Intelligent Information Retrieval Course | Fall 2024
Instructor: Dr. A. Shakery | Chef TA: Samaneh PaymaniRad

Hands-On Retrieval Augmented Generation (RAG)

Introduction to LanceDB and LangChain

About Me

- MSc Student, Information Technology Engineering, University of Tehran
- BSc Computer Engineering, University of Isfahan
- Graduate Researcher at
  - NLP Lab

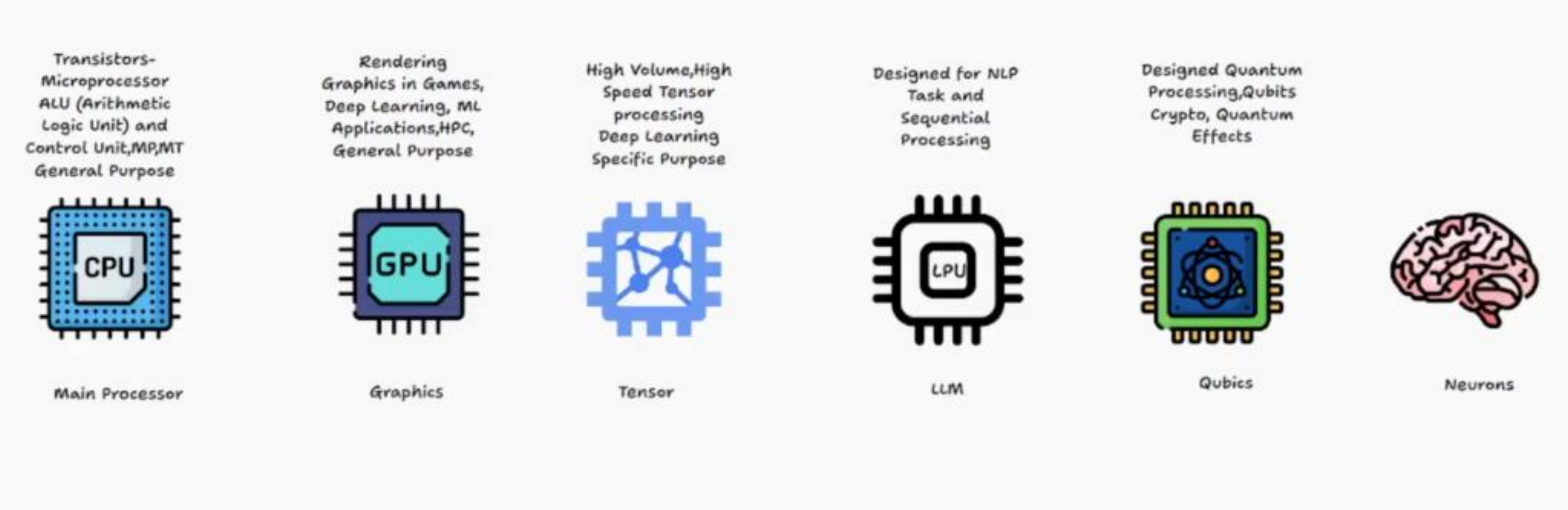
    Supervisor: Dr. Hesham Faili
  - Computational Cognitive Psychology Lab

    Supervisor: Dr. Hadi Moradi
- LLM Researcher
- Research Field: LLM base Language Agents and Psychology
- AI/Software Engineer

Workshop Plan

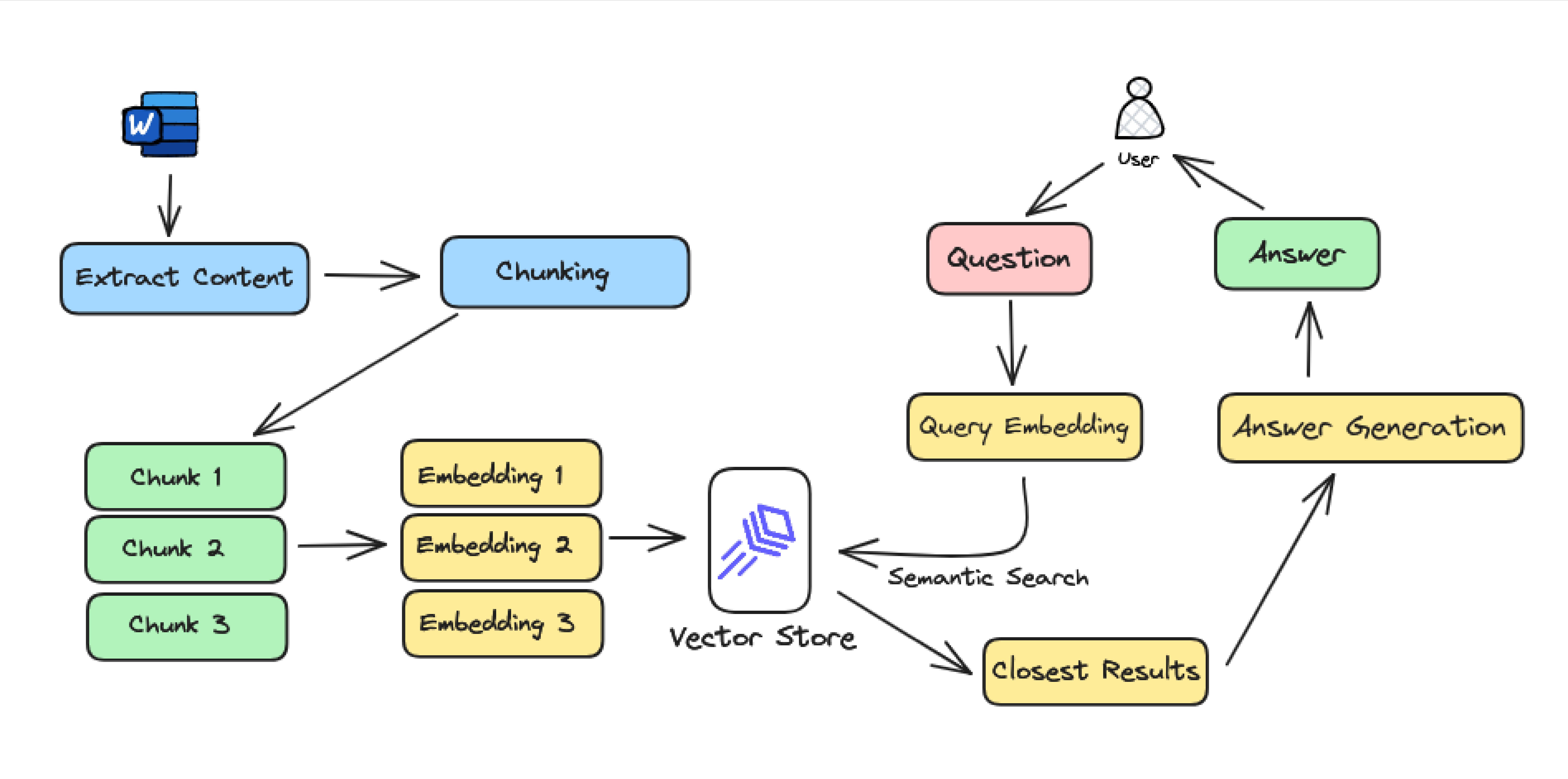
- Introduction to RAG
- How to work with LanceDB
- How to work with LLMs using APIs and Langchain framwork

