Team 10: Smart Document Retrieval & The AI-Assisted Build

Pacera Hack Days 2026

Aleksejs, Nikolajs, Olegs, Yurii

We used AI to build a better way to use AI.

The Vision: Velocity Above All

“Use anything and everything that speeds us up.”

Strategy: Orchestrate AI agents to bypass traditional coding bottlenecks.

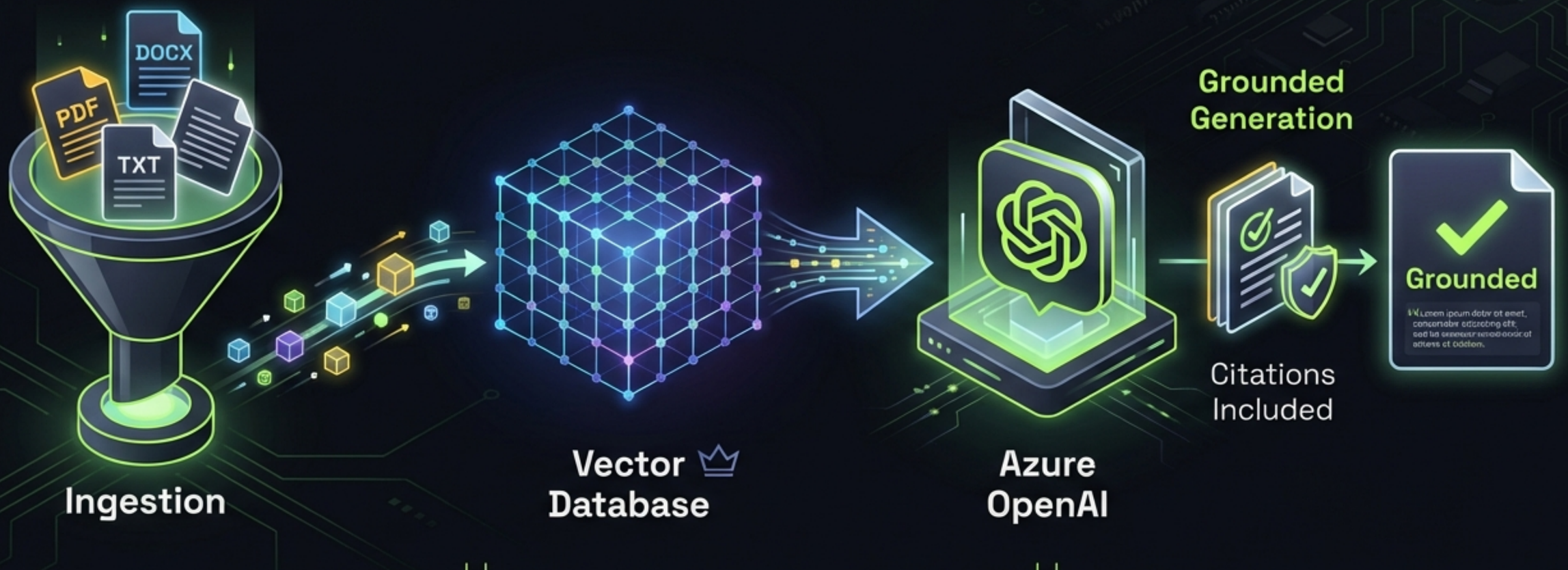
The Problem: Data Overload & LLM Amnesia

≠

The Black Box: Opaque Collections.

Context Limits & Hallucinations.

The Solution: Retrieval-Augmented Generation (RAG)

