IDUG

2026

Sydney | March 16 - 18

AU Db2 TECH CONFERENCE

Db2 AI Strategy Update for Developers

Dale McInnis, *IBM*

Session Code: B05

Platform:

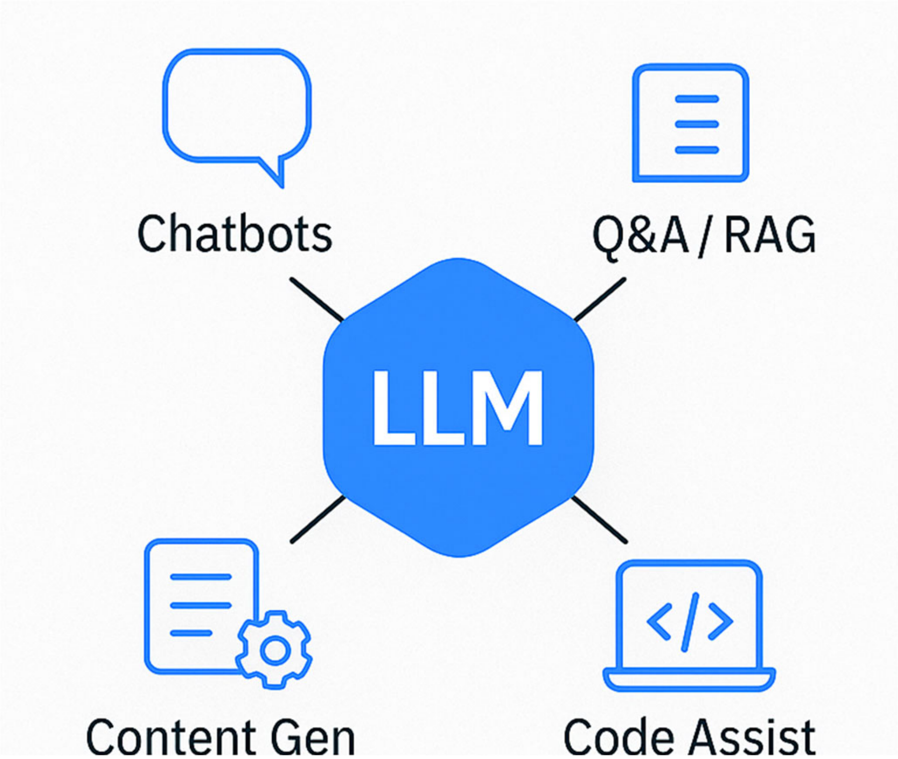
Db2 LUW

AGENDA

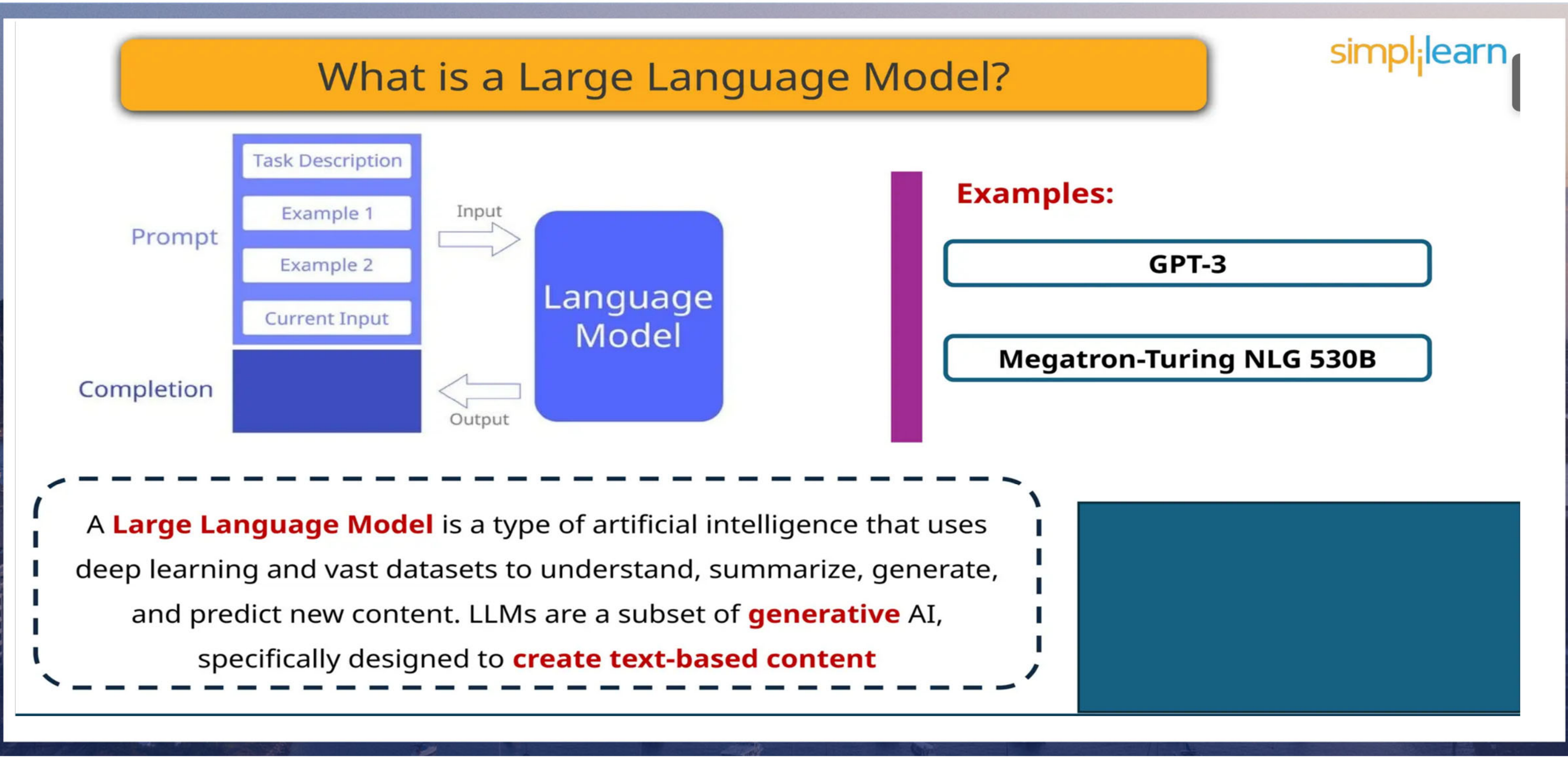
- Concepts / Definitions
- Db2 Vector Features
- Python LLM Frameworks Support
- EAP Features
- Demo: RAG Search with IBM Db2

LLM Applications

Apps that use **LLMs** to understand, generate, and interact with natural language.



What is a LLM?



What is a Large Language Model?

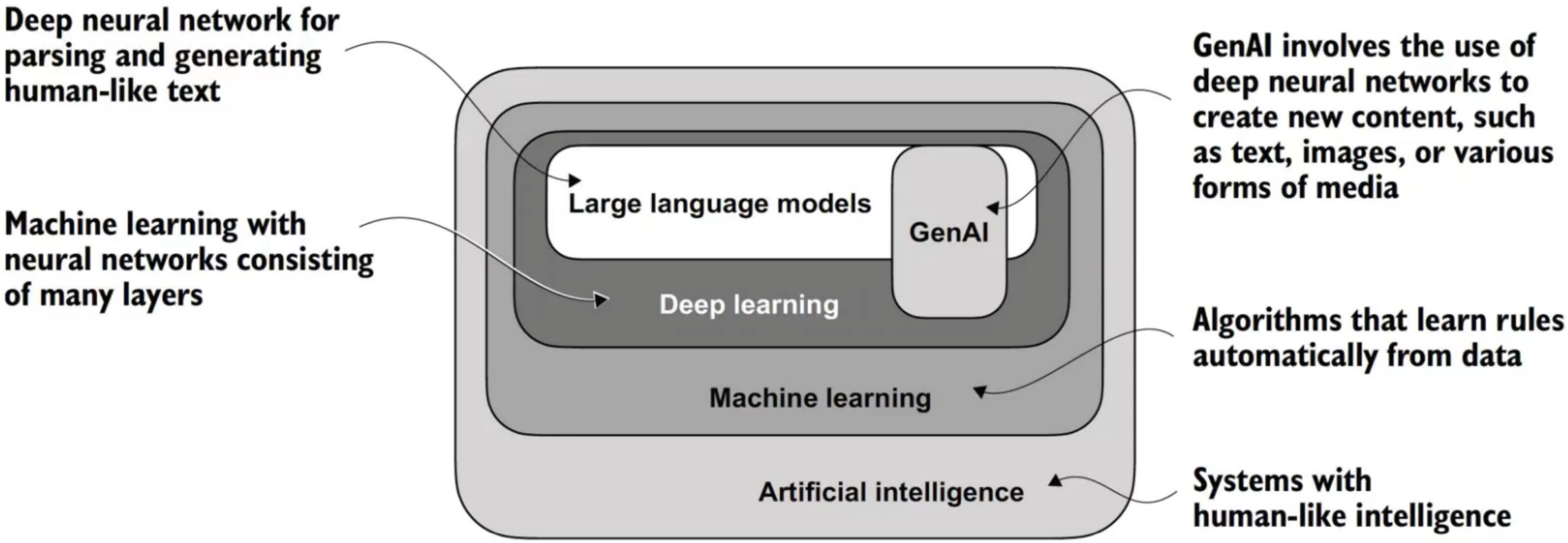
simplilearn

Examples:

- GPT-3
- Megatron-Turing NLG 530B

A **Large Language Model** is a type of artificial intelligence that uses deep learning and vast datasets to understand, summarize, generate, and predict new content. LLMs are a subset of **generative** AI, specifically designed to **create text-based content**

Where do LLMs fit in the AI eco-sphere?