LLMs, but not just input/output


Transistors- Microprocessor ALU (Arithmetic Logic Unit) and Control Unit,MP,MT General Purpose
CPU
Main Processor
Rendering Graphics in Games, Deep Learning, ML Applications,HPC, General Purpose
GPU
Graphics
High Volume,High Speed Tensor processing Deep Learning Specific Purpose
Tensor
Designed for NLP Task and Sequential Processing
LPU
LLM
Designed Quantum Processing,Qubits Crypto, Quantum Effects
Qubics
Neurons

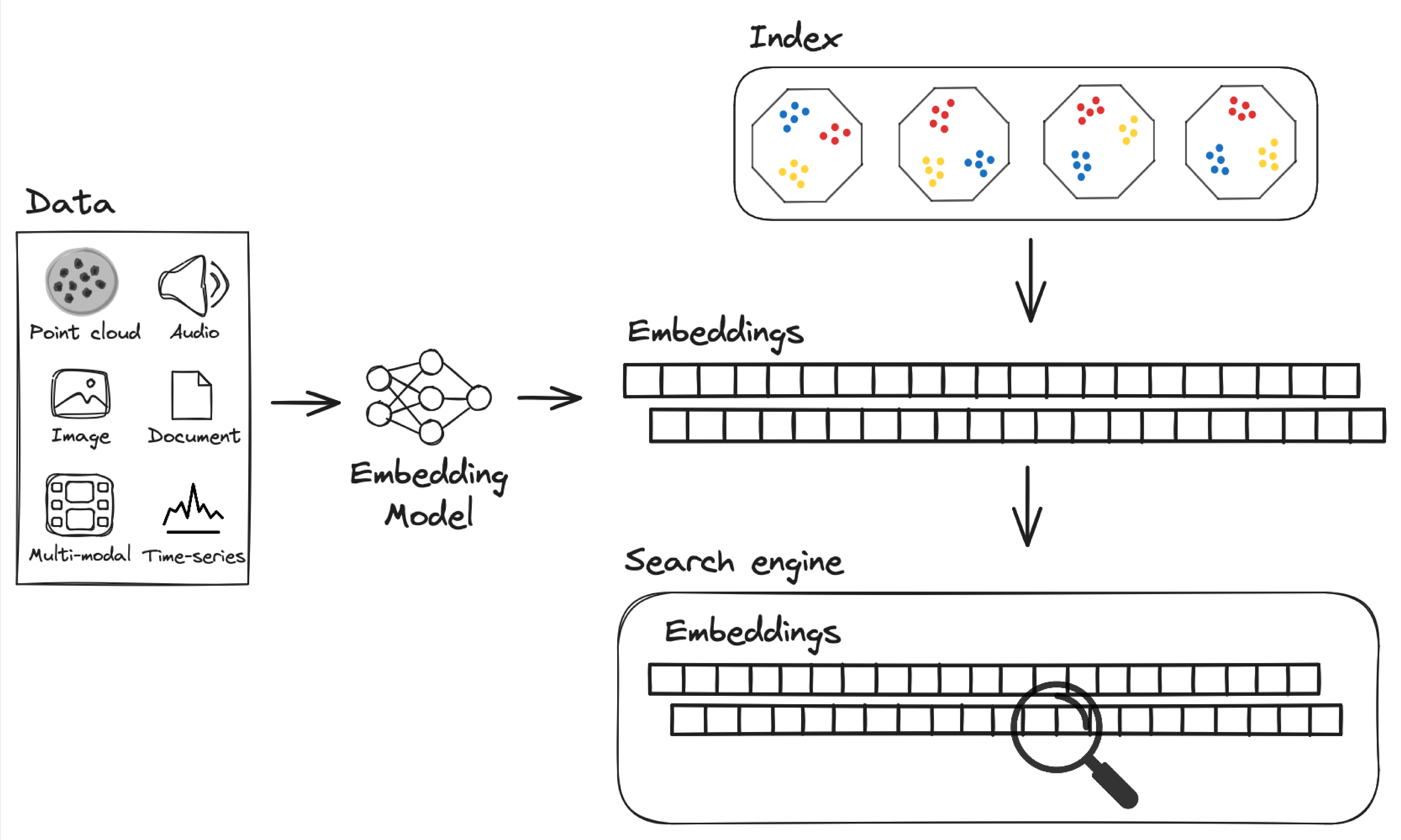
Source: JCharisTech

RAG