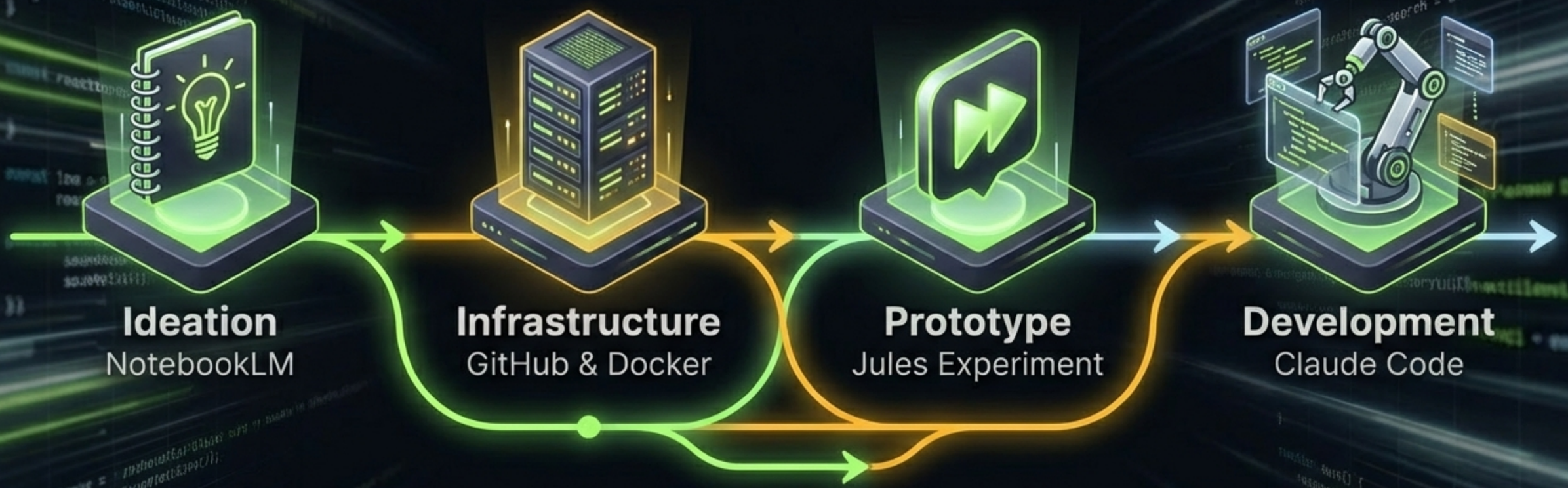


1. Index: Chunk & Embed

2. Retrieve: Semantic Search

3. Generate: Grounded Answers

The Build Strategy: AI as a Team Member

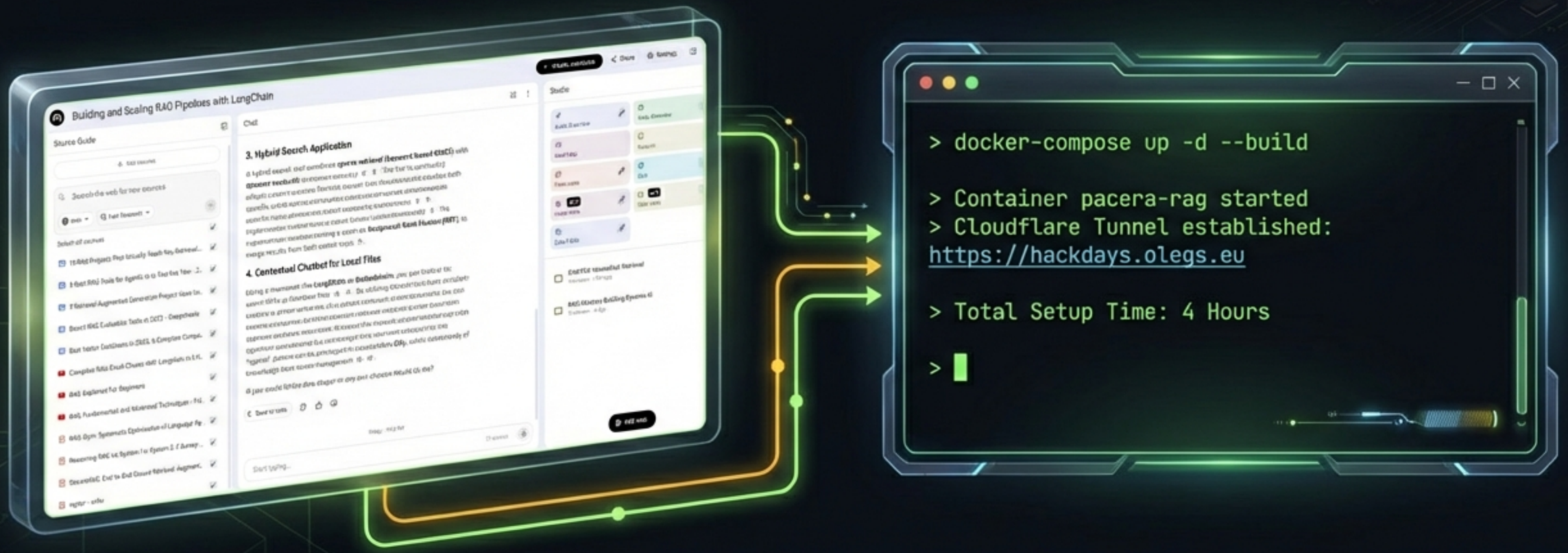


AI was used for everything—from **ideation** to deployment.

