Getting to Know:

& Graph Database

TABLE OF CONTENTS

01

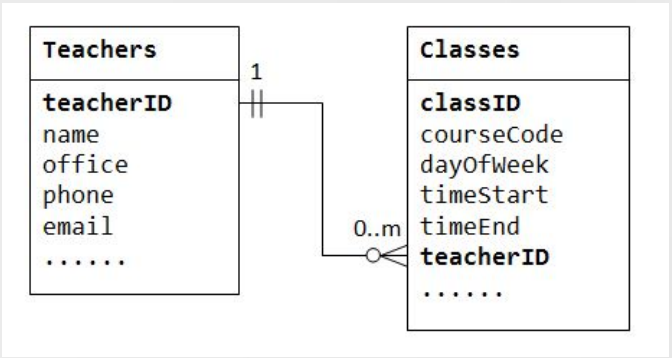
INTRODUCTION TO GRAPH DATABASE

DBMS

(Database Management System)

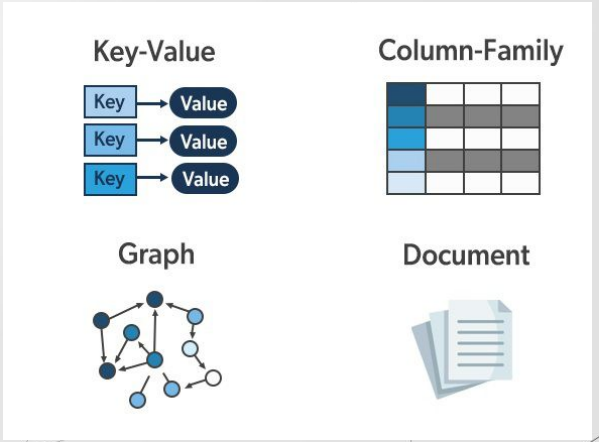
Relational

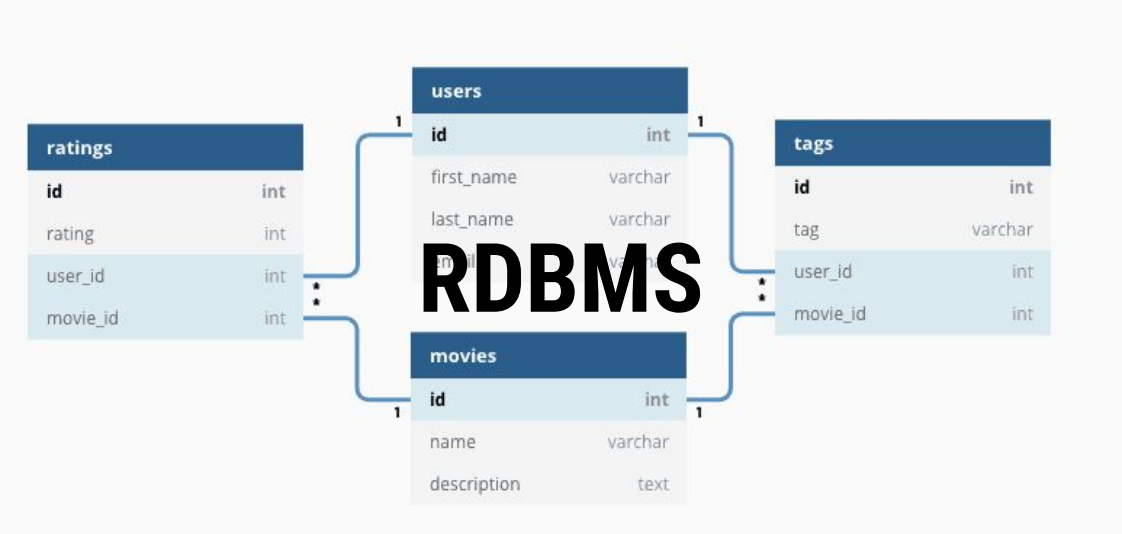
SQL



Non-Relational

No SQL



