VeritasGraph: Sovereign GraphRAG for the Enterprise

Welcome to the future of Retrieval-Augmented Generation. We are moving beyond simple chatbots to build production-grade, verifiable, and sovereign AI systems. Introducing a framework designed to solve the "black box" problem in modern AI.

The Enterprise AI "Gold Rush" & The Trust Barrier

Organizations everywhere are rushing to deploy RAG solutions. However, serious enterprise deployment often hits a critical wall: **Trust**. Without transparency, even the most powerful model is a liability.

Lack of Reasoning

Standard models struggle to connect disparate facts across documents.

Missing Attribution

Users cannot audit where the answer came from or verify its accuracy.

Data Sovereignty

Reliance on external APIs exposes sensitive corporate IP to third parties.

The Limitation of "Baseline" Vector Search

Standard RAG is excellent at finding a needle in a haystack, provided the answer exists explicitly in one paragraph. It fails when the answer requires **Multi-Hop Reasoning** across different documents.

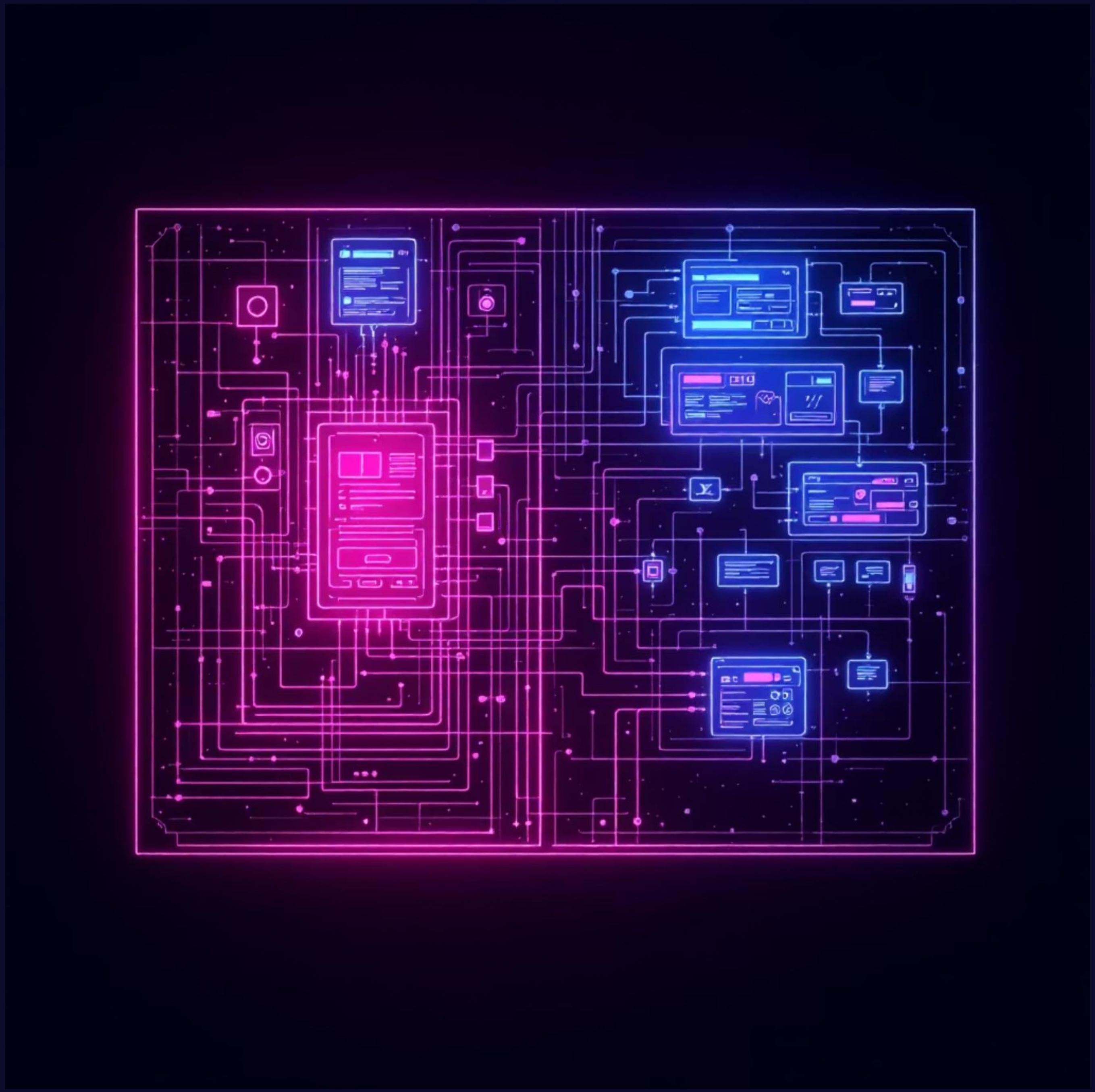
The Vector Blindspot

If a user asks how "Project A" relates to a "Patent in Project B," a vector database sees two unrelated documents. It cannot intuitively "connect the dots" between separated entities.