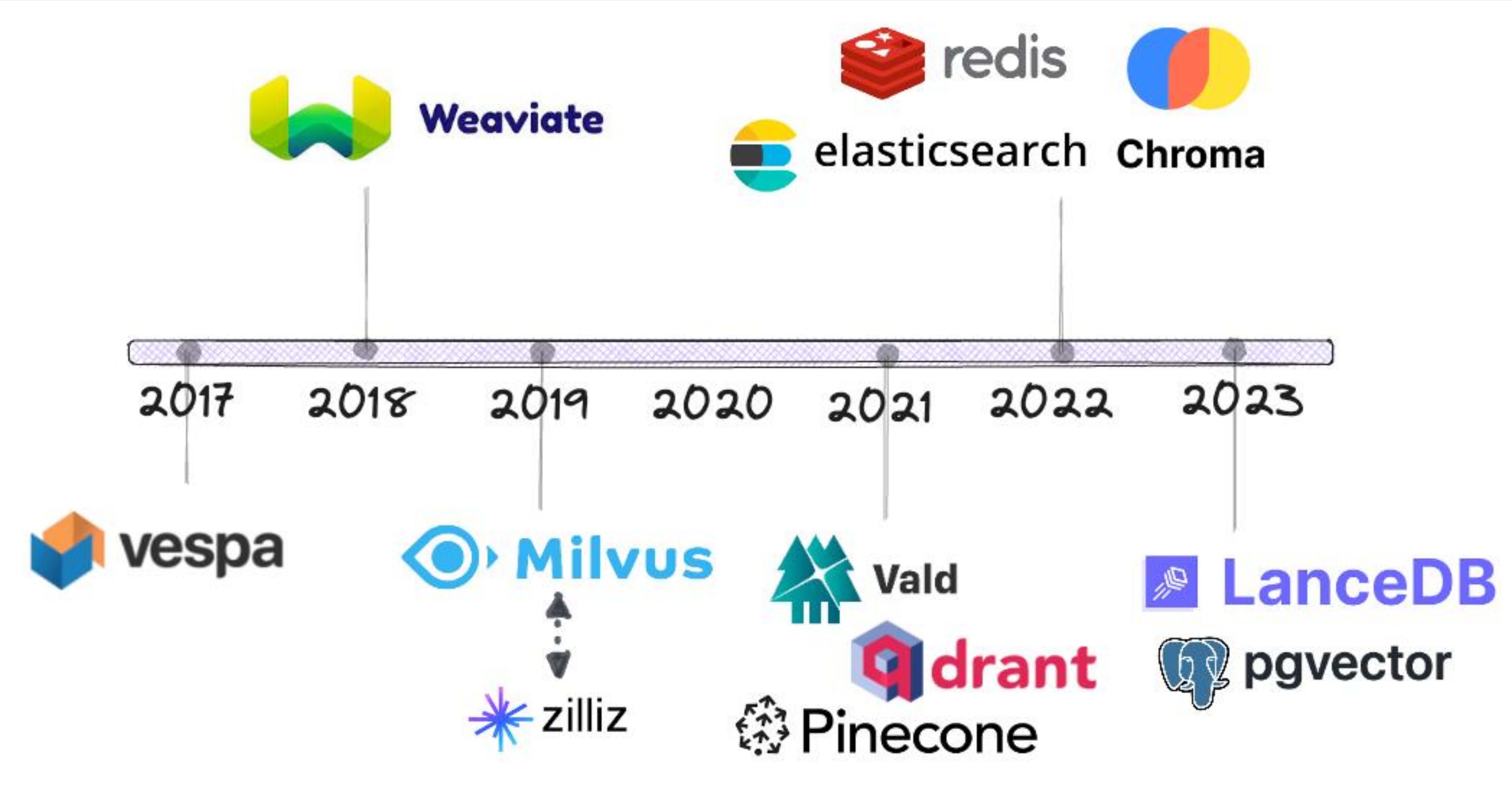
Source: LanceDB github

Vector Search



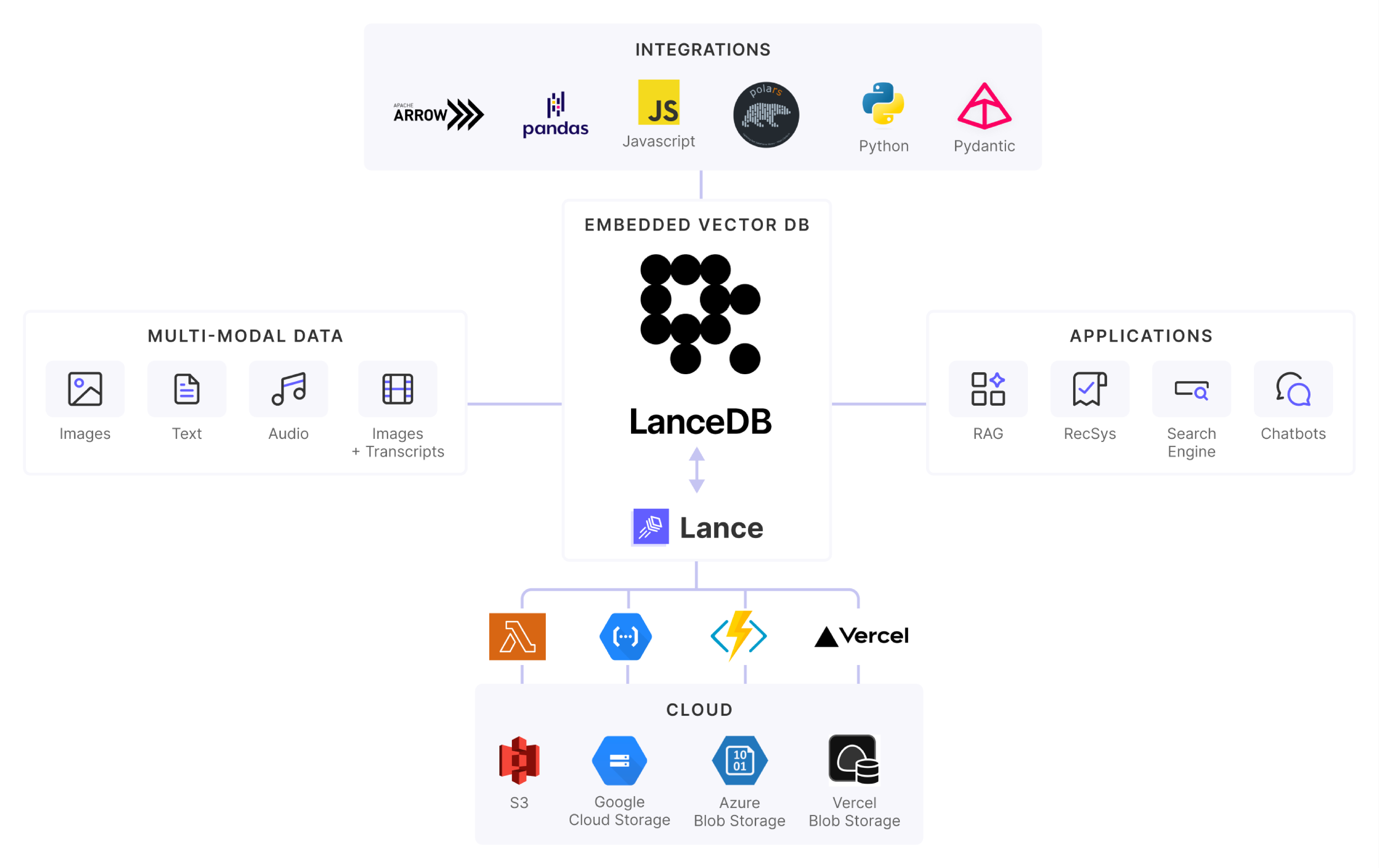
Source: LanceDB github

Vector Databased



Source: The Data Quarry blog

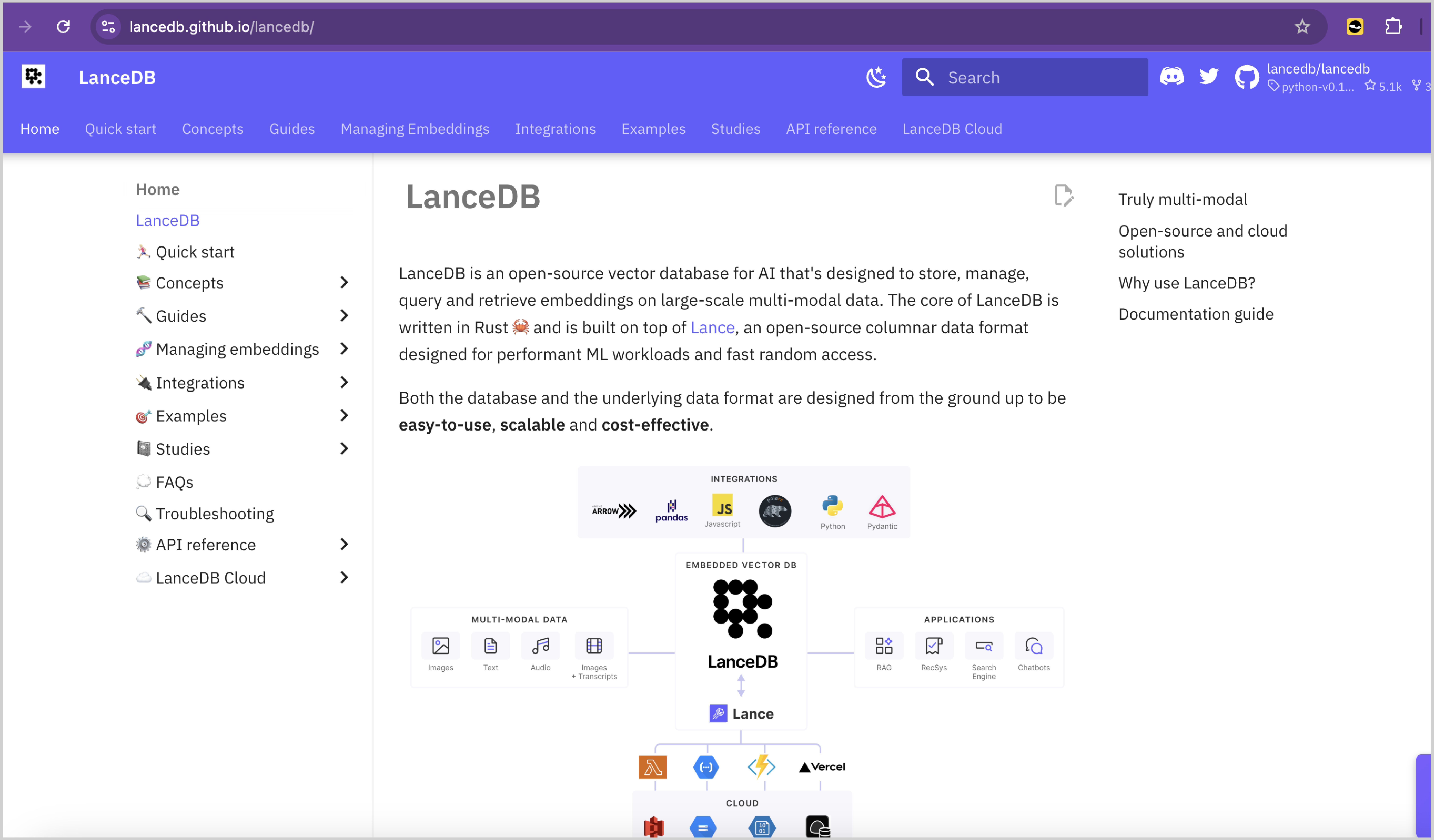
LanceDB



Source: LanceDB github

Let’s code with LanceDB

LanceDB Documentation



lancedb.github.io/lancedb/

Home

LanceDB

- Quick start
- Concepts
- Guides
- Managing embeddings
- Integrations
- Examples
- Studies
- FAQs
- Troubleshooting
- API reference
- LanceDB Cloud

LanceDB

