Graph-Powered AI:

Unlocking Insights with Knowledge Graphs

Sathishkumar Madhanagopal | Udhayakumar Ulaganathan

thoughtworks

Agenda

Understanding Graphs & Relationships

Introduction to Knowledge Graphs

Neo4j as a Knowledge Graph Platform

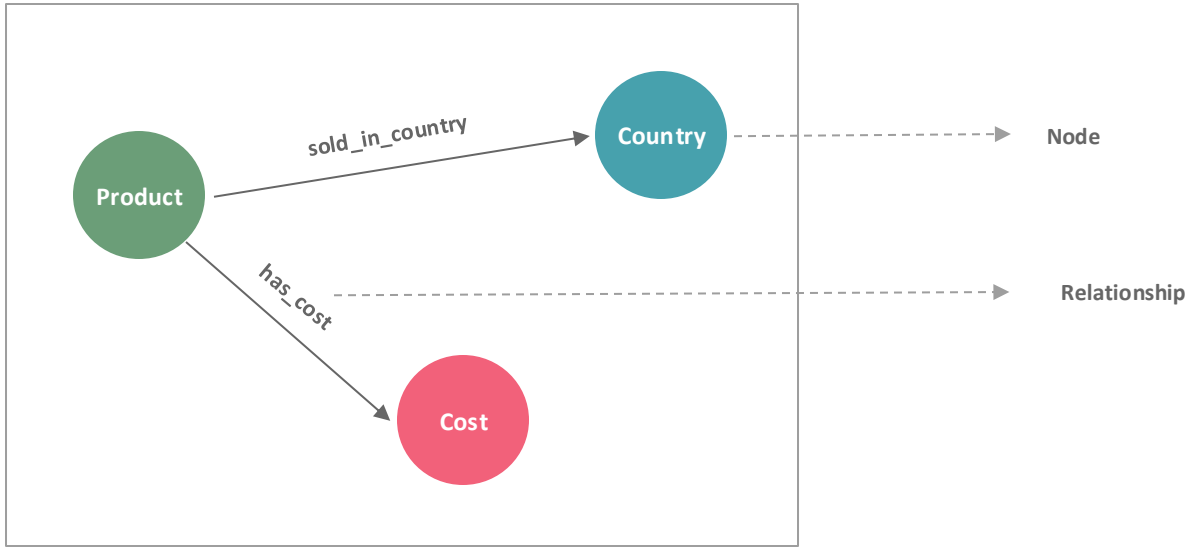
Explore Graph RAG with Knowledge Graph

Graph RAG Demo

Understanding Graphs & Relationships

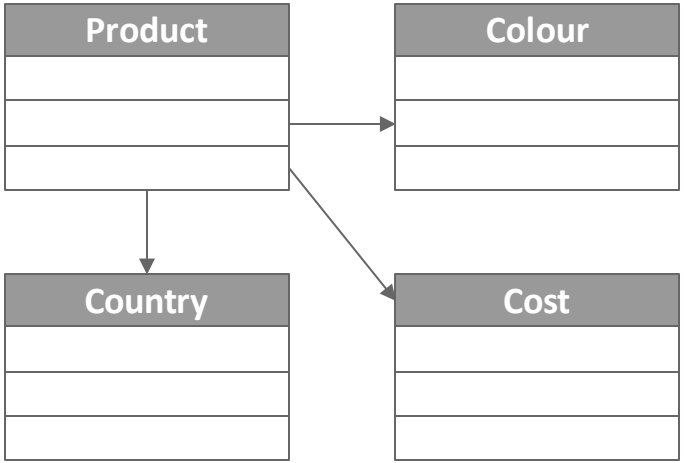
Graph DB

A type of database that stores data in the form of nodes connected with relationships.