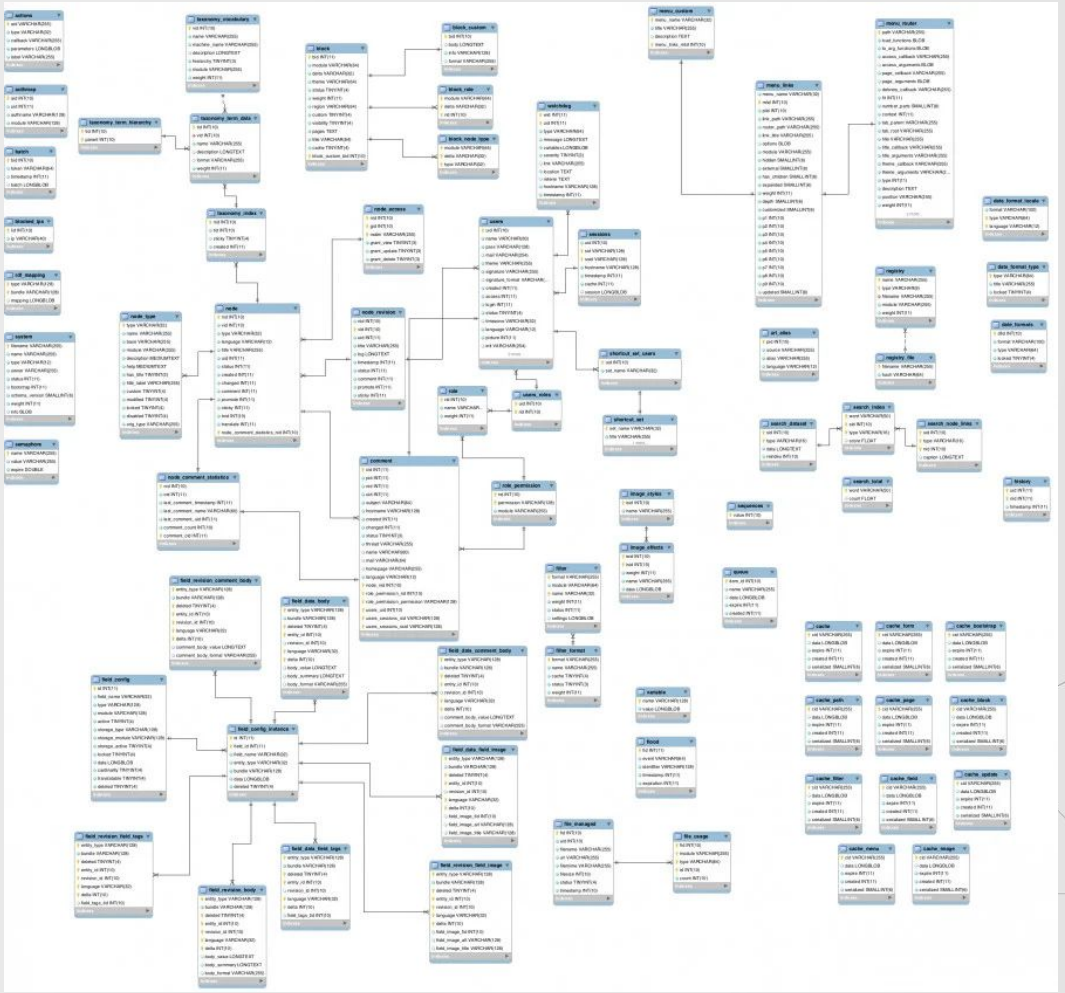


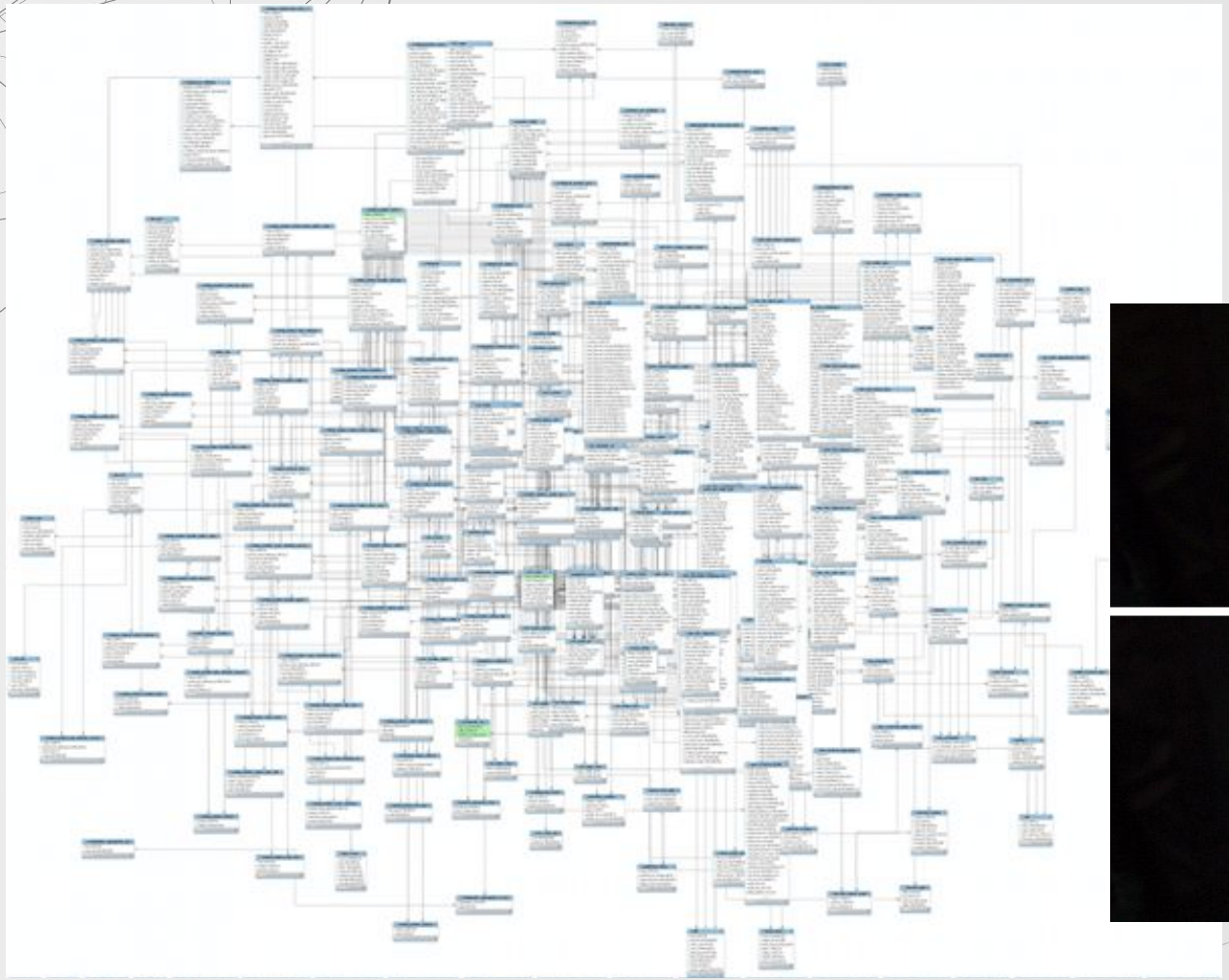
Pro's

- Easy to use
- Well understood structure
- Good precision
- Operational database