Deep neural network for parsing and generating human-like text

Machine learning with neural networks consisting of many layers

GenAI involves the use of deep neural networks to create new content, such as text, images, or various forms of media

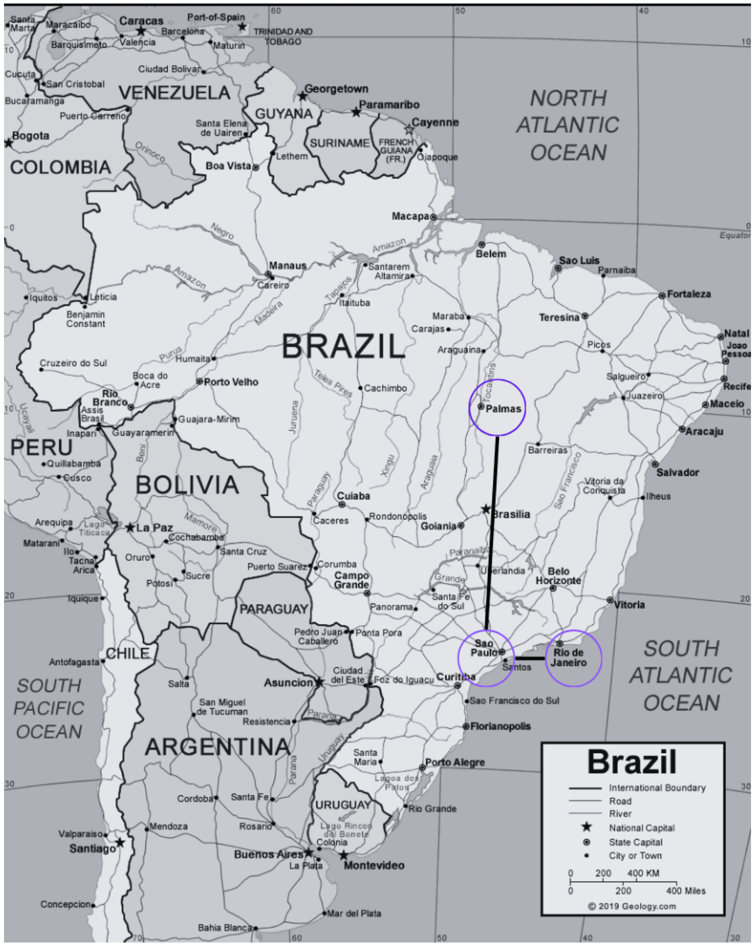
Algorithms that learn rules automatically from data

Systems with human-like intelligence

[Source: Raschka, Sebastian. *Build a large language model (from scratch)*. Simon and Schuster, 2024.]

What is a vector?

- A list of numbers, like (1, 2)
- Represents a point in space
- Like map coordinates for cities
- You can measure distance between two vectors



3 Common Retrieval Techniques

Keyword-based

Finds exact word matches such as IDs, emails

E.g., search query: "laptop repair"
Finds: "laptop repair guide", "repair@email" (no ranking)
Misses: "fixing laptops" (fixing ≠ repair), "repairing computers"

Full-text

Matches word variations, ranks results

e.g., search query: "laptop repair"
Ranked: "Fixing Laptops" > "Computer Tips" > "Support email"
Misses: "Device troubleshooting" (no word overlap)

Vector-based

Find similar ideas, not just word matches

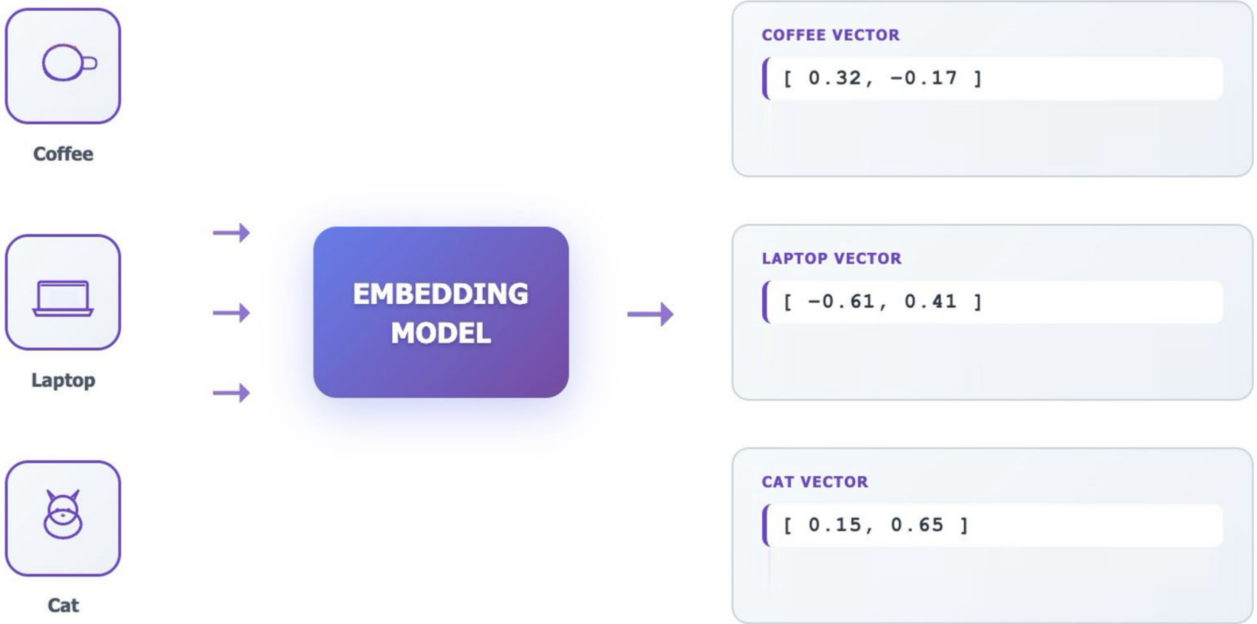
e.g., search query: "laptop repair"
Finds: "Fixing Laptops", "Device Troubleshooting"
✓ Matches meaning - works with any vocabulary

Which Two Are More Similar?

It Depends!

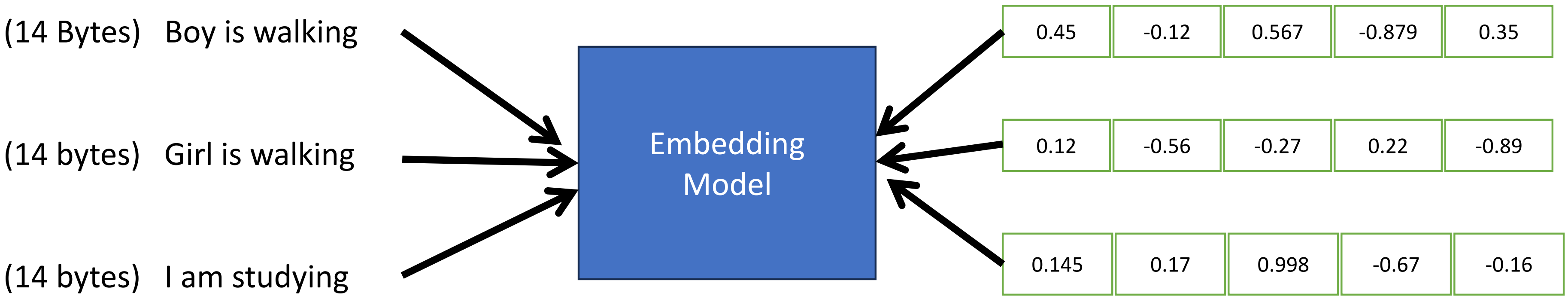
Vector Embedding: Turning Data Into Vectors

Transforming real-world objects into numerical representations



Word of Caution – Vector Bloat!

OpenAI embeddings: 1536 (6,144 bytes)
Sentence-BERT: 768 (3,072 bytes)
Vision Embedding: 2048-4096 (8,192 – 16,384 bytes)



(14 Bytes) Boy is walking

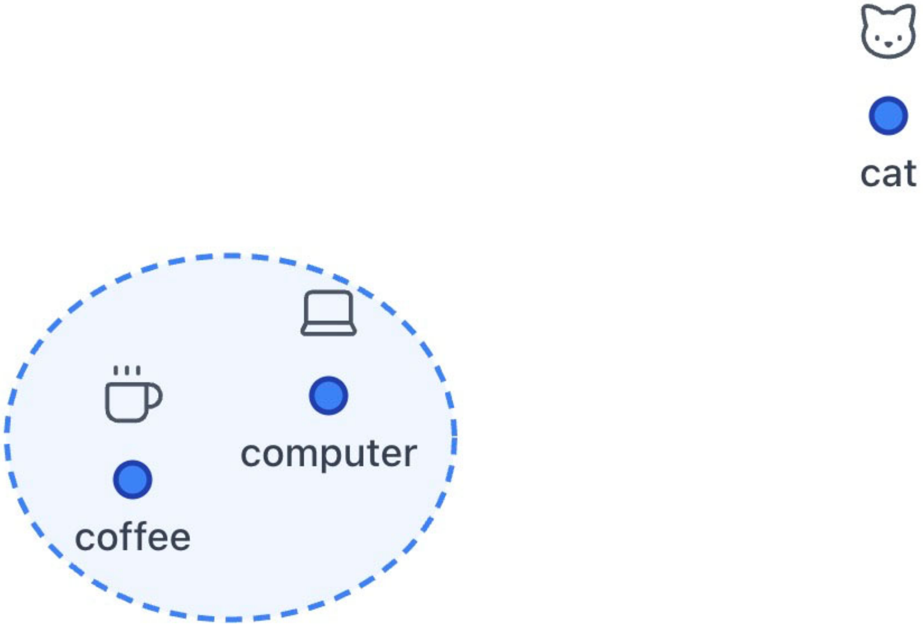
(14 bytes) Girl is walking

(14 bytes) I am studying

Embedding Model

0.45	-0.12	0.567	-0.879	0.35
0.12	-0.56	-0.27	0.22	-0.89
0.145	0.17	0.998	-0.67	-0.16

Real-world Entities on Vector Space



So, what? What can Vectors do for you?

