Graph Databases

Jose Andrés Campos Castro
Roberto Gutiérrez Sánchez

Agenda

- Graphs
- Types of graphs.
- History of Graph Databases.
- DBMS Examples.
- Graph Databases in Neo4j.
- Example.

Graphs

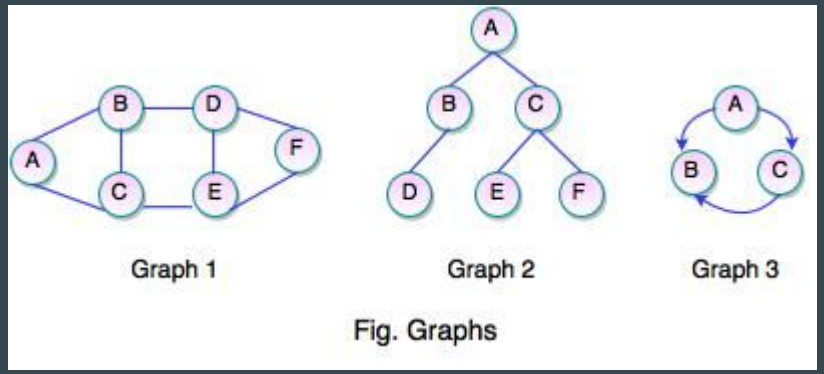
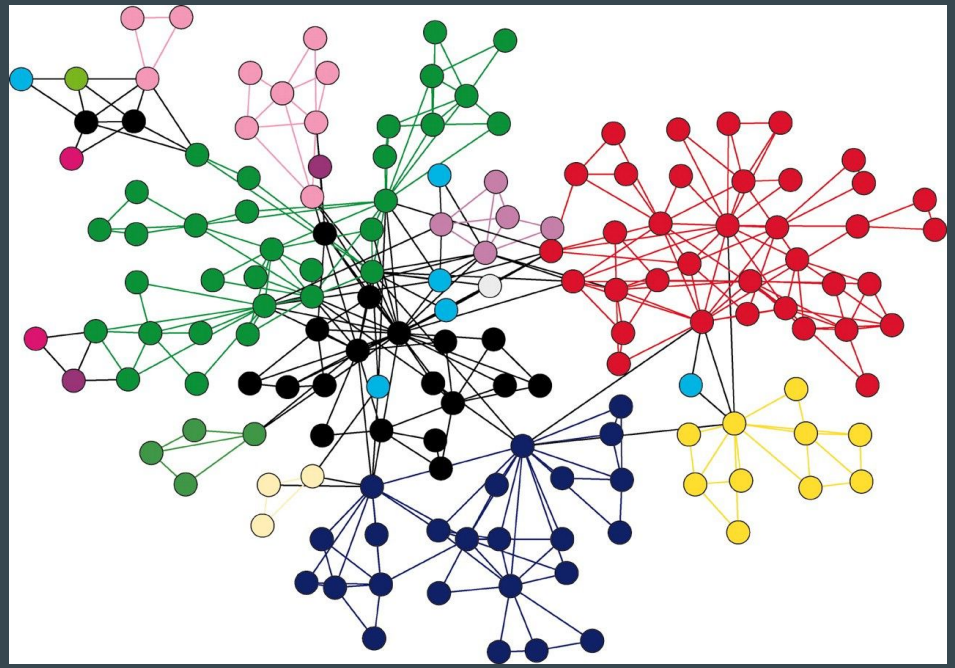
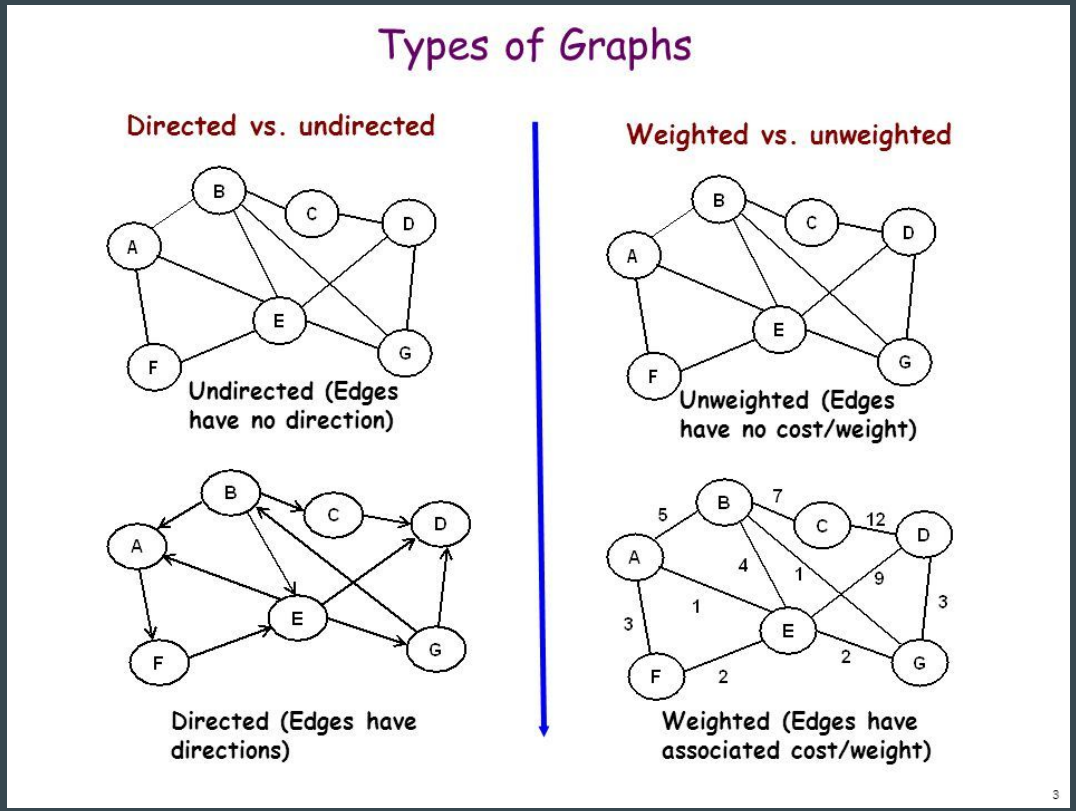