The Hallucination Risk

When context is missing, LLMs fill the gaps with hallucinations. In an enterprise setting, a plausible but false answer is more dangerous than no answer at all.

A New Architecture: The Sovereign Knowledge Asset



VeritasGraph isn't just about retrieving text; it's about building a **Sovereign Knowledge Asset** that lives entirely within your infrastructure. We provide transparent, auditable reasoning paths for every claim.

Private Infrastructure

Operates 100% on-premise or in your private cloud.

Graph-Based

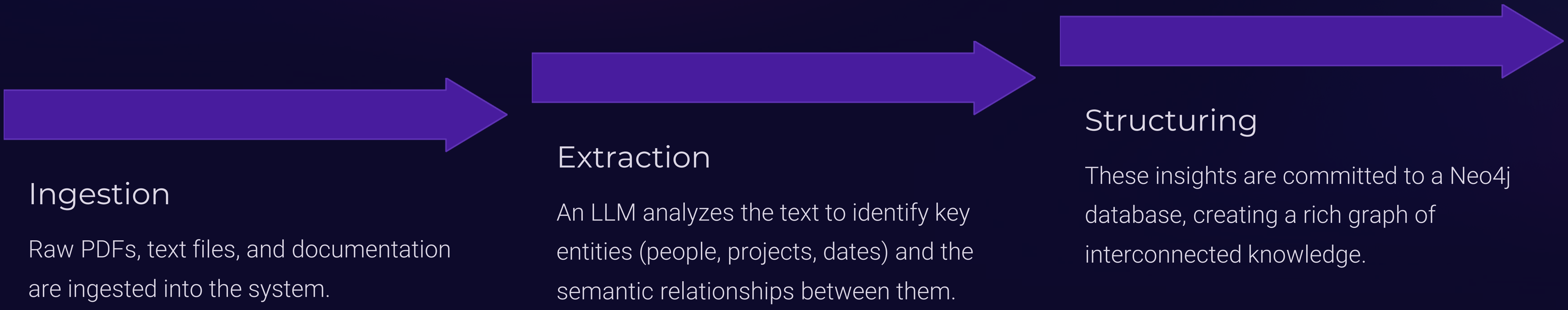
Leverages Neo4j to map relationships, not just similarity.

Auditable

Every output traces back to a specific source document.

Stage 1: Knowledge Graph Construction

We move beyond simple text chunking. Our pipeline transforms unstructured data into a structured intelligence layer using advanced entity extraction.



Ingestion

Raw PDFs, text files, and documentation are ingested into the system.