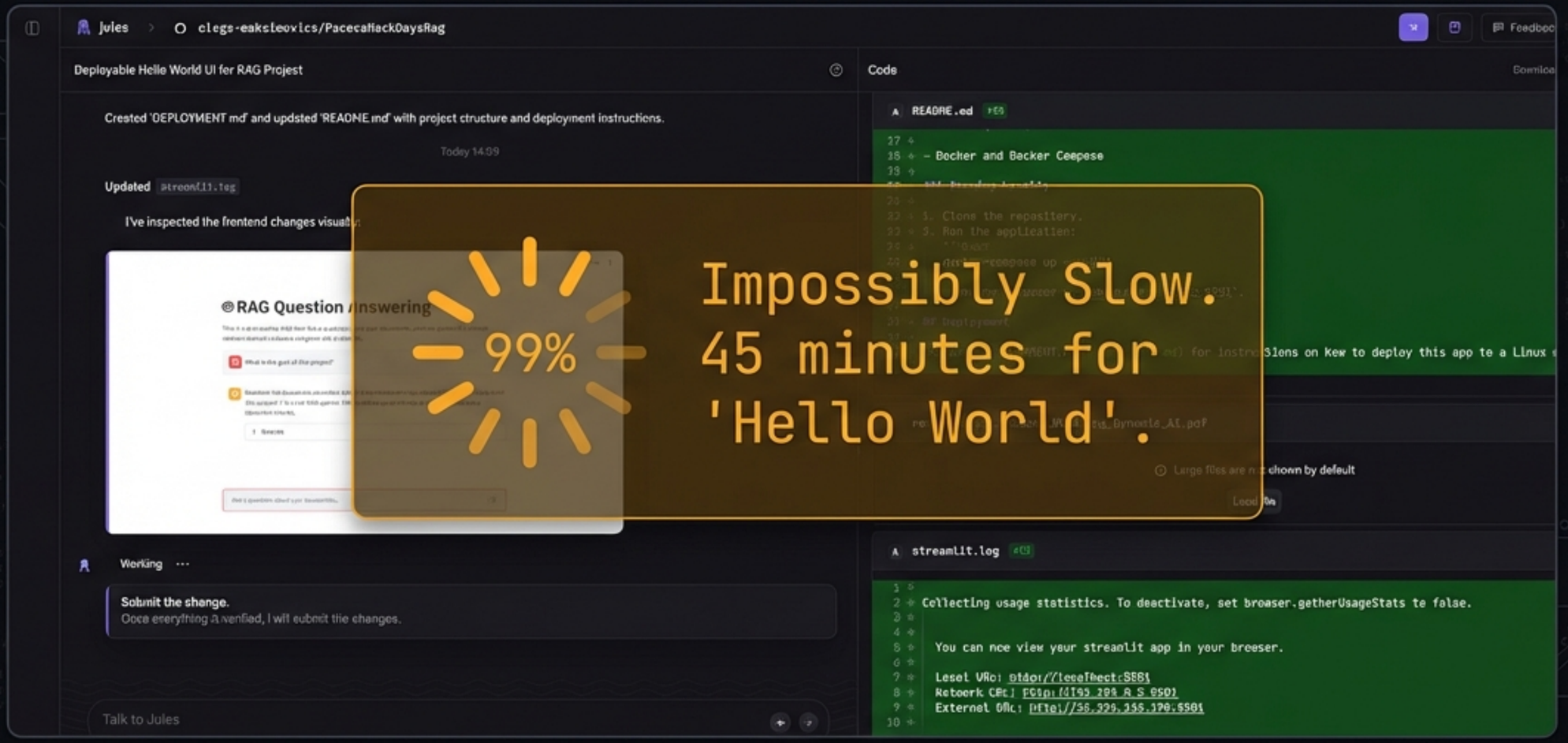
Step 1 & 2: Instant Ideation to Deployed Infra

NotebookLM:
Ideation & RAG Concepts

Deployment:
Infrastructure as Code



The Failure: The ‘Jules’ Experiment



Insight: Not all AI tools are ready for rapid prototyping.

The AI Developer Tool Showdown

Claude Code

9/10

JetBrains Mono

The MVP. Excellent planner & implementer.

Google Antigravity

8/10

JetBrains Mono

Copilot on steroids. Generous limits.

GitHub Copilot

7/10

JetBrains Mono

Standard, capable.

