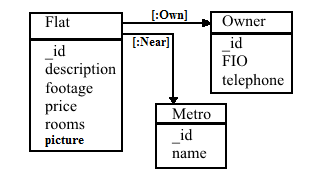
Neo4j



Модель состоит из 3-х сущностей:

1. Квартира:
   * “\_id”, int – идентификатор. 4В
   * “description”, String – Краткое описание. 100В
   * “footage”, int – размер квартиры. 4В
   * “price”, int – предполагаемая цена. 4В
   * “rooms”, int – число комнат. 4В
   * “picture”, String – название изображения – 15В
2. Владелец:
   * “\_id”, int – идентификатор. 4В
   * “FIO”, String – Имя владельца. 16В
   * “telephone”, String – контактные данные владельца. 10В
3. Метро:
   * “\_id”, int – идентификатор. 4В
   * “name”, String – название станции. 16В

«Чистый»: Пусть в базе хранится n станций метро, всего одна станция. У каждой станции продаётся 10 квартир, каждую продаёт 1 уникальный человек. Тогда объём всей БД равен: 1620\*n

«Фактический»: Сущность квартира занимает в среднем 817В, владелец – 415В, метро – 216В

Тогда объём – 12536В

«Избыточность»: Чистый объем делить на фактический.

Избыточность: 12536/1620 = 7,74

Запросы:

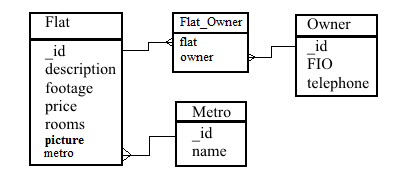
1. Количество комнат квартиры

MATCH(n) WHERE id(n)=1 return n.rooms;

1. Квартиры, продаваемые определённой станцией

MATCH(:Metro{name:”Автово”})--(f:Flat) RETURN id(f);

SQL



Модель состоит из 4-х таблиц:

1. Квартира:
   * “\_id”, int – идентификатор. 4В
   * “description”, String – Краткое описание. 100В
   * “footage”, int – размер квартиры. 4В
   * “price”, int – предполагаемая цена. 4В
   * “rooms”, int – число комнат. 4В
   * “picture”, String – название изображения – 15В
   * “metro”, int – идентификатор станции. 4В
2. Владелец:
   * “\_id”, int – идентификатор. 4В
   * “FIO”, String – Имя владельца. 16В
   * “telephone”, String – контактные данные владельца. 10В
3. Метро:
   * “\_id”, int – идентификатор. 4В
   * “name”, String – название станции. 16В
4. Квартира\_Владелец
   * “flat”, int – идентификатор квартиры. 4В
   * “owner”, int – идентификатор владельца. 4В

«Чистый»: Пусть в базе хранится n станций метро, у каждой станции продаётся 10 квартир, каждую продаёт 1 уникальный человек. Тогда объём всей БД равен: 1750\*n В

«Фактический»: Запись в таблице квартира занимает в среднем 214В, владелец – 80В, метро – 24В, квартира\_владелец – 8В

Тогда объём – 3044\*n В

Избыточность: 3044/1750 = 1,74

Запросы:

1. Количество комнат квартиры

SELECT rooms FROM Flat WHERE \_id=1

1. Квартиры, продаваемые рядом с определённой станцией

SELECT Flat.\_id FROM Flat

LEFT JOIN Metro ON Metro.\_id=Flat.metro

Вывод

1. Запросы на SQL-языке являются более громоздкими, чем на языке Neo4j Cypher.
2. В реализации реляционной модели данных требуется создать больше таблиц для связи всей информации.
3. Neo4j-БД требует больше памяти по сравнению с SQL-БД для хранения полей.

Для рассматриваемой задачи NoSQL модель данных предпочтительнее SQL модели в плане сложности запросов к БД, однако уступает в плане требуемой для хранения данных памяти