Con's

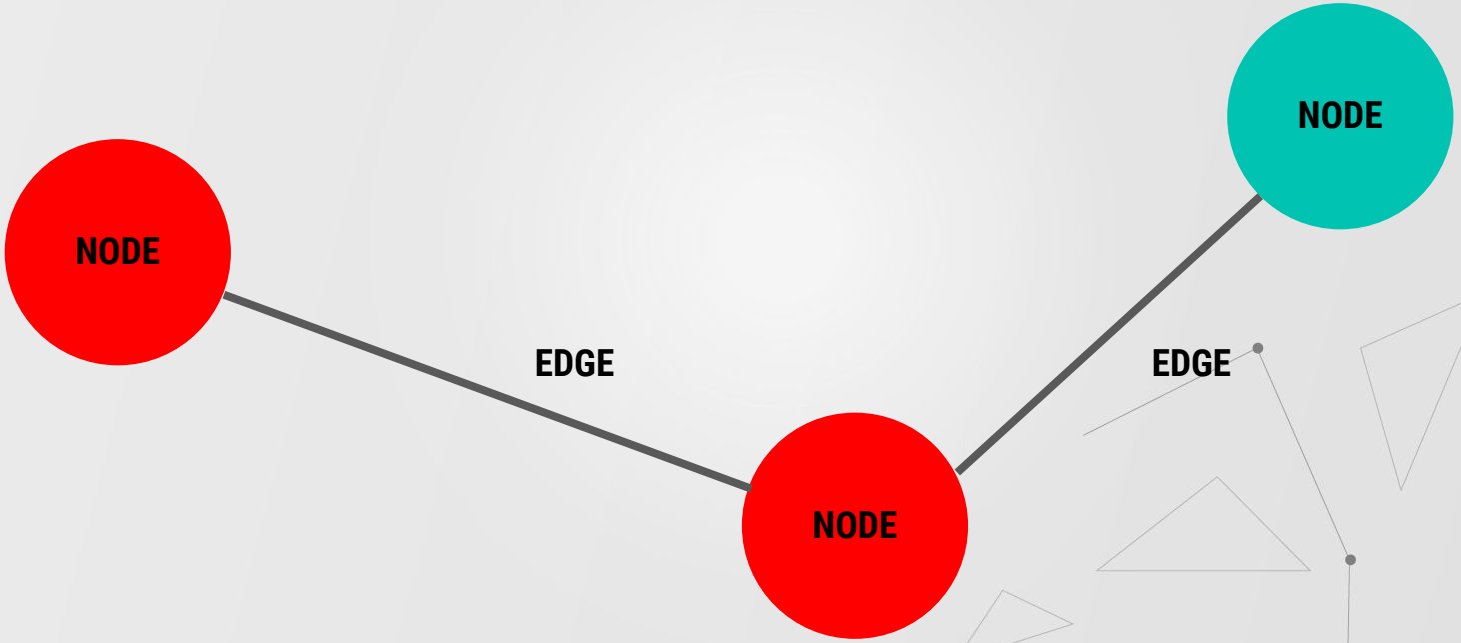
- Can't change frequently
- Rigid schema
- More computing power

