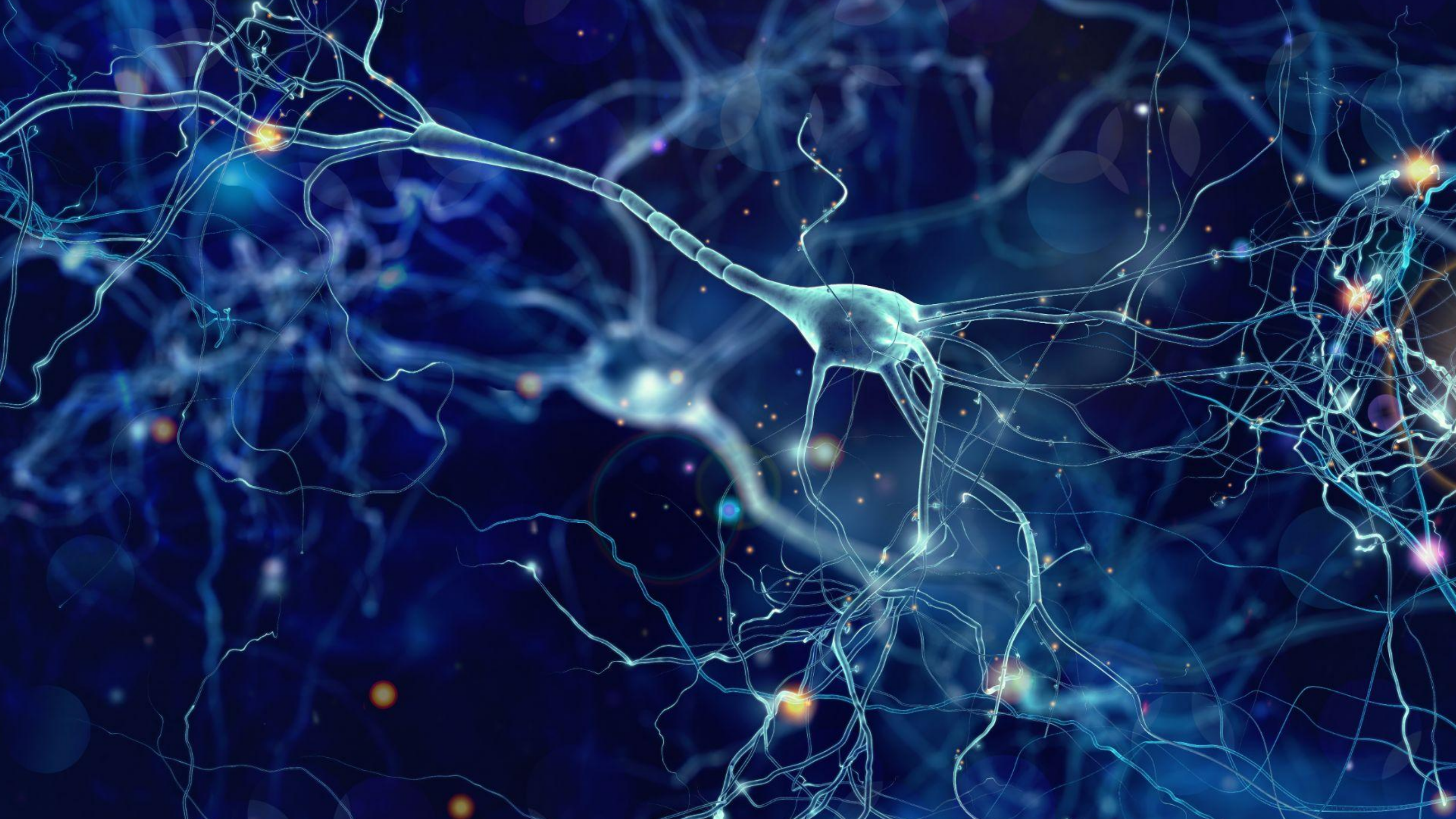


# Graphs + AI: Future Proof your Business

**Emil Eifrem (CEO & Founder Neo4j)**  
**Brad Bebee (GM AWS Neptune)**



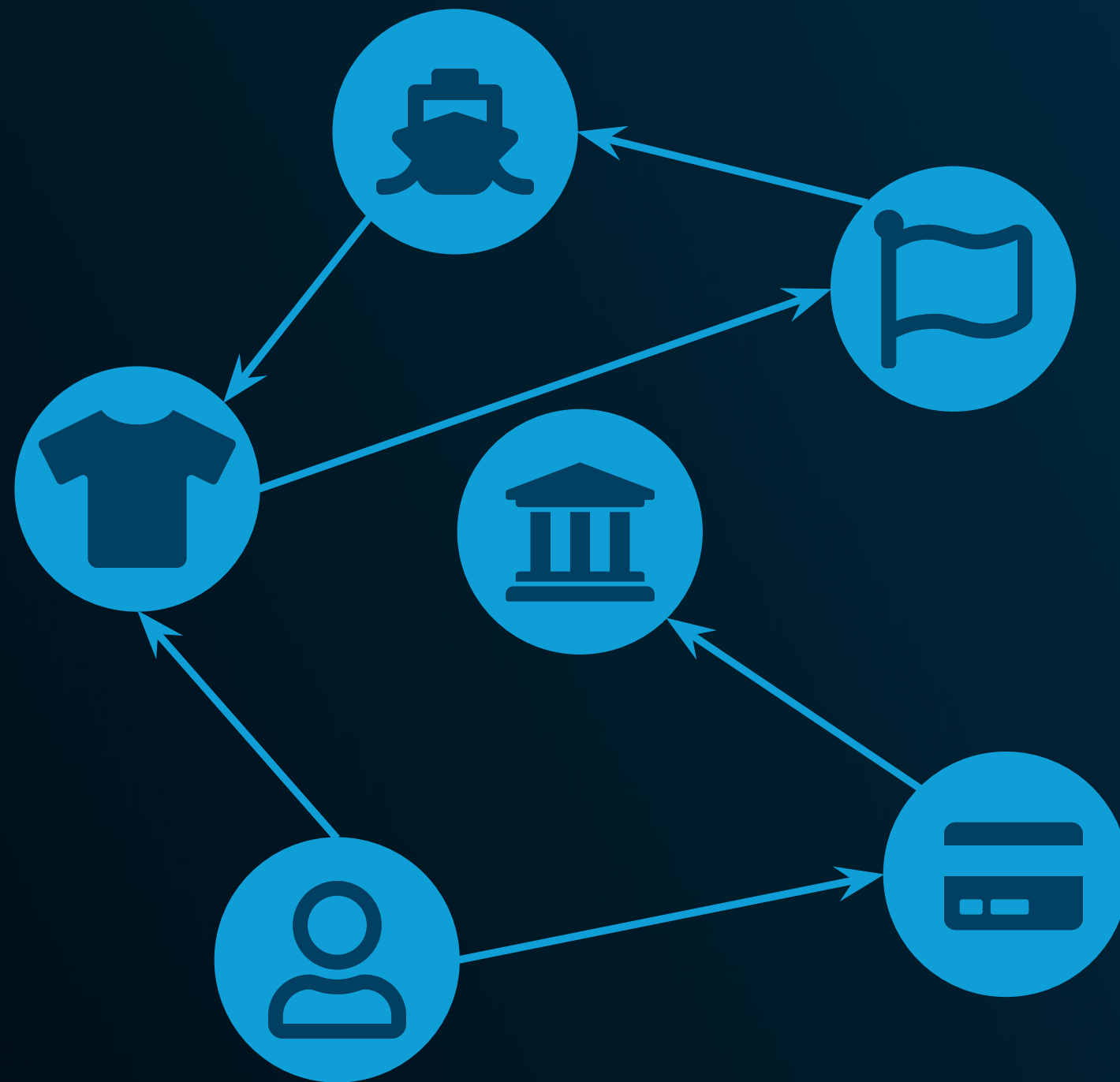




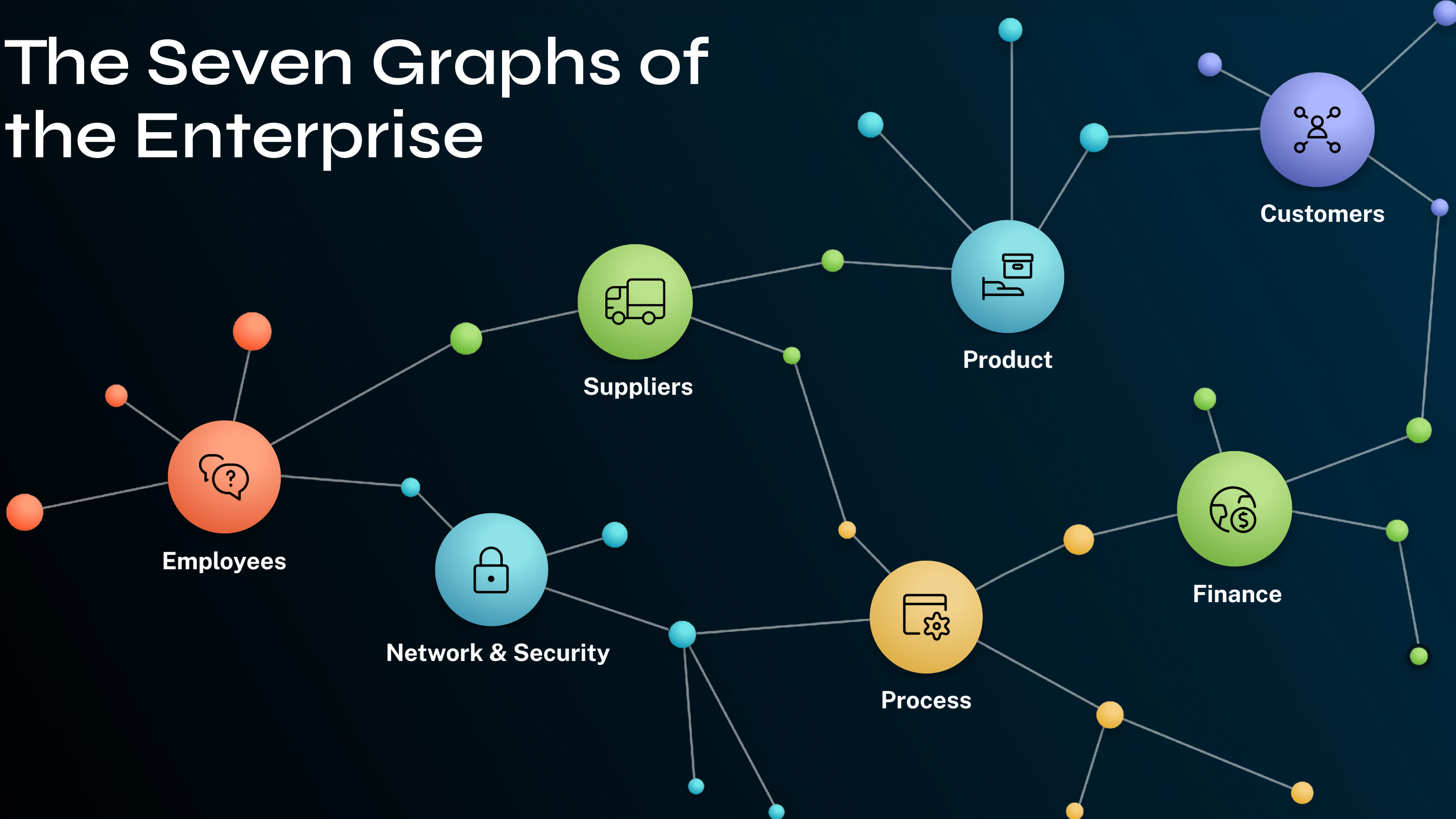




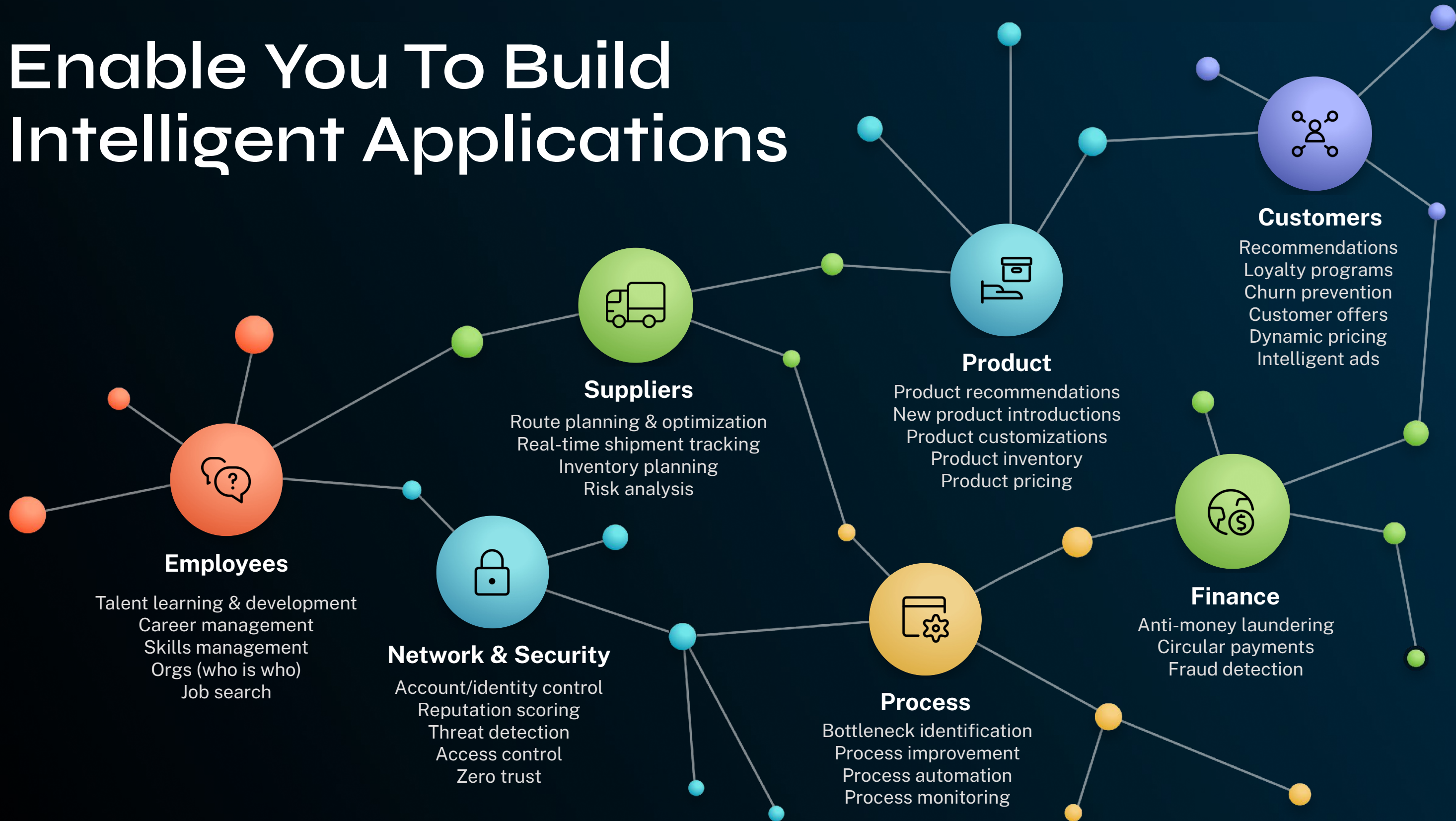


# The Seven Graphs of the Enterprise



# Enable You To Build Intelligent Applications



# AI is changing how we develop applications. **Quickly.**

**AXIOS**

"AI is fundamentally disrupting software development ..."

- Innovation Endeavors (VC) / Axios, June 2025

**HARVARD BUSINESS REVIEW**

"[GenAI]...will disrupt the very software that enabled the last wave of transformation."

