

Transforming Domino Applications with LLMs: DominoIQ and Beyond

Engage 2025 – The Hague





Today...

- A very short introduction to LLMs
- Some basics
- HCL Domino IQ
- The Bigger Picture
- The Knowledge Gap: What LLMs don't know
- Retrieval-augmented generation
- Langchain4j Domino Project
- Emerging concepts in the LLM World
- Conclusion and Q&A



Serdar Basegmez

- Developer/Half-blooded Admin
- New(ish) Londoner - Ex-Istanbulite
- Freelancer at Developi UK
- Member Director at OpenNTF Board
- Notes/Domino since 1999
- IBM Champion Alumni (2011-2018)
- HCL Ambassador (2020-2025)
- Blog: LotusNotus.com / Bluesky: @serdar.uk
- Also tweets/writes/speaks/podcasts on scientific skepticism





Everything Open Source!

Demos, source codes, libraries, integrations, datasets...



 <https://github.com/sbasegmez>

OpenNTF Project Metadata v1 2024-11-...			
OpenNTF Project Metadata v1			
	Name	Created	Chiefs-Cooks
	Audit Manager	10/10/2006 11:58:53	Neil Gower,Ulrich Krause,Per Henrik Lausten
	AuthServer	18/12/2010 23:51:21	Stan I. Uptagrafft
	Auto Logins for IBM Domino/ XWork server	24/06/2012 12:46:44	Mark Leusink
	Automated Communication	05/03/2009 10:07:53	Davy Carmanis
	Automated Dead Mail Deletion App	05/03/2014 06:20:52	Shankar Venkatachalam1,Niklas Heidloff
	Automation - SCN Hired Custom Group Limitation	10/09/2014 06:27:13	Shankar Venkatachalam1,Niklas Heidloff

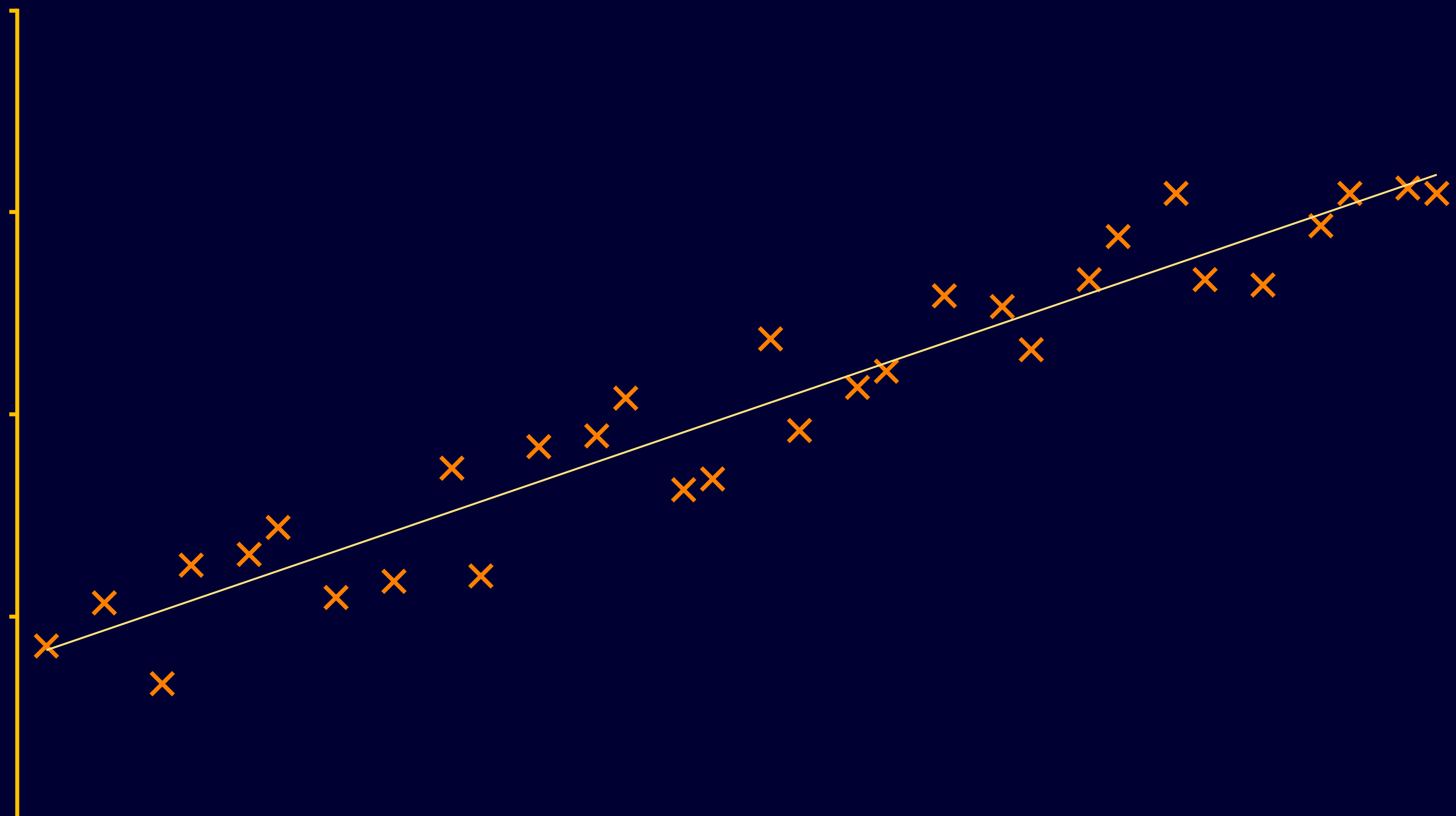
form	project	
sourceDbPath	projects/pmt.nsf	source database in Server/FilePath format
sourceUnid	6B42E3F3F57BF4E986257A130040B415	source document unid
sourceUrl	https://www.openntf.org/main.nsf/project.xsp?r=project/Auto+Logins+for+IBM+Domino%2F+XWork+server	uri pointing the original project
importedOn	18/11/2024 12:30:34	
importedBy	CN=Serdar Basegmez/O=developi	

id	6B42E3F3F57BF4E986257A130040B415	Original id (unid)
name	Auto Logins for IBM Domino/ XWork server	"ProjectName"
overview	"Remember me" option for IBM Domino/ XWork server logins	"ProjectOverview": Short summary of the project
details	You've reached the project site of the 3rd runner up in the 3rd OpenNTF Development Contest! Visit my blog to view my other projects or read other interesting articles. Auto Logins for IBM Domino/ XWork server is an application/ component that you can use to add a "Remember me" feature to the login screen of an IBM Domino/ Xwork server. If a user (within the specified number of days) tries to access the server again, he is recognized (based on a token in a cookie). If the token is valid and hasn't expired, the user is automatically signed in.	"Details": Long explanation (rt/mime)



Large Language Models

A very short introduction...

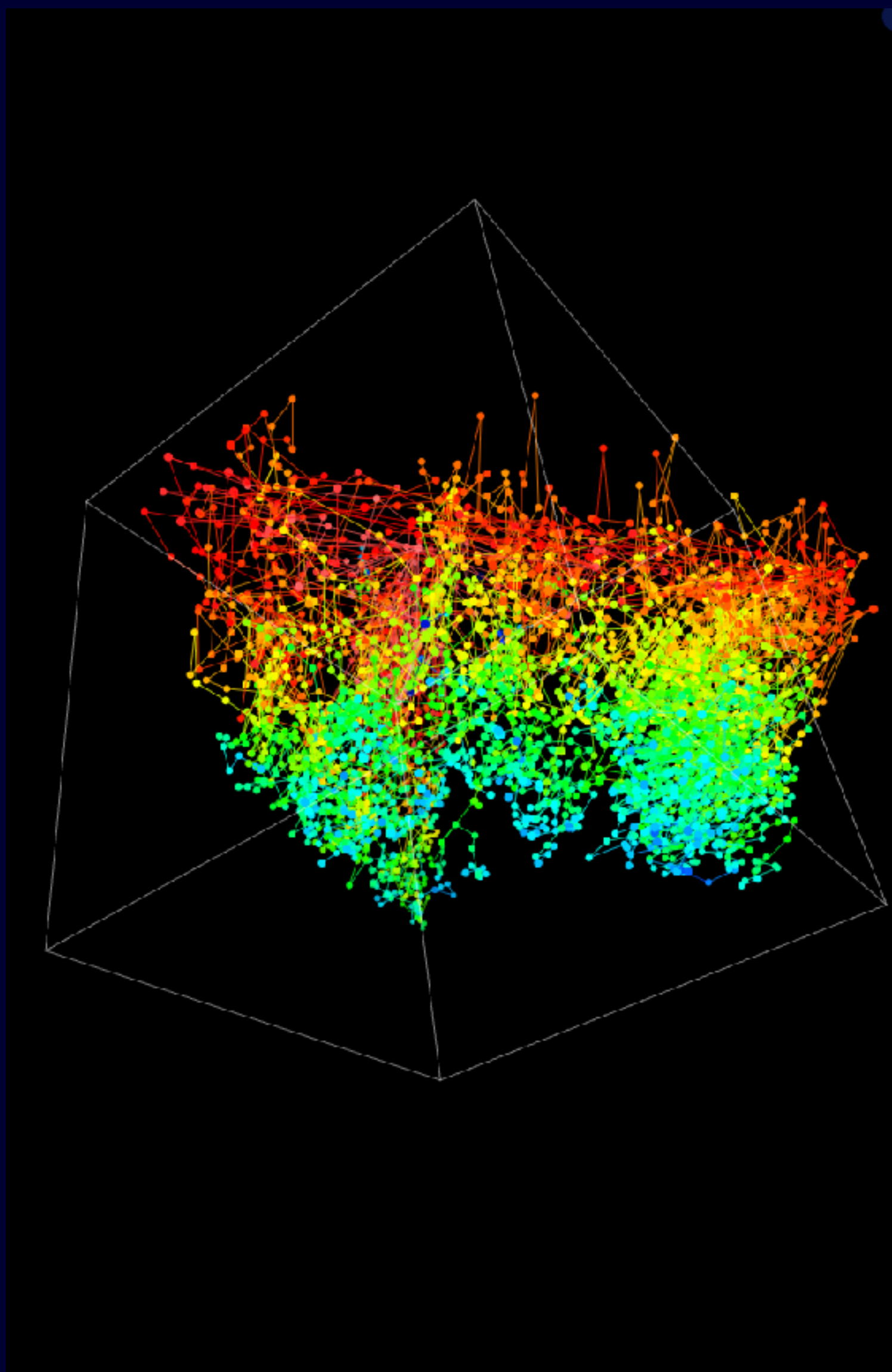


- <https://www.cs.cmu.edu/~dst/WordEmbeddingDemo/tutorial.html>





LLM Under the Hood



Intelligence

is

the ability to

a

defined as

is

generally

fundamentally

typically

the

Intelligence

often

not

capacity

possession

cognitive

multifaceted

complex

the

defined

the

defined

the

is

regarded

as

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

learn,

acquire,

be

grasp

analyze,

think,

understand

comprehend,

use

gather

discover

for

an

individual

to

acquire,

understand,

and

apply

knowledge,

and

skills

It

learn,

acquire,

be

THIS IS YOUR MACHINE LEARNING SYSTEM?

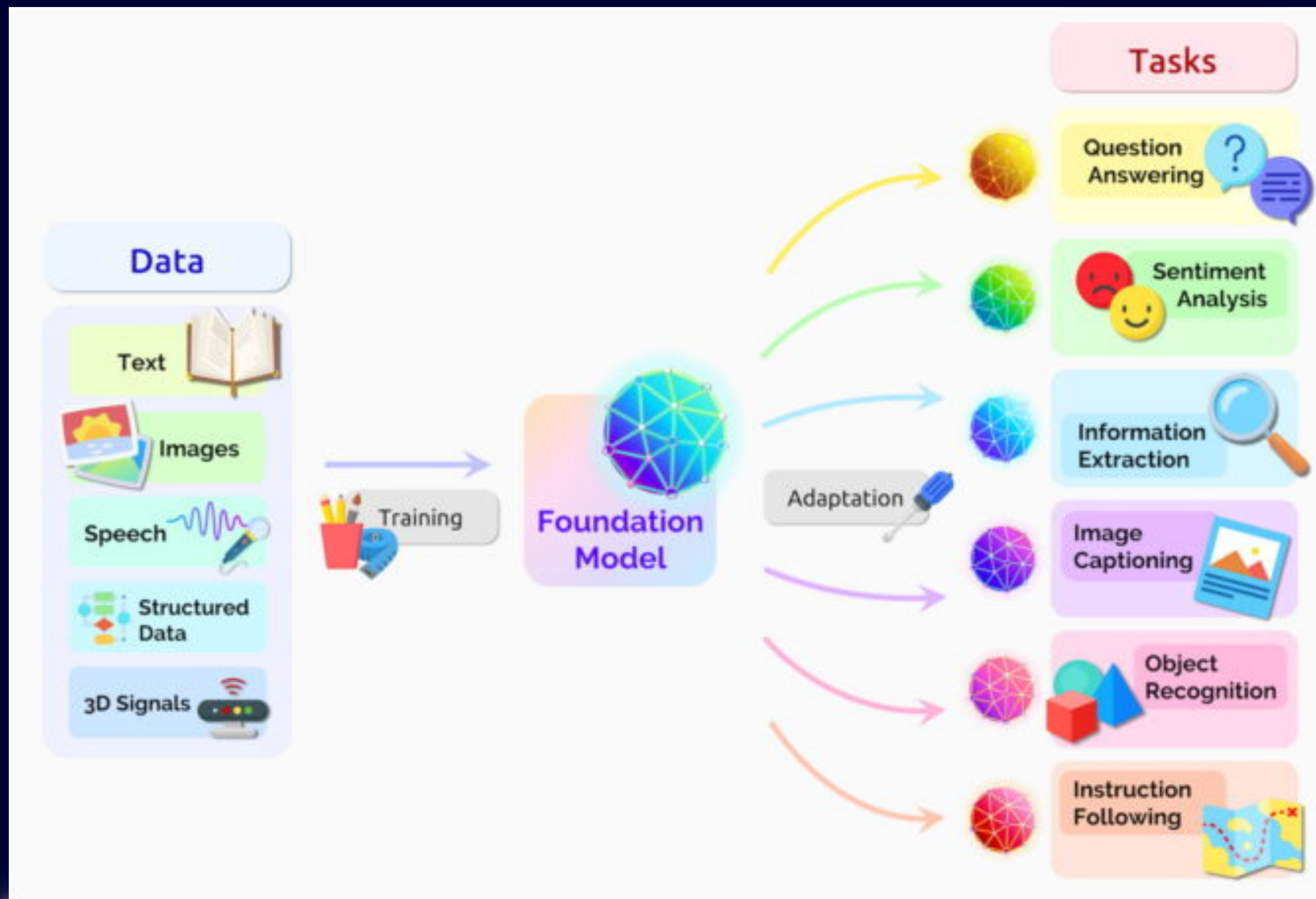
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.

