Introduction to Graph Databases

by Swami M

Agenda

- Why Graph Databases?
- neo4j
- Demo
- Optimization

What is a Database?

- Organize data for easy access

Types of Databases

- SQL Databases (MySQL, PostgresSQL, Google CloudSQL etc...)
- NoSQL Datbases (MongoDb, Google Datastore etc...)
- Graph Databases (Neo4j, ArangoDB)

Let's design database for an ecommerce application (promotions domain)

Specifications (promotions domain)

1. Business should be able to create Promotions on their inventory
2. Consumers should know the list of applicable SKUs/Variants for a given Promotion

SKU - Stock Keeping Unit

Inventory Specification

- Merchant: Single Business
- Category: Mobile Phones, Electronics etc...
- Brand: Apple, LG etc...
- Product: iPhone, LG OLED TV etc...
- Variant: iPhone 65 GB Space Grey, LG Smart OLED 4K 65B9PUA

Whiteboard View of the Problem

flow diagram

Will SQL Databases work well?

Schema

How about NOSQL Databases?

Schema

Finally…

Introduction to Graph Databases :)

Why would Graph Databases work?

- Relationships are first-class citizens
- Closely represent whiteboard models
- Cheap Traversals

The Cypher Query Language

Let's create a simple graph using Cypher queries

model

CREATE Clause

```
CREATE (appleBrand:Brand{id:'apple_123', name:'Apple'}) RETURN appleBrand;
```

Relationships

```
MATCH (appleBrand:Brand{id:'apple_123'})
MATCH (iphoneVariant:Variant{id:'variant_123'})
CREATE (appleBrand)-[:has_variant]->(iphoneVariant);
```

Demo

Optimizations

how neo4j works

Further Reading

- Indexes in Neo4j click here
- Match Query Optimizations click here
- Neo4j Architecture click here

Thank you for your time

:)

Questions?