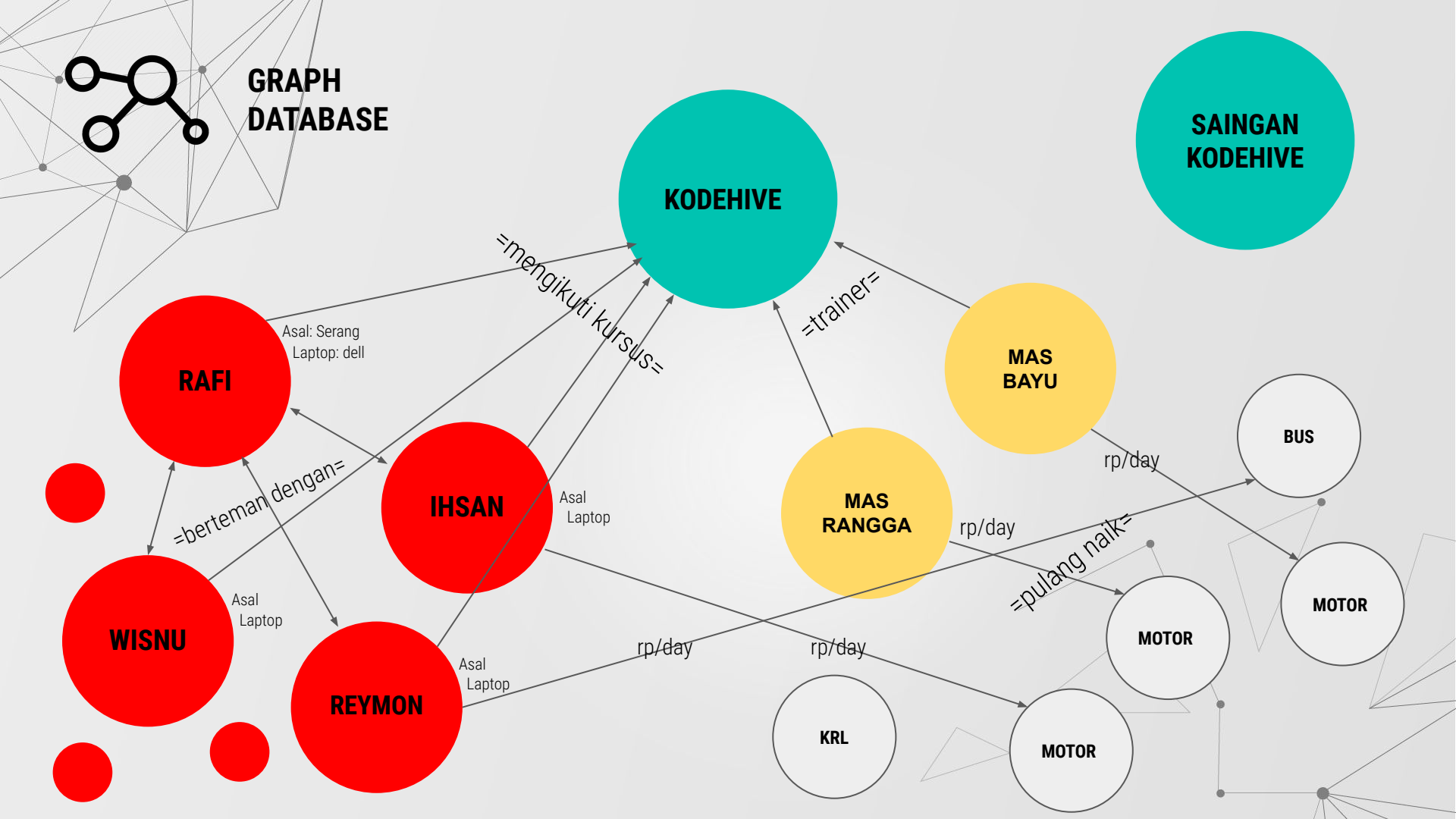





DEVELOPER
He's so ugly!
I love him!

GRAPH DATABASE




GRAPH DATABASE
KODEHIVE
SAINGAN KODEHIVE
=mengikuti kursus=
=trainer=
RAFI
Asal: Serang
Laptop: dell
MAS BAYU
=berteman dengan=
IHSAN
Asal
Laptop
MAS RANGGA
BUS
rp/day
rp/day
=pulang naik=
MOTOR
MOTOR
WISNU
Asal
Laptop
REYMON
Asal
Laptop
rp/day
rp/day
KRL
MOTOR

GRAPH DATABASE

PROBLEM THAT SOLVED BY GRAPH DATABASE

- Solves Many-to-Many relationship problems
- When relationships betweens data elements are more important
- Low latency with large scale data

