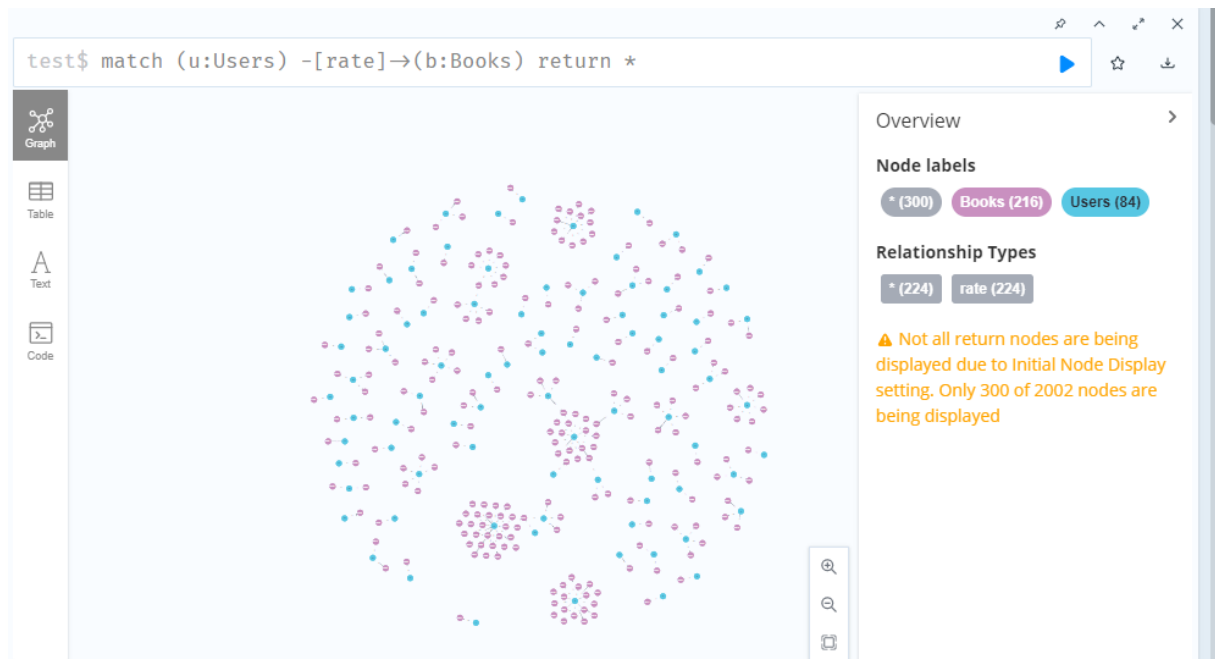
```

booksdb$ match (n:Books) where n.book_author = "A. C. Crispin" return n.isbn ,n.book_title, n.year_of_publication

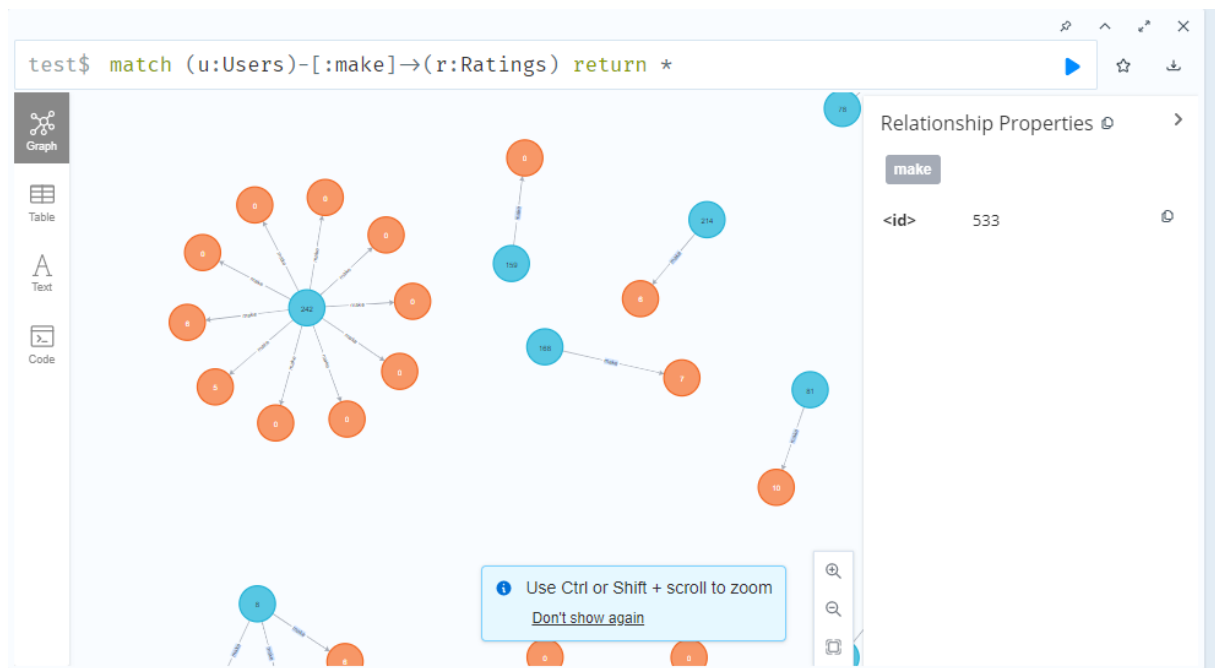
```

	n.isbn	n.book_title	n.year_of_publi
1	"553574167"	"The Hutt Gambit (Star Wars: The Han Solo Trilogy, Vol. 2)"	"1998"
2	"553574159"	"The Paradise Snare (Star Wars: The Han Solo Trilogy, Volume 1)"	"1998"