Common Use Cases for Vector Search

Find similar songs you'll like

Shop by taking a photo

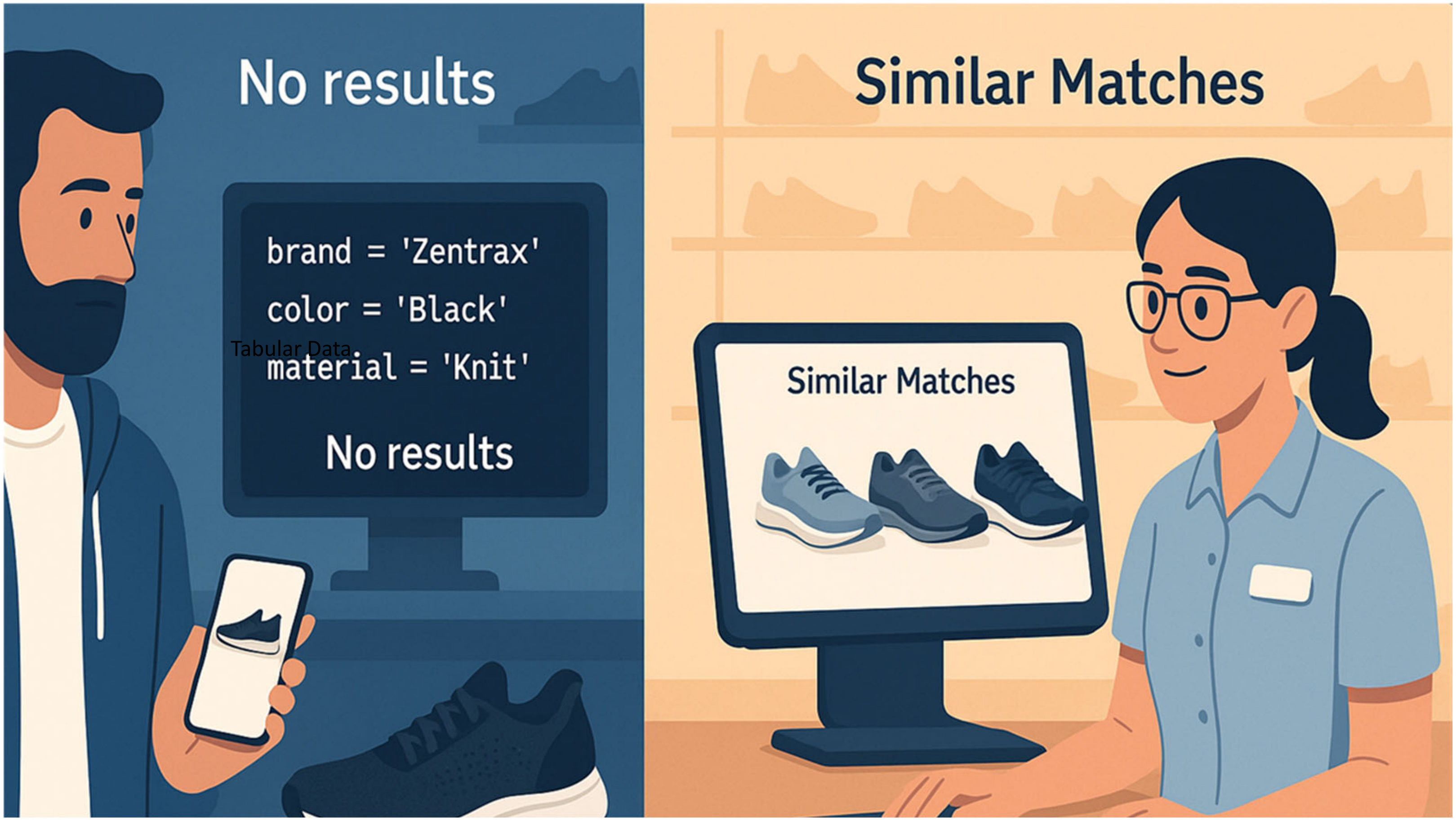
Movies you might enjoy

Find your photos by faces

Search using pictures

Ask questions, get answers

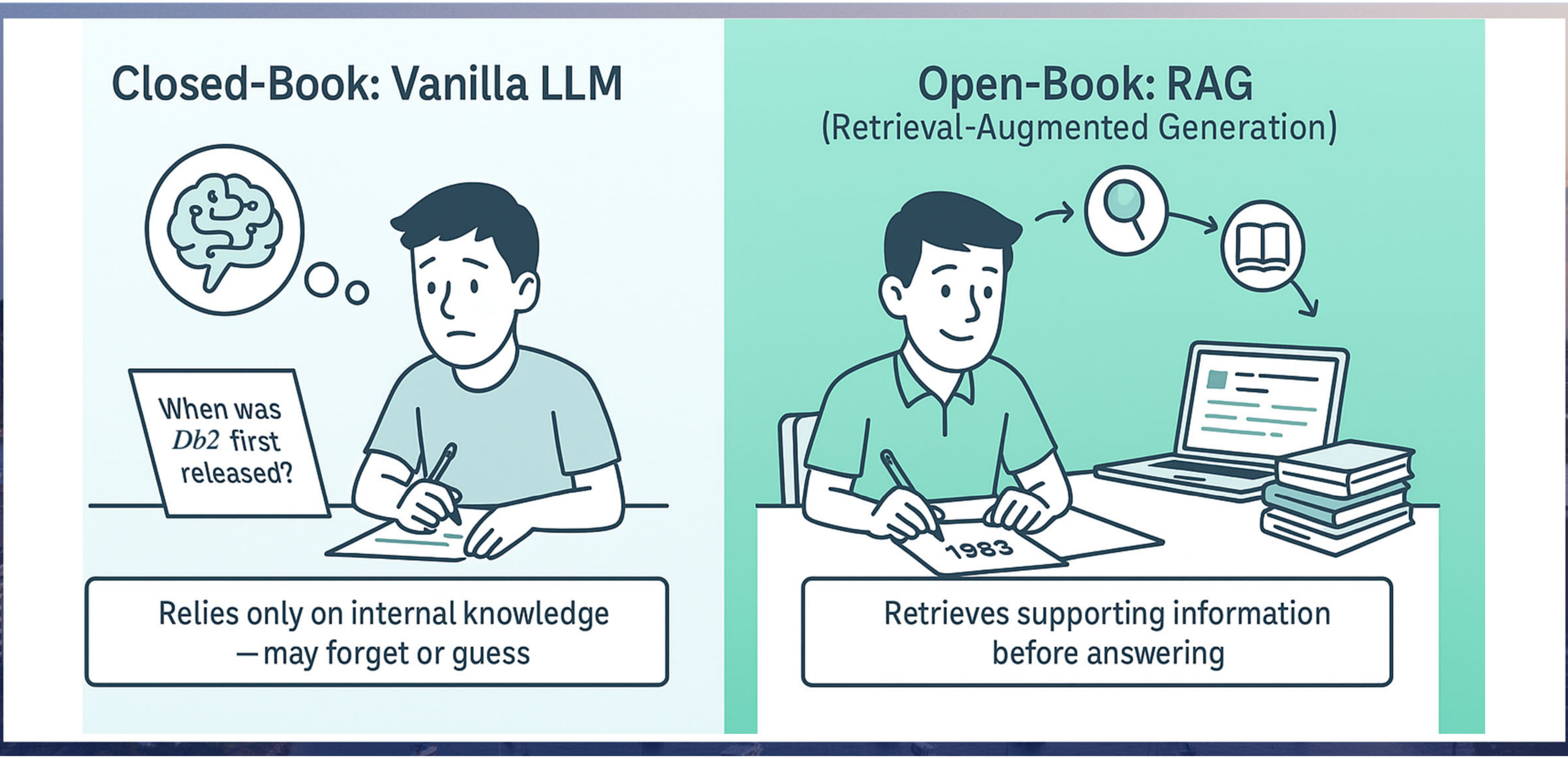
Semantic Search Over



```
brand = 'Zentrax'
color = 'Black'
material = 'Knit'
```

Tabular Data

Why is RAG?