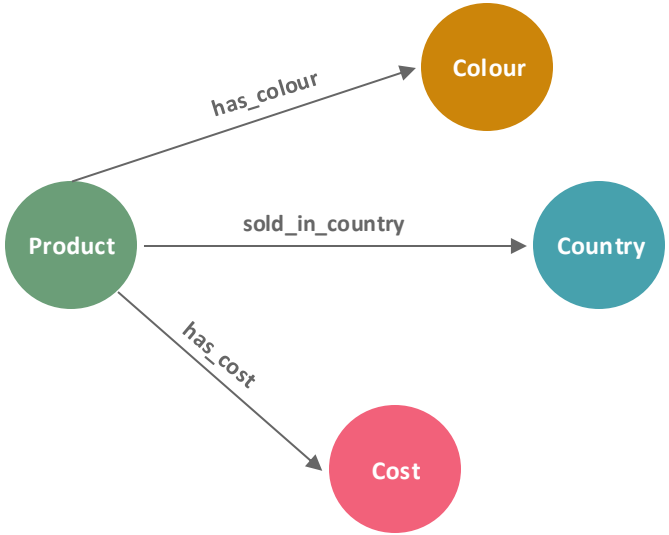


Relational DB vs Graph DB

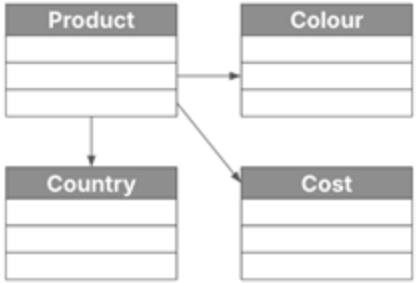
Relational DB



Graph DB

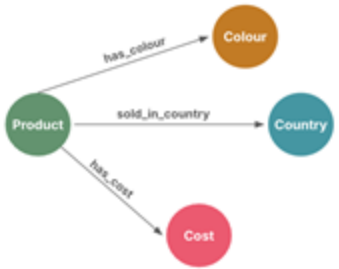


Cypher Query



SQL

```
SELECT p.ProductName, c.ColorName, co.CostAmount, cn.CountryName
FROM ProductTable p, ColorTable c, CostTable co, CountryTable cn
WHERE
    p.ColorID = c.ColorID
    AND p.CostID = co.CostID
    AND p.CountryID = cn.CountryID;
```



Cypher

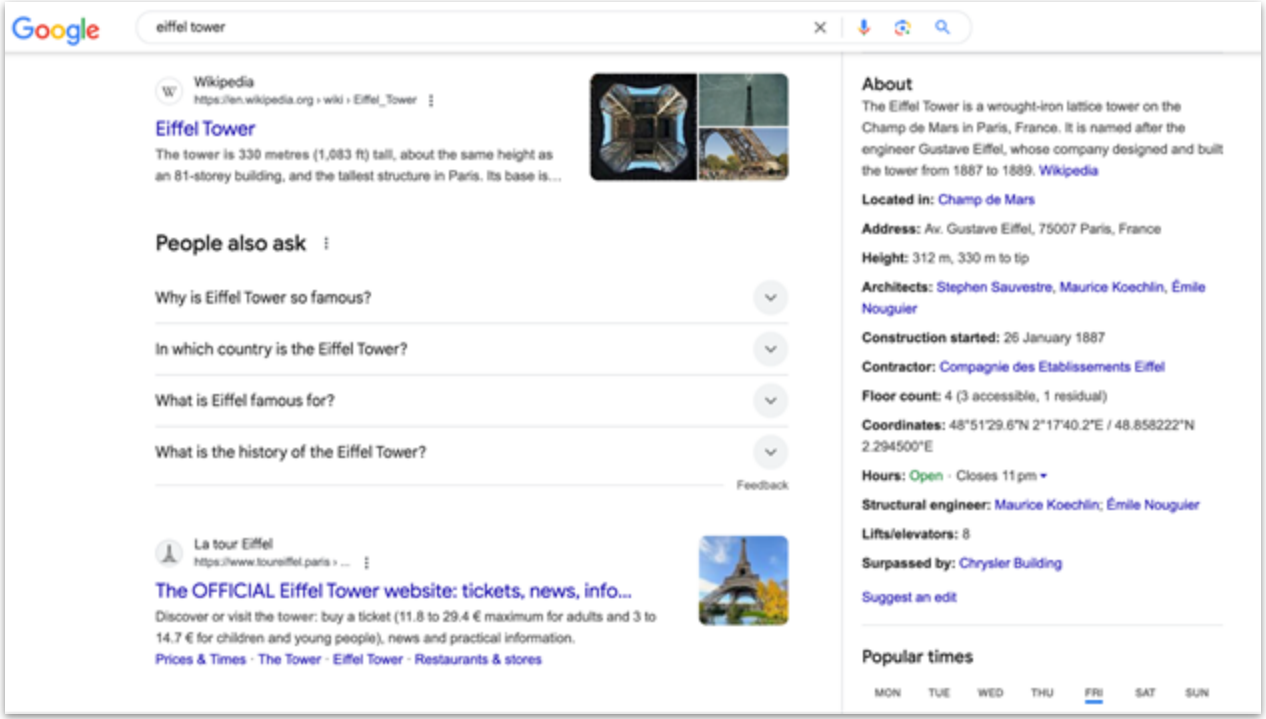
```
MATCH (p:Product)-[:HAS_COLOUR]->(c:Color),
      (p)-[:HAS_COST]->(co:Cost),
      (p)-[:SOLD_IN_COUNTRY]->(cn:Country)
RETURN p, c, co, cn;
```