Fig. Graphs



Types of Graphs



History of Graph Databases

70's → Tabular Databases

80's → Relational Databases

90's → NoSQL.

BASE	ACID
Basic **A**vailability	**A**tomic: Everything in a transaction succeeds or the entire transaction is rolled back.
Soft-state	**C**onsistent: A transaction cannot leave the database in an inconsistent state.
Eventual consistency	**I**solated: Transactions cannot interfere with each other.
	Durable: Completed transactions persist, even when servers restart etc.

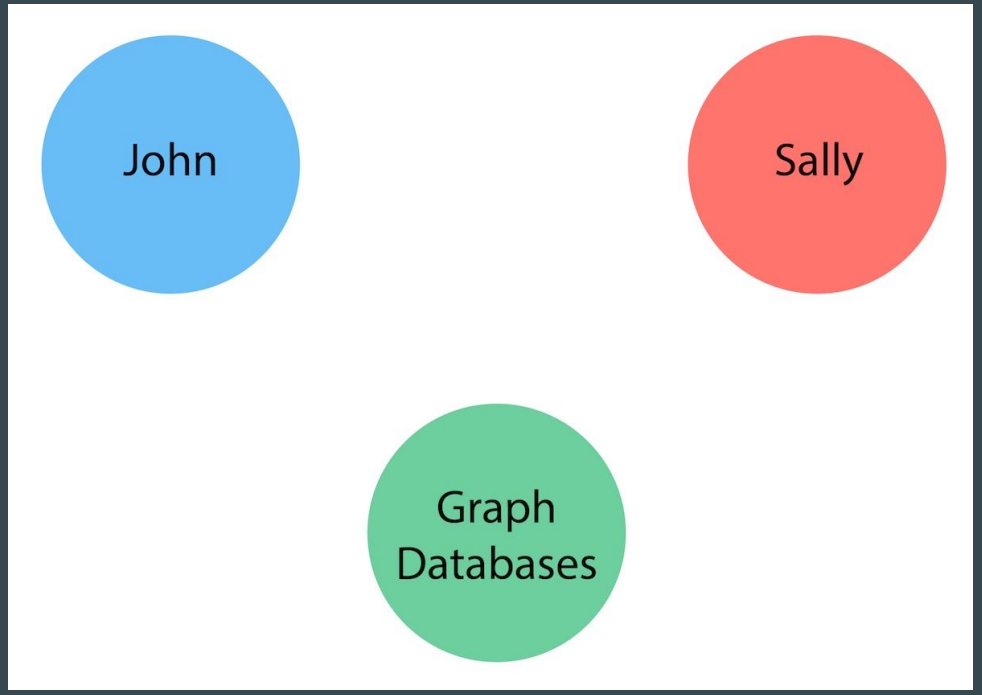
DBMS Examples.

neo4j	Cypher	Most famous graph database, Cypher O(1) access using fixed-size array
DSE Graph DATASTAX AURELIUS	Gremlin	Distributed graph system based on Cassandra
ArangoDB	AQL	Multi-model database (Document + Graph)
OrientDB	OQL	Multi-model database (Document + Graph)

Graph Databases in Neo4j.

Nodes

The first entities that we will identify in our domain are the nodes. Nodes are one of two fundamental units that form a graph (the other fundamental unit is relationships).
Nodes are often used to represent entities, but can also represent other domain components, depending on the use case.