- Harvard Business Review, May 2025

**Gartner**

"Organizations with rich, [connected] knowledge graphs will have a significant competitive advantage..."

- Gartner Perception is the New Superpower For the Future of Analytics and BI, 2025

**IDC**

"...AI pipelines need dynamic schemas to [handle] evolving sources."

- 2025 Future Enterprise Resiliency & Spending, IDC

**VentureBeat**

"A business data fabric reduces latency and powers advanced AI..."

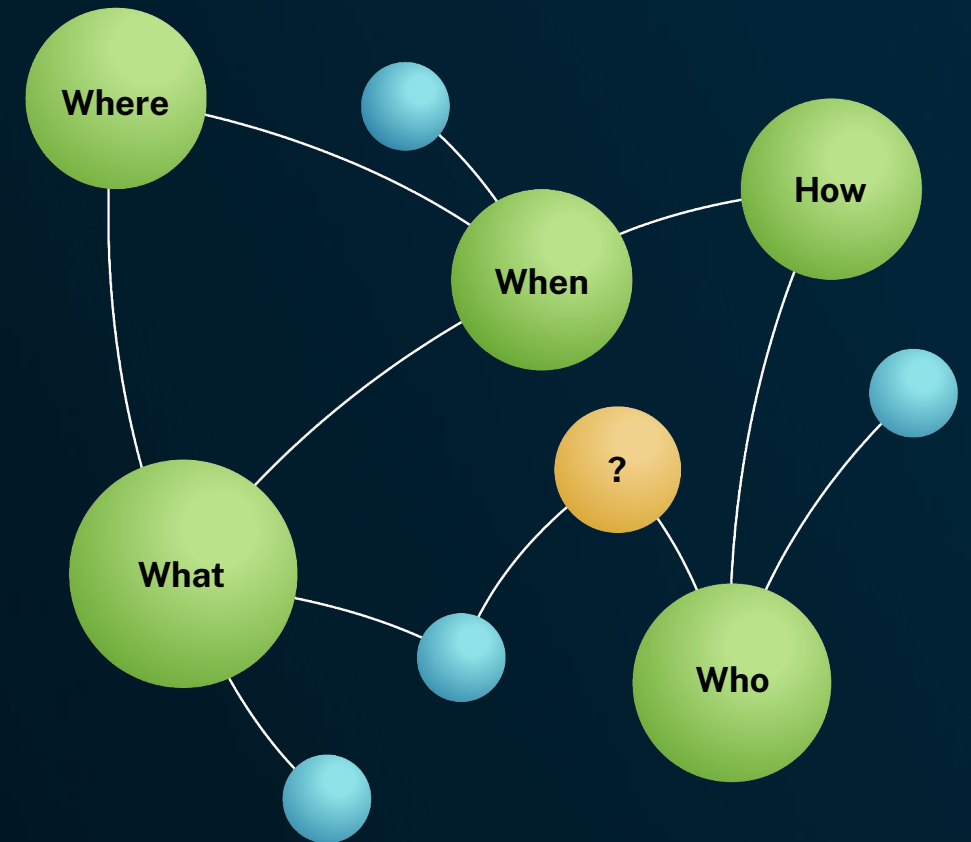
- Venture Beat, Enhancing business agility with rapid data integration and advanced AI, 2025

