

Автор:

Мизгирев Лев Сергеевич

студент

Научный руководитель:

Ахмадулина Юлия Сергеевна

старший преподаватель

ФГБОУ ВО «Ивановский государственный

политехнический университет»

г. Иваново, Ивановская область

ХРАНЕНИЕ ДАННЫХ В ГРАФОВОЙ БАЗЕ ДАННЫХ NEO4J

Аннотация: в данной работе рассматривается пример применения графовой системы управления базами данных Neo4j для проектирования базы данных социальной сети.

Ключевые слова: графовая база данных, Neo4j, граф.

Базы данных являются неотъемлемой частью любой информационной системы. Существует огромное количество разновидностей баз данных. В этой статье речь пойдёт о графовых базах данных.

В математике и компьютерных науках графами называют абстрактные структуры данных, описывающие структурные отношения между объектами. Сегодня графы широко используются для моделирования данных в различных предметных областях, требующих выяснения и определения правил и схем отношений объектов [1].

Графовая модель базы данных представляет собой обобщение сетевой модели данных и отличается сильными связями между узлами.

Компоненты графовой базы данных:

1. Узел (node) – объект базы данных, представляющий собой ячейку с хранимой информацией.
2. Метка (node label) – условное обозначение типа данных узла.

3. Ребро (связь) – связь между двумя узлами. Количество связей ограничено.

4. Свойства узла – набор данных, которые можно назначить узлу.

Neo4j – графовая система управления базами данных с открытым исходным кодом, реализованная на Java. Она предоставляет интерфейс программирования приложений для СУБД на многих языках программирования, включая Java, Python, Clojure, Ruby, PHP, также реализовано REST API. Расширить программный интерфейс можно как с помощью серверных плагинов, так и с помощью расширений (unmanaged extensions); плагины могут добавлять новые ресурсы к REST-интерфейсу для конечных пользователей [2].

В СУБД используется собственный язык запросов – Cypher, но запросы можно делать и другими способами, например, напрямую через Java API. Cypher является не только языком запросов, но и языком манипулирования данными.

Примером реализации хранения данных в графовой базе данных Neo4j послужит социальный граф. Социальный граф – граф, узлы которого социальные объекты, такие как пользовательские профили с различными атрибутами (например, имя, день рождения, родной город), сообщества медиаконтента и так далее, а рёбра – социальными связями между ними.

Создадим граф, запрос на создание:

CREATE

//Задаём сущности (узлы) базы
(sl:Person {name:'Александр'}),
(lm:Person {name:'Лев'}),
(ak:Person {name:'Анастасия'}),
(no:Person {name:'Наталья'}),
(ug:Person {name:'Юрий'}),

//Зададим связи (рёбра) – кто кого знает и год знакомства

(lm)-[:KNOWS {since:2014}]->(sl),
(lm)-[:KNOWS {since:2014}]->(ak),
(ak)-[:KNOWS {since:2014}]->(sl),

(sl)-[:KNOWS {since:2014}]->(no),

(no)-[:KNOWS {since:2014}]->(ug),

(ak)-[:KNOWS {since:2014}]->(ug)

Результатом работы запроса на создание является граф, представленный на рис. 1.

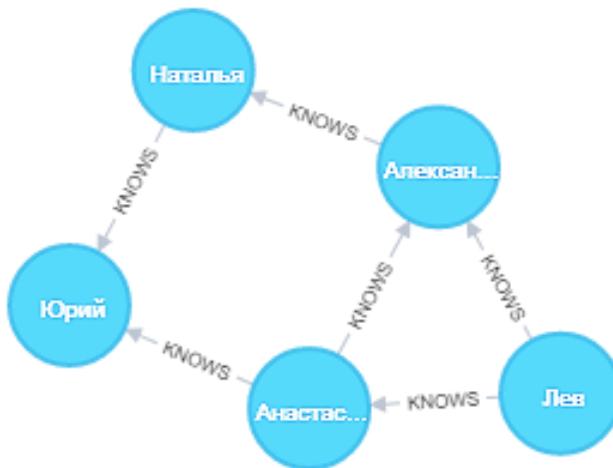


Рис. 1. Графическое представление базы друзей

Графовые базы данных лучше всего подходят для реализации проектов, предполагающих естественную графовую структуру данных. Например, для создания социальных сетей, семантических паутин, баз интернет-магазинов, схем маршрутов и т. п. В подобных задачах они сильно опережают реляционные БД по производительности. Их достоинствами также являются простота внесения изменений и наглядность представления информации. У некоторых баз данных существуют механизмы специальной оптимизации для работы с SSD-накопителями. Для работы с достаточно большими графиками используются алгоритмы, предполагающие частичное помещение графа в оперативную память.

Список литературы

1. Обработка больших объемов графовых данных: путеводитель по современным технологиям [Электронный ресурс]. – Режим доступа: <https://www.ibm.com/developerworks/ru/library/os-giraph/index.html>
2. Neo4j // Википедия [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Neo4j>