LanceDB is an open-source vector database for AI that's designed to store, manage, query and retrieve embeddings on large-scale multi-modal data. The core of LanceDB is written in Rust 🦀 and is built on top of Lance, an open-source columnar data format designed for performant ML workloads and fast random access.

Both the database and the underlying data format are designed from the ground up to be **easy-to-use**, **scalable** and **cost-effective**.

Truly multi-modal

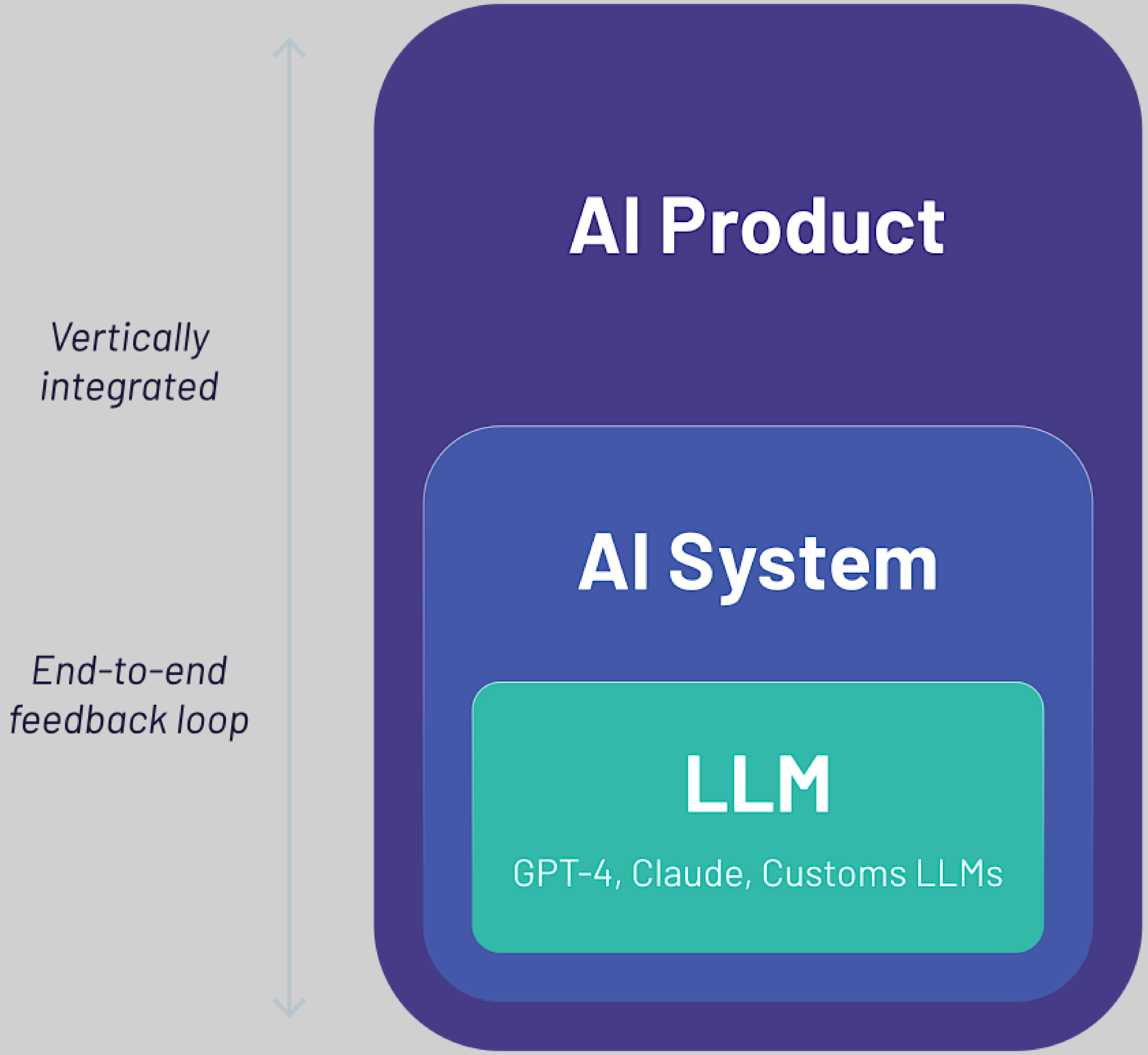
Open-source and cloud solutions

Why use LanceDB?