# Why we build applications is NOT changing

**Problem to solve**

**Business process to map**

**Questions to answer**







Predefined workflows on simple CRUD data models

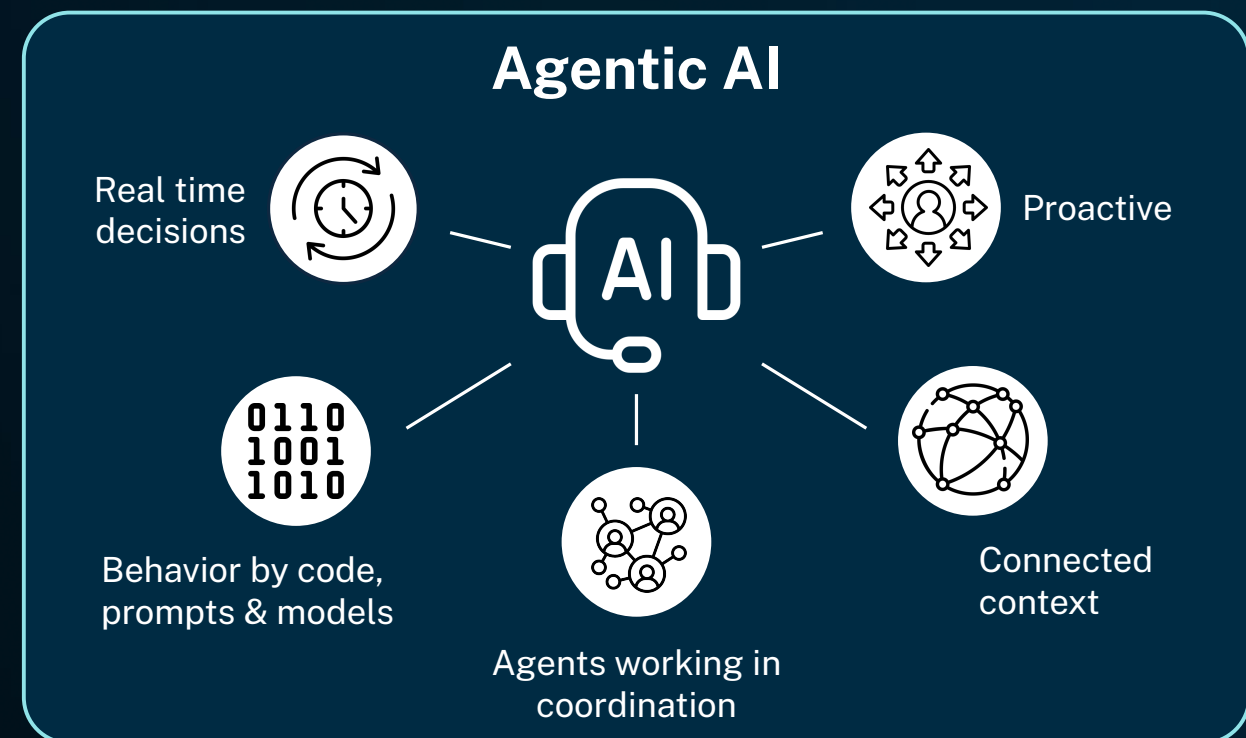
Dynamic, context-aware & autonomous applications

● ● **CRUD** (Create, Read, Update, Delete)

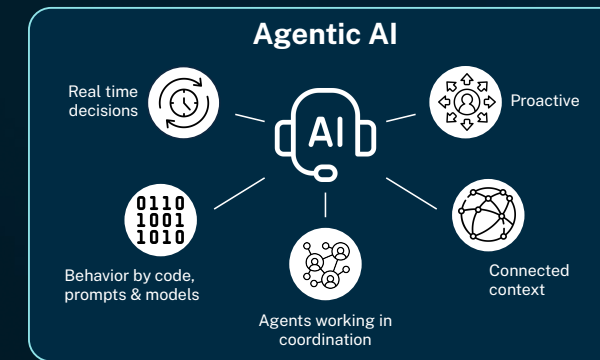
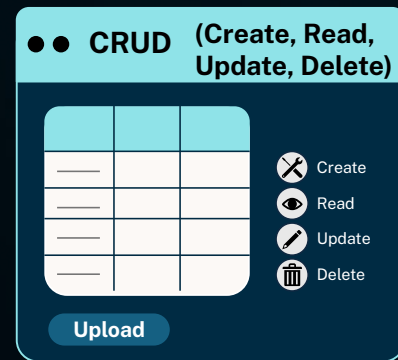
—	—	—
—	—	—
—	—	—
—	—	—

-  Create
-  Read
-  Update
-  Delete

**Upload**



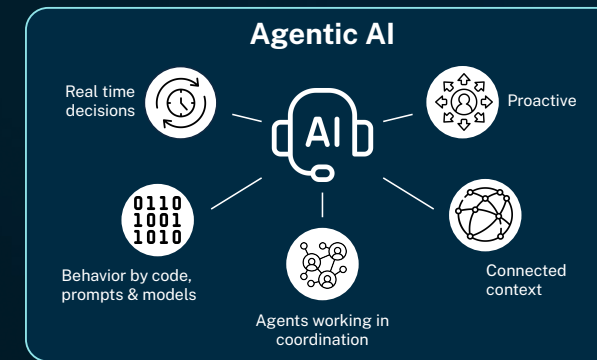
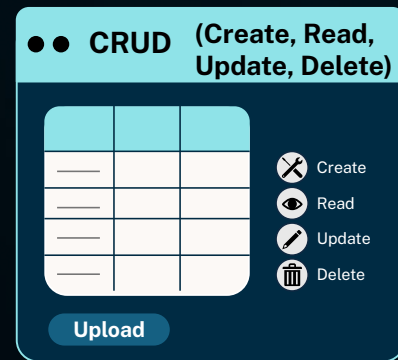
# ① From Static Logic to Dynamic Reasoning



Hardcoded  
rules defined at  
compile-time

Reason, adapt & make  
decisions in real time  
based on goals, context  
& changing inputs

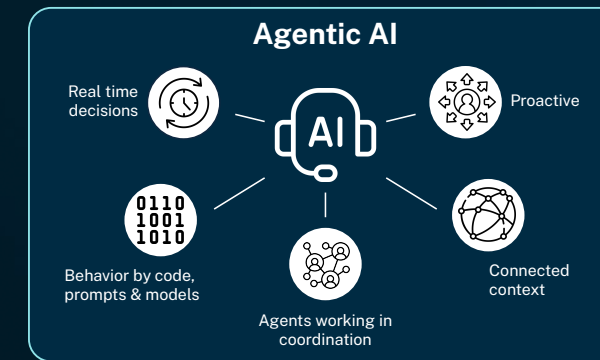
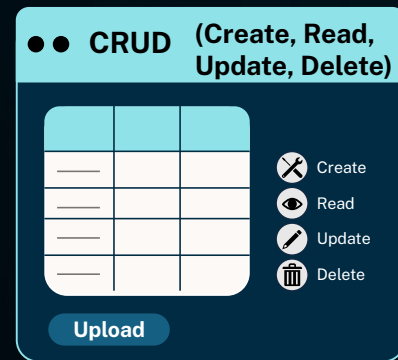
# ② From Request-Response to Goal-Oriented



React to  
user input

Proactively pursue goals,  
orchestrate tasks, and  
learn from feedback

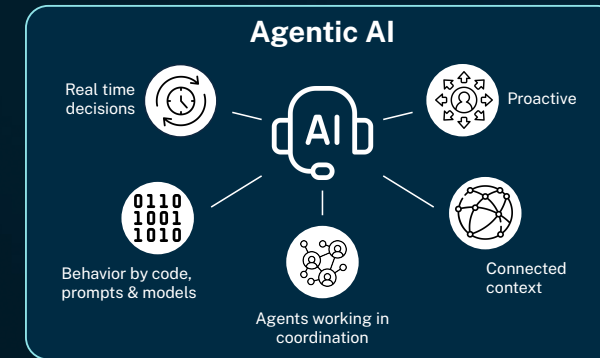
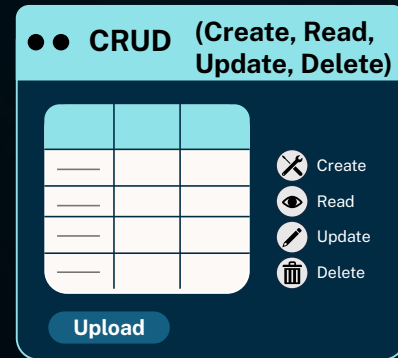
# ③ From Monoliths to Composable Systems



Tightly coupled architecture

Multiple specialized agents working in coordination

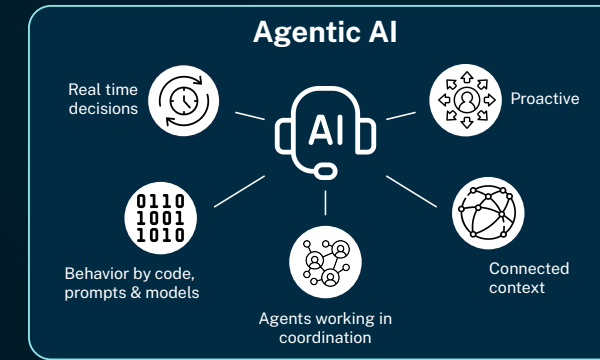
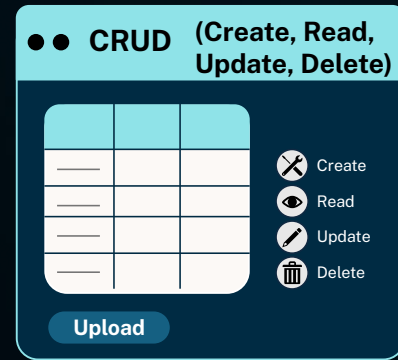
# ④ From Code-Centric to Prompt-Driven