Google Gemini Assistant

0/10

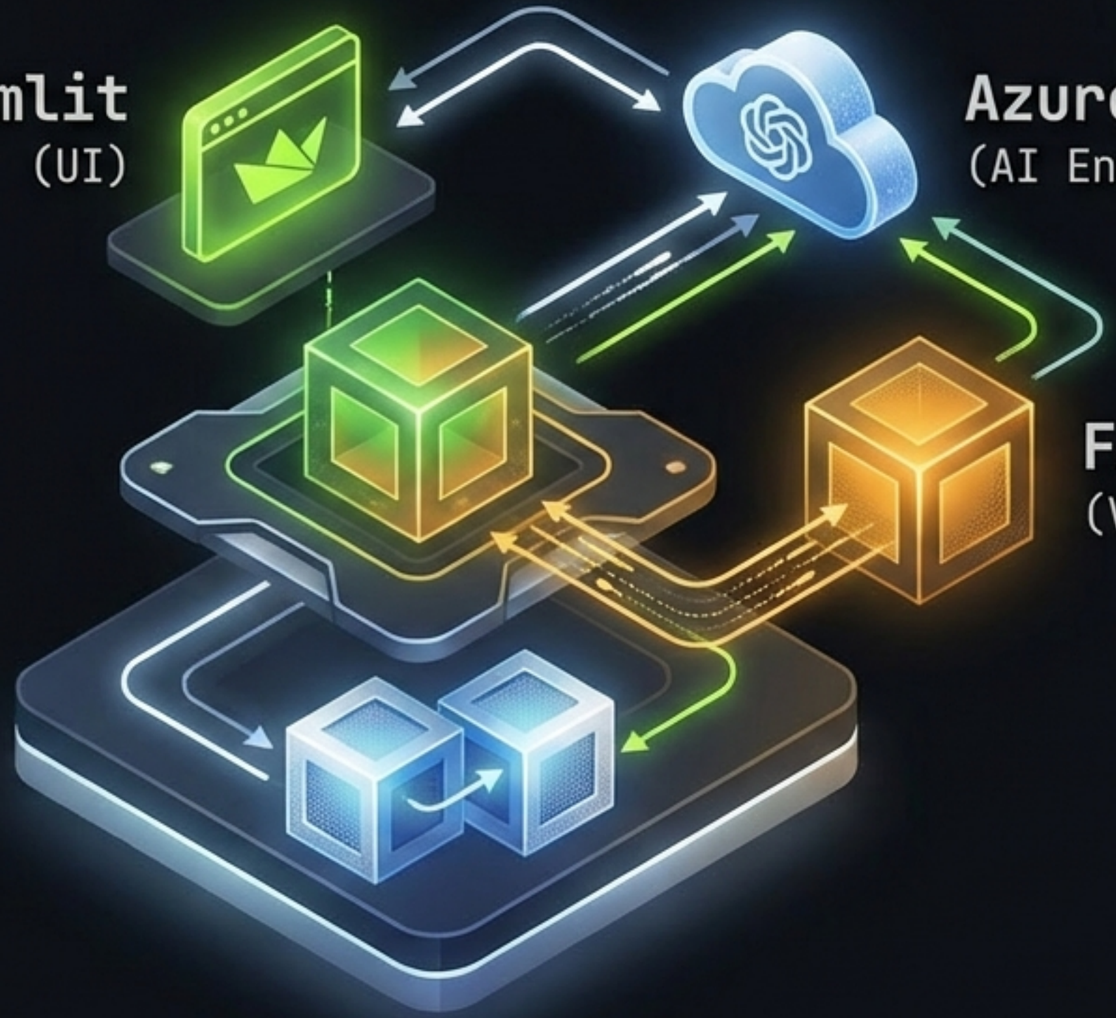
JetBrains Mono

More often doesn't work.

Under the Hood: The Tech Stack

Streamlit
(UI)

Azure OpenAI
(AI Engine)