Documentation guide

Source: LanceDB github

LLM based systems and products



AI Products re-bundle user workflow around AI systems (e.g. feedback to AI)

AI Systems combine multiple AI components (including LLMs) to do things like access private data, automate workflows, ...

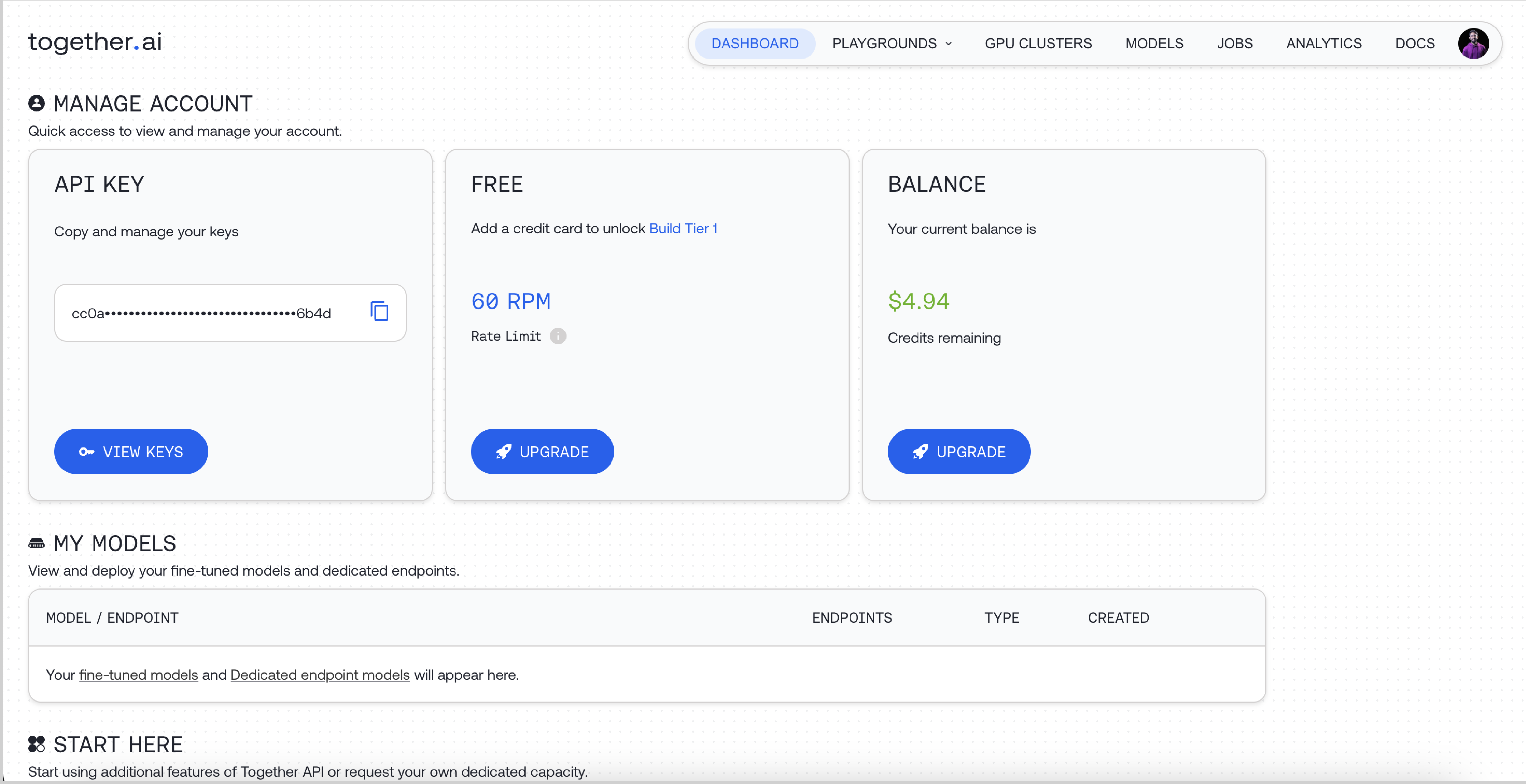
LLMs are the core ("intel inside")

Source: BoxCars AI: Tabrez Syed

Model Providers

together.ai

TogetherAI



MANAGE ACCOUNT

Quick access to view and manage your account.

API KEY

Copy and manage your keys

cc0a•••••••••••••••••••••••••••••••••6b4d

FREE

Add a credit card to unlock Build Tier 1

60 RPM

Rate Limit

BALANCE

Your current balance is

$4.94

Credits remaining

MY MODELS

View and deploy your fine-tuned models and dedicated endpoints.