Other Graph Query Languages - ***Gremlin, SparQL, GQL***

Cypher's Advantage - ***Declarative nature, rich ecosystem, optimised for deeper traversals***

Introduction to Knowledge Graphs

Knowledge Graph - Things, not Strings



Ref: https://blog.google/products/search/introducing-knowledge-graph-things-not/

Knowledge Graphs

Graph Database efficiently stores and queries relationships

Knowledge Graph goes beyond by **adding meaning, semantics, and reasoning** to the data.

Components of Knowledge Graphs:

1. **Nodes** - Entities
2. **Relationships** - between entities
3. **Organising Principles** - Concept map

Taxonomy -[WILL_DERIVE] -> OrganisingPrinciple

Product Recommendation System => Product Taxonomy is the Organising Principle

Why This Matters for AI?

- ✅ **Enhanced Search & Recommendations**
- ✅ **AI-powered Insights & Predictions**
- ✅ **Automated Reasoning & Decision-Making**

"A Graph Database tells you what exists, but a Knowledge Graph helps you understand and reason about it!"

Neo4J as a Knowledge Graph Platform

Neo4J

Neo4j is a leading native graph database platform designed to store and query highly connected data efficiently.

Key Features:

- **Native Graph Storage**
- **Cypher Query Language**
- **Fully managed cloud-based offering** : **Neo4j Aura**

Other Graph DB Offerings *:*

- Azure Cosmos DB, Amazon Neptune, Tiger Graph, Arango DB

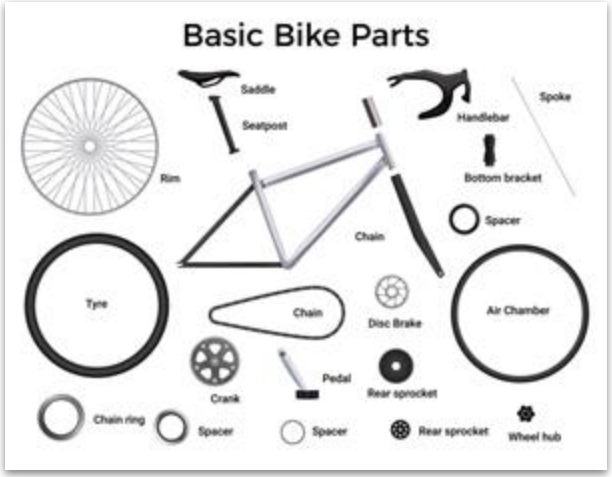
Let’s have a quick look at Neo4J

Our Journey with Neo4J

BigQuery and BOM Data. The application's performance began to decline as the volume of data increased.

Challenges Faced:

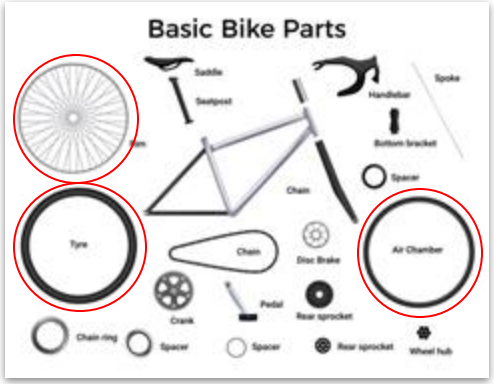
- **Performance Issues**: Recursively querying hierarchical BOM data in SQL caused significant delays and large joins across tables impacted query performance and readability.
- **Complexity**: Difficult to visualize data relationships directly.
- **Scalability Concerns**: As data grew, SQL query execution times worsened.