Behavior defined  
exclusively  
by code

Behavior defined by a  
combination of code,  
prompts and AI models

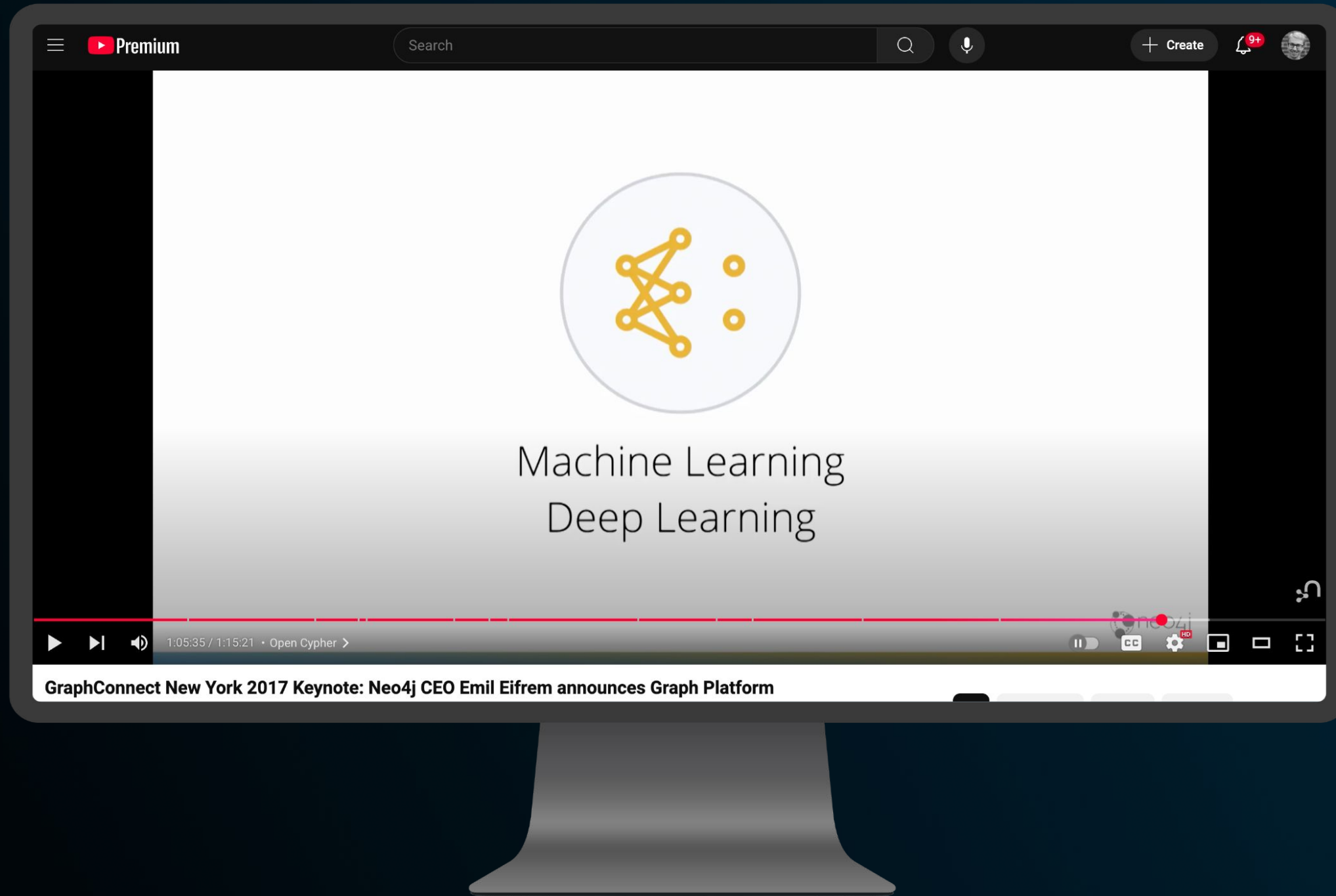
# 5 From CRUD to Context



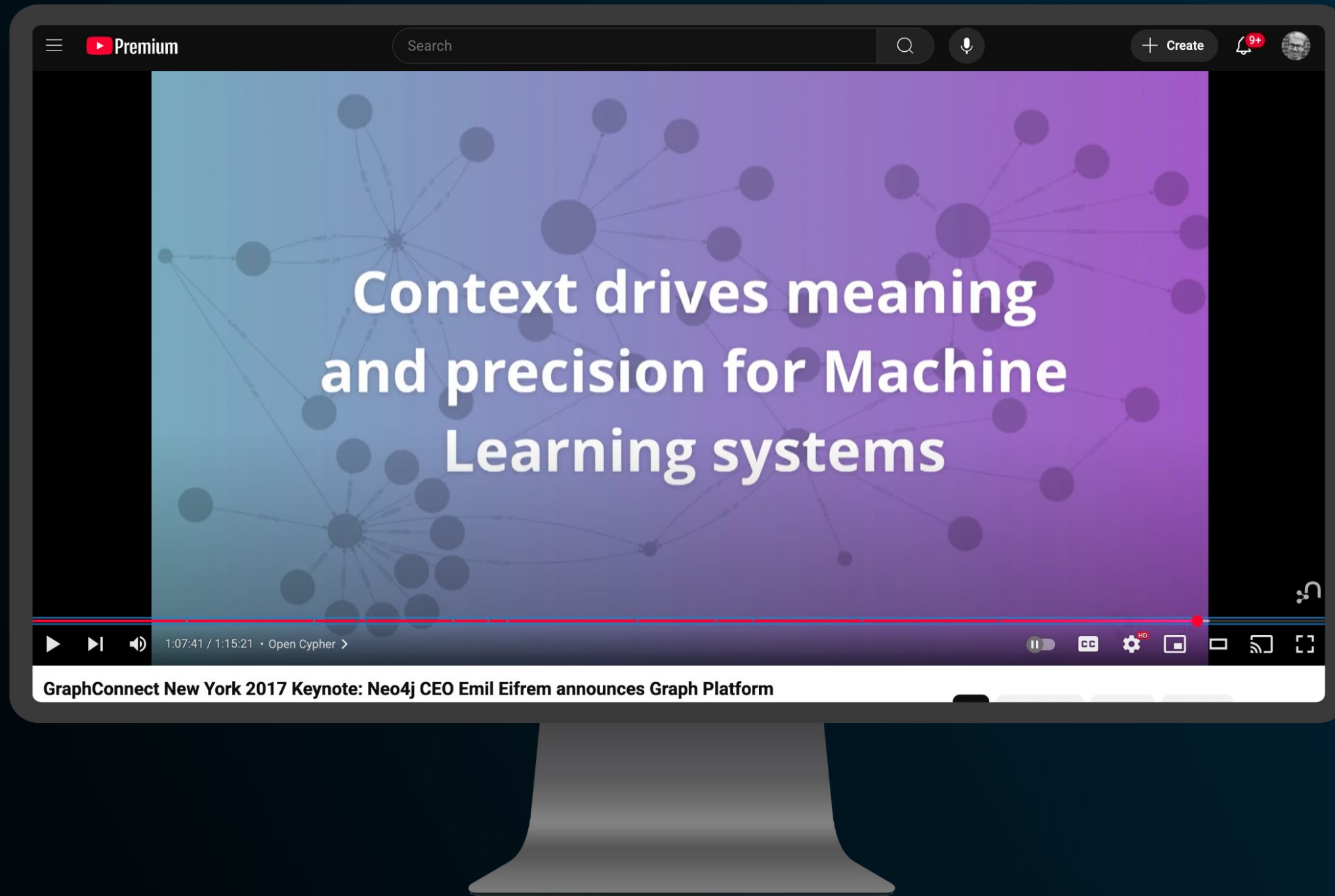
Logic centered  
around  
Create Read  
Update Delete

Logic set by reasoning,  
goals & rich, connected  
context (users, tasks,  
history, environment)

# From GraphConnect NYC (2017)



# From GraphConnect NYC (2017)



# From GraphConnect NYC (2017)

The diagram illustrates a central node (a person's profile) connected to several other nodes via labeled relationships:

- MARRIED\_TO**: A wedding couple icon.
- PLAYS**: A concert hall icon.
- WATCHES**: A television icon.
- WORKS\_AT**: A globe icon.
- BORN\_IN**: A map of Argentina icon.
- FATHER\_OF**: A family icon with a stroller.
- DRIVE**: A car icon.

The video player interface shows the video is at 1:08:48 / 1:15:21. The title is "GraphConnect New York 2017 Keynote: Neo4j CEO Emil Eifrem announces Graph Platform".

# Software engineering is changing

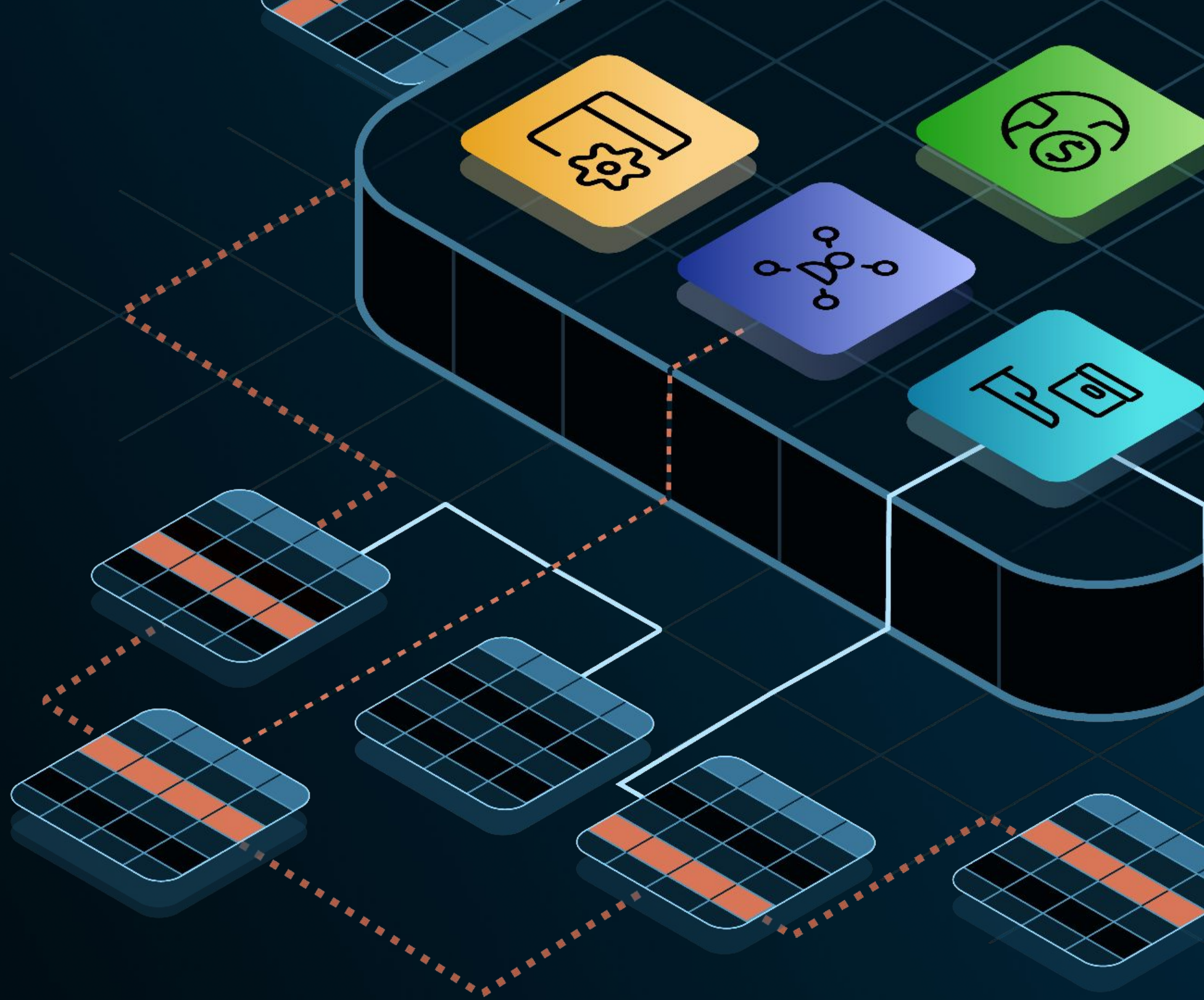
- 1 From Static Logic to Dynamic Reasoning
- 2 From Request-Response to Goal-Oriented
- 3 From Monoliths to Composable Systems
- 4 From Code-Centric to Prompt-Driven
- 5 From CRUD to Context