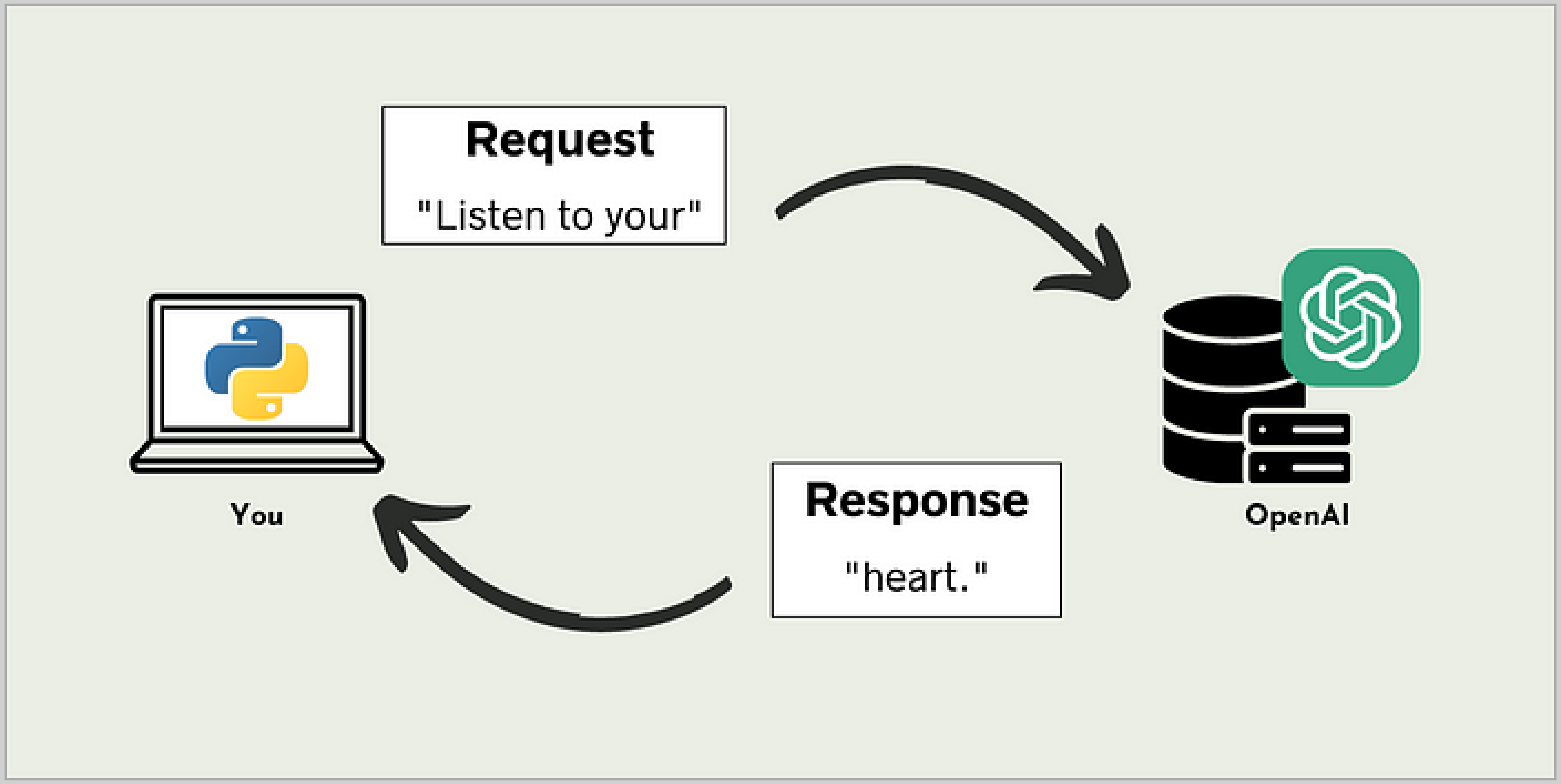
MODEL / ENDPOINT	ENDPOINTS	TYPE	CREATED
Your fine-tuned models and Dedicated endpoint models will appear here.			

START HERE

Start using additional features of Together API or request your own dedicated capacity.

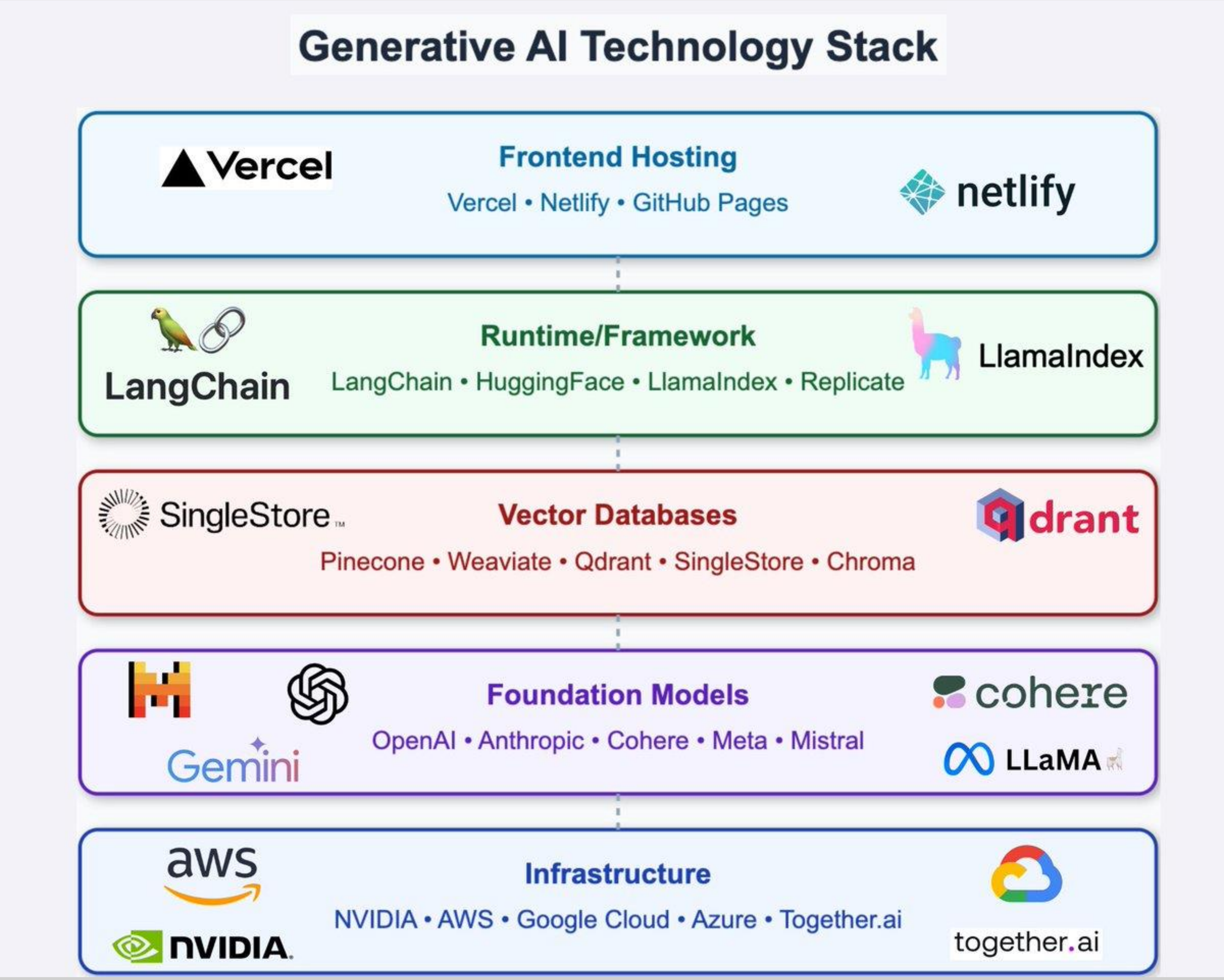
Source: TogetherAI

What does ‘client’ mean here?