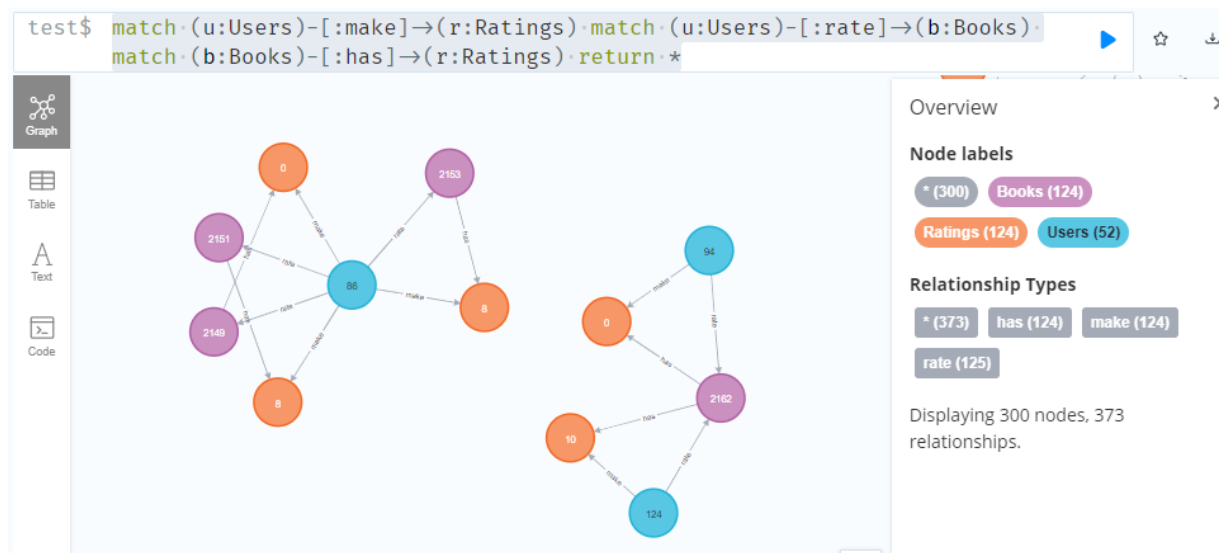
Récupérer les livres qu'a noté un user



Récupérer les notes qu'a donné un user



Récupérer les livre noté par un user et la note donnée



Update

Avant modification



User après modification

```
1 MATCH (u:Users {user_id:"1"})
2 SET u.age = 22
3 RETURN u
```

Graph

Table

Text

Code

u

```
{
  "identity": 0,
  "labels": [
    "Users"
  ],
  "properties": {
    "user_id": "1",
    "location": "nyc, new york, usa",
    "age": 22
  }
}
```

Création des index

Nous créons les index grâce à la commande create index, en exécutant les requêtes avant et après l'ajout des différents index on remarque que le temps d'exécution est réduit et que la réponse est plus rapide, le système des index est semblable au système de cache des données.

```
1 CREATE INDEX book_id for (n: Books) ON (n.isbn);
2 CREATE INDEX user_id for (n: Users) ON (n.user_id);
3 CREATE INDEX ratings_id for (n: Ratings) ON (n.user_id,n.isbn);
```

neo4j\$ CREATE INDEX book_id for (n: Books) ON (n.isbn)	✓
neo4j\$ CREATE INDEX user_id for (n: Users) ON (n.user_id)	✓
neo4j\$ CREATE INDEX ratings_id for (n: Ratings) ON (n.user_id,n.isbn)	✓

Suppression des données sans relation

```
MATCH (u : Users {name : ' Name'})
```

```
DELETE u
```

Remarque : On utilise delete que lorsque notre nœud n'intervient dans aucune relation

Suppression des données avec relation

```
test$ match (u:Users {user_id:'9'}) detach delete u
```

Deleted 1 node, deleted 6 relationships, completed after 7 ms.

Table

Code

Suppression de tout le contenu de la base de données

MATCH (p) DETACH DELETE p

Démarrer neo4j avec Docker

Par défaut, l'image Docker expose trois ports pour l'accès à distance :

- 7474 for HTTP
- 7473 for HTTPS
- 7687 for Bolt

```
docker run \  
  --name testneo4j \  
  -p7474:7474 -p7687:7687 \  
  -d \  
  -v $HOME/neo4j/data:/data \  
  -v $HOME/neo4j/logs:/logs \  
  -v $HOME/neo4j/import:/var/lib/neo4j/import \  
  -v $HOME/neo4j/plugins:/plugins \  
  --env NEO4J_AUTH=neo4j/test \  
  neo4j:latest
```

Utilisation de neo4j avec jupyter notebook

- Installer le package neo4J : `pip install neo4j`
- Importer le driver GraphDatabase : `from neo4j import GraphDatabase`
- Etablir la connection

Sauvegarde de la base de données Neo4j

Pour faire une sauvegarde de la base de données il suffit de créer un dossier de sauvegarde dans lequel on copie les fichiers de notre base de données

Restauration de la base de données Neo4j

Arrêter le serveur de la base de données, vider le dossier de la base de données actuel, copier les fichiers de notre sauvegarde dans le dossier de la base et lancer le serveur.