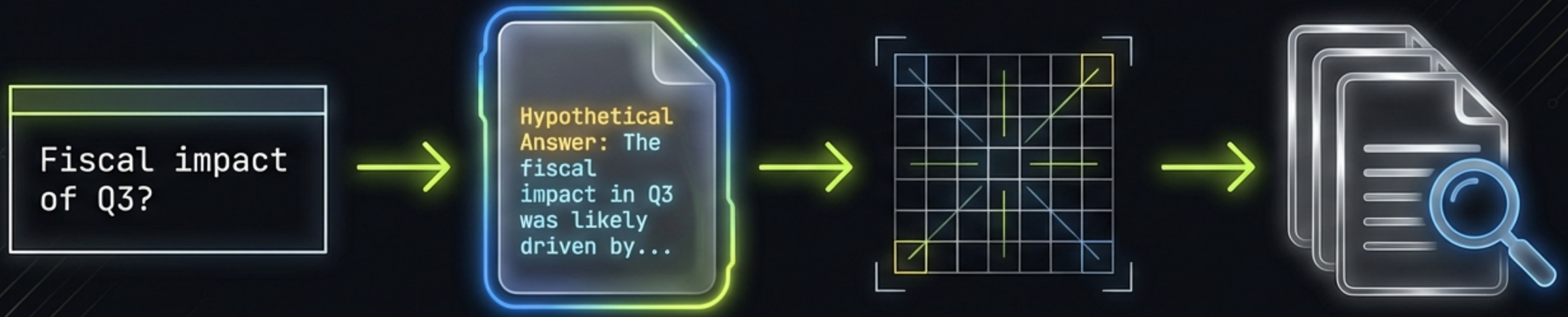


Logic
Python 3.9+ & LangChain

FAISS
(Vector DB)

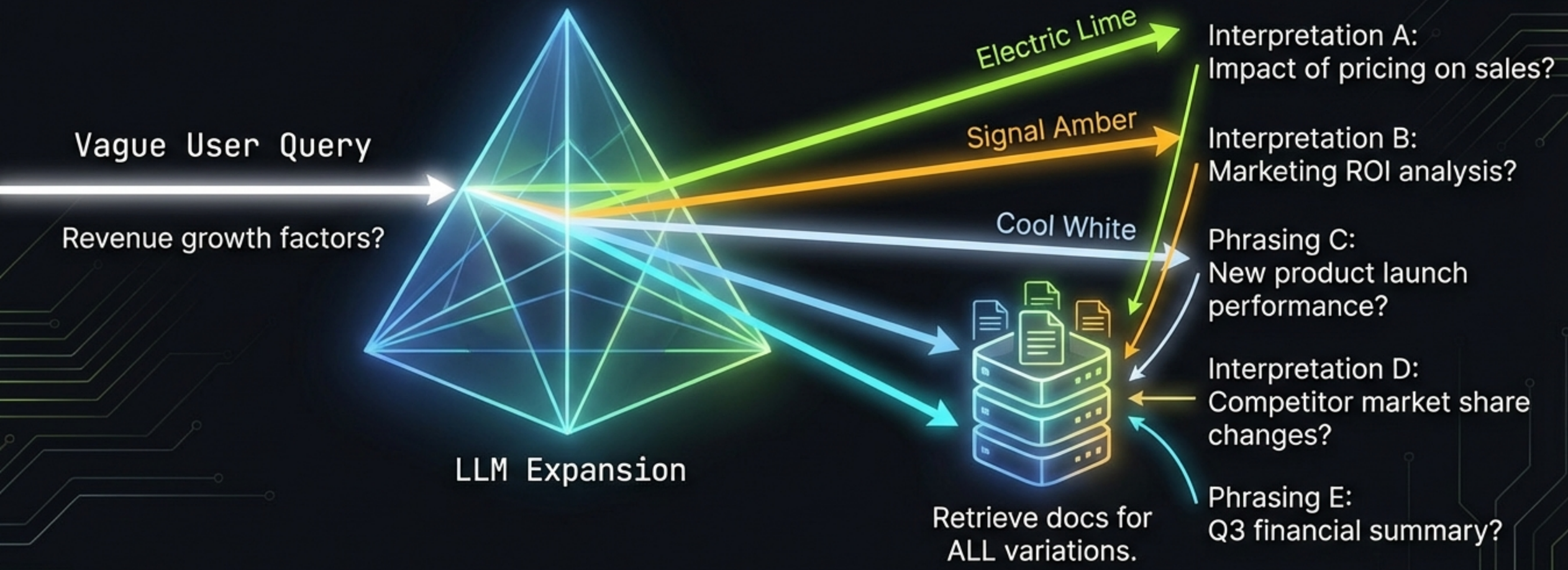
Infrastructure
Docker Compose +
Cloudflare Tunnel