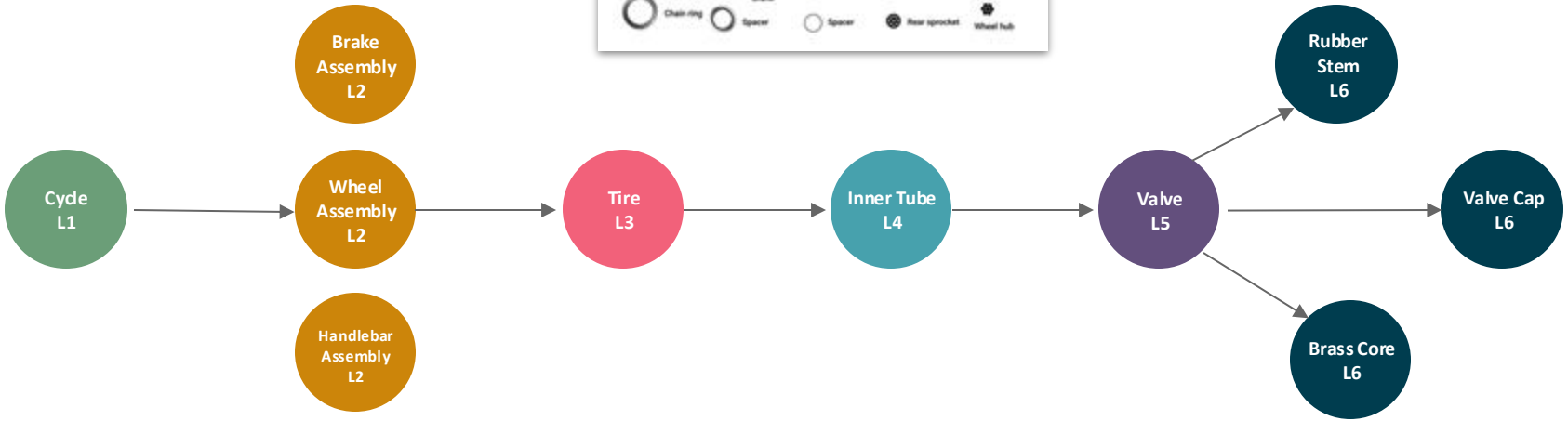


Designed by Freepik

BOM (Bill of Materials) is a hierarchical structure of assemblies, sub-assemblies, and parts, often used in manufacturing.

Hierarchical BOM Data





Neo4J vs BigQuery

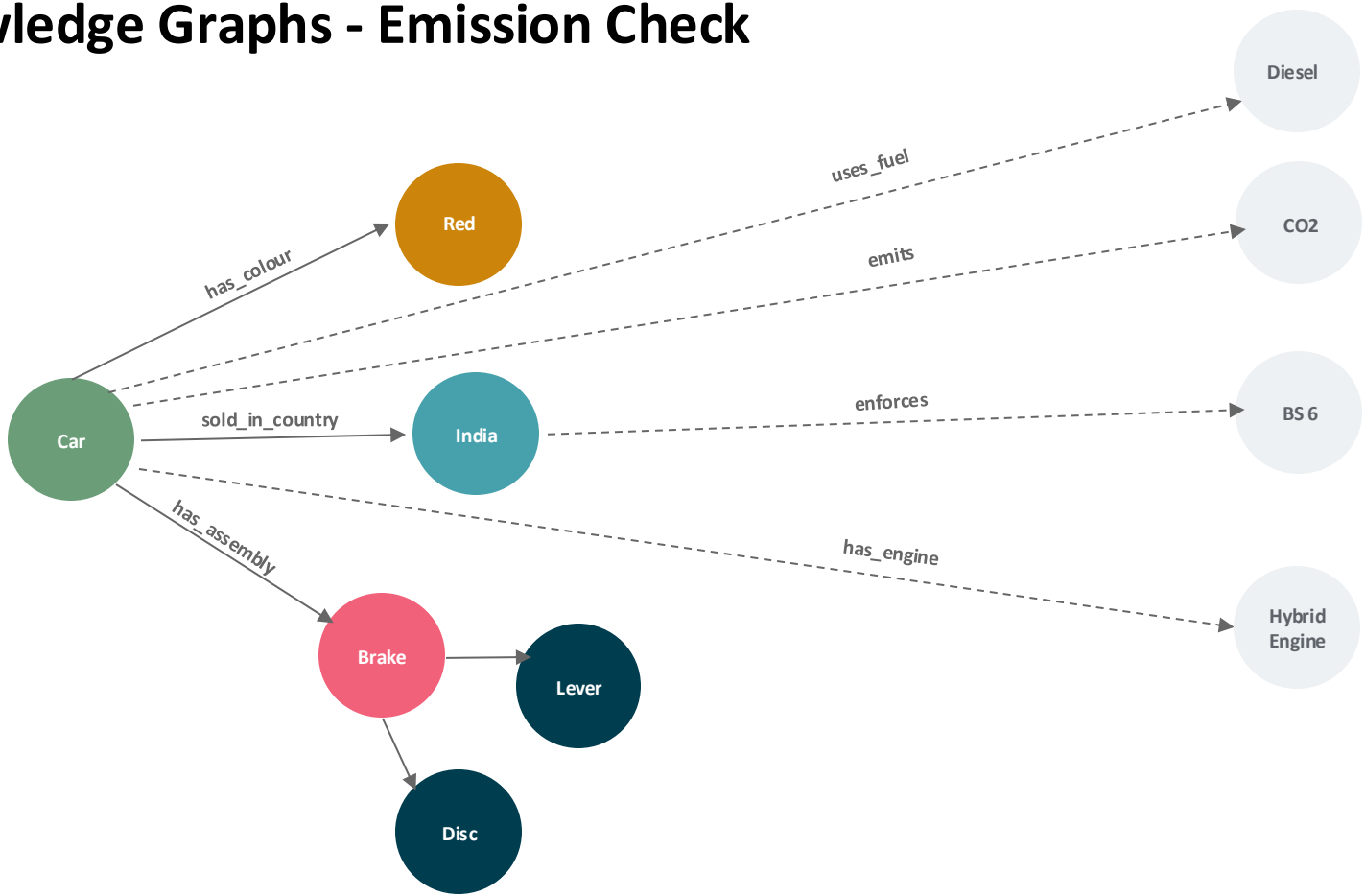
Neo4j's significant edge in scenarios requiring deep, recursive traversals