Source: TowardsDataScience: Shaw Talebi

Stack



Generative AI Technology Stack

Frontend Hosting
Vercel • Netlify • GitHub Pages

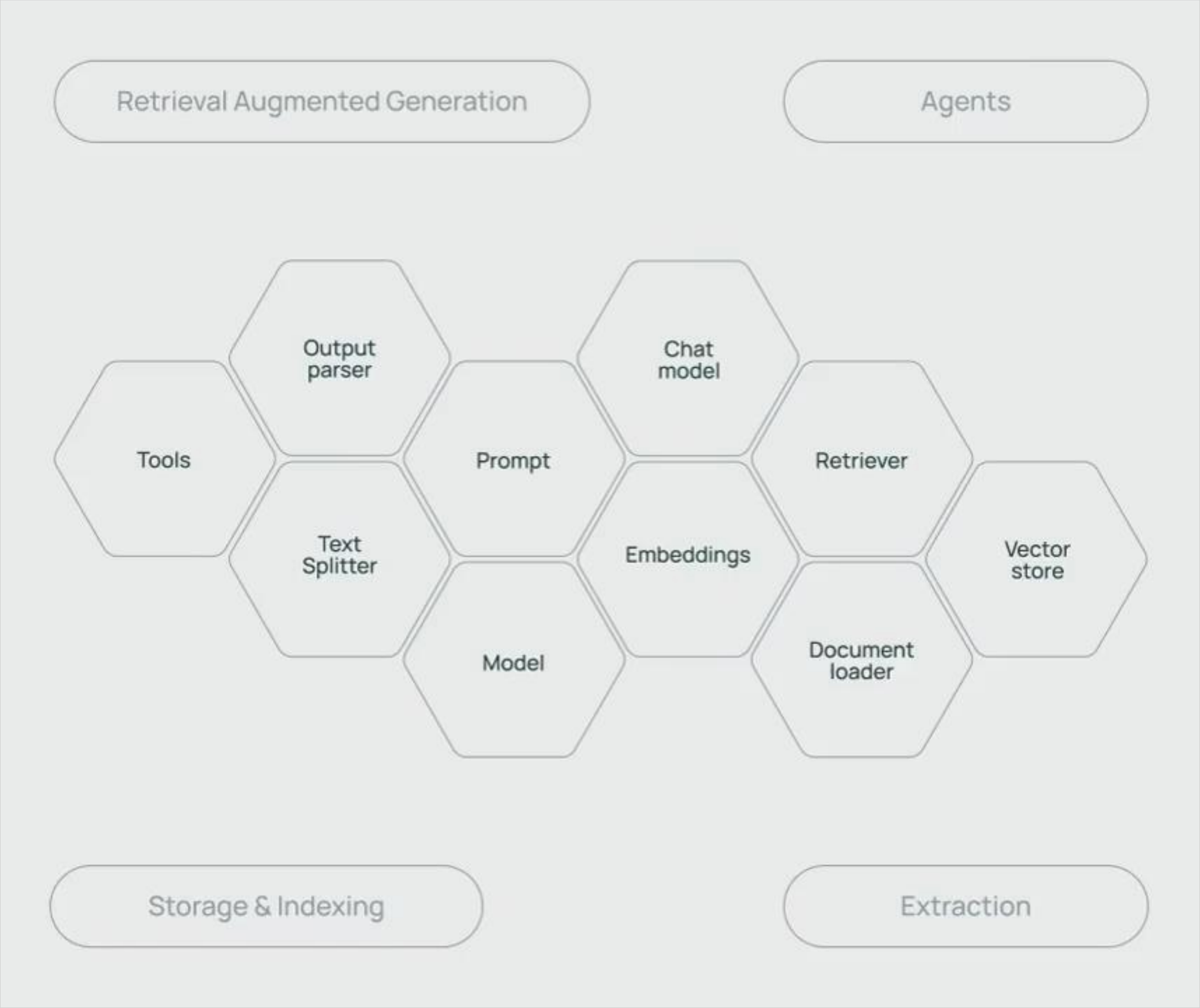
Runtime/Framework
LangChain • HuggingFace • LlamaIndex • Replicate

Vector Databases
Pinecone • Weaviate • Qdrant • SingleStore • Chroma

Foundation Models
OpenAI • Anthropic • Cohere • Meta • Mistral

Infrastructure
NVIDIA • AWS • Google Cloud • Azure • Together.ai

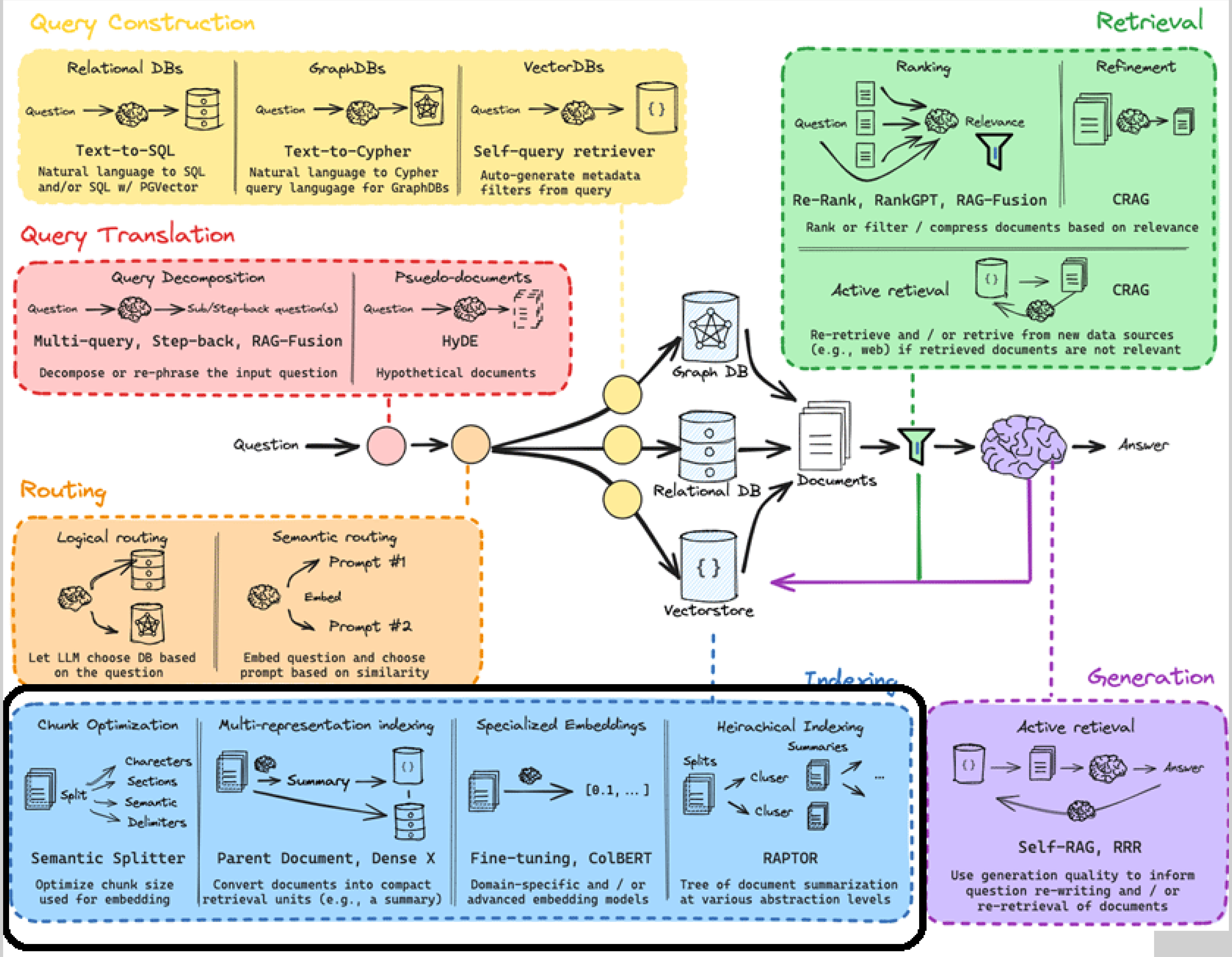
Source: Pavan Belagatti™


Retrieval Augmented Generation
Agents
Output parser
Chat model
Tools
Prompt
Retriever
Text Splitter
Embeddings
Vector store
Model
Document loader
Storage & Indexing
Extraction