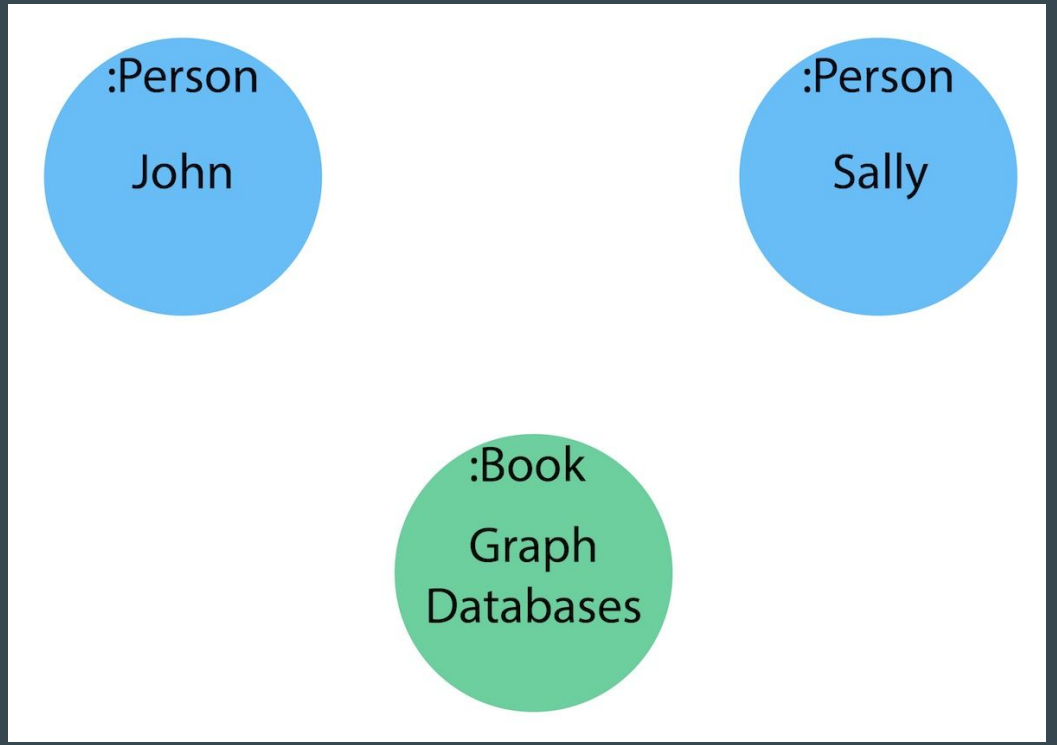


Graph Databases in Neo4j.

Labels

Now that we have an idea of what our nodes will be, we can decide what labels (if any) to assign our nodes to group or categorize them.

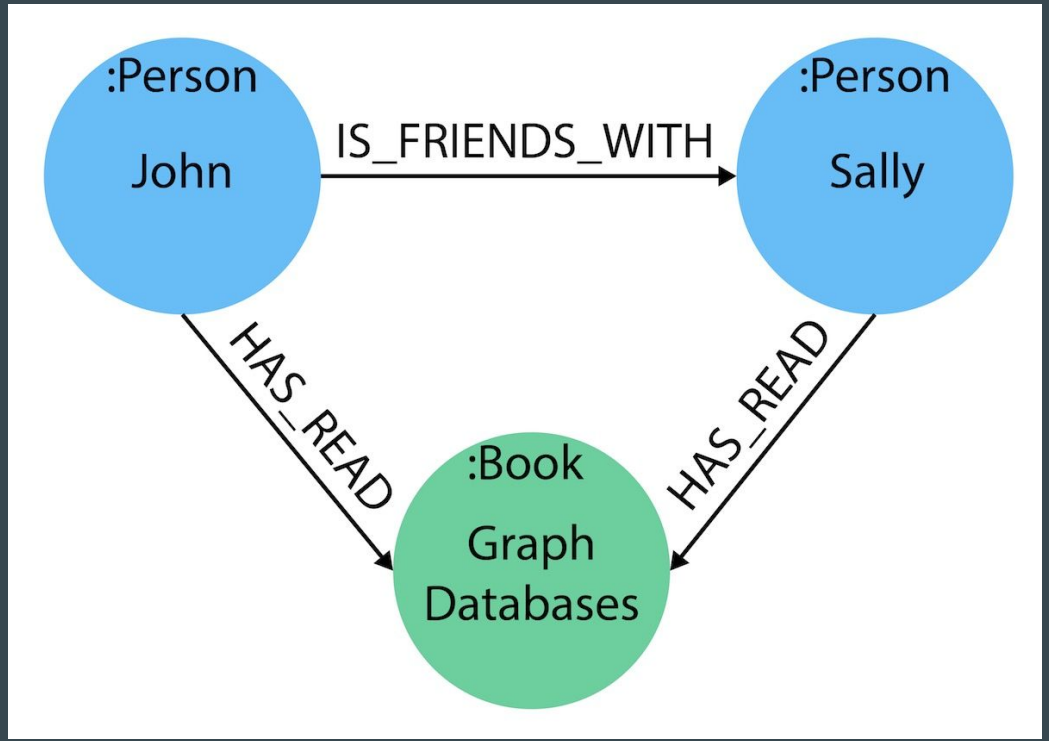
A label is a named graph construct that is used to group nodes into sets. All nodes labeled with the same label belongs to the same set.