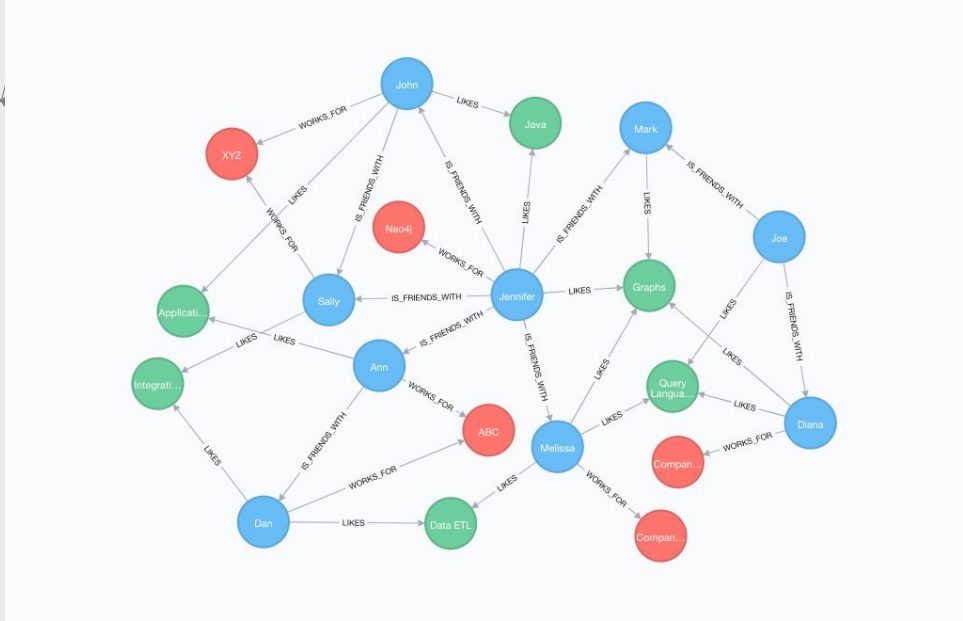
GRAPH DATABASE

Factor	Knowledge Graph	Relational Database
Storage approach	Entities and Relationships are stored as Nodes and Edges respectively	Data is stored in tables as rows and columns. Joins are created between tables for fast querying. The relationships between the columns of a table are inferred, but never stored separately.
Schema	Schema-free. Unstructured.	Rigid schema. Data Structure and format are pre-defined.
Purpose	Solely for uncovering hidden insights. Doesn't serve operational purposes.	Serves both operational and analytics purposes
Performance	Blazing fast even for large sets of data	Relatively slower than Knowledge Graphs
Maintenance	A lot easy, as they are schema-free	Difficult and often cumbersome, as minor changes could affect the entire structure

02

WHEN WE USE GRAPH DATABASE?

WHEN WE USE GRAPH DATABASE?



USED FOR:

- Complex independent connection
- Data model continuously changed
- Retrieving data more important than storing
- Not a tabular data
- Huge data collection

DONT USE WHEN:

- Data is disconnected and relationships do not matter
- Storing data more important than retriev
- Data model/schema stay consistent and not changed
- Tabular data

03

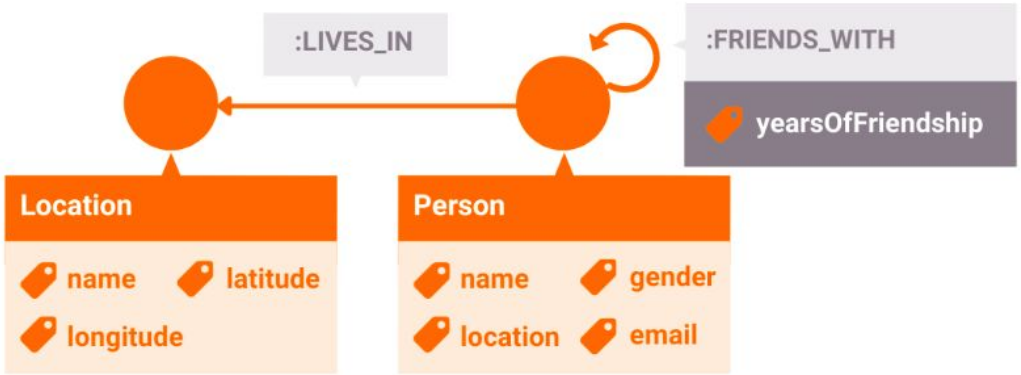
HOW IS IT DIFFERENT FROM OTHER DB's

The Graph Data Model

First things first! To decide if you need a graph database, you need to be familiar with the basic terminology. The fundamental components of a graph database are:

1. **Nodes** - the main entities in a graph. You can think of them as rows in a relational database.
2. **Relationships** - the connections between those entities. These would be foreign keys in a relational database.
3. **Labels** - attributes that group similar nodes together.
4. **Properties** - key/value pairs stored within nodes or relationships.

In a typical social network graph, the nodes represent people in different social groups and their connections with one another. Every person is represented with a node that's labeled as `Person`. These nodes contain the properties `name`, `gender`, `location` and `email`. The relationships between people in this network are of the type `FRIENDS_WITH` and contain a `yearsOfFriendship` property to specify the duration of the friendship connection. Each person is assigned a location through `:LIVES_IN` relationships with nodes labeled `Location`.



04

THANKS

Does anyone have any questions?


slidesgo