LangChain

Let’s code with OpenAI client and Langchain

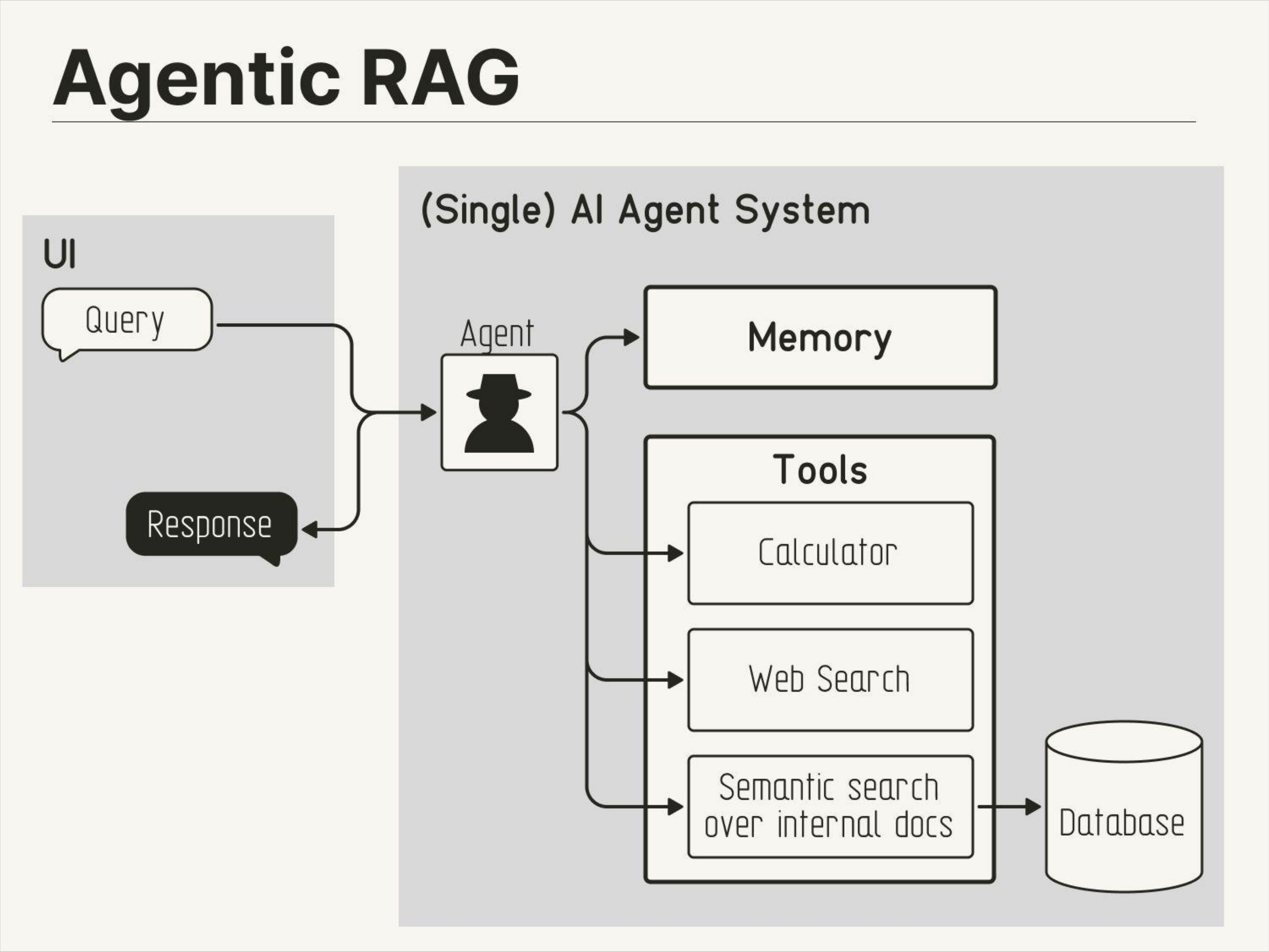
Advanced RAG techniques


Query Construction
Relational DBs
Question
Text-to-SQL
Natural language to SQL and/or SQL w/ PGVector
GraphDBs
Question
Text-to-Cypher
Natural language to Cypher query langugage for GraphDBs
VectorDBs
Question
Self-query retriever
Auto-generate metadata filters from query
Query Translation
Query Decomposition
Question
Sub/Step-back question(s)
Multi-query, Step-back, RAG-Fusion
Decompose or re-phrase the input question
Psuedo-documents
Question
HyDE
Hypothetical documents
Routing
Logical routing
Let LLM choose DB based on the question
Semantic routing
Prompt #1
Embed
Prompt #2
Embed question and choose prompt based on similarity
Question
Graph DB
Relational DB
Vectorstore
Documents
Answer
Retrieval
Ranking
Question
Relevance
Re-Rank, RankGPT, RAG-Fusion
Refinement
CRAG
Rank or filter / compress documents based on relevance
Active retrieval
CRAG
Re-retrieve and / or retrive from new data sources (e.g., web) if retrieved documents are not relevant
Indexing
Chunk Optimization
Split
Charecters
Sections
Semantic
Delimiters
Semantic Splitter
Optimize chunk size used for embedding
Multi-representation indexing
Summary
Parent Document, Dense X
Convert documents into compact retrieval units (e.g., a summary)
Specialized Embeddings
[0.1, ...]
Fine-tuning, ColBERT
Domain-specific and / or advanced embedding models
Heirachical Indexing
Splits
Summaries
Cluser
Cluser
...
RAPTOR
Tree of document summarization at various abstraction levels
Generation
Active retrieval
Answer
Self-RAG, RRR
Use generation quality to inform question re-writing and / or re-retrieval of documents

Source: Latest and Greatest blog

div.beehiiv.com

Agentic RAG



Source: X.com/
@helloiamleonie

Good Luck with LLMs!

you'll need it ;)

For further discussions or questions, feel free to reach out via email:

miladmohammadi@ut.ac.ir