WHAT IF THE ANSWERS ARE WRONG?

JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.



Large Language Models





Major Large Language Models (LLMs)

ranked by capabilities, sized by billion parameters used for training

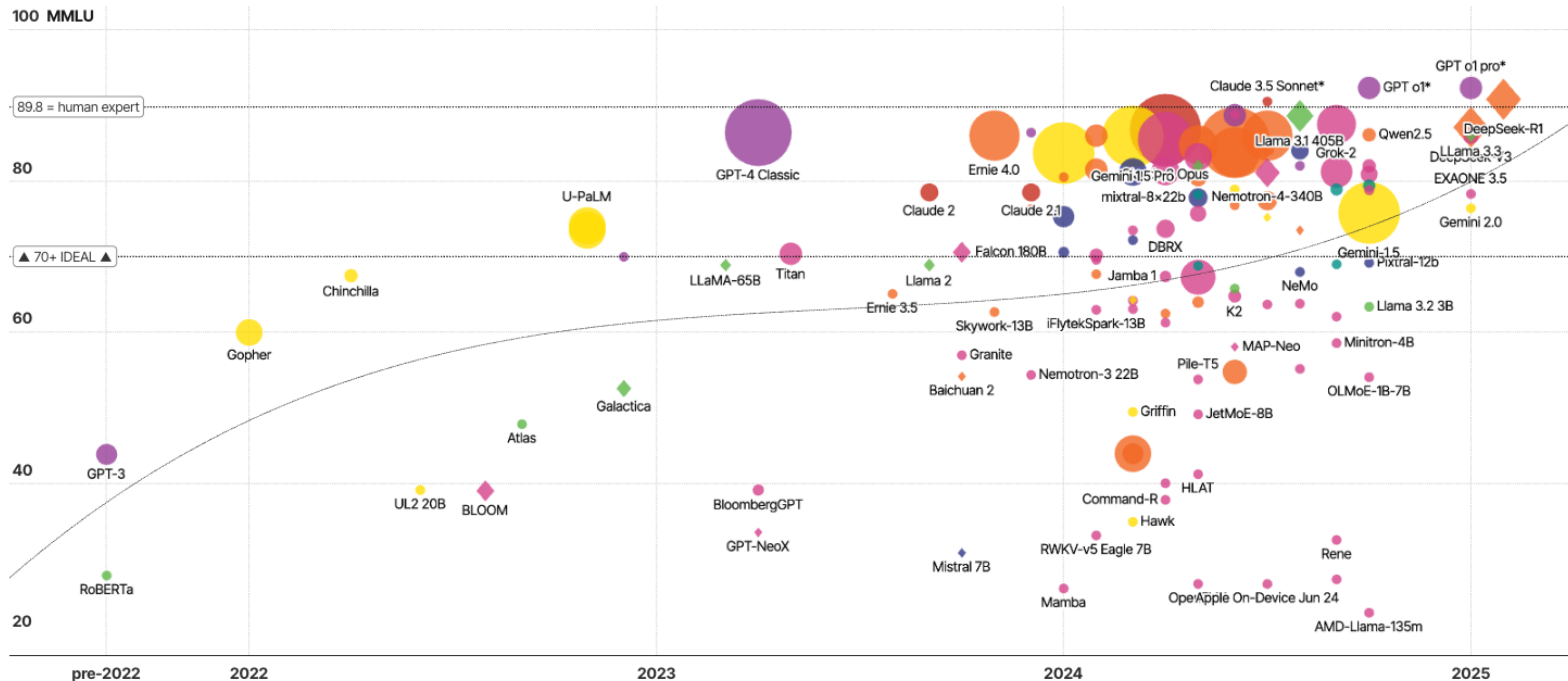
CLICK LEGEND ITEMS TO FILTER

anthropic chinese google meta microsoft mistral openAI other

Parameters (Bn) open access

search...

show only: all



David McCandless, Tom Evans, Paul Barton
Informationisbeautiful // Jan 2024

MMLU = benchmark for measuring LLM capabilities
* = parameters undisclosed // source: [LifeArchitect](#) // [data](#)

MADE WITH VIZsweet

MADE WITH VIZsweet



LLMs Make Text Tasks Easier

- Summarisation / Simplification
- Sentiment analysis
- Chatbots / Conversational AI
- Classification / Categorisation
- Semantic Search
- Speech recognition
- Recommendation
- Content Generation
- Text-to-speech synthesis
- Spell/Grammar correction
- Translation
- Fraud detection
- Code generation
- AI Agents

Picking the Right Model

Right tool for the right job...





Model Options

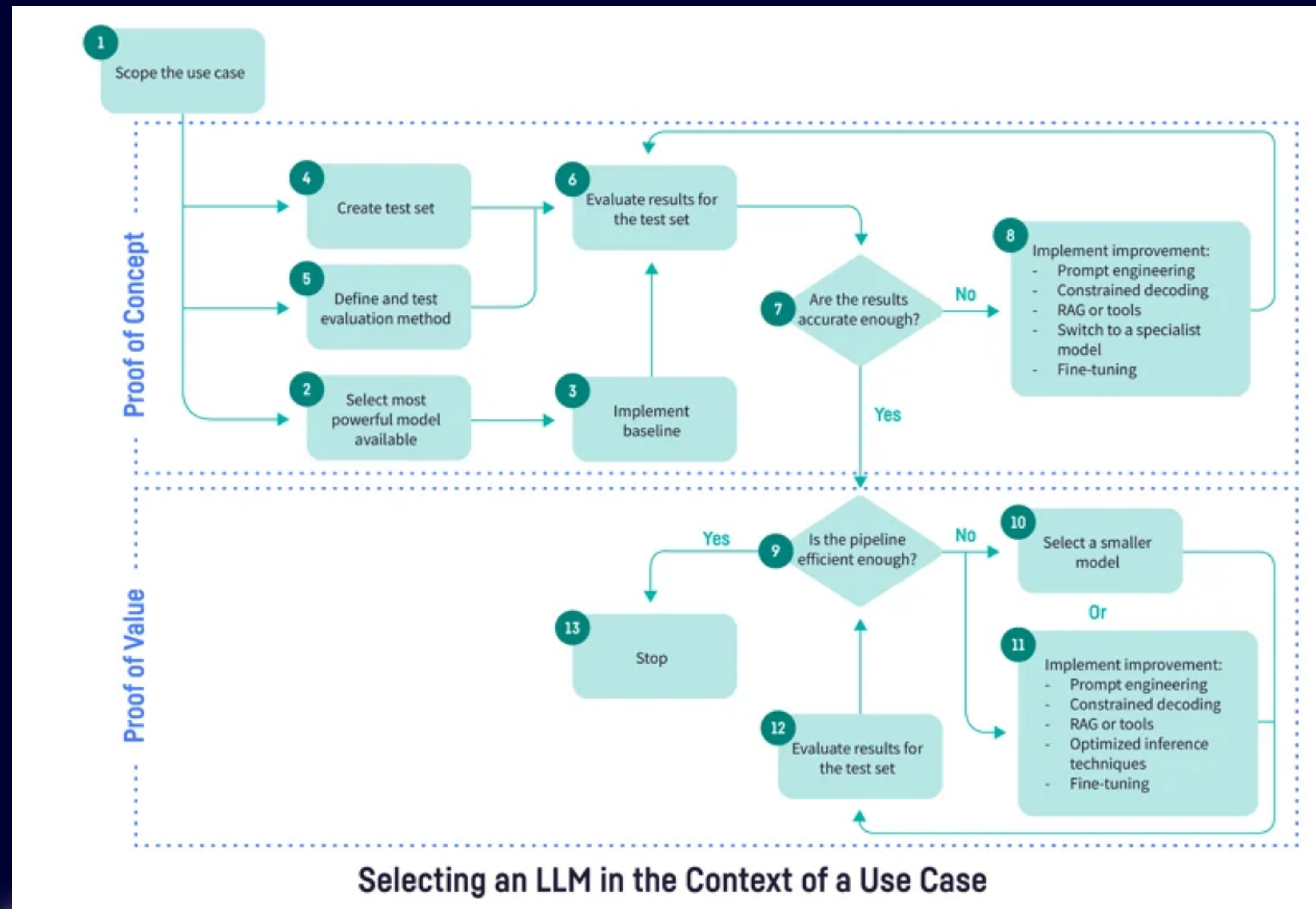
Local LLM (Model Runners) (e.g. Domino IQ, Ollama...)

- ✓ Data won't leave the server
 - ✓ Most are free with permissive licenses
 - ✓ No vendor lock-in
 - ✓ No cost per operation
 - ✓ Model Runners can be flexible
-
- ✗ Model files are huge.
 - ✗ LLM tasks are resource-intensive
 - ✗ Maintenance can be a hassle
 - ✗ Less capable models

Cloud LLM (API-based) (e.g. OpenAI, Vertex AI...)

- ✓ Managed services
 - ✓ Pay-per-use model
 - ✓ Scalable / Available
 - ✓ High performance / High quality
 - ✓ Much better in complicated tasks
-
- ✗ Privacy and security concerns
 - ✗ Network latency
 - ✗ High costs for very busy systems
 - ✗ Vendor lock-in

Decide and Test the Model



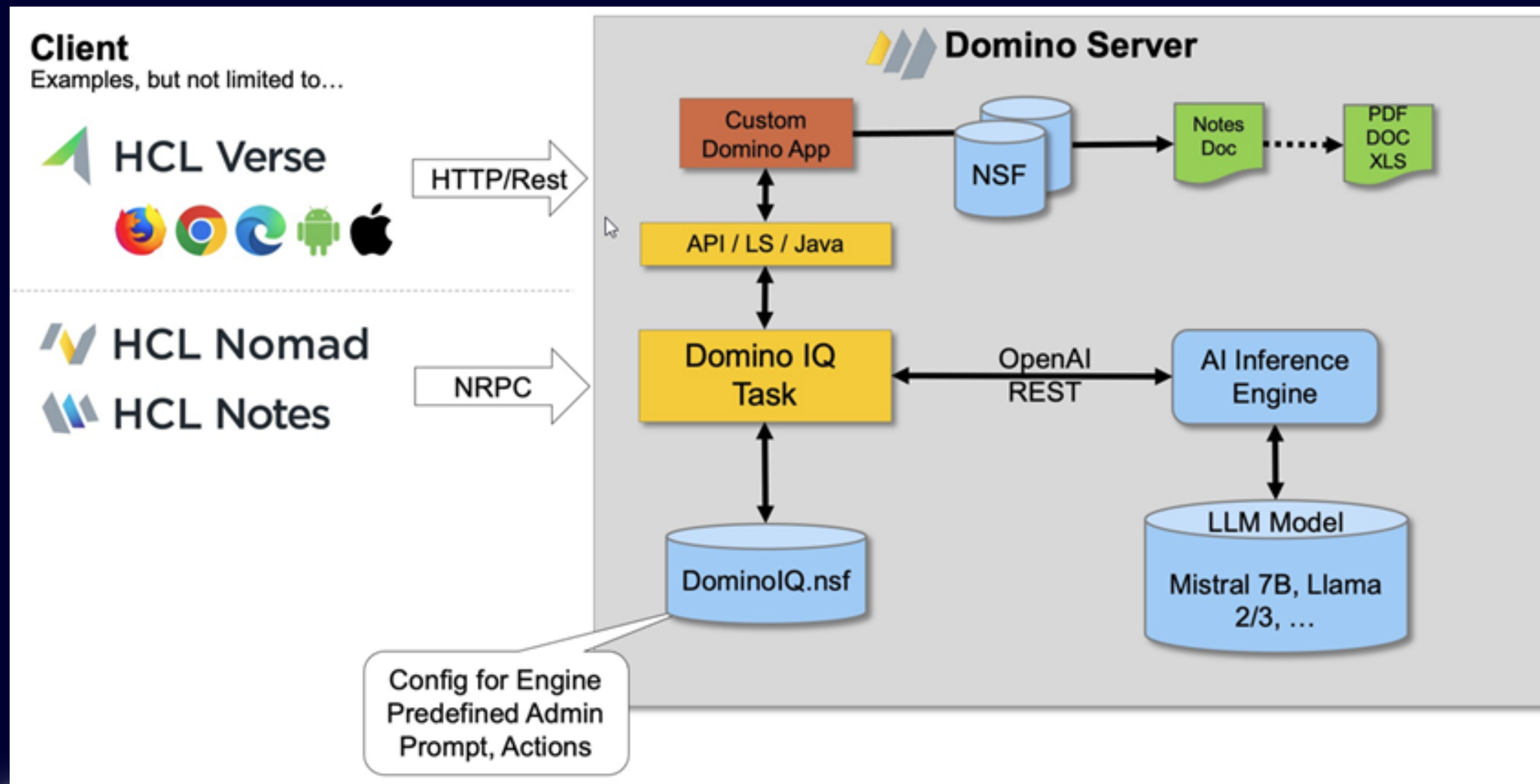
Selecting an LLM in the Context of a Use Case