Graph Databases in Neo4j.

Relationships

A relationship connects two nodes and allows us to find related nodes of data. It has a source node and a target node that shows the direction of the arrow.

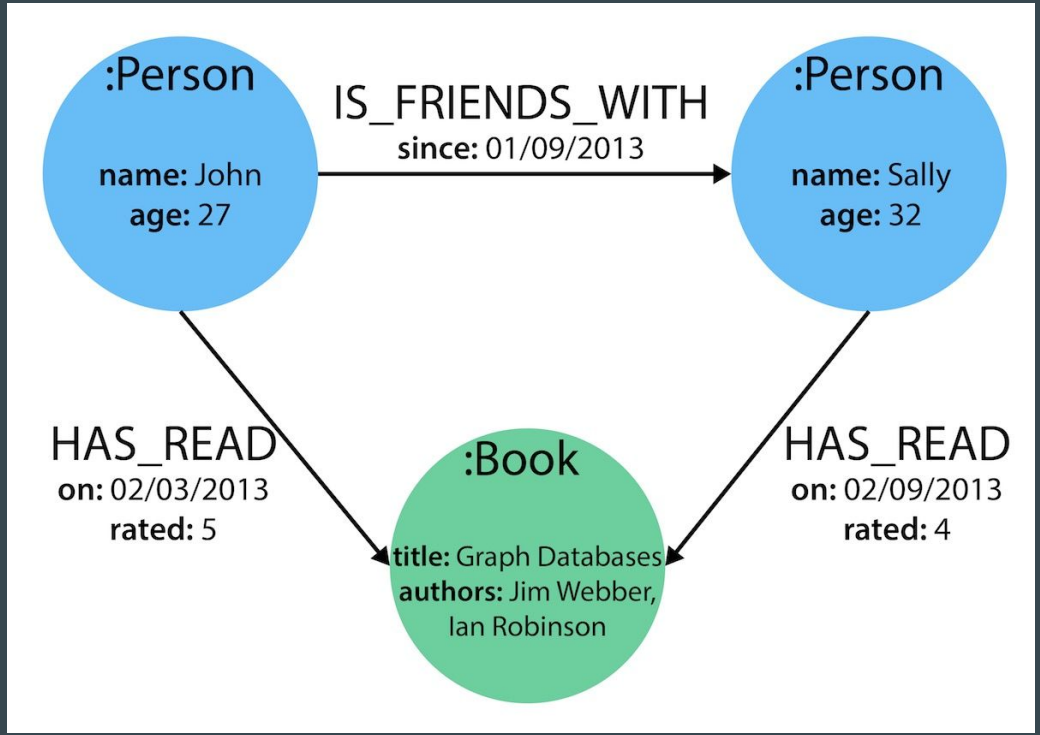


Graph Databases in Neo4j.

Properties

Properties are name-value pairs of data that you can store on nodes or on relationships. Most standard data types are supported as properties.

Properties allow you to store relevant data about the node or relationship with the entity it describes. They can often be found by knowing what kinds of questions your use case needs to ask of your data.



Example

https://github.com/campos-97/GraphDBExample

https://neo4j.com/download/

Bibliography

- "History of Databases and Graph Database - Bitnine Global Inc.", Bitnine Global Inc [Online]. Available: https://bitnine.net/blog-graph-database/history-of-databases-and-graph-database/. [Accessed: 7- Oct- 2018].
- "Impossible Is Nothing: The History (& Future) of Graph Data [GraphConnect Recap] - Neo4j Graph Database Platform", Neo4j Graph Database Platform. [Online]. Available: https://neo4j.com/blog/history-and-future-of-graph-data/. [Accessed: 7- Oct- 2018].
- "Relational Databases vs. Graph Databases: A Comparison", Neo4j Graph Database Platform .[Online]. Available: https://neo4j.com/developer/graph-db-vs-rdbms/. [Accessed: 7- Oct- 2018].
- J. Cook, "ACID versus BASE for database transactions", Johndcook.com. [Online]. Available: https://www.johndcook.com/blog/2009/07/06/brewer-cap-theorem-base/. [Accessed: 7- Oct- 2018].