Solving Vocabulary Mismatch: HyDE



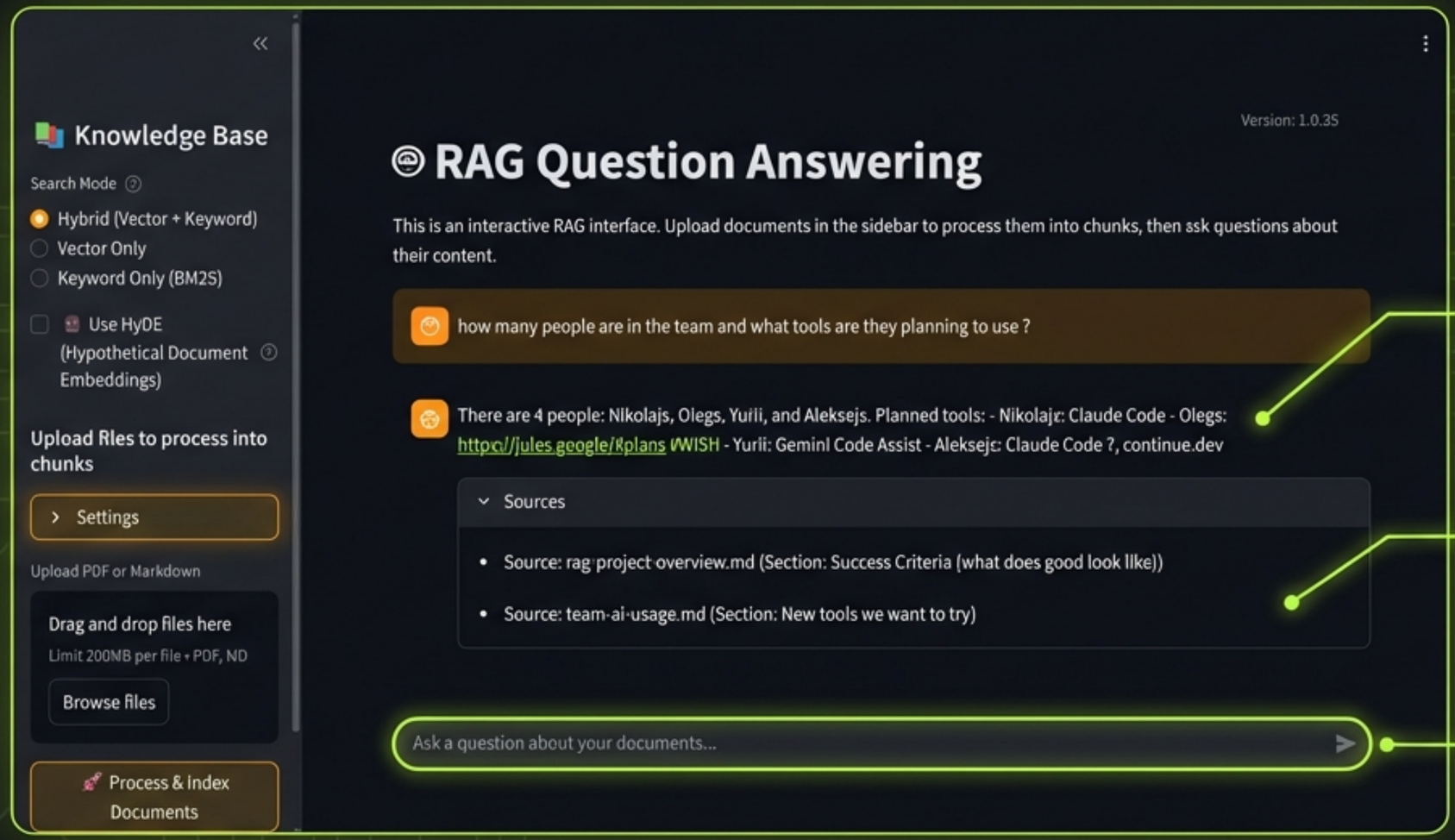
Hypothetical Document Embeddings: Search for what the answer *looks* like.

Solving Ambiguity: Multi-Query Expansion



Significantly increases recall by addressing multiple potential interpretations of the user's intent.

The Result: Smart, Grounded Retrieval



Performance Metrics

Retrieval Relevance (>90%)

Latency (<5s)

Accuracy (>85%)

Scalability (1,000+ Docs)

Key Learnings & Surprises

What Worked

RAG is efficient. Claude Code is a productivity beast.

The Surprise

Autonomous planning capabilities of agents are higher than expected.

The Challenge

Vocabulary mismatch (User vs. Docs) is the hardest problem. Solved by HyDE.

Ready to Deploy

github.com/AleksejsKudrins/PaceraHackDaysRag

TEAM 10: ALEKSEJS • NIKOLAJS • OLEGS • YURII