# Apps will change. So must the data layer.

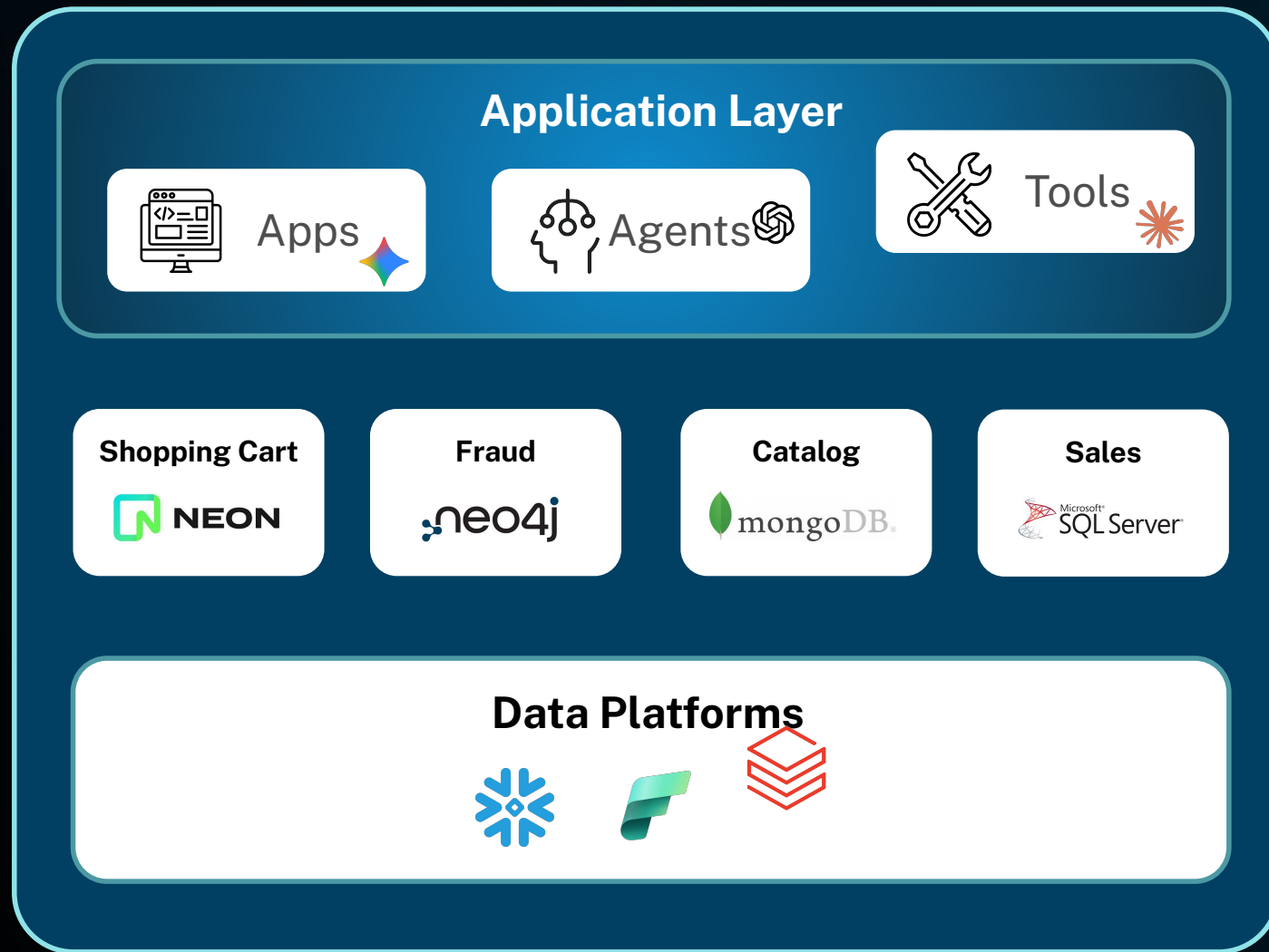
Most data is still  
stored in rows &  
columns