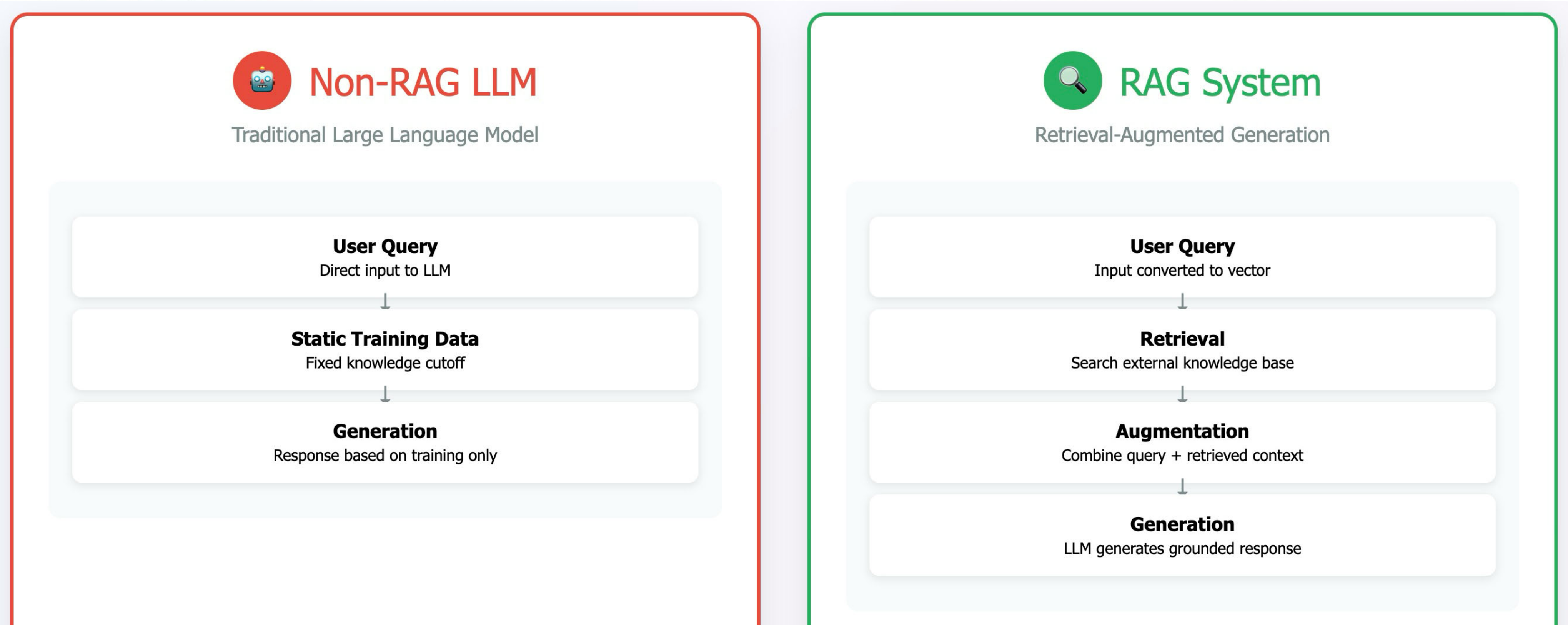


Closed-Book: Vanilla LLM

Relies only on internal knowledge
— may forget or guess

Open-Book: RAG
(Retrieval-Augmented Generation)

Retrieves supporting information
before answering



🤖 Non-RAG LLM

Traditional Large Language Model

User Query
Direct input to LLM

↓

Static Training Data
Fixed knowledge cutoff

↓

Generation
Response based on training only

🔍 RAG System

Retrieval-Augmented Generation

User Query
Input converted to vector

↓

Retrieval
Search external knowledge base

↓

Augmentation
Combine query + retrieved context

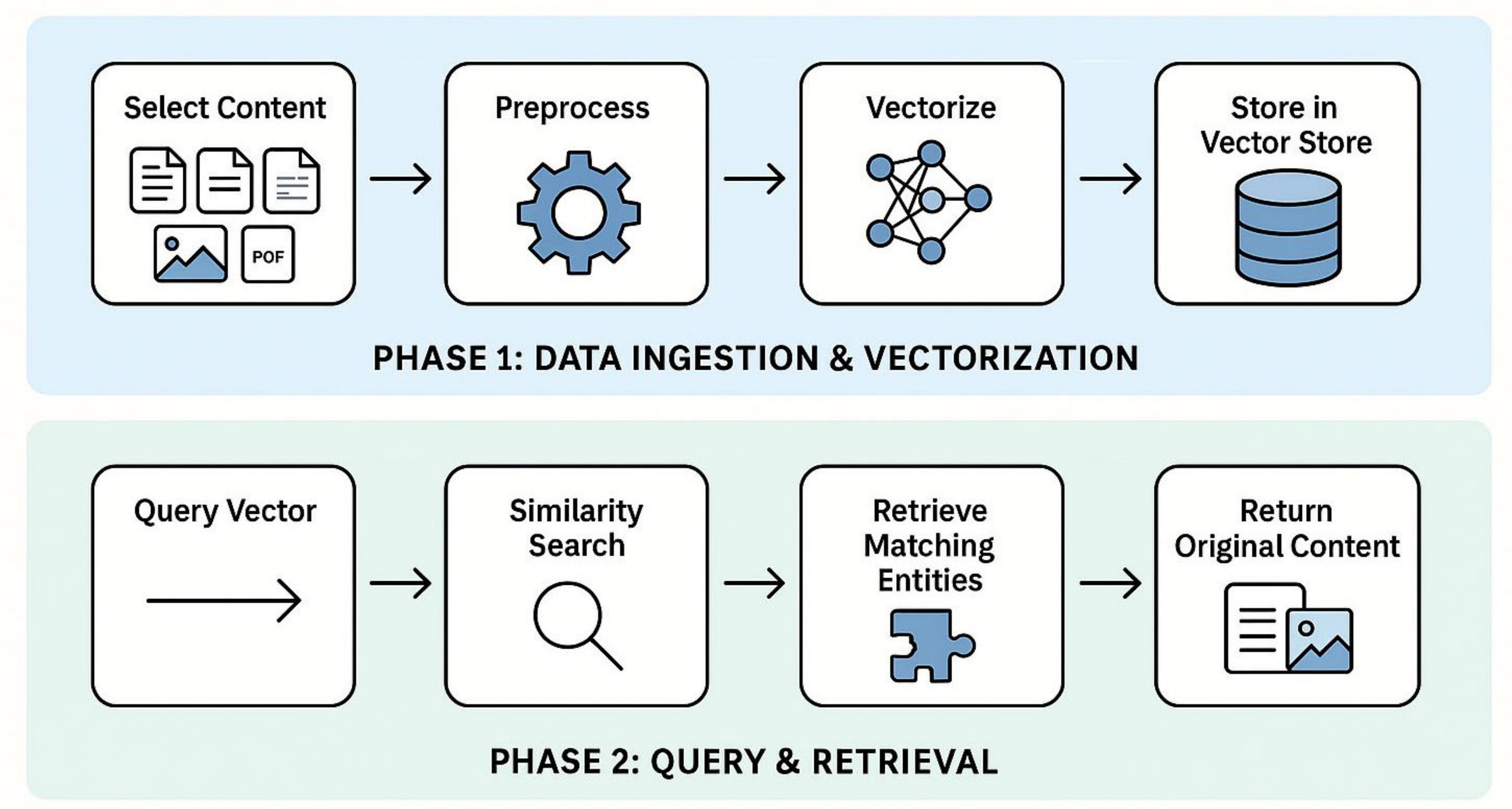
↓

Generation
LLM generates grounded response

Knowledge-Cutoff / Static Knowledge, Hallucinations

Dynamic Knowledge, Domain-Knowledge, Source citation

Workflow of Vector Search Use Cases



Select Content → Preprocess → Vectorize → Store in Vector Store

PHASE 1: DATA INGESTION & VECTORIZATION

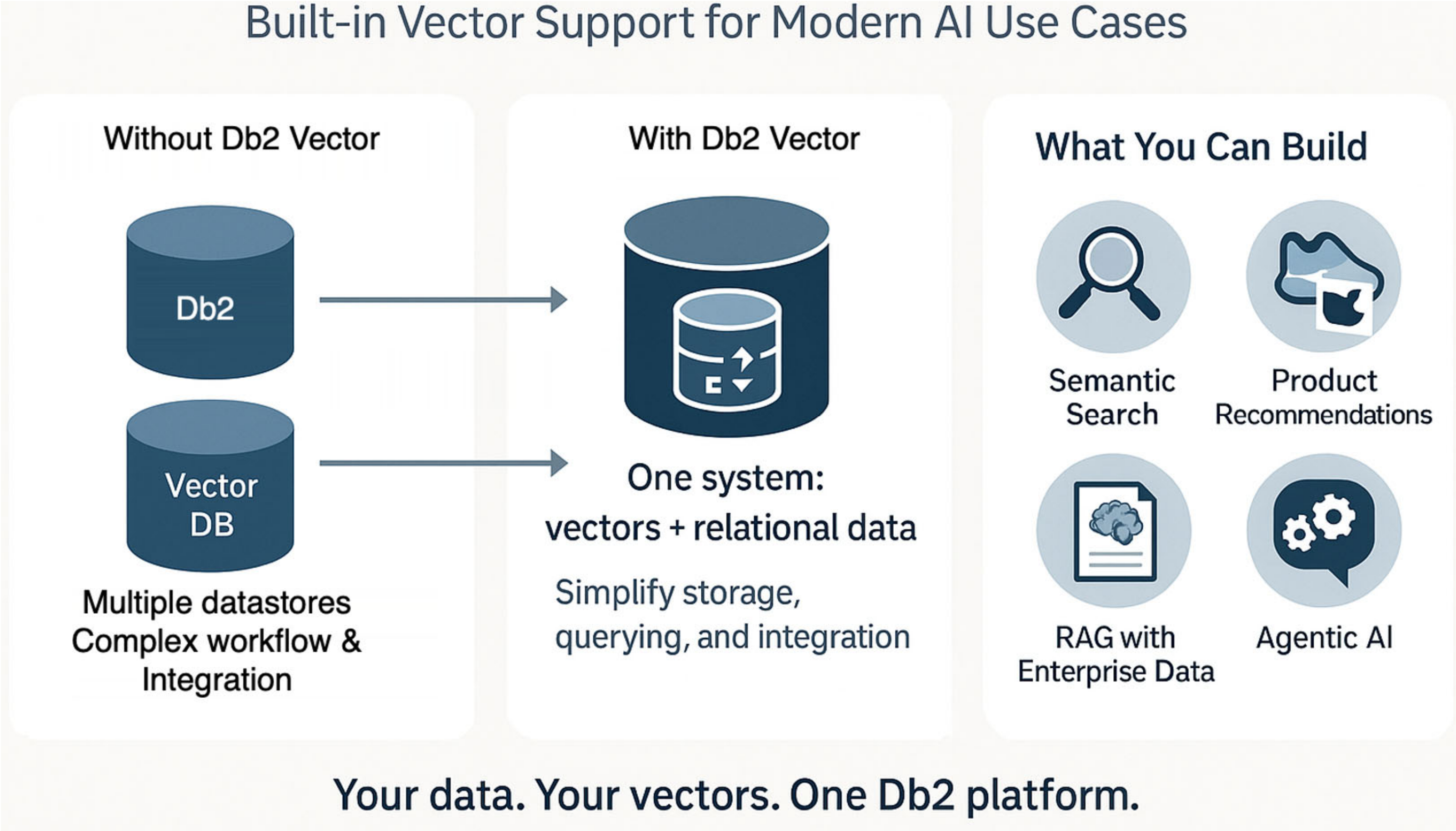
Query Vector → Similarity Search → Retrieve Matching Entities → Return Original Content