HCL Domino IQ

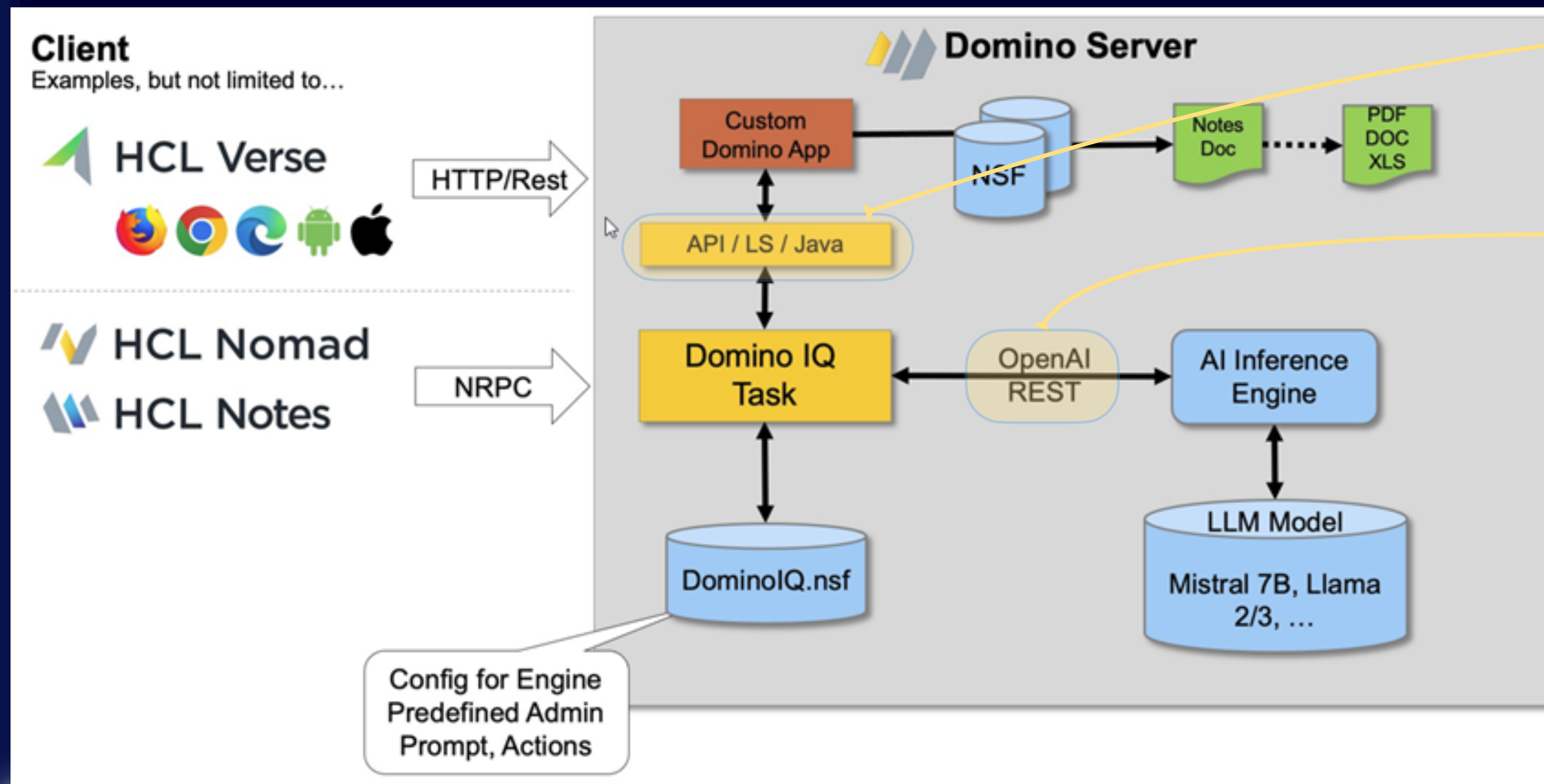
Domino Server meets Large Language Models

HCL Domino IQ: GA at Domino 14.5





HCL Domino IQ: Native Access

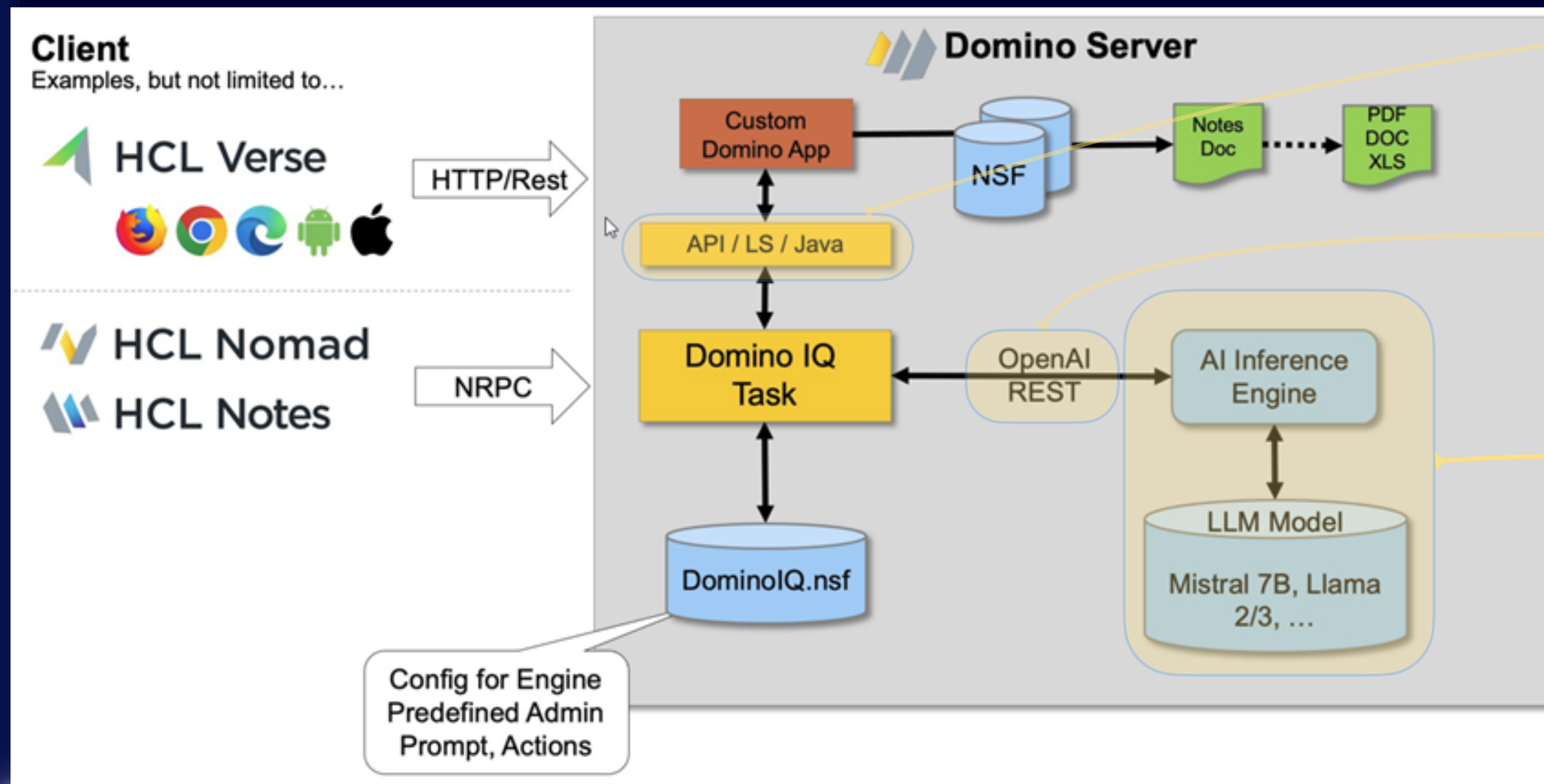


New LS/Java Bindings
NotesLLMRequest / NotesLLMResponse

External inference engine
via native DB Threads



HCL Domino IQ: Inference Engines



New LS/Java Bindings
NotesLLMRequest / NotesLLMResponse

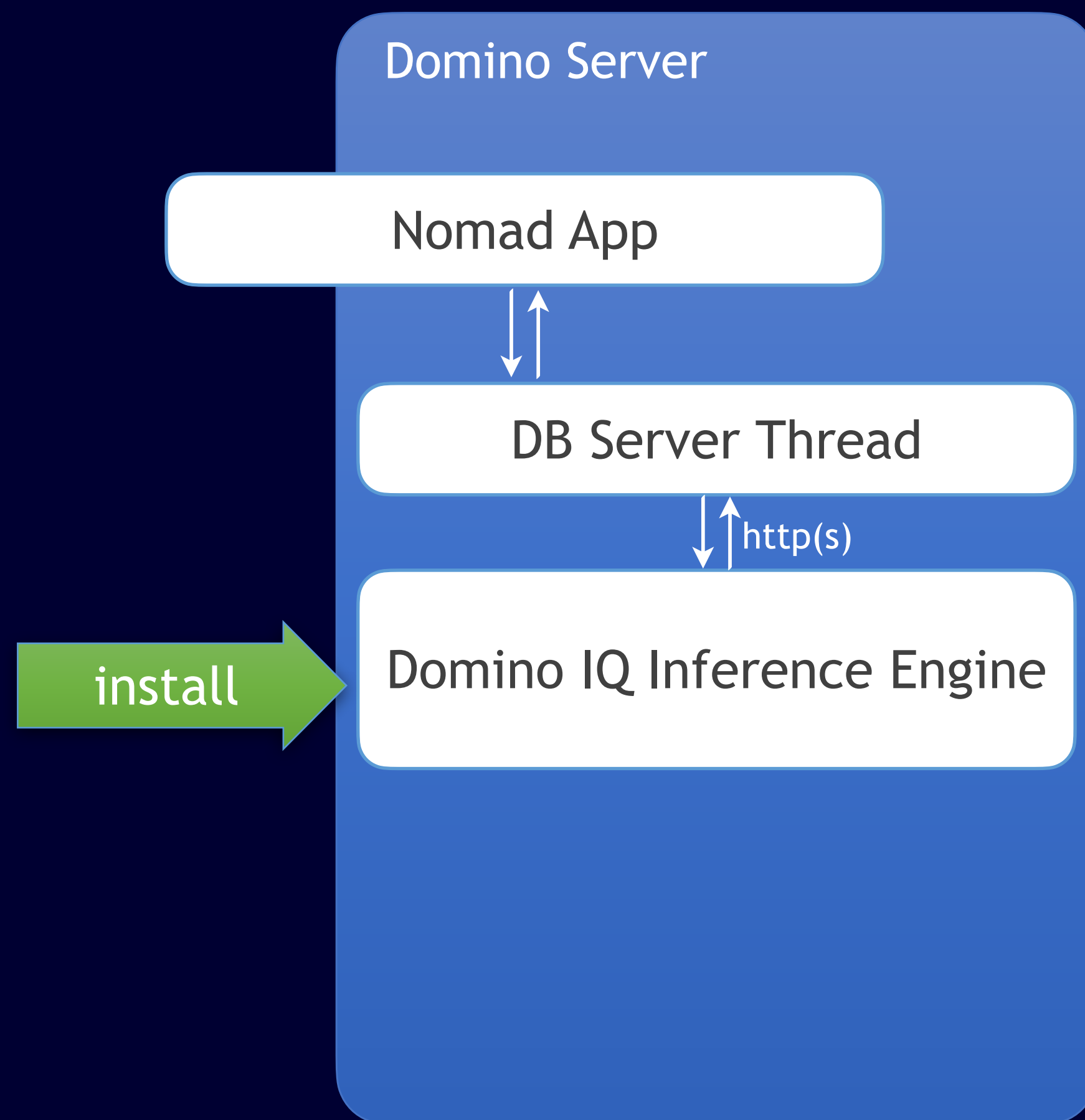
External inference engine
via native DB Threads