Metric	BigQuery (SQL)	Neo4j (Graph)
Traversal Depth	2 levels: ~2 second	2 levels: ~150 ms
	5 levels: ~15 seconds	5 levels: ~300 ms
	7 levels: ~40+ seconds	7 levels: ~600 ms
Scalability	Exponential increase in query time	Linear increase in query time
Reason for Difference	Intermediate joins and computation	Direct node and relationship traversal

*Data size - **~5 million nodes and ~10 million relationships***

Building a Knowledge Graph

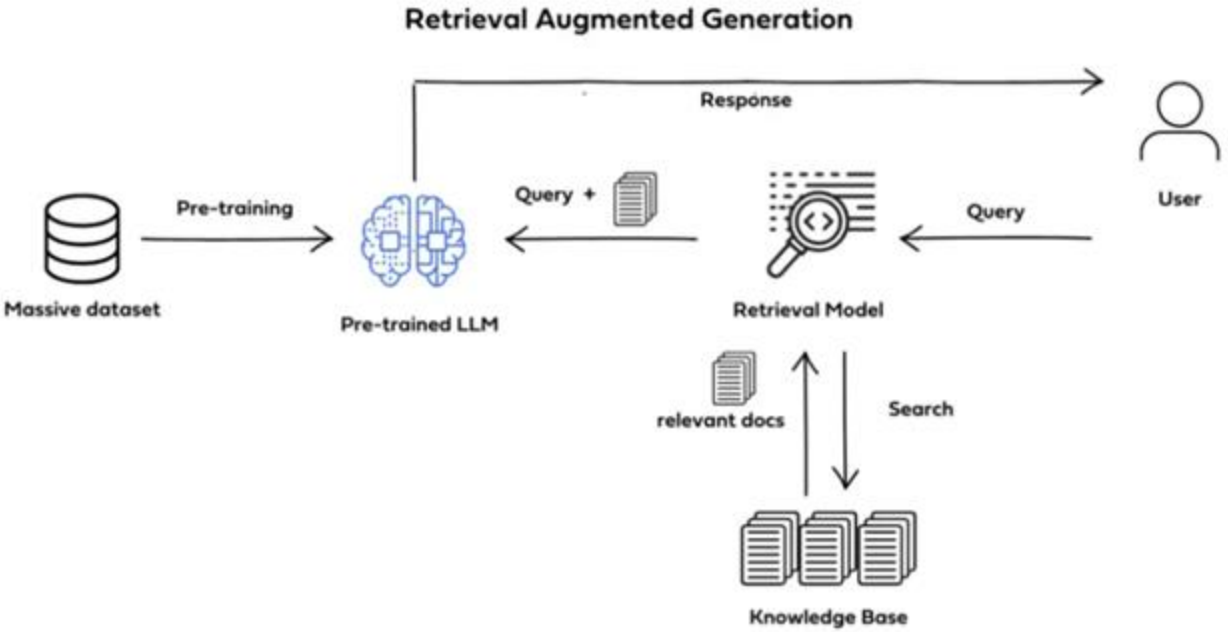
Knowledge Graphs - Emission Check



This transforms your **basic Graph Database** into a **true Knowledge Graph**, making it more **intelligent, inferential, and useful for AI-driven applications**.