PHASE 2: QUERY & RETRIEVAL

IDUG 2026 Australia Db2 Tech Conference

Db2 unlocks Semantic Search and GenAI workflows

#IDUGDb2



Built-in Vector Support for Modern AI Use Cases

Without Db2 Vector

Multiple datastores
Complex workflow &
Integration

With Db2 Vector

One system:
vectors + relational data

Simplify storage,
querying, and integration

What You Can Build

Your data. Your vectors. One Db2 platform.

Vector Support in Db2

VECTOR type:

- FLOAT32 vectors
- INT8 vectors
- Max dimension:
  - 8168 (FLOAT 32)
  - 32672 (INT8)

VECTOR Functions:

- VECTOR_DISTANCE – between vectors
- VECTOR constructor: string -> vector
- VECTOR_SERIALIZE – vector -> string
- *VECTOR_NORM* – vector magnitude
- VECTOR_DIMENSION_COUNT – number of dimensions

LLM Apps Dev Support:
LangChain Connector

🎉 NEW RELEASE

Db2 12.1.3
Production-Ready Vector Support

Data movement utilities • SQL routines • LLM framework integration

🦜

LangChain & LlamaIndex

Native Python packages for RAG apps

⚙️

SQL Routines

Custom AI logic at database layer

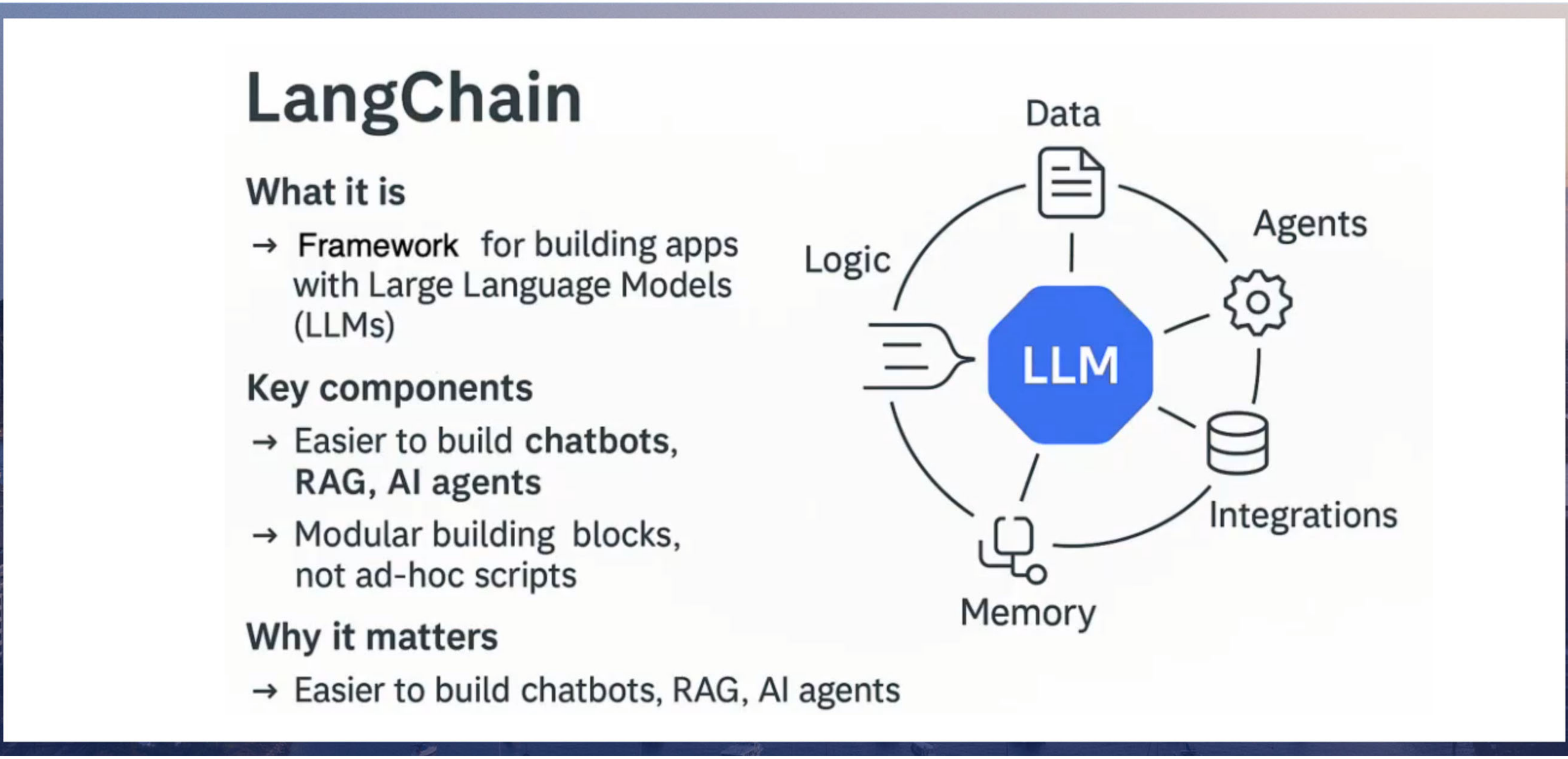
🚀

Data Movement

LOAD, backup, schema migration

Columnar Engine **VECTOR Data Type** **Enterprise Scale**

Open Source Framework for LLM Apps



LangChain

What it is

→ **Framework** for building apps with Large Language Models (LLMs)