Domino IQ Inference Engine
Separate Install

Prompt, Actions
Predefined Admin
Config for Engine

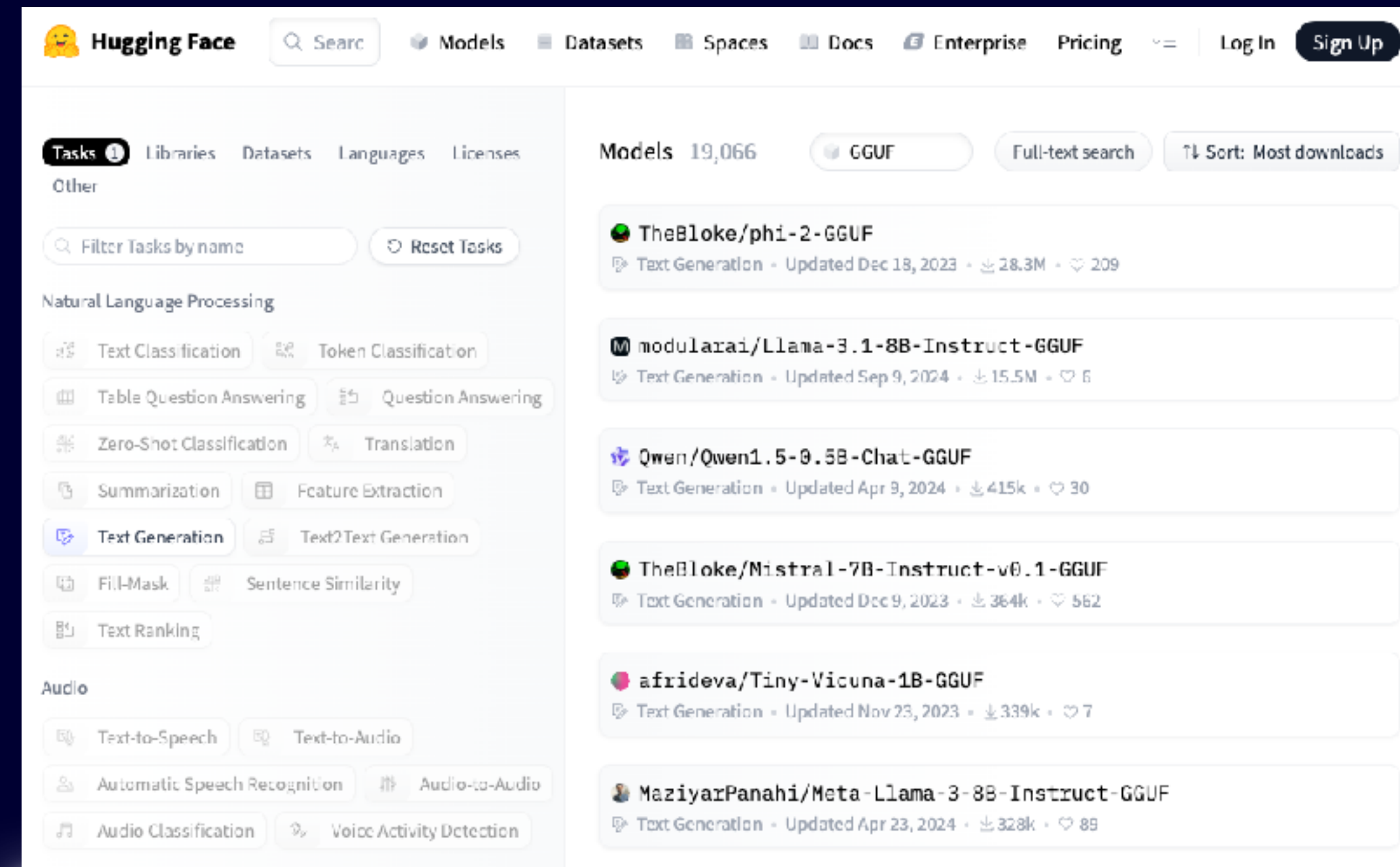
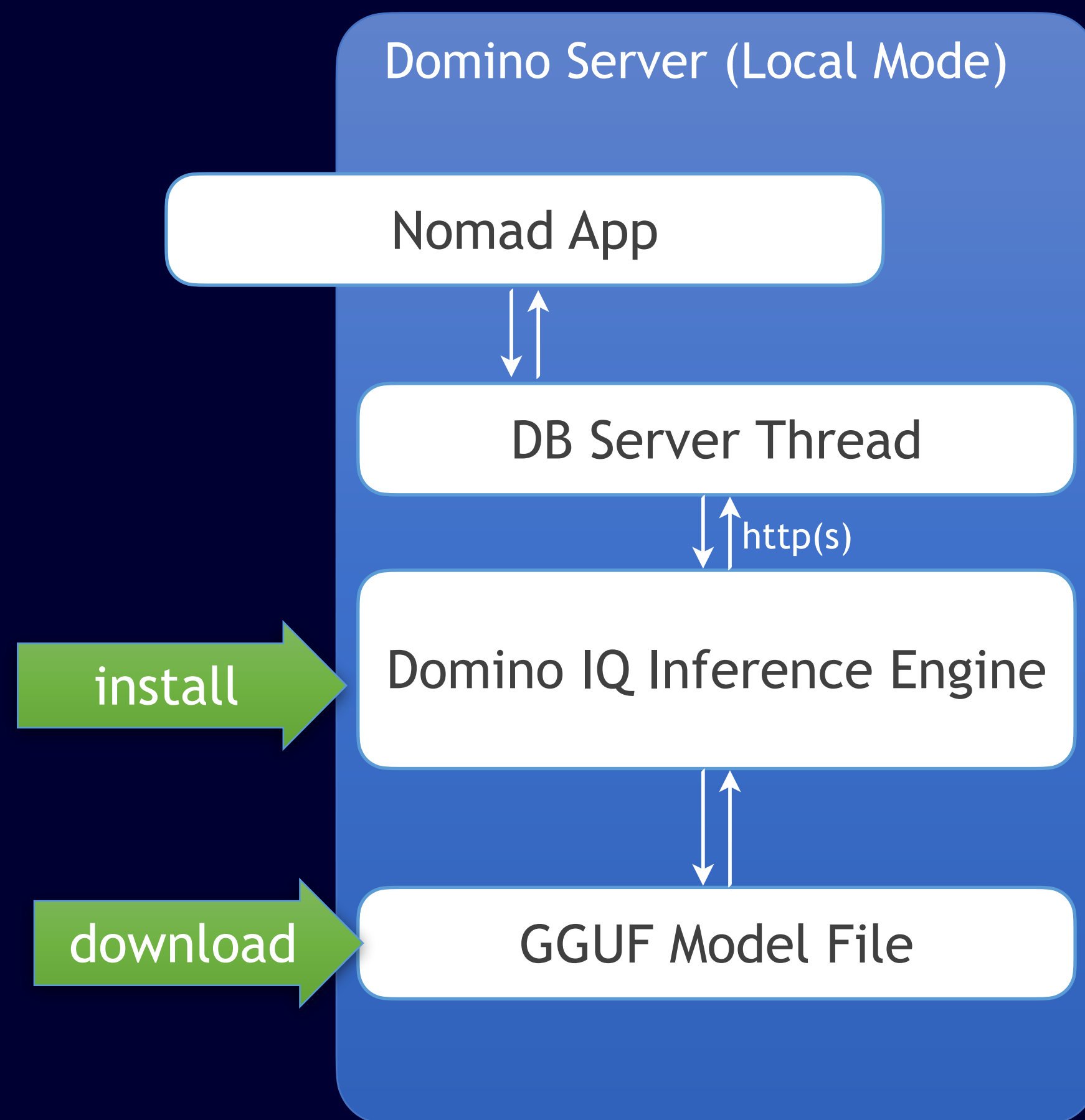


Domino IQ Inference Engine





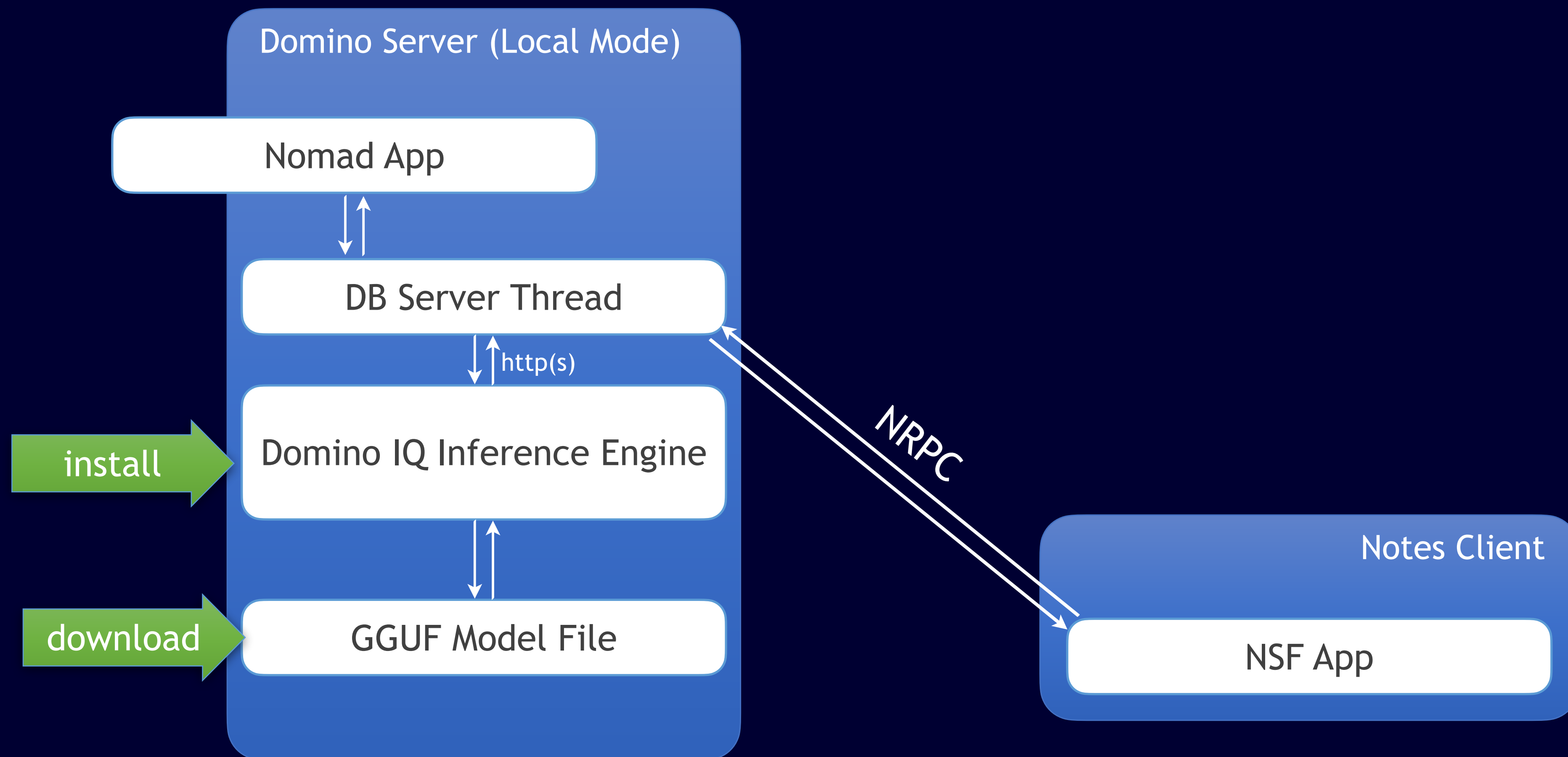
Domino IQ Inference Engine



Requires GPU!

