When & Why You Need

Neo4j with AI

The graph database that makes AI smarter, grounded and explainable

Graph Data **Vector Search** **Grounded AI**

Turn connected data into trustworthy answers

The Big Idea

AI is brilliant at language — but it needs facts it can trust. Neo4j stores your world as it truly is: things and the relationships between them. Together, they deliver answers that are fast, grounded and explainable.



Connected by design

Relationships are stored, not computed — following them is instant.



Grounds the AI

Real facts and paths stop the model from guessing or hallucinating.



Explains itself

Every answer comes with the exact path that produced it.

AI Alone Has Blind Spots

AI Without a Knowledge Graph

- Can hallucinate confident but wrong answers
- No memory of how your data connects
- Struggles with multi-hop "how is X linked to Y?"
- Answers you cannot audit or trust
- Re-reads raw documents every single time
- Facts go stale the moment data changes

AI + Neo4j

- Answers grounded in verified facts
- Relationships are first-class and queryable
- Multi-hop questions become one traversal
- Every answer shows its exact path
- Retrieve only what matters, instantly
- Always current — the graph is the source of truth

When You Need Neo4j with AI

Your data is highly connected

Customers, products, accounts, dependencies — the value is in the links.

Questions span many hops

"Who influences whom?", "What breaks if this fails?", "How are these related?"

You need trustworthy answers

Regulated, high-stakes, or customer-facing AI that must be explainable.

RAG keeps returning noise

Flat vector search misses context; the graph adds the missing relationships.

Data changes constantly

Live systems where the AI must reflect the latest reality, not a snapshot.

Fraud, risk & recommendations

Patterns hide in relationships — exactly what graphs are built to reveal.

Graph + Vector in One Store

Neo4j combines two superpowers in a single database — so your AI finds the right things AND understands how they connect.

🔎 Vector Search

Semantic search over embeddings finds the most relevant items by meaning — "things like this" — even without exact keywords.

Perfect for: finding the starting point from a natural-language question.

Graph Traversal

From those starting points, follow real relationships to gather the full context — neighbors, dependencies, paths and patterns.

Perfect for: expanding a match into a complete, connected answer.

How GraphRAG Works

1. Ask	A user asks a question in plain language.
2. Find	Vector search finds the most relevant nodes by meaning.
3. Expand	The graph traverses relationships to gather connected context.
4. Ground	That real, connected context is handed to the AI.
5. Answer	The AI replies — accurate, specific and on-topic.
6. Explain	The exact path is returned, so the answer is auditable.

Why Teams Love It

Accurate answers

Grounded in facts, not probabilities — dramatically fewer hallucinations.

Blazing fast

Relationships are stored, so deep queries stay fast no matter the data size.

Fully explainable

Show the path behind every answer — audit, trust and compliance built in.

One source of truth

Graph + vectors + facts live together, always current, always consistent.

Faster to build

Intuitive model mirrors the real world; Cypher queries read like sentences.

Scales with you

From a laptop container to billions of relationships in production.

Where It Shines

Use Case	What the Graph + AI Delivers
Enterprise knowledge assistant	Answers grounded in your documents, systems and how they relate
Fraud & risk detection	Spot suspicious patterns hidden across chains of relationships
Recommendations	Personalized suggestions from real connections, not just similarity
Impact & dependency analysis	"What breaks if this changes?" answered in a single traversal
Customer 360	Unify every touchpoint into one connected, queryable view
Legacy & code understanding	Map large systems as a graph so AI reasons in your domain terms

Queries That Read Like Questions

Cypher lets you ask the graph in a way that looks like what you mean — so humans and AI can both write and read it.

```
MATCH (c:Customer)-[:BOUGHT]->(p:Product)<-[:BOUGHT]-(other)
RETURN other.name, count(*) ORDER BY count(*) DESC
```

"Customers who bought what this customer bought" — a recommendation in one readable line.

Natural-language → AI writes Cypher → runs on Neo4j → returns the answer with its path. No guesswork at query time.

The Payoff

↓

Hallucinations

trustworthy AI

answers grounded in real facts

Faster

deep queries

relationships stored, not joined

100%

explainable

every answer shows its path

Better answers, built faster, that your business and regulators can trust.

Easy to Get Started



Pick a question

Choose one connected-data problem AI struggles with today.



Model the graph

Map your entities and relationships — it mirrors the real world.



Add vectors + AI

Embed your data, wire up GraphRAG, ask in plain language.



Scale with trust

Roll out explainable AI across teams and use cases.

Start small — a single Docker container runs Neo4j locally in minutes.

AI Gets Smarter

When It Knows How Things Connect

Neo4j + **AI** = answers you can trust

Grounded. Fast. Explainable. Let's build it together.