Key components

→ Easier to build **chatbots, RAG, AI agents**

→ Modular building blocks, not ad-hoc scripts

Why it matters

→ Easier to build chatbots, RAG, AI agents

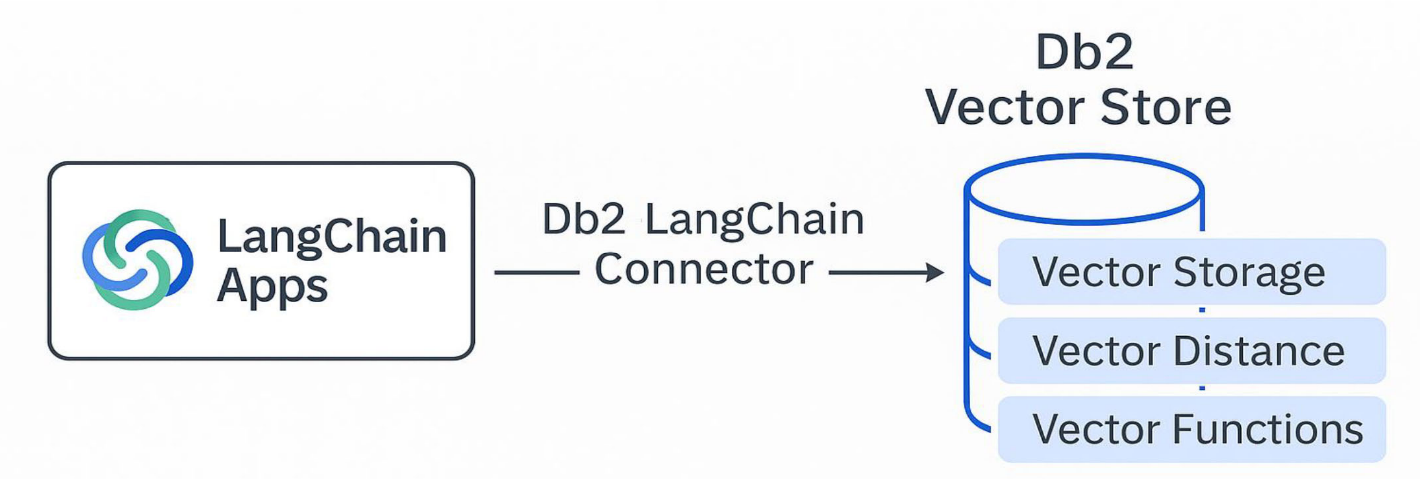
Db2 LangChain Connector

What it is

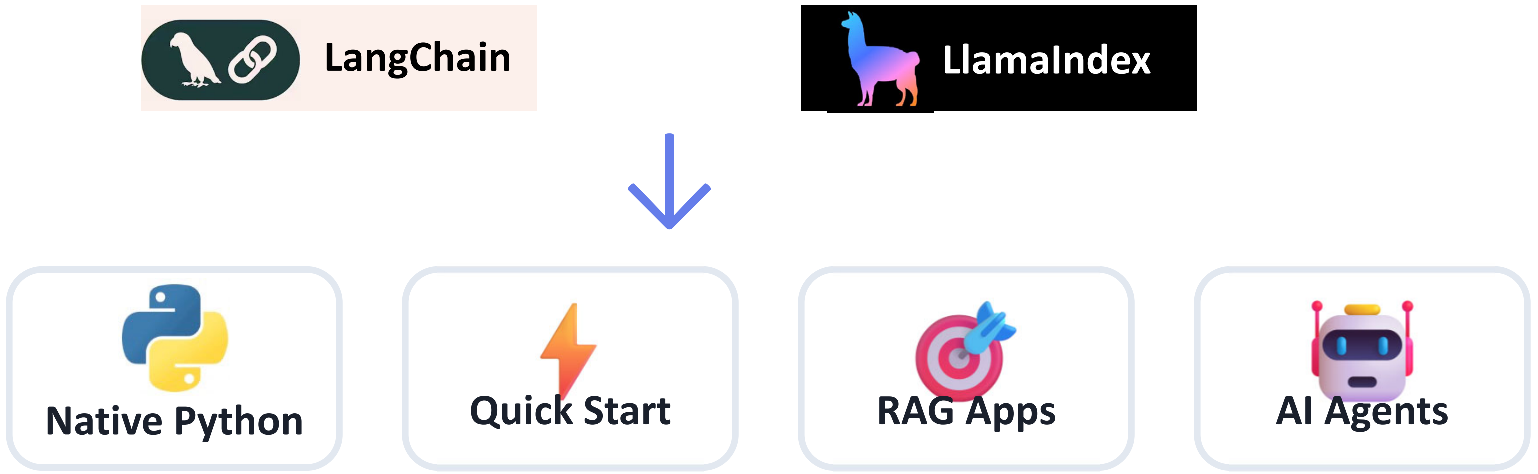
→ Open-source Python connector linking **IBM Db2** and **LangChain**
→ Powered by Db2's **native vector support** (storage, similarity search)

Why it matters

→ Brings **enterprise-grade Db2** (scale, security, reliability) to AI apps
→ Enables **RAG and AI agents** with familiar LangChain tools



Build LLM Apps with Db2



LangChain

LlamaIndex

Native Python

Quick Start

RAG Apps

AI Agents

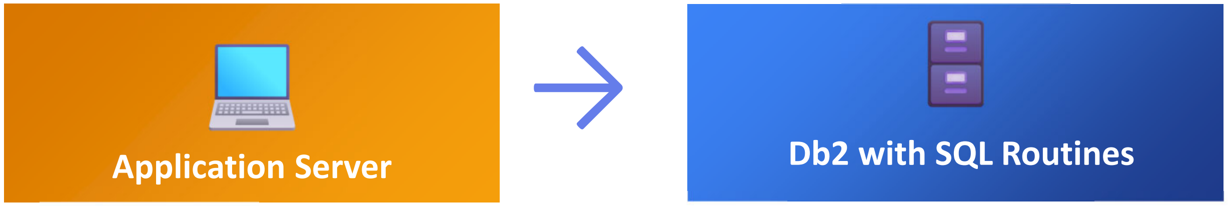
Db2 VECTOR Data Type

Cosine • Dot Product • Euclidean

✓ Enterprise Reliability ✓ Python-Native APIs ✓ No Low-Level SQL

Custom AI Logic in SQL Routines

Encapsulate AI application logic at the database layer where your data lives



Domain Logic
Custom metrics

Reusable Operations
Normalization, validation

Hybrid Queries
Vector + structured filters

Batch Processing
Compute statistics

Consistency
Same logic across apps

Columnar Engine
Native optimization

✓ Write logic once in SQL • Execute where data resides • No network overhead
UDFs & Stored Procedures with VECTOR data types

Production-Grade Vector Data Movement

Db2 12.1.2
IMPORT • EXPORT

→

Db2 12.1.3
+ 6 More Utilities

LOAD
High-volume insertion

ADMIN_COPY_SCHEMA
Clone schemas

ADMIN_MOVE_TABLE
Relocate tables

db2move
Migrate databases

External Tables
Query in data lakes

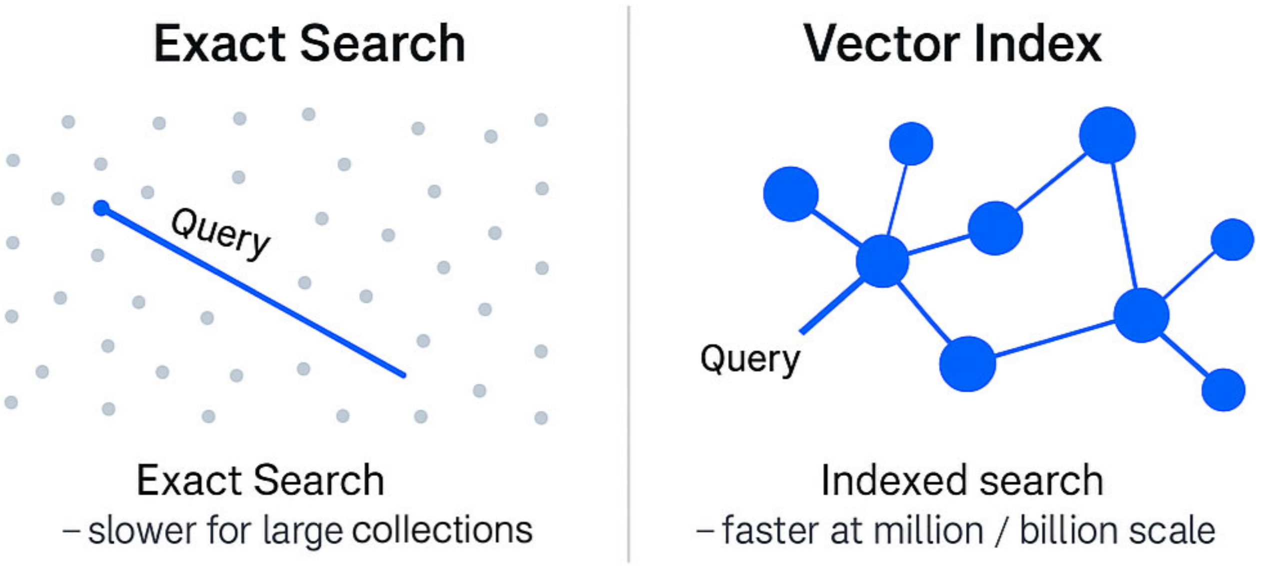
Logical Backup
Schema-level backups

Key Capabilities

📦 Full & Incremental 🎯 Table-Level 🌐 External File ⚙ Production-Ready

✓ Move vector data using the same tools as traditional data types

EAP Feature: Db2 Vector Index – Scaling Vector Search



Exact Search

Vector Index

Exact Search
– slower for large collections

Indexed search
– faster at million / billion scale

Faster vector search with Db2 Vector Index

IDUG
2026 Australia Db2 Tech Conference

Why Brute-Force KNN Doesn't Scale

#IDUGDb2

⚠ Brute-force KNN (available since 12.1.2) performs a table scan — every vector is compared. Works for small datasets or queries with predicates that reduce the search space. But interactive workloads (RAG, semantic search, fraud detection) issue multiple queries on millions of vectors — this is impractical.

The Curse of Dimensionality (Garcia-Arellano, Oct 2025):

Exponential Volume

As dimensions increase, the search space grows exponentially. Vectors become sparse even with billions of points.

Distance Degeneracy

In high-dimensional spaces, distances between points converge — hard to distinguish truly similar from distant vectors.

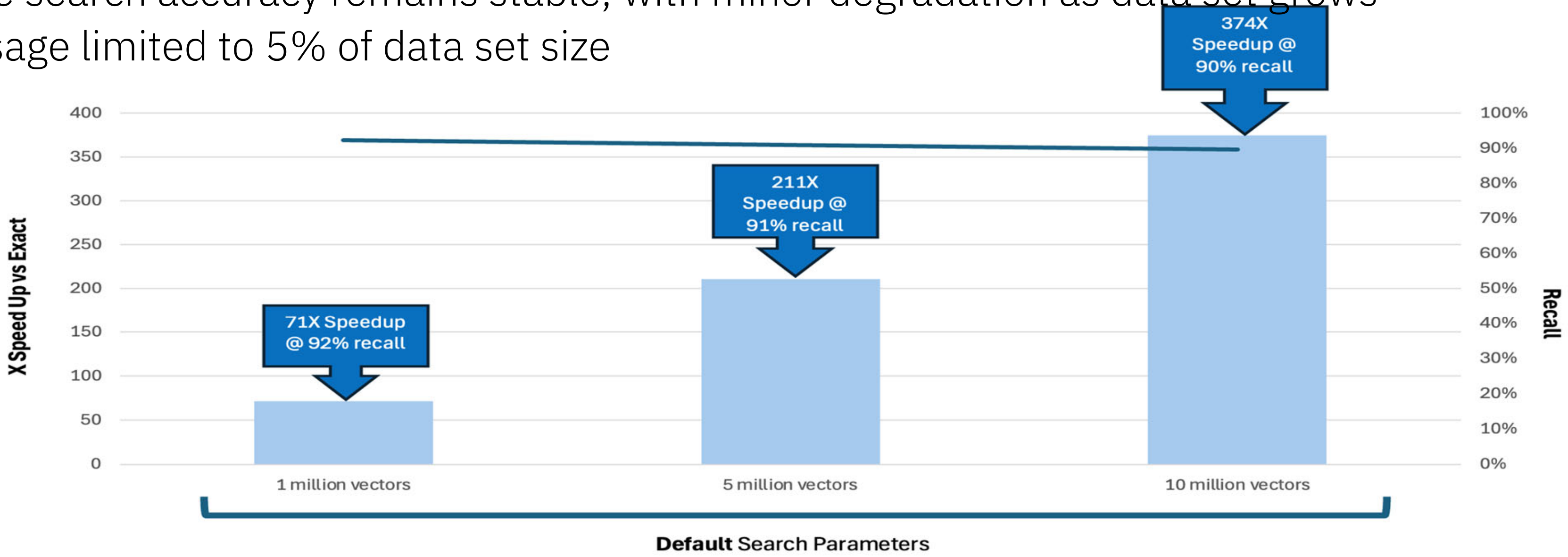
CPU Cost Scales

Computing one VECTOR_DISTANCE call scales linearly with dimension. 768–1500 dims per vector is expensive at millions of rows.