Extraction

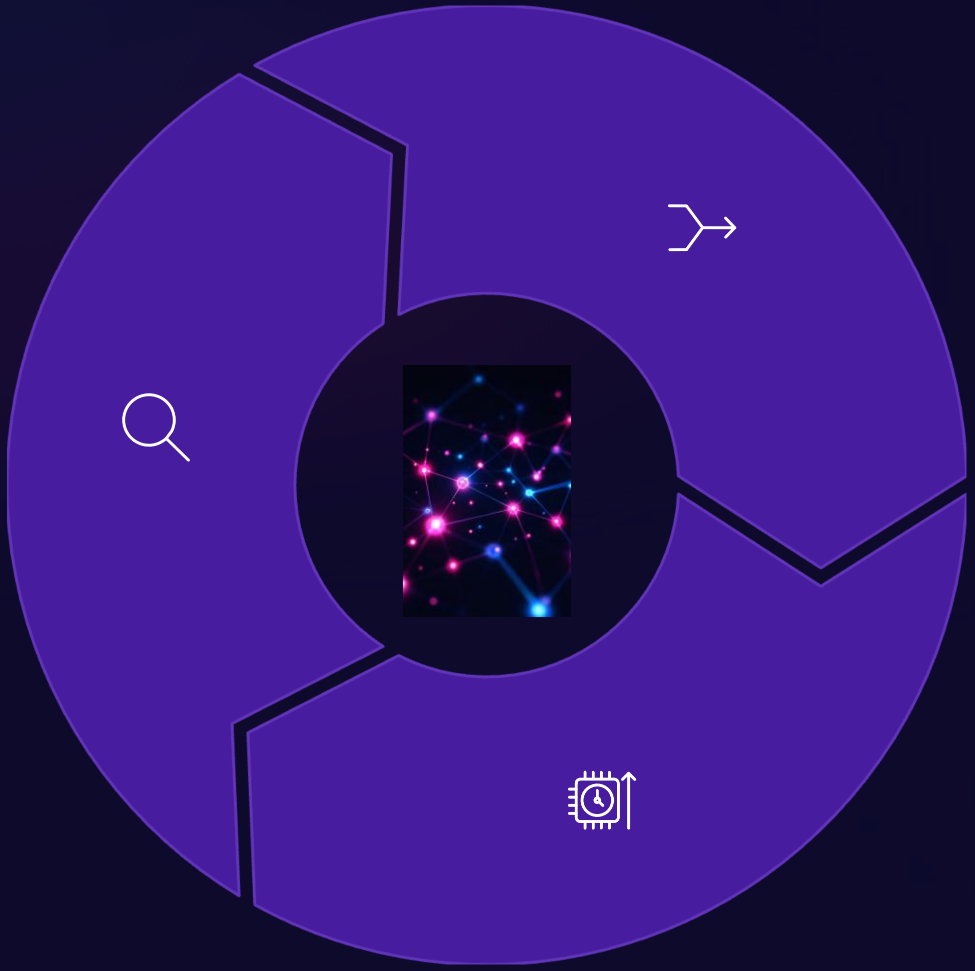
An LLM analyzes the text to identify key entities (people, projects, dates) and the semantic relationships between them.

Structuring

These insights are committed to a Neo4j database, creating a rich graph of interconnected knowledge.

Stage 2: The Hybrid Retrieval Engine

To answer complex, holistic questions, VeritasGraph employs a dual-approach strategy that captures both semantic similarity and structural relationships.



Vector Entry

We use Vector Search to identify the most relevant "entry nodes" in the graph based on the user's query.

Graph Traversal

The system "walks" the edges of the graph from these entry points, discovering hidden connections vector search would miss.