Explore Graph RAG with Knowledge Graph

RAG



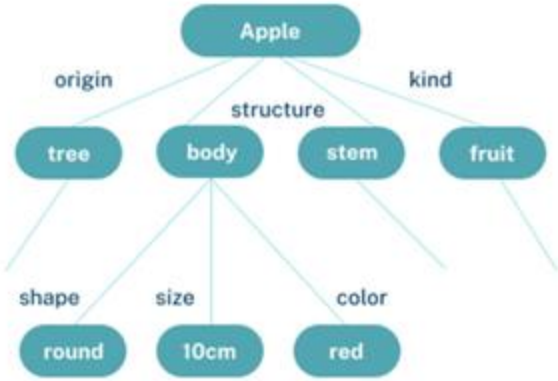
Enterprise Data

Vector and Graph

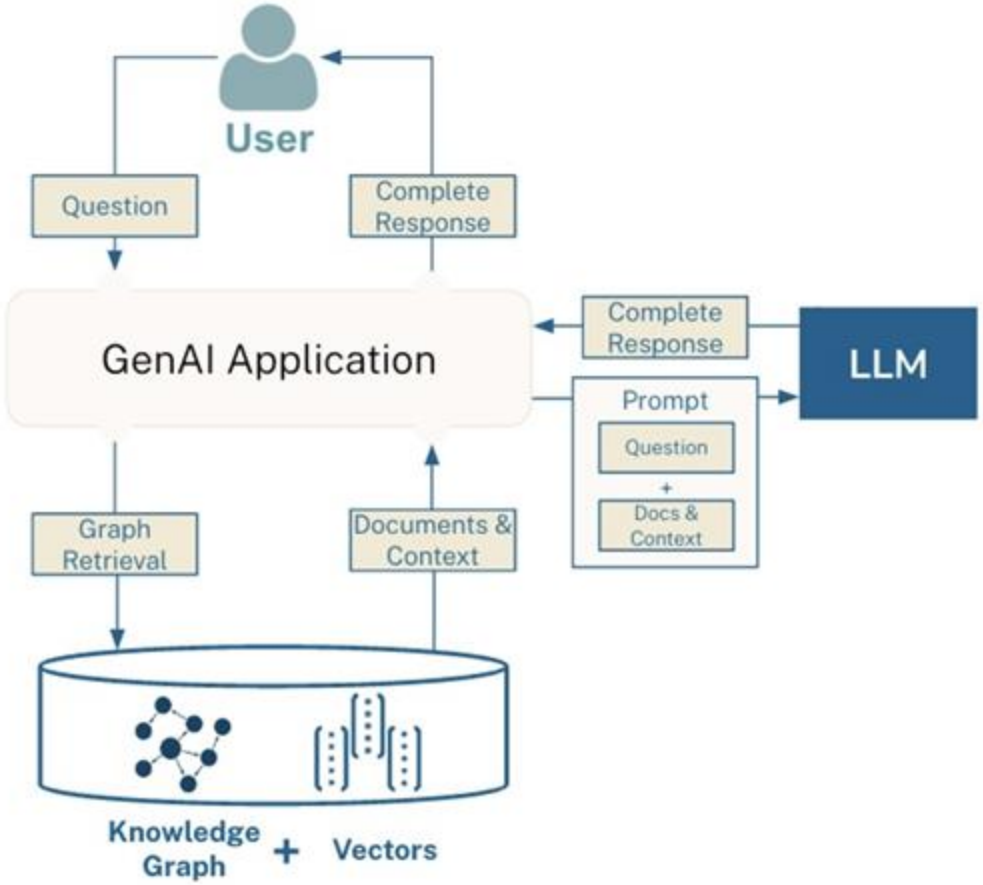
Human View of an Apple

Vector View of an Apple

Knowledge Graph View of an Apple



Graph RAG


User
Question
Complete Response
GenAI Application
Complete Response
LLM
Prompt
Question
+
Docs & Context
Graph Retrieval
Documents & Context
Knowledge Graph
+
Vectors

DEMO

Reference / Further Reading

https://neo4j.com/docs/getting-started/

https://neo4j.com/use-cases/knowledge-graph/

https://github.com/docker/genai-stack

https://neo4j.com/blog/graphrag-manifesto/

Thank you.