	Exact KNN	ANN (Vector Index)
Accuracy	100%	98%+ recall (tunable)
Complexity	O(N) table scan	Sub-linear graph traversal
Scalability	Poor at 1M+ rows	Handles very large datasets
Db2 Support	GA (12.1.2+)	EAP — planned for future GA

Vector Similarity Search Indexing Insights #1

- Db2 Vector Indexing search performance improves as data set grows
  ✗ Exact search grows linearly with data set size, as it uses brute force approach that needs to scan all data and compute all distances
  ✓ Vector Search grows logarithmically with data set size, as graph navigation absorbs data set growth, reducing the number of random IO operations and distance computations.
- Approximate search accuracy remains stable, with minor degradation as data set grows
- Resource usage limited to 5% of data set size



Entity Customer data set: 1, 5 and 10 million vectors, 768 float32 dimensions, embeddings generated using SLATE-125 model from JSON documents sourced from TPC-DS relational tables

In-database LLM Integration (EAP)

The Feb 2026 EAP enables you to register watsonx.ai (and OpenAI-compatible) models directly inside Db2. Use built-in SQL functions TO_EMBEDDING and TEXT_GENERATION to generate embeddings and LLM responses without leaving the database.

CREATE EXTERNAL MODEL

Register watsonx.ai or OpenAI-compatible models in the Db2 catalog with DDL. Store API key, URL, model ID, and type (TEXT_EMBEDDING or TEXT_GENERATION).

TO_EMBEDDING function

Built-in SQL function. Generate a VECTOR from a VARCHAR input using a registered embedding model — directly in SQL, no Python pipeline needed.

TEXT_GENERATION function

Built-in SQL function. Generate text from a prompt using a registered LLM (e.g., granite). Supports watsonx.ai and OpenAI API spec.

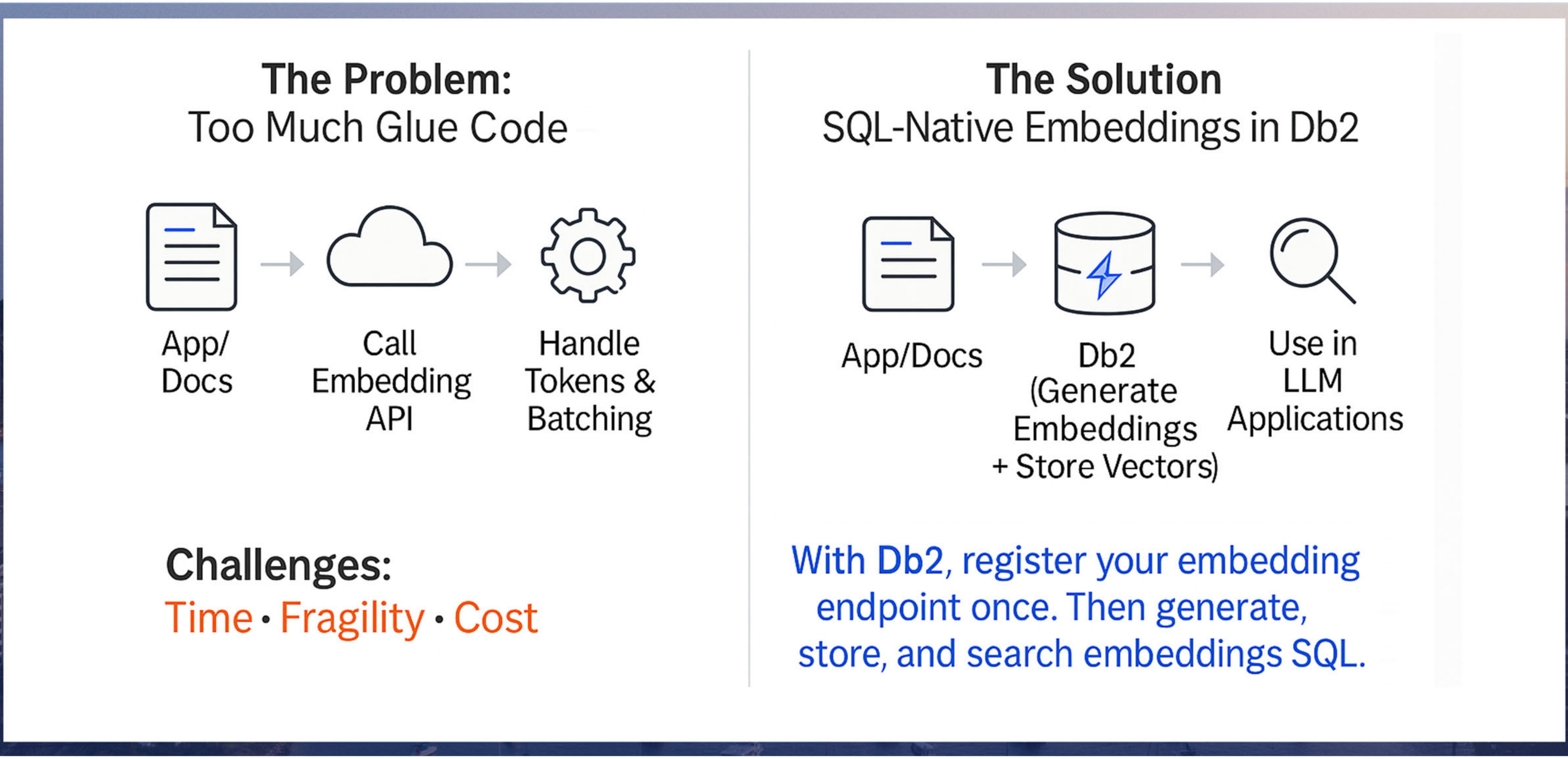
Catalog views

SYSCAT.EXTERNALMODELS, SYSCAT.EXTERNALMODELOPTIONS, SYSCAT.EXTERNALMODELAUTH — track metadata, options, and access privileges.

IDUG
2026 Australia Db2 Tech Conference

SQL-Native Embedding Generation (EAP)

#IDUGDb2



The Problem:
Too Much Glue Code

App/ Docs → Call Embedding API → Handle Tokens & Batching

Challenges:
Time • Fragility • Cost

The Solution
SQL-Native Embeddings in Db2

App/Docs → Db2 (Generate Embeddings + Store Vectors) → Use in LLM Applications

With **Db2**, register your embedding endpoint once. Then generate, store, and search embeddings SQL.

Registering Language Models

SQL

```
-- Register a watsonx.ai embedding model in the Db2 catalog:
CREATE EXTERNAL MODEL aschema.granite_embed
  PROVIDER WATSONX
    KEY        'api-key-xxxx'
    ID         'ibm/slate-30m-english-rtrvr'
    TYPE       TEXT_EMBEDDING RETURNING VECTOR(1024, FLOAT32)
    URL        'https://us-south.ml.cloud.ibm.com/ml/v1/text/embeddings'
    PROJECT_ID ‘YOUR_PROJECT_ID';

-- Register a text generation model (watsonx.ai):
CREATE EXTERNAL MODEL aschema.granite_llm
  PROVIDER WATSONX
    KEY             'api-key-xxxx'
    ID              'ibm/granite-13b-instruct-v2'
    TYPE            TEXT_GENERATION RETURNING VARCHAR(4096)
    URL             'https://us-south.ml.cloud.ibm.com/ml/v1/text/generation'
    PROJECT_ID      ‘YOUR_PROJECT_ID'
    MAX_NEW_TOKENS  512
    TEMPERATURE     0.7;

-- Grant usage to a user:
GRANT USAGE ON EXTERNAL MODEL aschema.granite_embed TO USER LISA;
```

PROVIDER WATSONX

Uses watsonx.ai REST API. Db2 assembles the full endpoint URL per watsonx conventions.

PROVIDER OPENAI

Supports any OpenAI API-compatible provider (private cloud). Feb 2026 EAP extends this to API key auth.