Structured for fixed  
questions and static data

Relationships are implied,  
or even lost



# But what about my existing data platform?

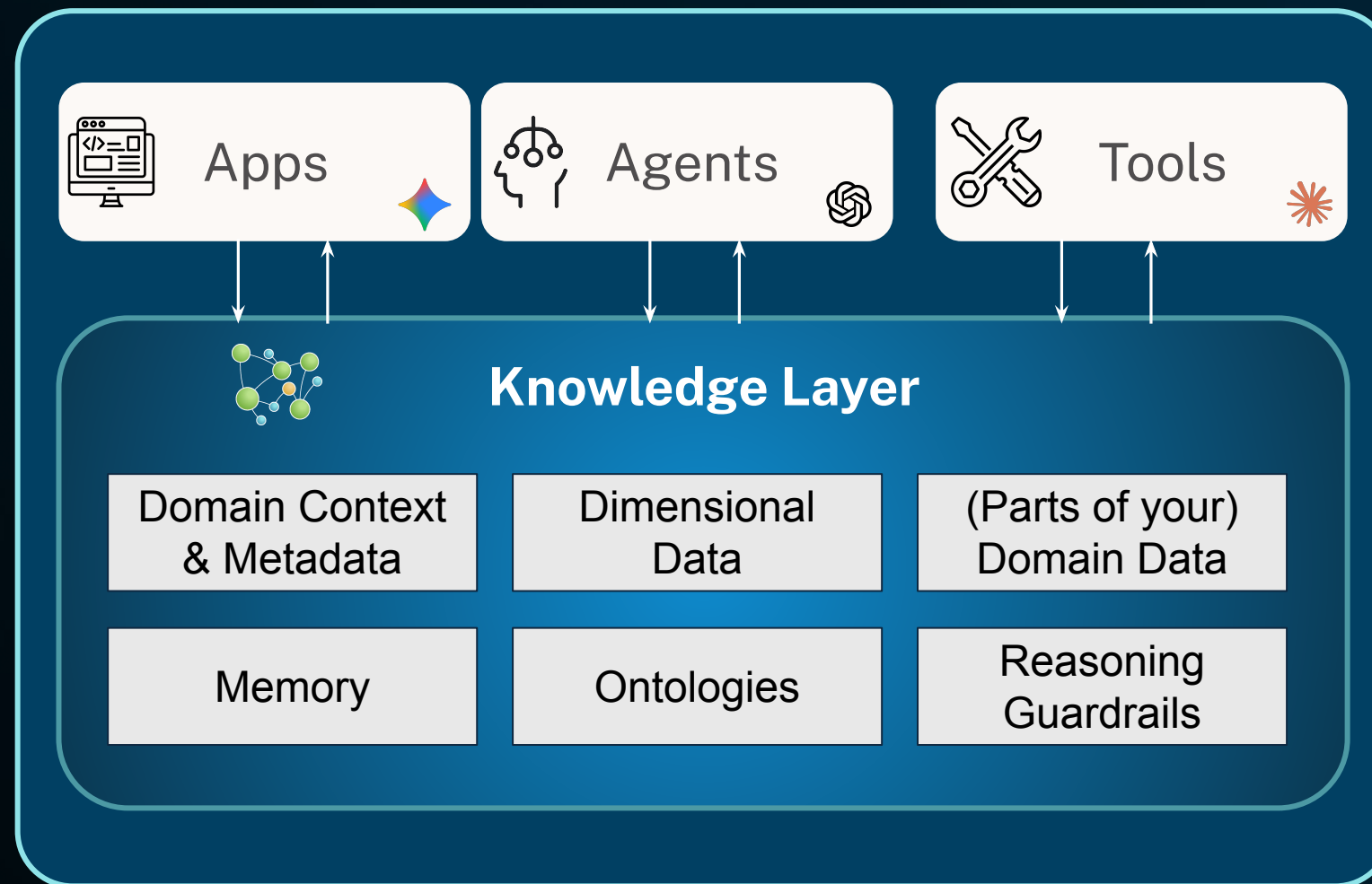


Powerful, but it doesn't support relationships and low latency queries

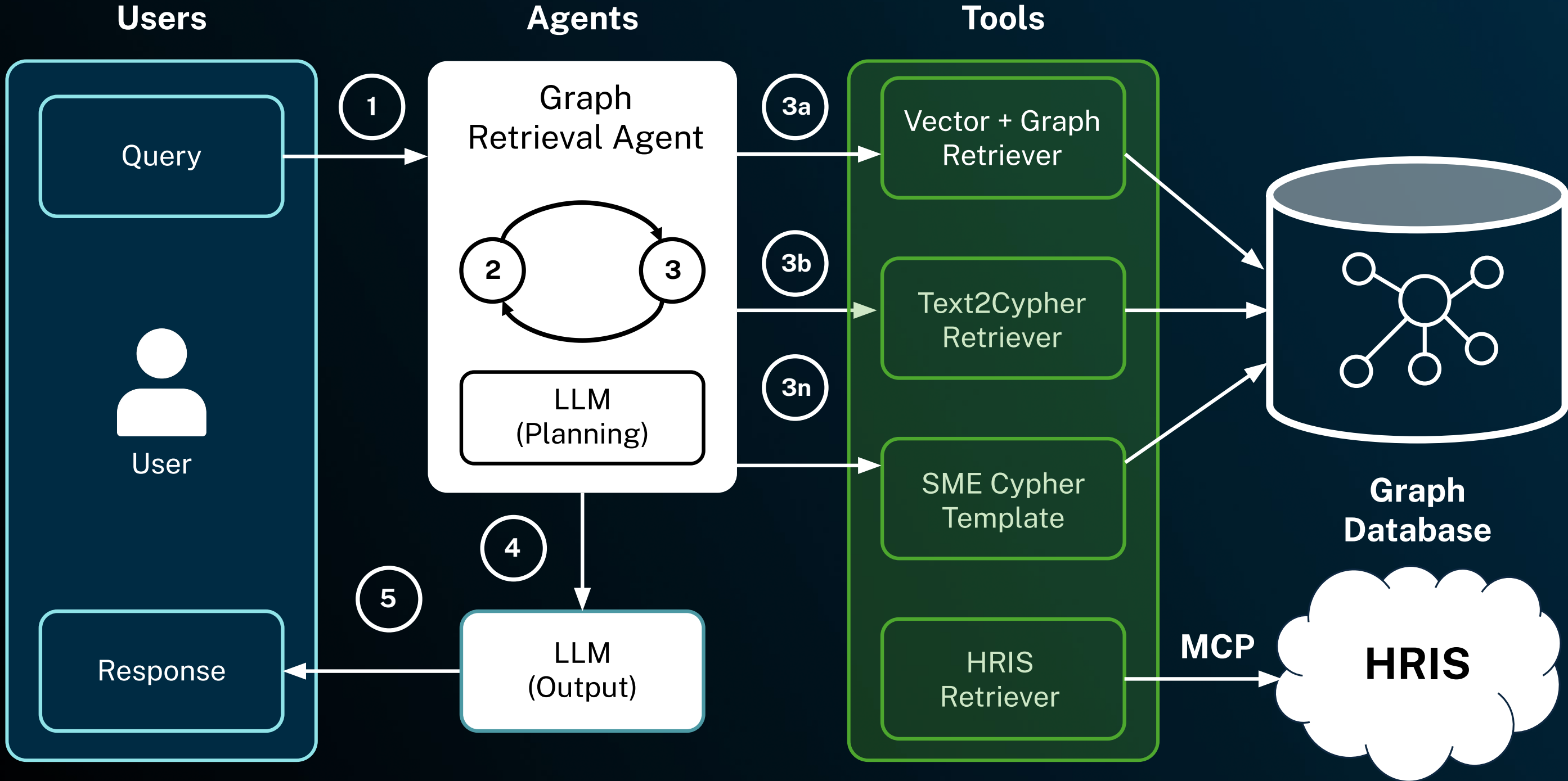
Relationships are crucial for context and speed makes that context usable in real time

# The next layer in the AI stack:

# KNOWLEDGE



# Example Agentic Architecture



# Wait, hmm, GraphRAG?

GraphRAG is RAG where the R path includes a knowledge graph.

# ① Higher Accuracy

GraphRAG delivers up to **3x higher accuracy** than traditional RAG, with better multi-hop reasoning for context-rich AI applications.

Lettria Analysis<sup>1</sup>

Writer Knowledge Graph<sup>2</sup>  
(RobustQA Benchmark)

RAG vs. GraphRAG: Multi-hop  
Question Answering<sup>3</sup>

GenUI Experiments  
(MultiHop-RAG Dataset)<sup>4</sup>

GraphRAG  
Performance

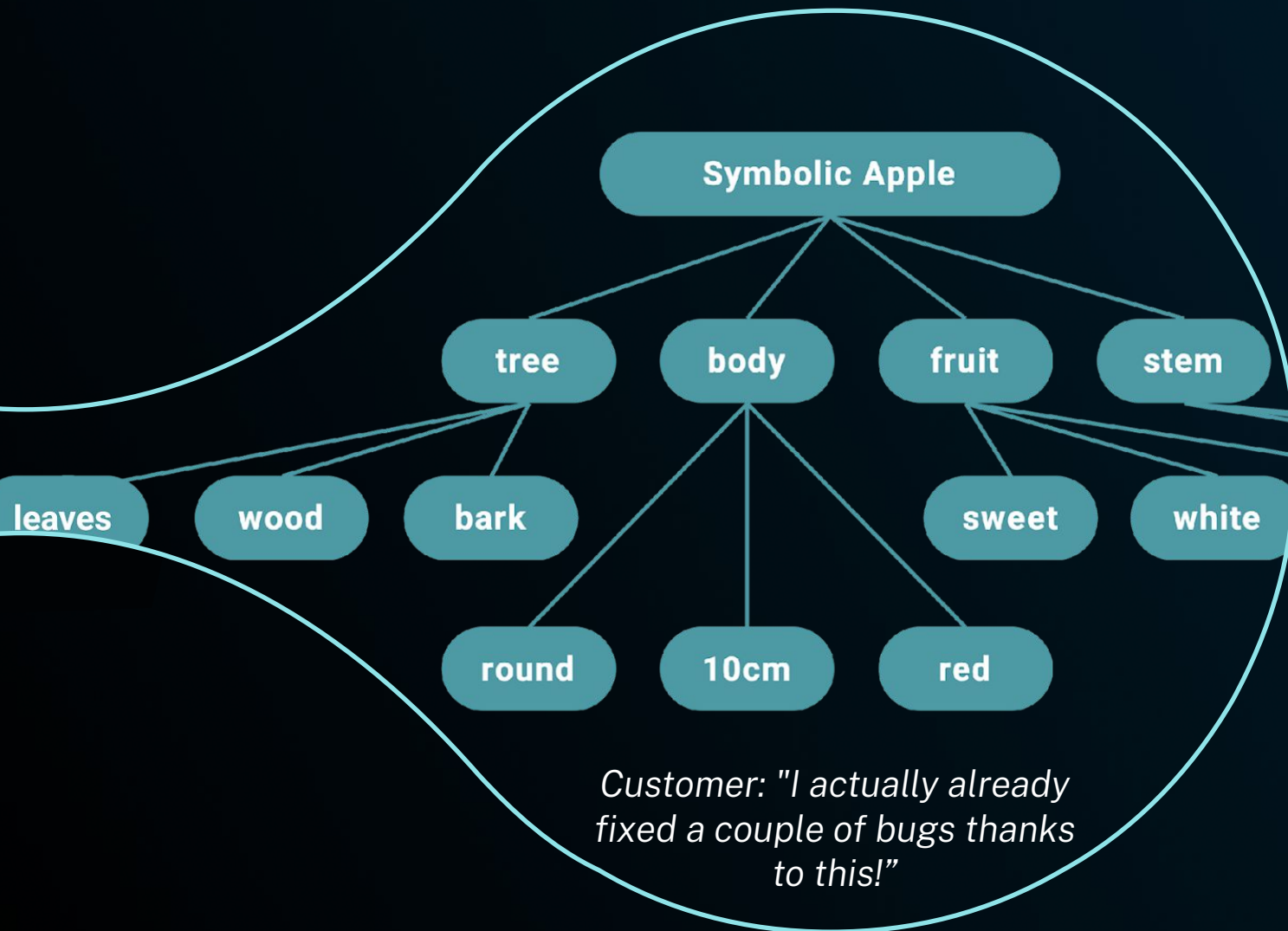
RAG  
Performance

81.67%	57.50%
86.31%	32.74%–75.89%
77%	66%
Successfully answered complex, multi-step queries	Struggled integrating data from multiple sources

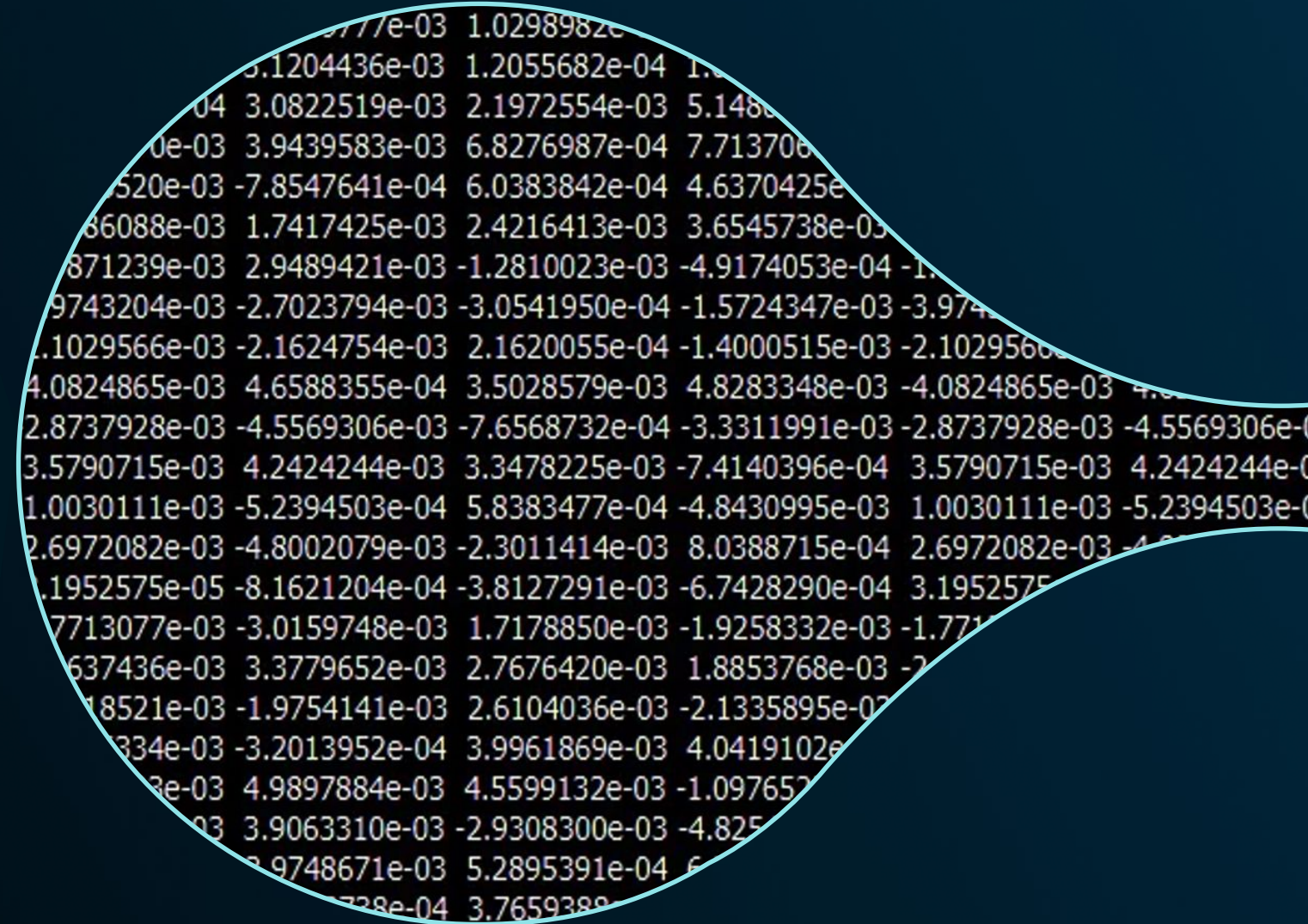
1) <https://writer.com/engineering/rag-benchmark/> 2) <https://www.lettria.com/blogpost/vectorrag-vs-graphrag-a-convincing-comparison>  
3) <https://arxiv.org/abs/2502.11371> 4) <https://www.genui.com/resources/graphrag-vs.-traditional-rag-solving-multi-hop-reasoning-in-llms>