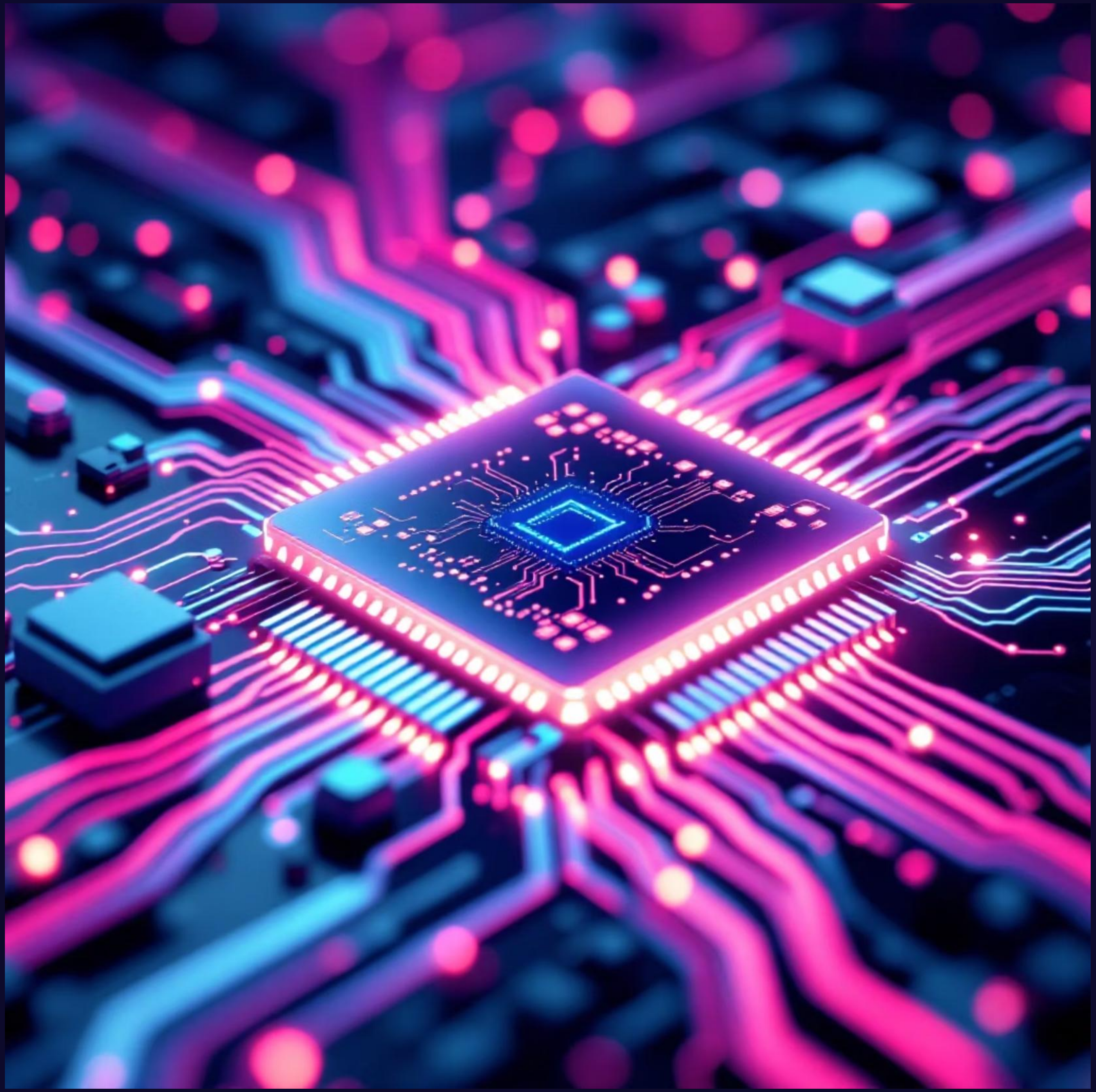
Context Assembly

Relevant nodes and edges are aggregated to form a comprehensive context window for the LLM.

Stage 3: Reasoning with LoRA-Tuned Models

Generic models are often too broad. VeritasGraph is optimized for use with **LoRA-Tuned LLMs** (Low-Rank Adaptation), specifically fine-tuned for logical reasoning and strict adherence to context.

- **Local Efficiency:** optimized to run on consumer hardware or standard enterprise servers.
- **Specialized Tuning:** Models like Llama 3 are tuned specifically for citation and graph interpretation.
- **Reduced Latency:** Faster inference times compared to massive general-purpose models.



Stage 4: Verifiable Attribution

This is the cornerstone of the framework. We eliminate the "trust me, bro" aspect of AI. We provide a structured JSON output containing the provenance of every fact.

