Graph Database

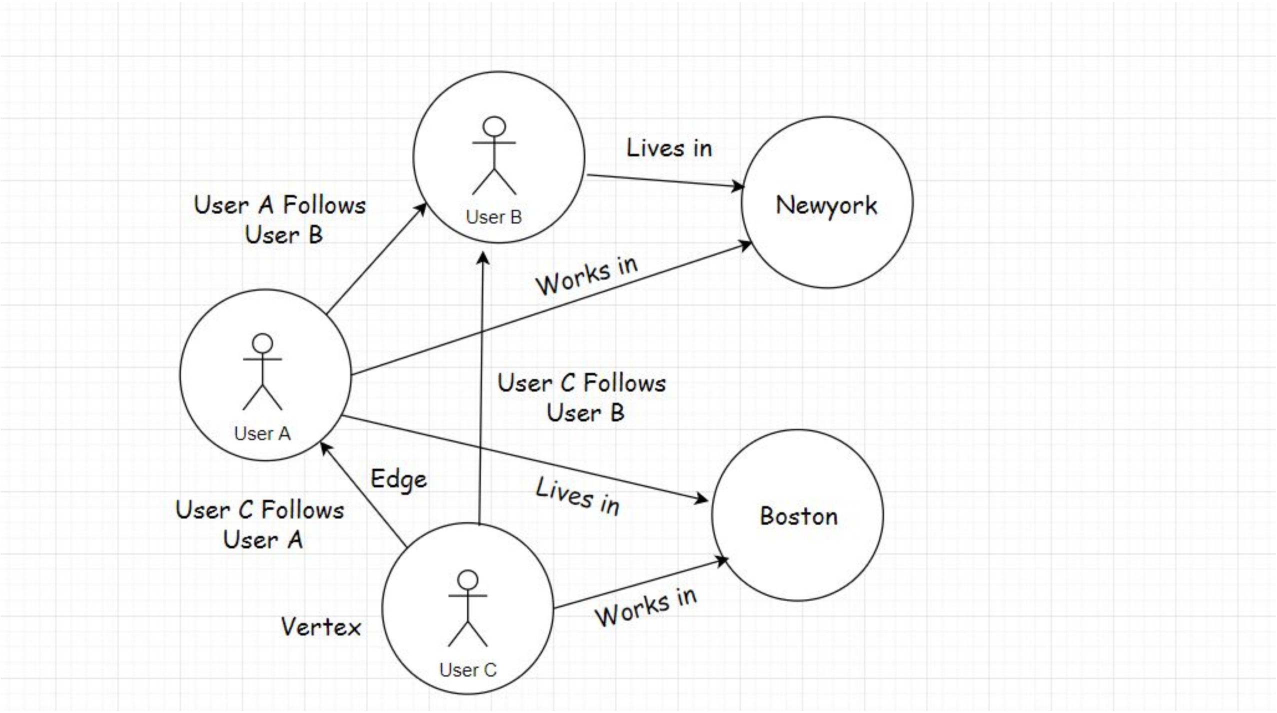
In this lesson, you will learn about the graph database and when to choose it for projects

We'll cover the following

- What is a graph database?
- Features of a graph database
- When do I pick a graph database?
- Real-life Implementations

What is a graph database?#

Graph databases are also a part of the *NoSQL* database family. They store data in *nodes/vertices* and *edges* in the form of relationships.



Each *node* in a graph database represents an entity. It can be a person, a place, a business, etc., and the *edge* represents the relationship between the entities.

But, why use a graph database to store relationships when we already have SQL based relational databases available?

Features of a graph database#

There are two primary reasons. The first is visualization. Think of that pinned board in the thriller detective movies where the pins are pinned on a board over several images connected via threads. It helps visualize how the entities are related and how things fit together. Right?

The second reason is the low latency. In graph databases, the relationships are stored a bit differently than how relational databases store relationships.

Graph databases are faster because the relationships in them are not calculated at query time, as it happens with the help of *joins* in the relational databases. Rather, the relationships here are persisted in the data store in the form of edges, and we just have to fetch them. No need to run any sort of computation at the query time.

A good real-life example of an application that would fit a graph database is Google Maps. *Nodes* represent the cities, and the *edges* represent the connections between them.

Now, if I have to look for roads between different cities, I don't need *joins* to figure out the relationship between the cities when I run the query. I just need to fetch the edges, which are already stored in the database.

just need to fetch the edges, which are already stored in the database.

When do I pick a graph database?#

Ideal use cases of graph databases are building social, knowledge, and network graphs, writing AI-based apps, recommendation engines, and fraud analysis apps, storing genetic data, etc.

Graph databases help us visualize our data with minimum latency. A popular graph database used in the industry is *Neo4J*.

Real-life Implementations#

Here are some of the real-life implementations of the tech:

- Walmart shows product recommendations to its customers in real-time using Neo4J graph database (https://neo4j.com/blog/walmart-neo4j-competitive-advantage/)
- NASA uses Neo4J to store "lessons learned" data from their previous missions to educate the scientists and engineers. (https://neo4j.com/blog/david-meza-chief-knowledge-architect-nasa/)

← Back

Document-Oriented Database

Next →

Key-Value Database

☑ Mark as Completed

Report an Issue