# ② Easier Development

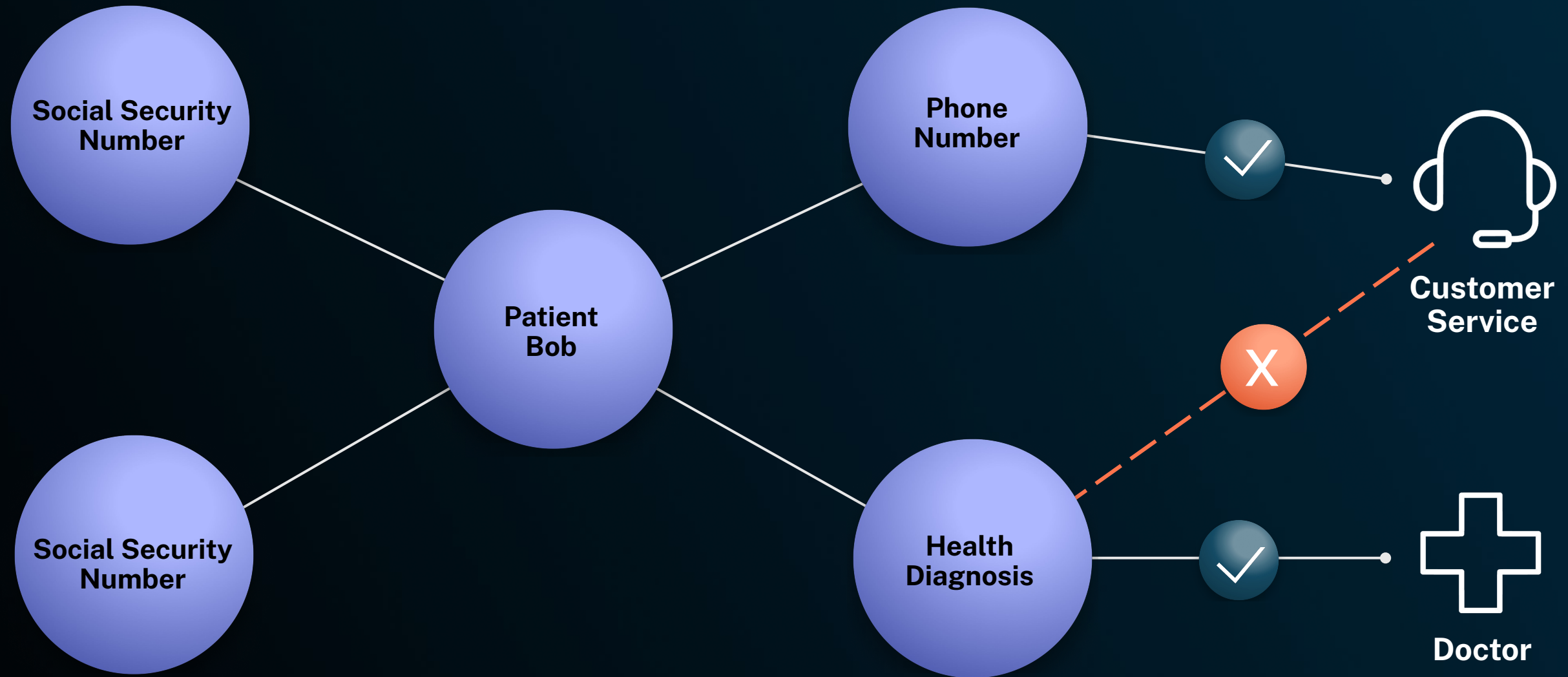
Transparent & Explainable



Opaque & Implicit



# ③ Improved Governance & Explainability



# Benefits of GraphRAG

**1**

Higher  
Accuracy

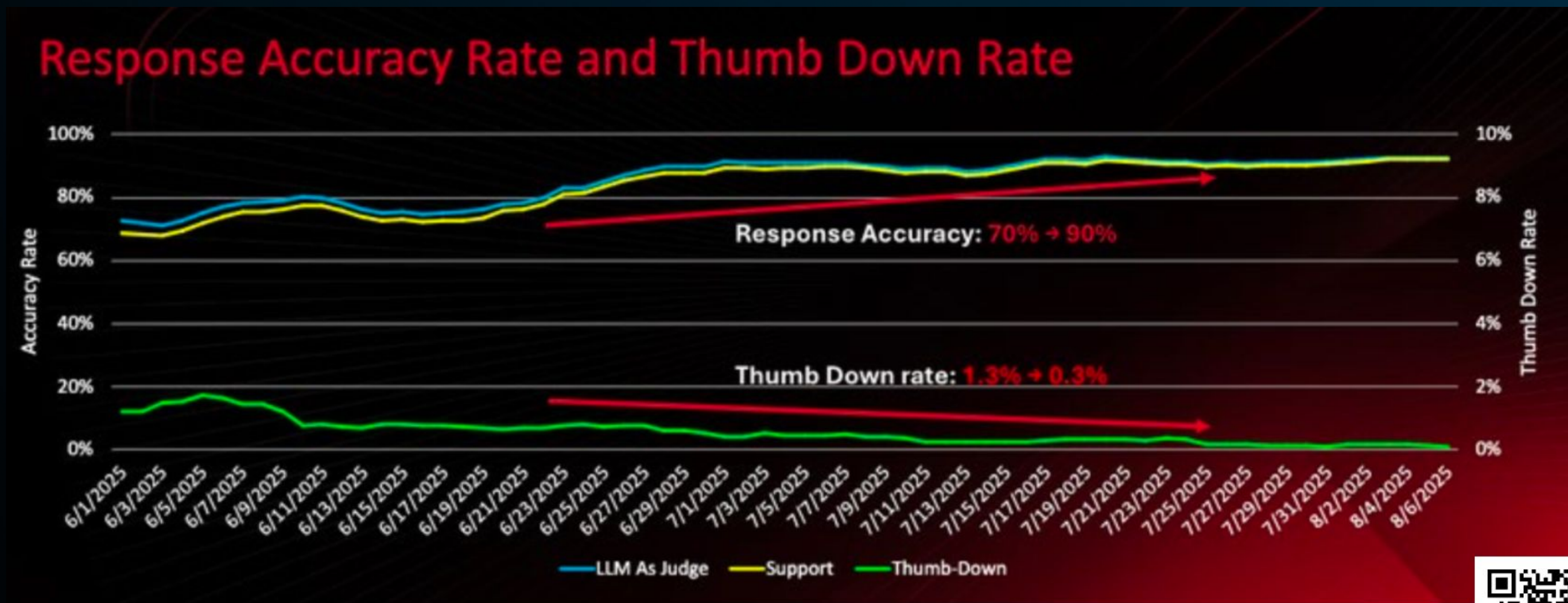
**2**

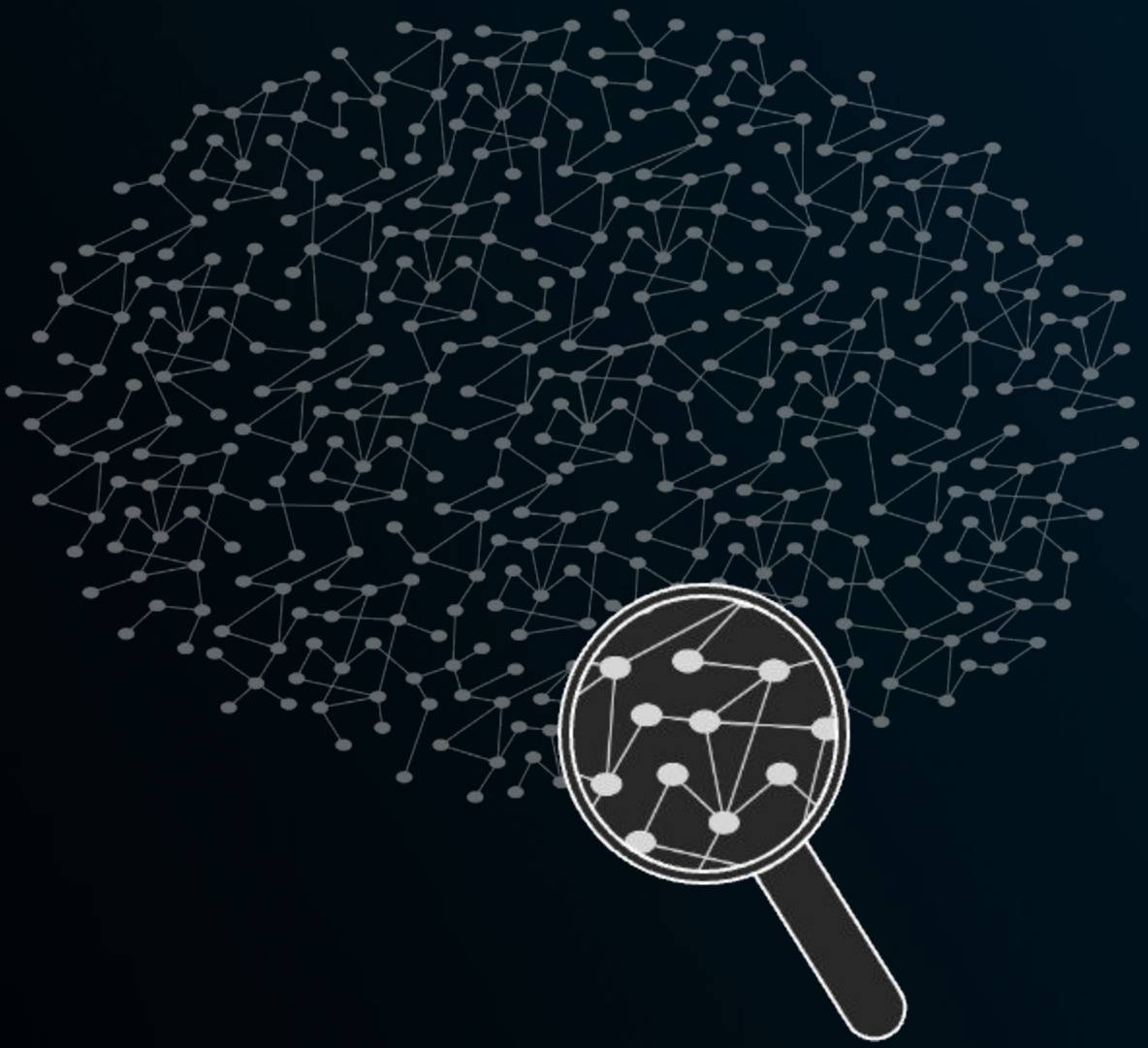
Easier  
Development

**3**

Improved  
Explainability

# Trend Micro GraphRAG improvements





+

aws





<https://github.com/aws-labs/graphrag-toolkit>

## 2.6 Query the data using graph-enhanced search

Run the cell below to use the GraphRAG Toolkit's graph-enhanced search:

```
In [3]: M %reload_ext dotenv
%dotenv

import os

from graphrag_toolkit.lexical_graph import LexicalGraphQueryEngine
from graphrag_toolkit.lexical_graph.storage import GraphStoreFactory
from graphrag_toolkit.lexical_graph.storage import VectorStoreFactory

with (
    GraphStoreFactory.for_graph_store(os.environ['GRAPH_STORE']) as graph_store,
    VectorStoreFactory.for_vector_store(os.environ['VECTOR_STORE']) as vector_store
):

    query_engine = LexicalGraphQueryEngine.for_traversal_based_search(
        graph_store,
        vector_store,
        streaming=True,
        tenant_id='ecorp',
        no_cache=True
    )

    response = query_engine.query('...')

response.print_results()
```

```
retrieve_ms: {in
answer_ms : {in
total_ms : {in
"""
```

Based on the search results regarding the...

The sales prospectus analysts have identified source-data/ecorp-toy store chains for SageMaker/graph...

Widget is predicted to be a success. The toolkit/source-data/ecorp-toy store chains must-have item until the end of the year. The Widget is absolutely essential for the Countdown to Christmas.

Example Corp has identified influencers to promote the Christmas.md. The influencers are predicted to be successful. The retailers\_prepare...

However, there are risks associated with the global distribution of the product. The global economy is currently blocked by...

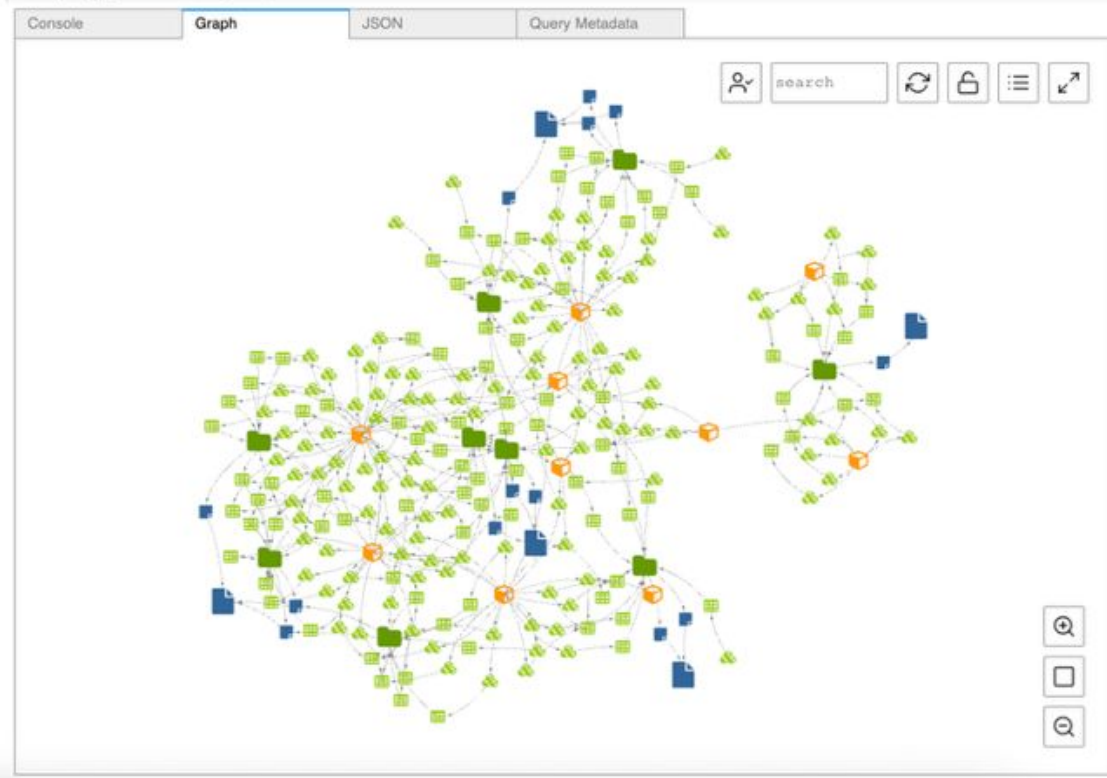
## 2.7 Visualize the results

You can view the results by running the visualisation below:

```
In [2]: M NB_CLASSIC = True

from graphrag_toolkit.lexical_graph.visualisation import GraphNotebookVisualisation

v = GraphNotebookVisualisation(nb_classic=NB_CLASSIC)
v.display_results(response)
```



# How can you add Agent Memory?



Amazon Bedrock AgentCore

## MEMORY MCP SERVERS



Constructs Knowledge Graphs from memories using Open Source MCP server

## MEMO



Deep personalized knowledge graph for individual users/agents

## COGNEE



Conversation focused memories create a lexical graph