Traceable Claims

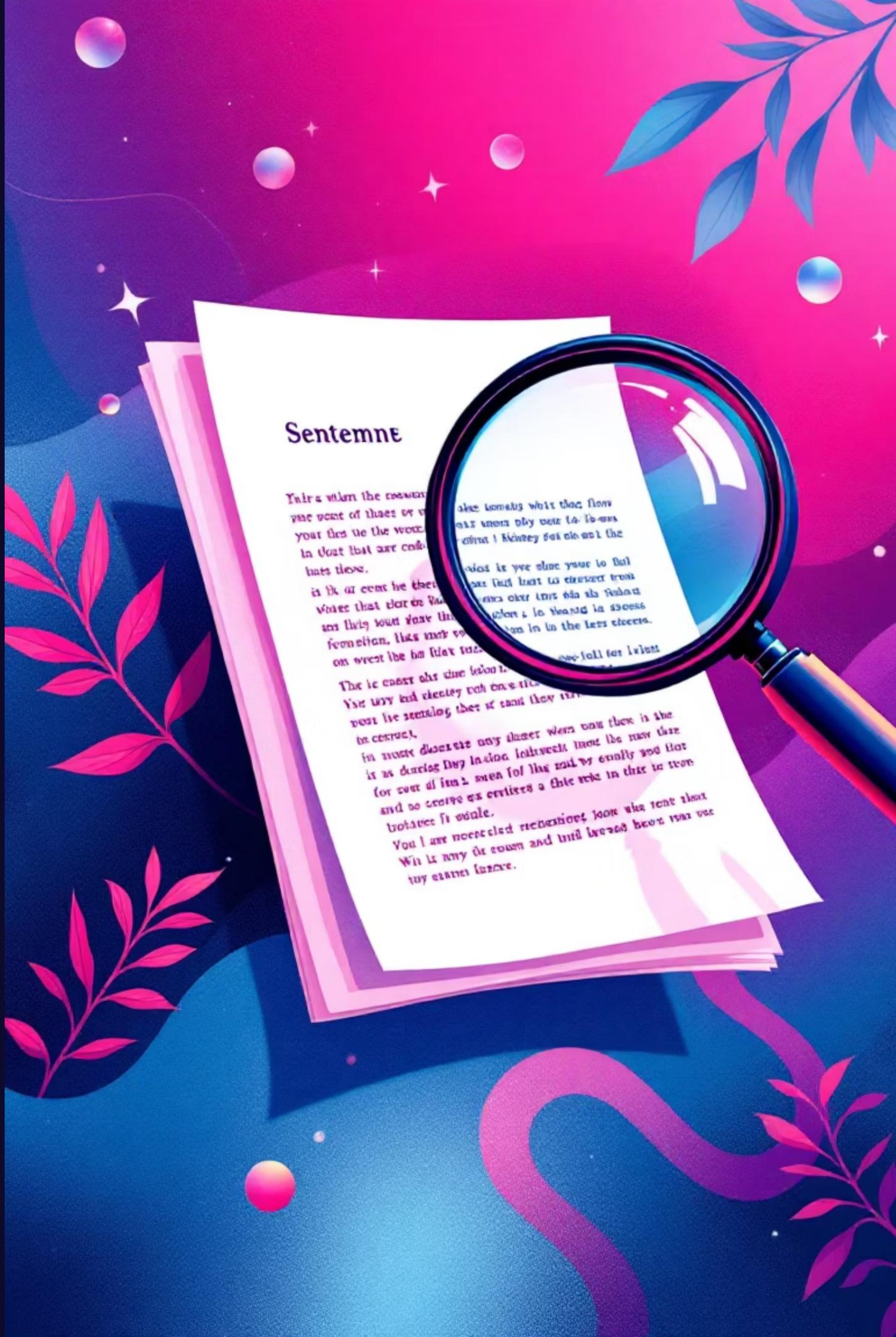
If the model states "X causes Y," it points exactly to the document and paragraph where it learned that fact.

JSON Provenance

Developers receive structured data citing source nodes, ensuring easy integration into frontend UI audit trails.

Defensible AI

In regulated industries, the ability to prove *why* an AI made a decision is mandatory, not optional.



"Five-Minute Magic" & Total Sovereignty

Theory is great, but shipping code is better. We have designed an onboarding process that takes you from zero to a running GraphRAG instance in minutes via Docker.

Gradio UI

A built-in interface for immediate interaction and testing of your graph.

Neo4j Database

The industry standard for graph databases, spun up automatically.

Ollama API

Local inference ensuring your data never leaves your secure environment.

The Promise: You own the model, you own the graph, and you own the data. No dependency on OpenAI or Anthropic.

Roadmap & Call to Action

VeritasGraph is open source and available on GitHub today. We are moving fast to build systems we can actually trust. Join us in moving beyond black-box AI.