Calling Language Models

SQL

```sql
-- 1. Generate embedding for a new document and insert in one SQL statement:
INSERT INTO documents(id, content, embedding)
VALUES (
  101,
  'Lightweight waterproof trail running shoe',
  TO_EMBEDDING('Lightweight waterproof trail running shoe’ USING aschema.granite_embed)
);

-- 2. Search using a user query embedded inline:
SELECT id, content,
       VECTOR_DISTANCE(
         TO_EMBEDDING('fast running shoe waterproof', aschema.granite_embed),
         embedding, COSINE
       ) AS similarity
FROM   documents
ORDER  BY similarity ASC
FETCH  APPROX FIRST 5 ROWS ONLY;

-- 3. Generate an LLM response directly in SQL:
SELECT TEXT_GENERATION(
  CONTENT USING aschema.granite_llm
) AS summary
FROM documents WHERE id = 101;
```

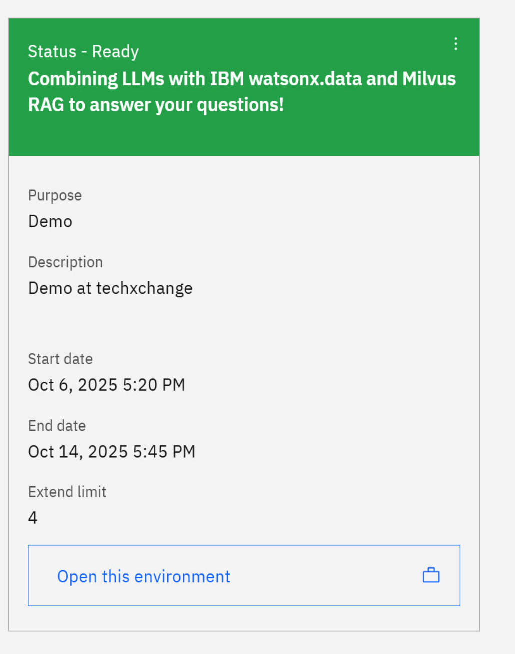
DEMO time

Keep your fingers crossed

Demonstration System Components

Available on IBM Techzone

- Watsonx.data 2.1.0 - IBM Data Lakehouse environment
- Presto 0.286 - Database engine used to query data in the lakehouse
- Milvus - Vector database included in watsonx.data
- Ollama - Platform for running LLMs locally
- Streamlit - Web interface framework
- Llama Index - Data framework for building LLM applications
- pyMilvus - Python SDK of Milvus
- prestodb - Presto client
- langchain - LangChain is a framework designed to simplify LLM applications
- sqlalchemy - SQLAlchemy is an Object Relational Mapper for database interactions
- sentence_transformer - Sentence Transformers provides modules for accessing, using, and training embedded models
- pandas - pandas provides data structures designed to work with relational or tabular data



Status - Ready

Combining LLMs with IBM watsonx.data and Milvus RAG to answer your questions!

Purpose
Demo

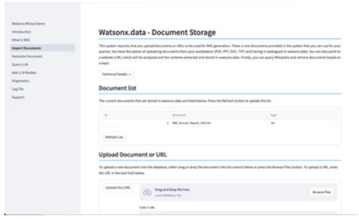
Description
Demo at techxchange

Start date
Oct 6, 2025 5:20 PM

End date
Oct 14, 2025 5:45 PM

Extend limit
4

System Architecture



Watsonx.data - Document Storage

This system requires that you upload documents or URLs to be used for RAG generation. There is one documents provided in the system that you can use for your queries. You have the option of uploading documents from your workstation (PDF, PPT, DOC, TXT) and having it catalogued in watsonx.data. You can also point to a website (URL) which will be analyzed and the contents extracted and stored in watsonx.data. Finally, you can query Wikipedia and retrieve documents based on a topic.

Technical Details ⌄

Document list

The current documents that are stored in watsonx.data are listed below. Press the Refresh button to update the list.

Id	Document	Type
1	IBM_Annual_Report_2023.txt	txt

Refresh List

Upload Document or URL

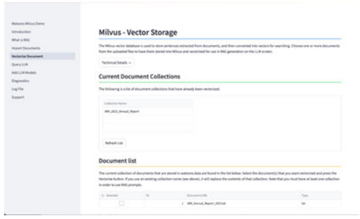
To upload a new document into the database, either drag or drop the document into the control below or press the Browse Files button. To upload a URL, enter the URL in the text field below.

Upload Doc/URL

Drag and drop file here
Limit 200MB per file

Browse Files

Enter a URL



Milvus - Vector Storage

The Milvus vector database is used to store sentences extracted from documents, and then converted into vectors for searching. Choose one or more documents from the uploaded files to have them stored into Milvus and vectorized for use in RAG generation on the LLM screen.

Technical Details ⌄

Current Document Collections

The following is a list of document collections that have already been vectorized.

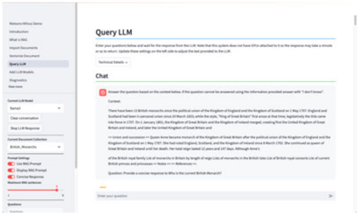
Collection Name
IBM_2023_Annual_Report

Refresh List

Document list

The current collection of documents that are stored in watsonx.data are found in the list below. Select the document(s) that you want vectorized and press the Vectorize button. If you use an existing collection name (see above), it will replace the contents of that collection. Note that you must have at least one collection in order to use RAG prompts.

Selected	Id	Document/URL	Type
☐	1	IBM_Annual_Report_2023.txt	txt



Current LLM Model: llama3

Clear conversation

Stop LLM Response

Current Document Collection: British_Monarchs

Prompt Settings
- Use RAG Prompt
- Display RAG Prompt
- Concise Response

Maximum RAG sentences: 1 – 9

Questions

Query LLM

Enter your questions below and wait for the response from the LLM. Note that this system does not have GPUs attached to it so the response may take a minute or so to return. Update these settings on the left side to adjust the text provided to the LLM.

Technical Details ⌄

Chat

Answer the question based on the context below. If the question cannot be answered using the information provided answer with "I don't know".

Context:

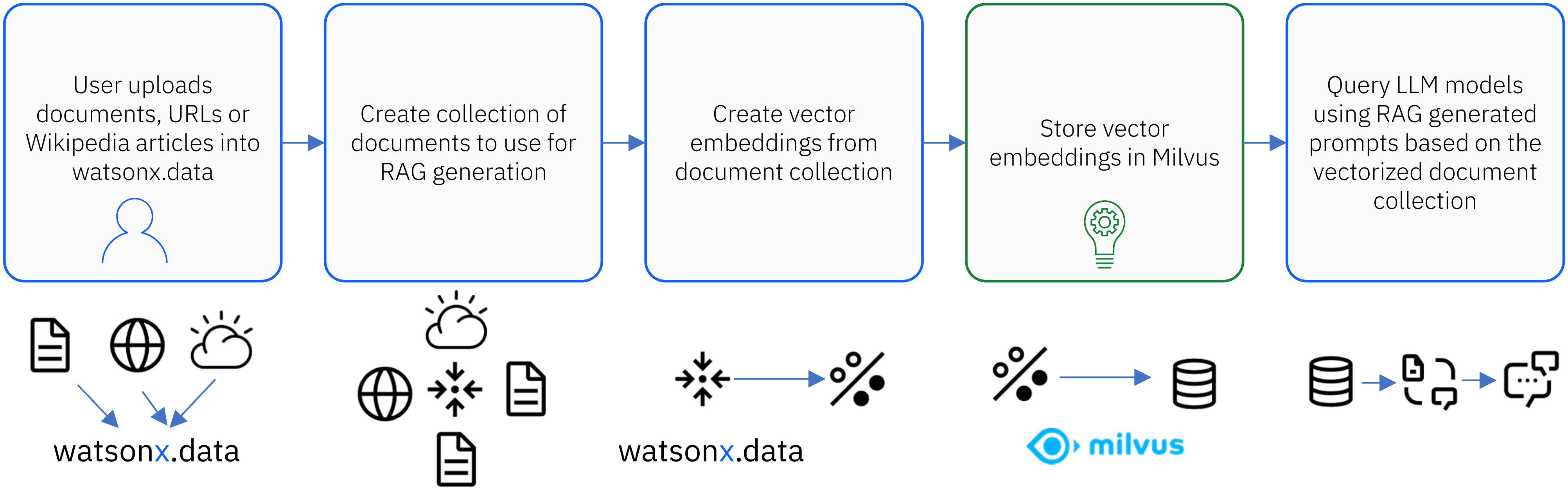
There have been 13 British monarchs since the political union of the Kingdom of England and the Kingdom of Scotland on 1 May 1707. England and Scotland had been in personal union since 24 March 1603, while the style, "King of Great Britain" first arose at that time, legislatively the title came into force in 1707. On 1 January 1801, the Kingdom of Great Britain and the Kingdom of Ireland merged, creating first the United Kingdom of Great Britain and Ireland, and later the United Kingdom of Great Britain and

== Union and succession == Queen Anne became monarch of the Kingdom of Great Britain after the political union of the Kingdom of England and the Kingdom of Scotland on 1 May 1707. She had ruled England, Scotland, and the Kingdom of Ireland since 8 March 1702. She continued as queen of Great Britain and Ireland until her death. Her total reign lasted 12 years and 147 days. Although Anne's

of the British royal family List of monarchs in Britain by length of reign Lists of monarchs in the British Isles List of British royal consorts List of current British princes and princesses == Notes == == References ==.

Question: Provide a concise response to Who is the current British Monarch?

Enter your question



User uploads documents, URLs or Wikipedia articles into watsonx.data

Create collection of documents to use for RAG generation

Create vector embeddings from document collection

Store vector embeddings in Milvus

Query LLM models using RAG generated prompts based on the vectorized document collection

LLM	TRAINING DAY
Gemma:2b	Unavailable
Instructlab/granite-7b-lab	June 2024
Instructlab/merlinite-7b-lab	May 2022
Llama3	December 2023
llama3.2:1b-instruct-q8_0	March 2023
Mistral-small	March 2023
phi:3.8b-mini-128k-instruct-q8_0	October 2023
tinyllama:1.1b-chat-v1-q8_0	September 2023

Ibm.biz/learndb2ai

ibm.biz/BuildingLLMAppsWithDb2

IDUG

2026

Sydney | March 16 - 18

AU Db2 TECH CONFERENCE

Db2 AI Strategy Update for Developers

Dale McInnis, *IBM*

Contact: dmcinnis@ca.ibm.com

Session Code: B05

Platform:

Db2 LUW

IDUG

2026 Australia **Db2** Tech Conference