## ZEP GRAPHITI



Temporal knowledge graph to track relationship evolution through time



STRANDS  
AGENTS



## Repo Research Agent

what is the url for the project mem0 and its most important metrics?

Based on my search, here's the information about the mem0 project:

### Project URLs:

- Main Repository: <https://github.com/mem0ai/mem0>
- Homepage: <https://mem0.ai>

### Most Important Metrics:

#### GitHub Repository Metrics:

- 🌟 Stars: 43,459 (very popular project)
- 🍴 Forks: 4,706 (high community engagement)
- 👁 Watchers: 43,459
- 🐛 Open Issues: 522 (active development)
- 📅 Created: June 20, 2023 (relatively recent but rapidly growing)
- 🔄 Last Updated: November 23, 2025 (actively maintained)

#### Project Details:

- Description: "Universal memory layer for AI Agents"
- Primary Language: Python
- License: Apache 2.0
- Repository Size: 43,657 KB

#### Key Topics/Tags:

- AI Agents, LLM, Memory Management, RAG (Retrieval-Augmented Generation)
- ChatGPT, GenAI, Long-term Memory, State Management
- Python, Hacktoberfest

The project appears to be highly successful and actively maintained, with impressive growth metrics considering it was created in mid-2023. The high star count and fork count indicate strong community interest and adoption in the AI/ML space.

Send a message



# Fireside Chat with Electronic Arts

**Bhaskar Sampathkumar**  
Principal AI Solutions Architect



Trusted by 84 of the  
**FORTUNE**  
**100**

**20 / 20**  
**Top US banks**



**10 / 10**  
**Top Automakers**



**9 / 10**  
**Top Aerospace & Defense**



**9 / 10**  
**Top Telcos**



**10 / 10**  
**Top Technology & Software**



**8 / 10**  
**Top Insurance Companies**



**8 / 10**  
**Top Retailers**



**9 / 10**  
**Top Pharmaceuticals**



The background features a complex, layered pattern of wavy, organic shapes in shades of light beige and cream. These shapes overlap and create a sense of depth and movement. In the corners, there are solid teal-colored shapes that resemble stylized water droplets or abstract organic forms, adding a pop of color to the otherwise neutral palette.

**Thank You!**

The background features a complex, layered pattern of wavy, concentric lines in shades of brown and tan, creating a sense of depth and movement. Overlaid on this are several large, organic, teal-colored shapes that resemble liquid droplets or abstract forms, positioned in the corners and along the edges.

**Relationships** are everywhere

They transform your **data** into **knowledge**

To power your **intelligent applications & systems**