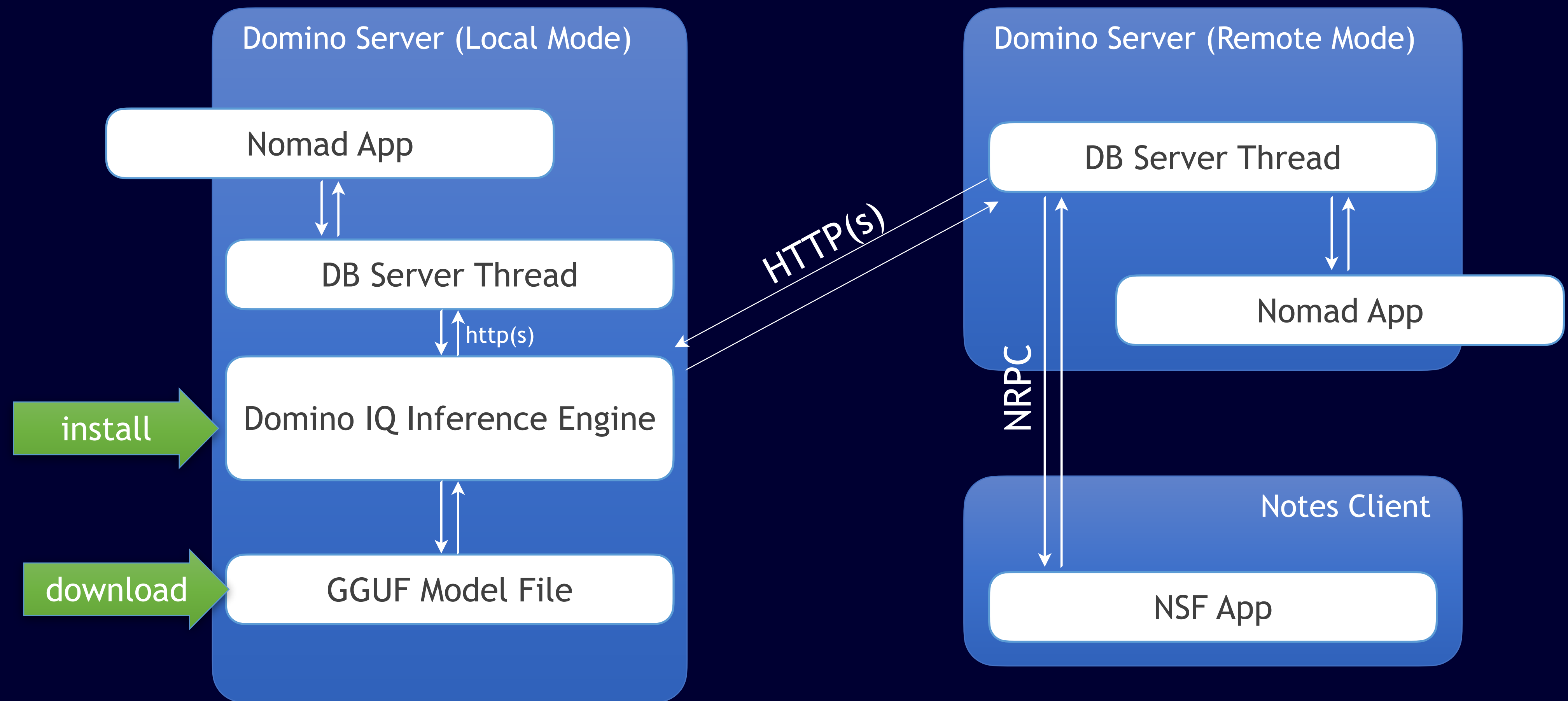


Domino IQ Inference Engine





Domino IQ Whole Picture



External inference engine *via native DB Threads*

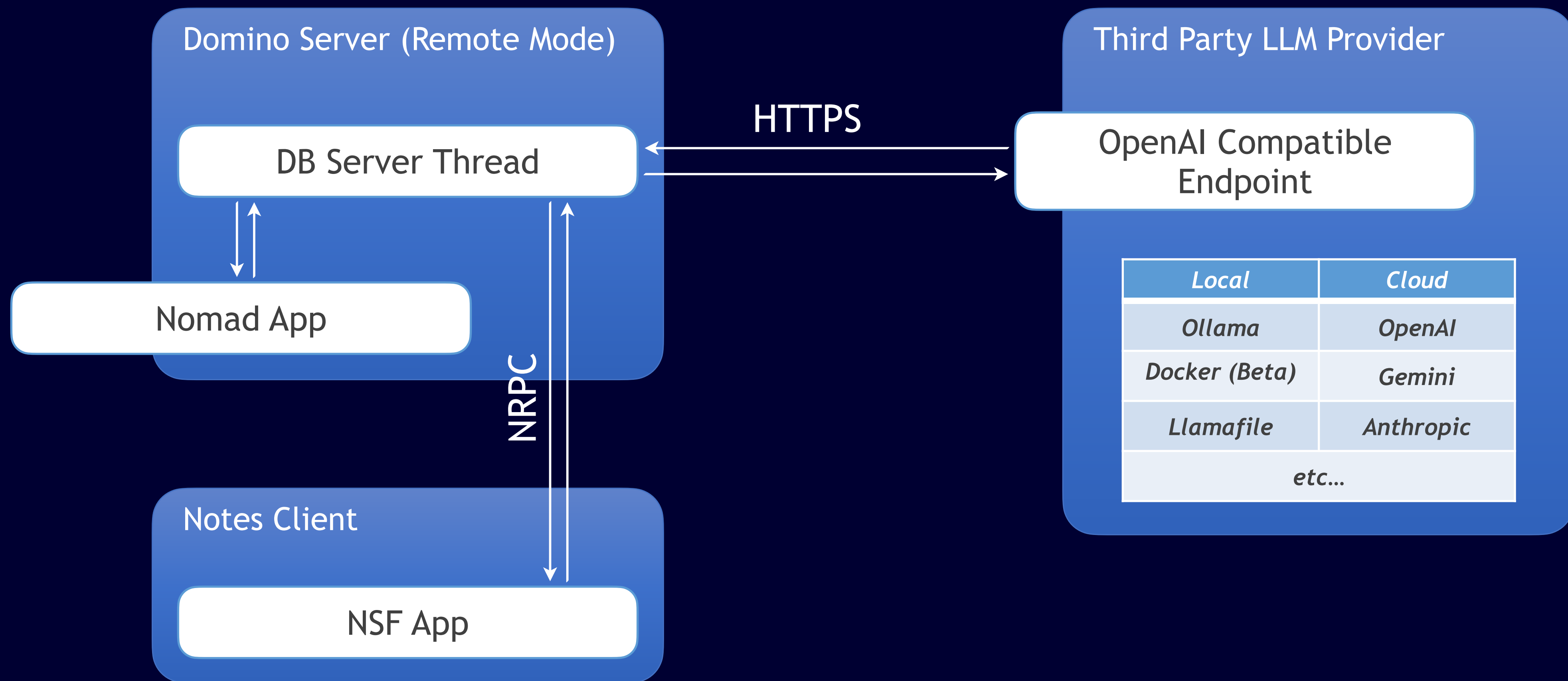
Domino IQ Inference Engine

OpenAI-Compatible API

OpenAI, Ollama server, etc.



HCL Domino IQ: Third Party LLM





Demo

Domino IQ on Notes Client



Domino IQ Demo

Generate Text with GPT4.1

Generate Text with GPT 4.1 mini

Project Name

NSF ODP Tooling

Summary

The NSF ODP Tooling project provides tools for managing NSF on-disk-projects using Maven and Eclipse, including a Maven plugin, Domino OSGi plugins, and Eclipse plugins. The project includes an ODP compiler to build NSFs without Domino Designer, an exporter to convert NSFs to PDF format, Eclipse tooling for autocompletion and project actions, and a deployment server to deploy NSFs to Domino servers. The tooling supports three modes of operation: remote (using a Domino server), containerized (using Docker), and local (using a local Notes or Domino installation). Additional features include an XSP transpiler for converting XPages and XSP Controls into Java source files, with some limitations. The project requires Maven 3.0 and specific versions of Domino or Notes depending on the operation mode, and it is licensed under Apache License 2.0.

Installation Steps

Installation steps:

1. Add the NSF ODP Maven plugin to your Maven configuration.
2. Configure your Maven 'settings.xml' with proxy information for remote execution.
3. For remote mode, install the Domino plugins.
4. For containerized mode, set up a Docker environment.
5. For local mode, point the plugin to your local Domino installation.

► Project Details

```
If mini Then
    cmdSuffix = "_mini"
Else
    cmdSuffix = ""
End If

Set uidoc = ws.Currentdocument
Set doc = uidoc.Document

Set llmreq = session.Createllmrequest()

commandName = "summarize_document" + cmdSuffix

Set llmresp = llmreq.Completion("moe/developi", commandName, getText(doc, "details"))

Call doc.ReplaceItemvalue("AutoSummary", Trim(StrLeft(llmresp.Content, "*****")))
Call doc.ReplaceItemvalue("AutoSteps", Trim(StrRight(llmresp.Content, "*****")))
```

Command Definition

Last Updated: Sun 11/05/2025 17:16:58

Display name: SummarizeAndWriteSteps

You are going to receive a detailed description for an OpenNTF Project. First, provide 5-sentence as a summary of the given text in plain British English. Then add a line with Five "*****" characters and installation steps in a simple form.

Here is an example for the format between two "-----" lines.

Prompt:

```
-----
<summary in five sentences>
*****
- Step 1 description
- Step 2 description
-----
```

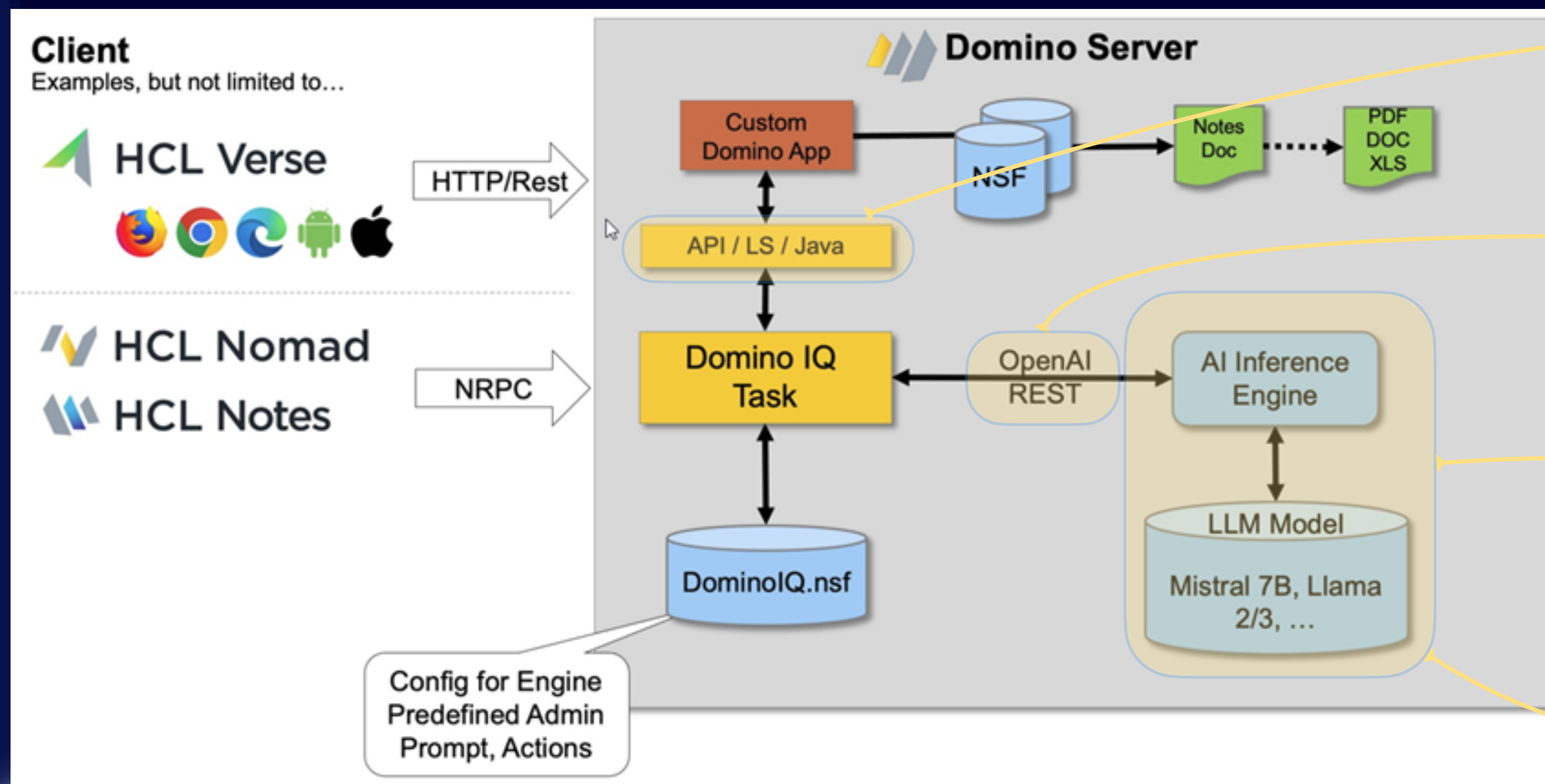


Domino IQ: What's the Catch?

- Only for **Chat Completion** for now... *(Watch this space!)*
- Number of Models to use
 - Local Mode can use a single model only
 - Remote Mode can use **multiple models** (only a single Remote)
- Dev/Testing platform issues
 - Local Mode requires a **GPU**
 - TLS is mandatory on remote mode
 - Use Remote mode with Ollama + Nginx *(See EA3 Forum)*



HCL Domino IQ: GA at Domino 14.5



New LS/Java Bindings

NotesLLMRequest / NotesLLMResponse

External inference engine

via native DB Threads

Domino IQ Inference Engine

Separate Install

OpenAI-Compatible API

OpenAI, Ollama server, etc.



The Knowledge Gap of LLMs

“You know nothing, Jon Snow”



The Knowledge Gap of LLMs

- LLMs lack access to **up-to-date** or **private data**
- Even with what they know, it can get **lost in the noise**
 - e.g. Make up the name and, make mistakes
- So LLMs confidently generate **incorrect** or **missing** details
 - It's not lying, just guessing

What ???

OpenNTF (Open Network for Transformation and Freedom) is a collaborative platform primarily focused on IBM and HCL technologies, offering community-driven applications and tools. If you are looking for an app to handle logging, here are a few OpenNTF solutions that might suit your needs.

1. XPages Log File Reader

- **Purpose:** This application allows developers and administrators to view log data through a web-based interface.

• Features:

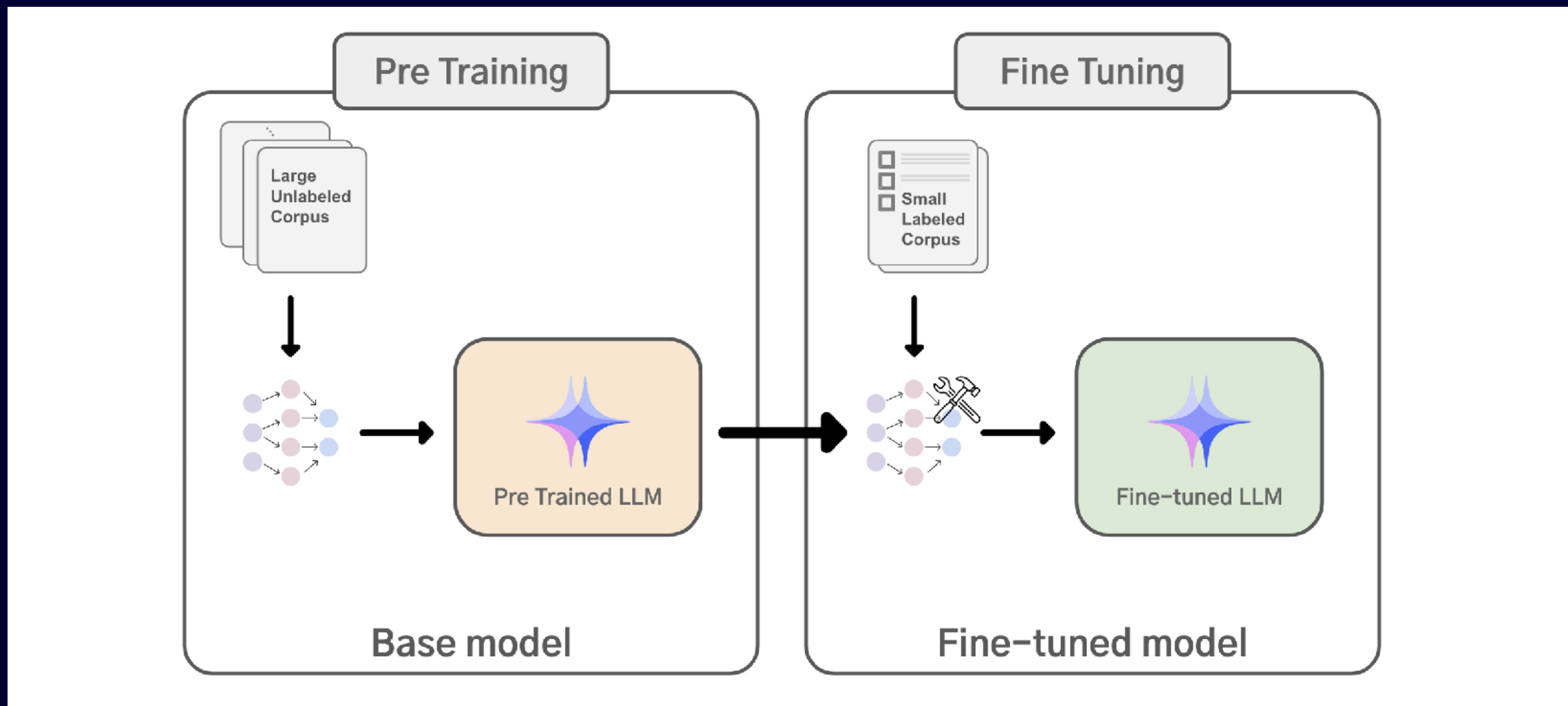
- Easy access to log data from Domino servers