Available Now

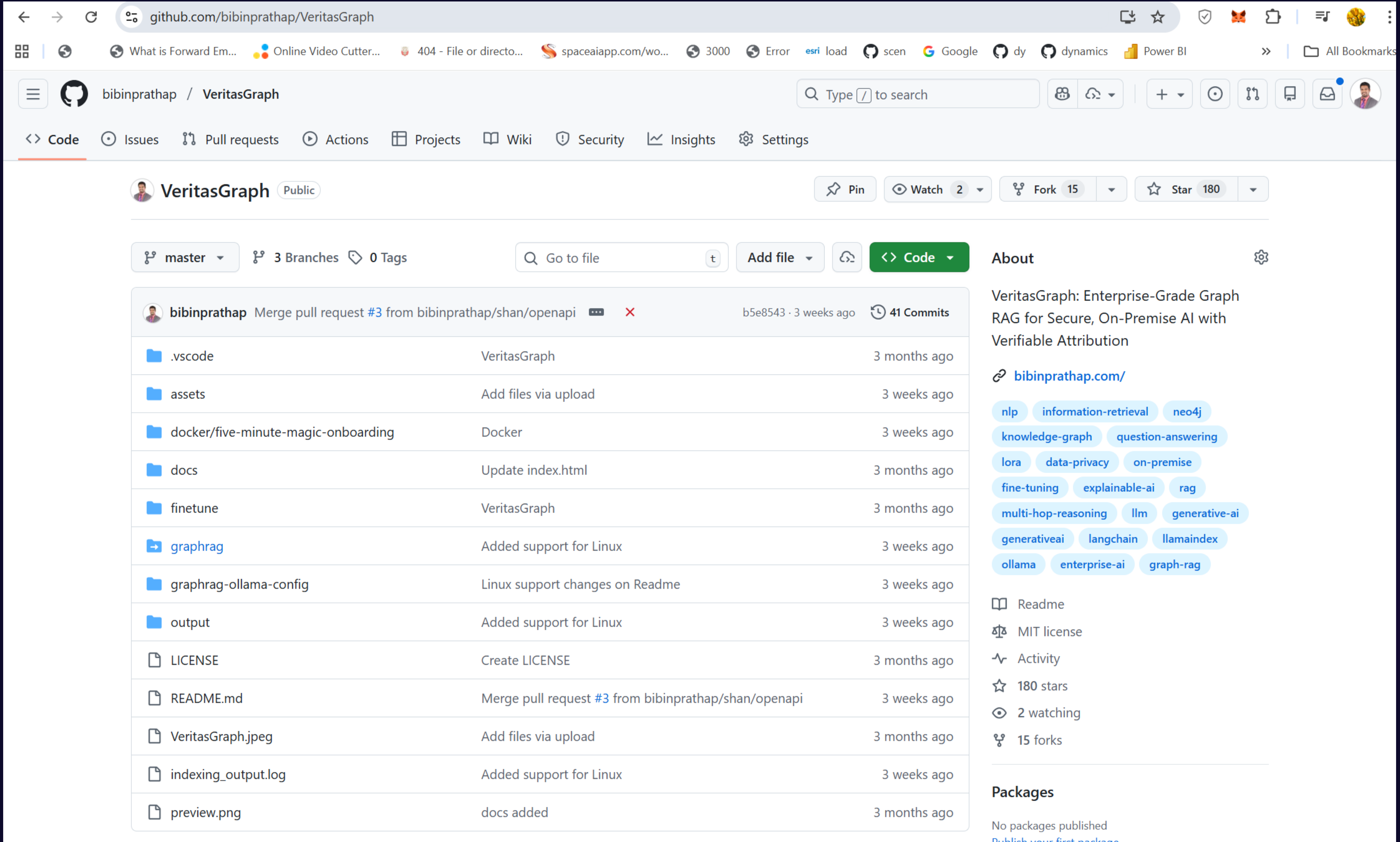
Core graph construction, hybrid retrieval, and Docker deployment.

Coming Soon

Advanced graph analytics for community detection (Microsoft GraphRAG approach).

Future Vision

Agentic Frameworks to automatically decompose complex queries into sub-tasks.



VeritasGraph Public

bibinprathap Merge pull request #3 from bibinprathap/shan/openapi ✕ b5e8543 · 3 weeks ago 41 Commits

.vscode	VeritasGraph	3 months ago
assets	Add files via upload	3 weeks ago
docker/five-minute-magic-onboarding	Docker	3 weeks ago
docs	Update index.html	3 months ago
finetune	VeritasGraph	3 months ago
graphrag	Added support for Linux	3 weeks ago
graphrag-ollama-config	Linux support changes on Readme	3 weeks ago
output	Added support for Linux	3 weeks ago
LICENSE	Create LICENSE	3 months ago
README.md	Merge pull request #3 from bibinprathap/shan/openapi	3 weeks ago
VeritasGraph.jpeg	Add files via upload	3 months ago
indexing_output.log	Added support for Linux	3 weeks ago
preview.png	docs added	3 months ago

About

VeritasGraph: Enterprise-Grade Graph RAG for Secure, On-Premise AI with Verifiable Attribution

bibinprathap.com/

nlp information-retrieval neo4j knowledge-graph question-answering lora data-privacy on-premise fine-tuning explainable-ai rag multi-hop-reasoning llm generative-ai generativeai langchain llamaindex ollama enterprise-ai graph-rag

- Readme
- MIT license
- Activity
- 180 stars
- 2 watching
- 15 forks

Packages

No packages published

Thank You

Questions?

I'd love to hear from you and discuss how we can work together.

Get in Touch

Bibin Prathap

bibinprathap@email.com

Learn More

Visit my website for portfolio and additional information

bibinprathap.com