Suggest an OpenNTF app for logging

Suggest an OpenNTF app for logging

XPages Log File Reader

Increase Knowledge: Fine Tune (Transfer Learning)



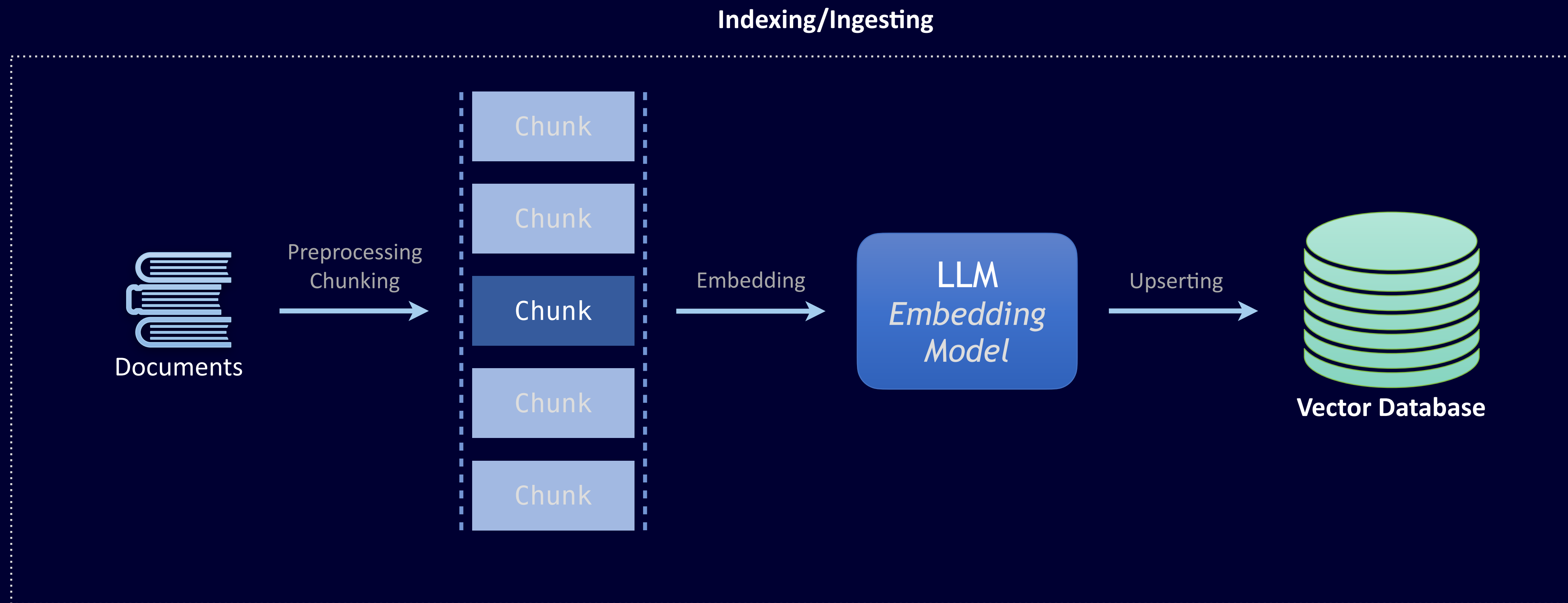


Add Context with RAG

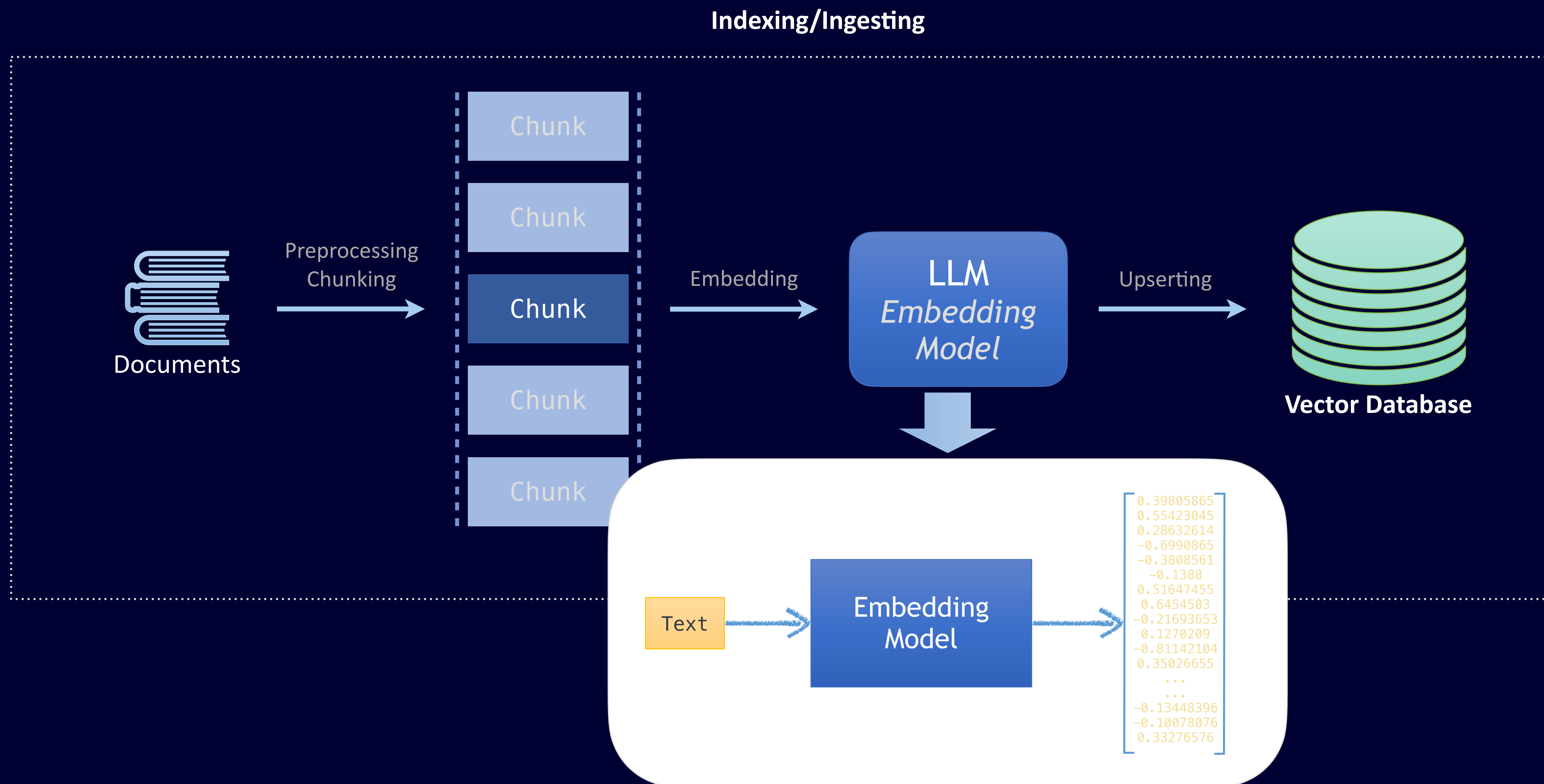
- Retrieval-augmented generation
- The common use case:
 - Domain Knowledge in documents, databases, etc.
 - **Ingest** all the knowledge to a memory
 - We receive a question for LLM
 - Retrieve and build a **context** from the **relevant knowledge**
 - Send question + the relevant context to the LLM...



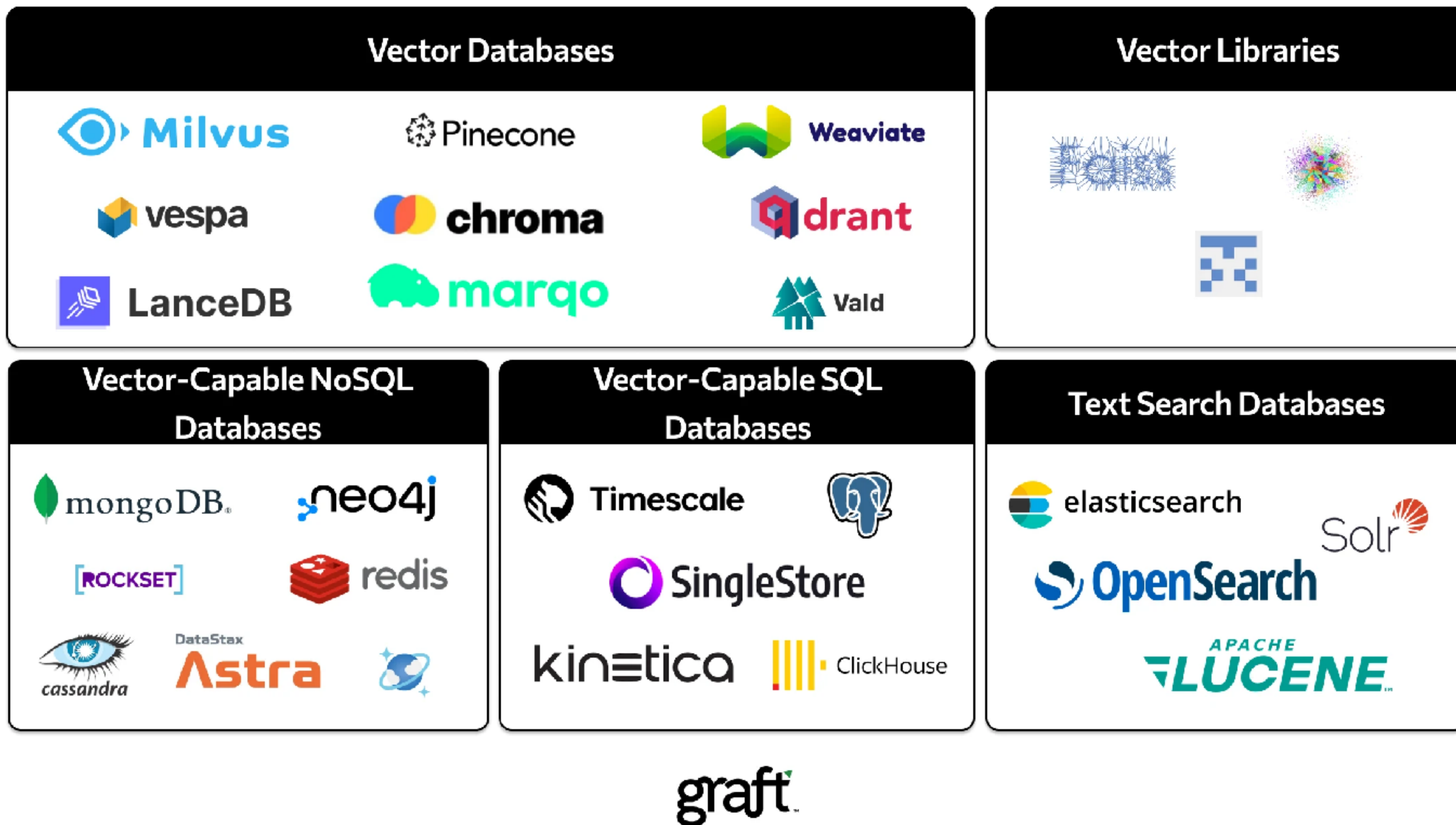
RAG Step 1: Ingestion



RAG Step 1: Ingestion

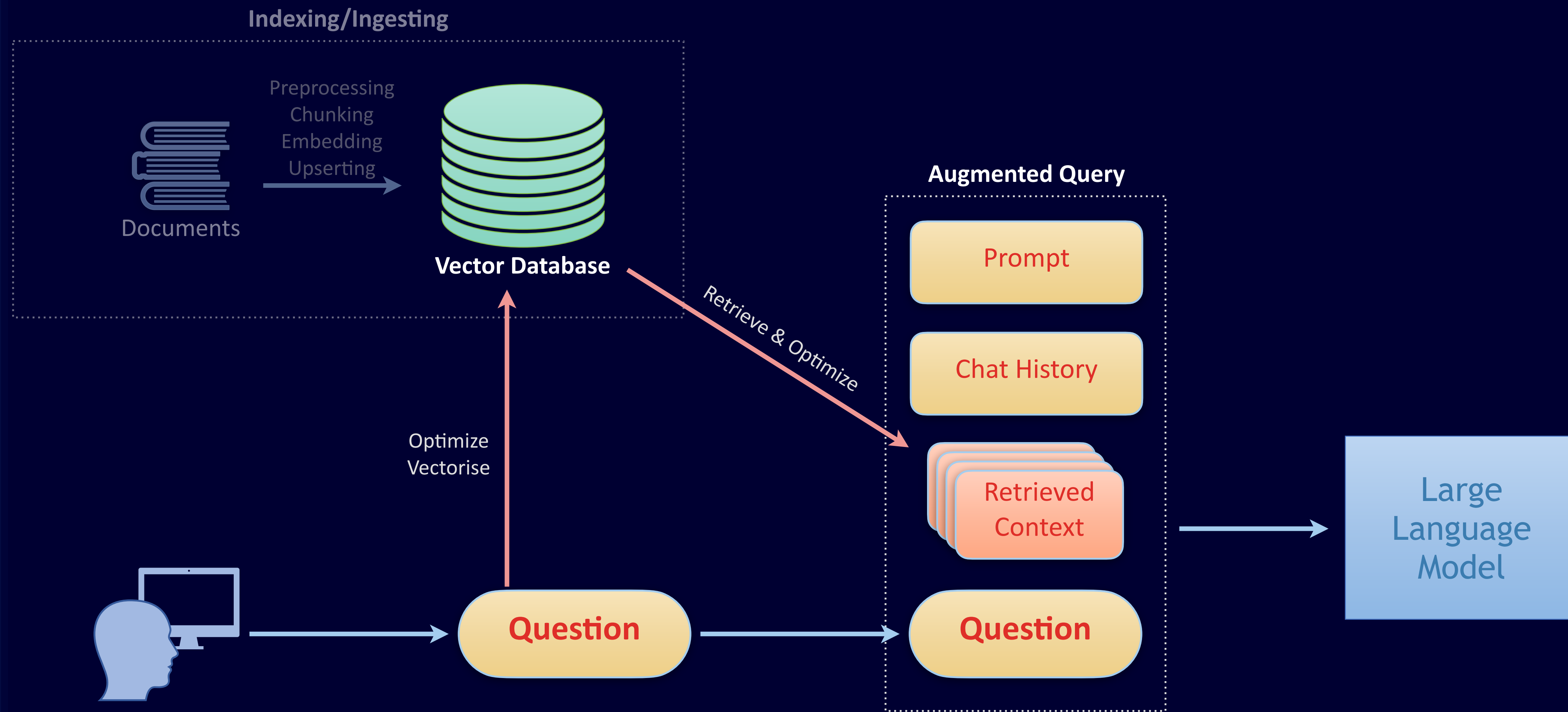


RAG Step 1: Vector Databases

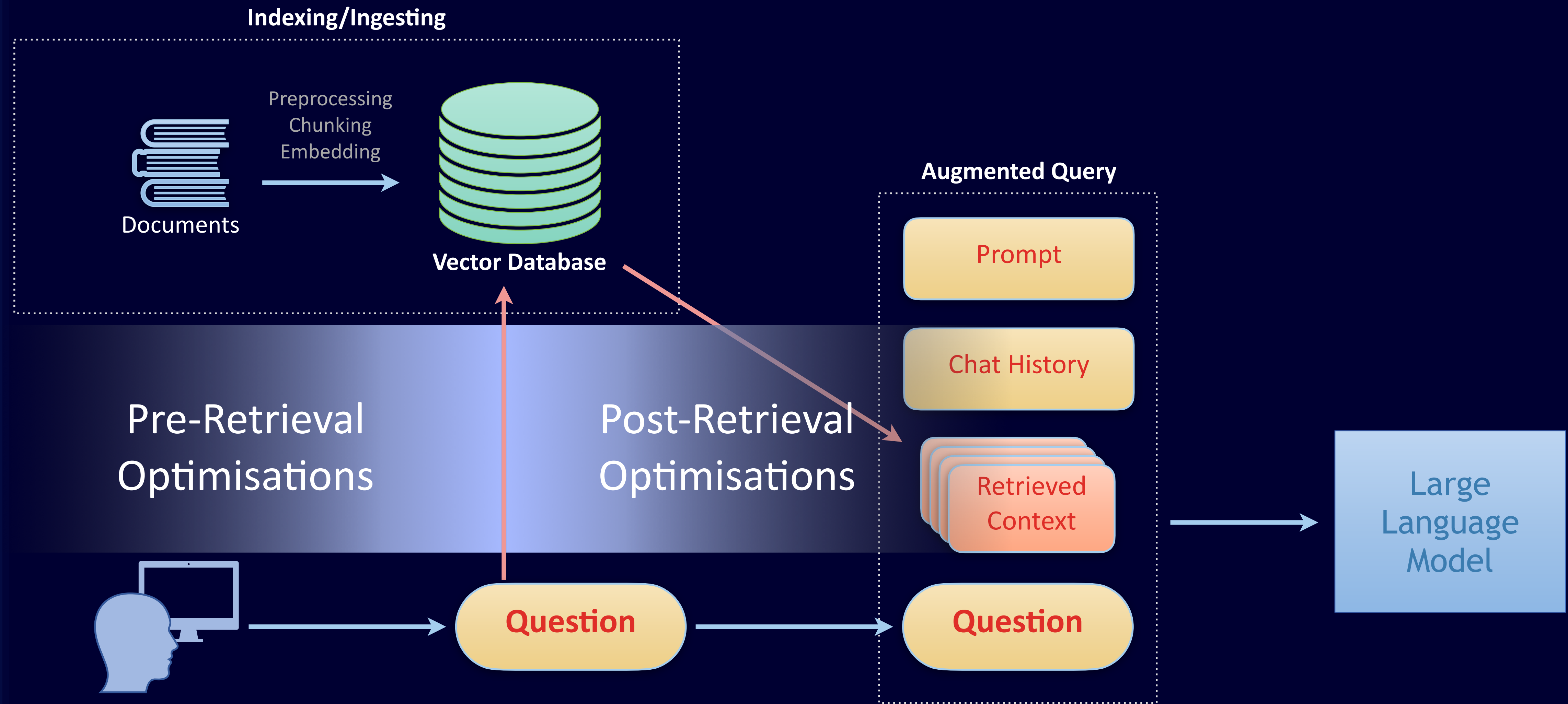




RAG Step 2: Retrieval

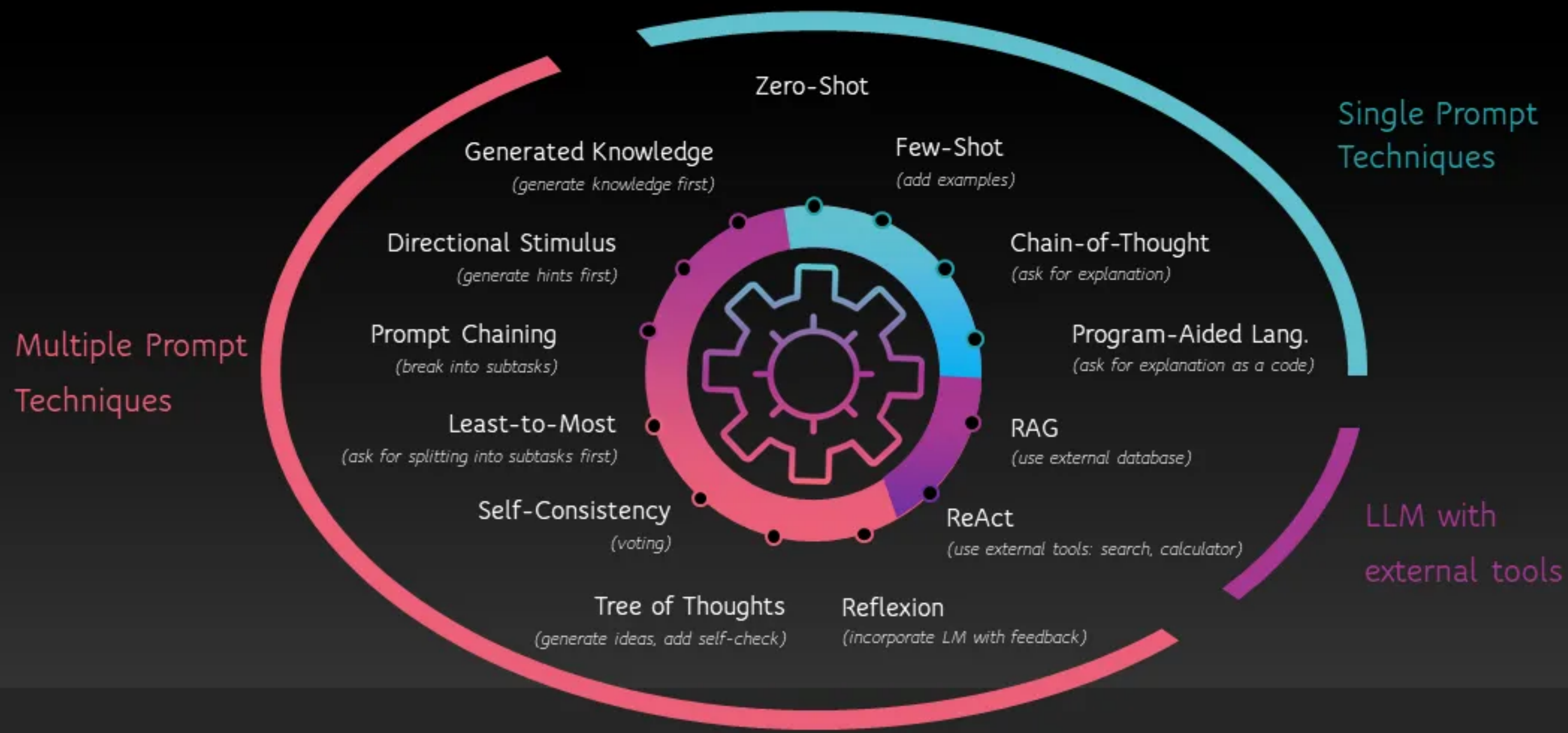


RAG Step 2: Retrieval

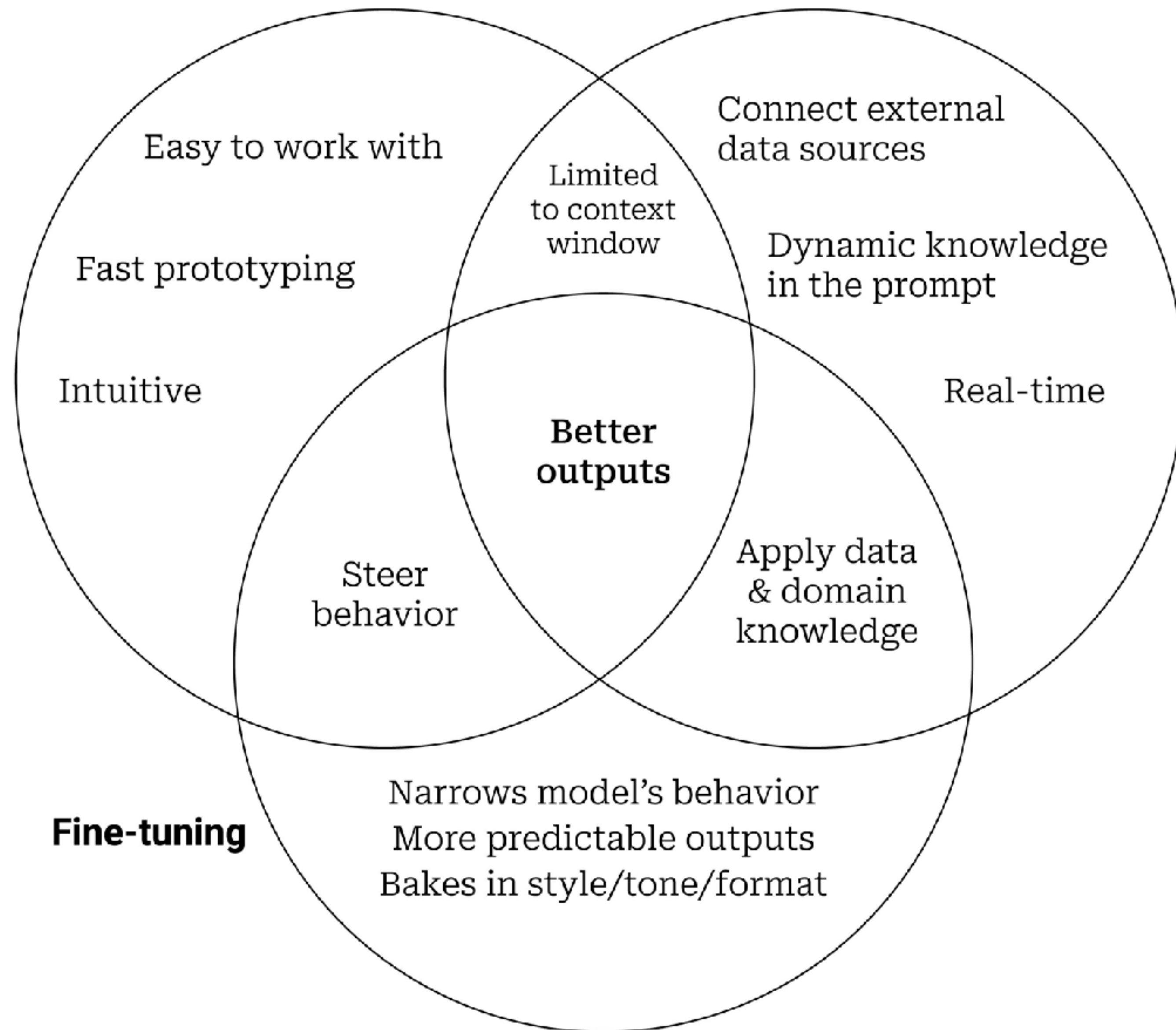


Mind the Instruction Gap: Prompt Engineering

Prompt Engineering Techniques



Prompt Engineering



RAG

Fine-tuning



LLM with Java and Domino

Programmability beyond Domino IQ...

For Java Developers

Current LangChain4j Integrations



LLM Integrations	
Amazon Bedrock	Google Vertex AI PaLM 2
Azure OpenAI	HuggingFace
ChatGLM	LocalAI
DashScope	Ollama
Google Vertex AI Gemini	OpenAI
	Mistral

Image Model Integrations

Azure OpenAI Dall-E

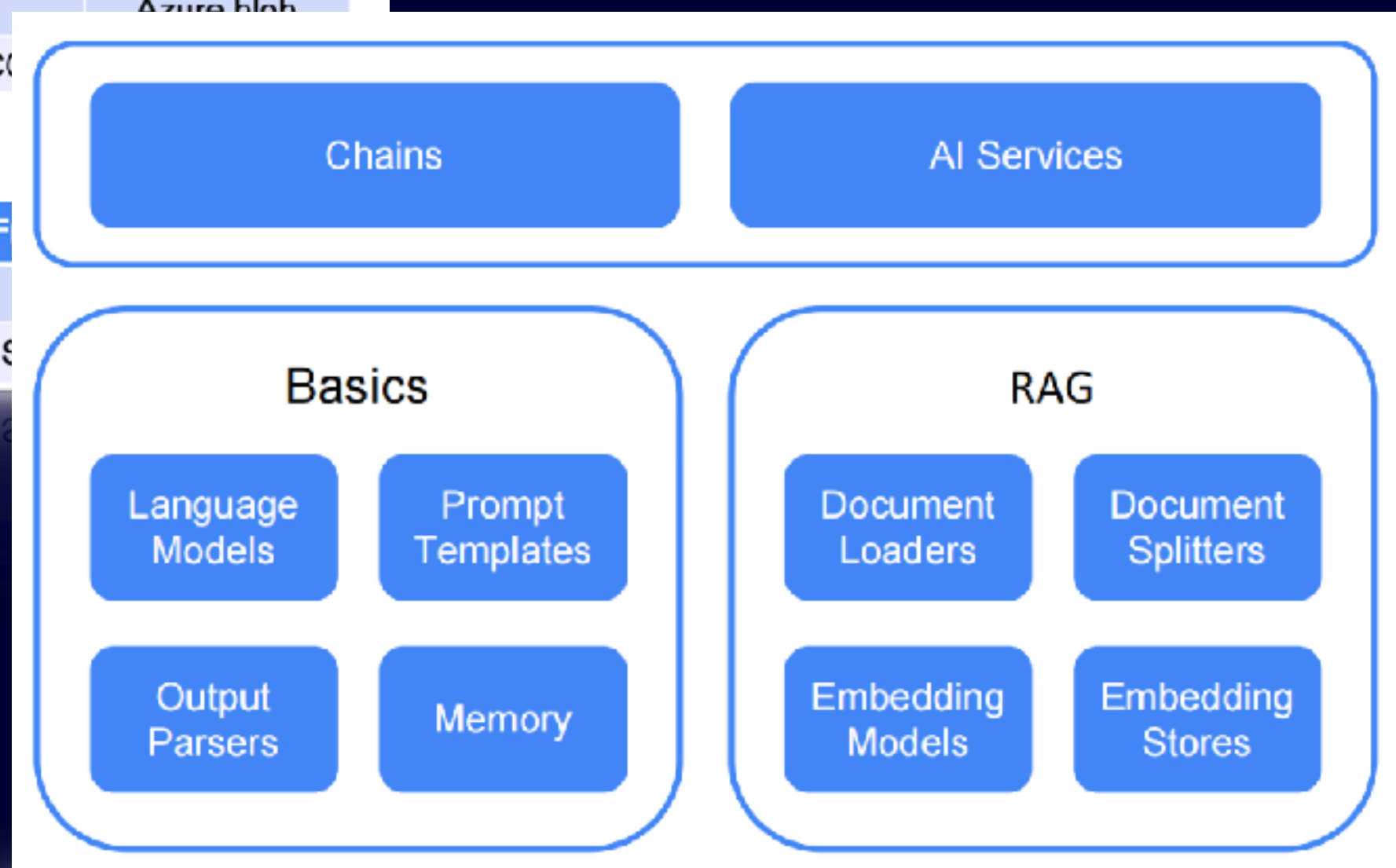
Current LangChain4j Integrations

Embedding Stores	
Chroma	Astra DB
Elasticsearch	Cassandra
Milvus	Neo4j
Pinecone	OpenSearch
Vespa	PGVector
Weaviate	MongoDB
Redis	Qdrant

Code Execution Engines

GraalVM Polyglot/Truffle
Judge0

Document Loaders	
txt	ppt
html	url
doc	github loader
pdf	S3
xls	Azure blob
Tencent COS	

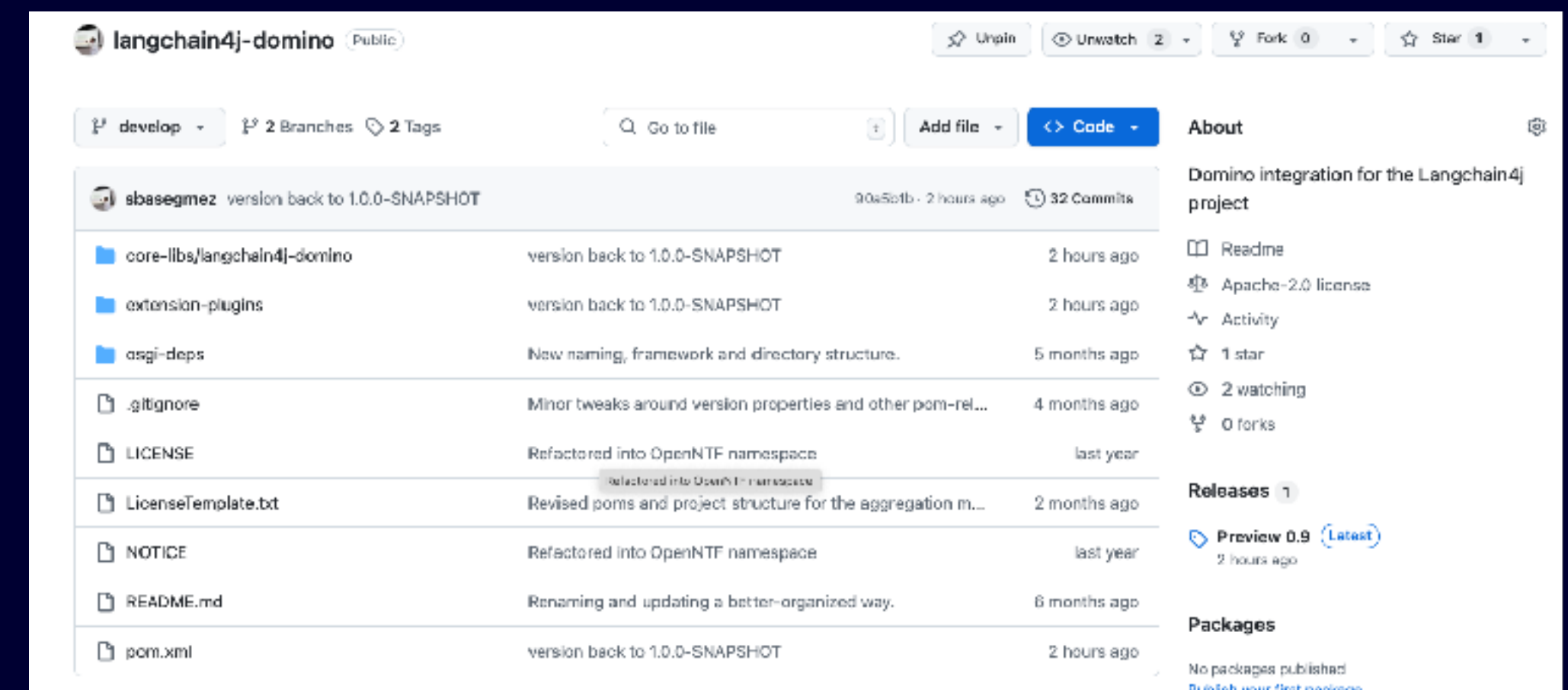


- LangChain4j is very promising



A New Project: Domino-LangChain4j

- Import langchain4j library into Domino
- Utilise ChatModel w/ Local or Cloud LLM
 - OpenAI and Ollama
- Vector Databases
 - Milvus and Chroma
- Quick and easy RAG
 - RAG document loaders for Domino documents





Domino-LangChain4j

```
// Ingestor will combine the ingesting elements
EmbeddingStoreIngestor ingestor = EmbeddingStoreIngestor.builder()
    .....embeddingModel(embeddingModel)
    .....embeddingStore(embeddingStore)
    .....documentSplitter(DocumentSplitters.recursive( maxSegmentSizeInChars: 8196, maxOverlapSizeInChars: 256))
    .....build();
```

```
var metadataDef = MetadataDefinition.builder(MetadataDefinition.DEFAULT)
    .....addString( fieldName: "Subject")
    .....build();
```

```
List<Document> docs = DominoDocumentLoader.create(metadataDef)
    .....database(database)
    .....noteIds(docIds)
    .....fieldName("Subject")
    .....fieldName("Body")
    .....loadDocuments();
```

```
System.out.println("Ingesting set of " + docs.size() + " documents");
```

```
AtomicInteger counter = new AtomicInteger( initialValue: 0);
```

```
docs.parallelStream().forEach( Document doc -> {
    .....ingestor.ingest(doc);
    .....if (counter.incrementAndGet() % 100 == 0) {
    .....    System.out.println(counter.get() + " docs ingested...");
    .....}
});
```

```
System.out.println("All ingested");
```

Ingest Documents into
Vector Database



Domino-LangChain4j

```
Set<Integer> projectNoteIds = database.openCollection( collectionName: "projects" ) Optional<...>
    .orElseThrow() DominoCollection
    .getAllIds( withDocuments: true, withCategories: false );

var metadataDef = MetadataDefinition.builder(MetadataDefinition.DEFAULT)
    .addString( fieldName: "name" )
    .addString( fieldName: "overview" )
    .addString( fieldName: "chefs" )
    .addString( fieldName: "sourceUrl" )
    .addString( fieldName: "category" )
    .build();

List<Document> docs = DominoDocumentLoader.create(metadataDef)
    .database(database)
    .fieldName("name")
    .fieldName("details")
    .noteIds(projectNoteIds)
    .loadDocuments();
```

DominoDocumentLoader

Load by Field

```
List<Document> docs = DominoDocumentLoader.create(metadataDef)
    .database(database)
    .noteIds(manifestoDocIds)
    .filePattern("*.pdf")
    .documentParser(new ApachePdfBoxDocumentParser())
    .loadDocuments();
```

Load Attachments



Domino-LangChain4j

DominoDocumentLoader

```
Set<Integer> projectNoteIds = database.openCollection( collectionName: "projects" ) Optional<...>  
    .orElseThrow() DominoCollection  
    .getAllIds( withDocuments: true, withCategories: false );
```

```
var metadataDef = MetadataDefinition.builder(MetadataDefinition.DEFAULT)  
    .addString( fieldName: "name" )  
    .addString( fieldName: "overview" )  
    .addString( fieldName: "chefs" )  
    .addString( fieldName: "sourceUrl" )  
    .addString( fieldName: "category" )  
    .build();
```

```
List<Document> docs = DominoDocumentLoader.create(metadataDef)  
    .database(database)  
    .fieldName("name")  
    .fieldName("details")  
    .noteIds(projectNoteIds)  
    .loadDocuments();
```

Customize Metadata

```
"addedtofile": "2024-11-18T12:30:50.7Z",  
"overview": "A small utility for managing expesne with analysis of data and better control",  
"unid": "F8B8D65D19C1BF1600258BD90044BDED",  
"created": "2024-11-18T12:30:50.69Z",  
"index": "0",  
"noteid": "0000F606",  
"chefs": "Anurag Sharma",  
"lastaccessed": "2024-11-18T12:30:50.7Z",  
"sourceUrl": "https://www.openntf.org/main.nsf/project.xsp?r=project/Expense+Management",  
"lastmodifiedinfile": "2024-11-18T12:30:50.71Z",  
"form": "project",  
"size": 1067,  
"lastmodified": "2024-11-18T12:30:50.7Z",  
"name": "Expense Management",  
"category": "Expense Tracking"
```



Demo

RAG chatbot with XPages in action



Notes Helpdesk Assistant...

You How do I create a local replica for a database?

Assistant Hello Jack,

To create a local replica of a database in HCL Notes, you can follow these steps:

1. From an open application, click File -> Application -> Make Available Offline.
- Note: This command is disabled if you are already focused on a local replica or on a server-based application for which you already have a local replica.
2. To create the replica immediately, accept all defaults and click OK.

Alternatively, for an advanced method:

1. Open the application you want to create a replica for.
2. Click File -> Replication -> New Replica. (You need to have Advanced Menus enabled from the View menu to see this command.)
3. If the application is on a Domino server, you can choose to create the replica locally or on a server by selecting the server name and specifying the file path.
4. If the application is local, you can create the replica on a server or locally by modifying the file path.
5. Optionally, click Replica Settings to configure encryption, full-text indexing, ACL copying, and replication timing.
6. Click OK to create the replica.

Notes will add a bookmark and an application entry to the Replication page for each local replica you create, which you can use to update your replicas.

Would you like me to open a ticket for you regarding this? If yes, please provide the severity level or I can infer it for you.

The database is huge. I don't want all of it.

Assistant Jack, just to confirm the details for the ticket:

You need assistance with creating and customizing a local replica of a large HCL Notes database. Specifically, you want help with advanced replication settings to save space and limit the amount of data replicated, including using the Space Savers page and other replication options.

Is that correct? Also, please let me know the severity level you would like to assign to this ticket: Low, Medium, or Critical. If you don't specify, I will infer it as Medium.

You Sure, it's OK

Assistant Thank you, Jack. I have created a ticket for your issue with problem ID DGQHG4. Please keep this ID for future reference. If you need any further assistance, feel free to ask!

|

Send



Domino-LangChain4j

Quick RAG

```
QueryTransformer queryTransformer = CompressingQueryTransformer.builder()
    .chatModel(getChatModel())
    .build();

ContentRetriever contentRetriever = EmbeddingStoreContentRetriever.builder()
    .embeddingStore(embeddingStore)
    .embeddingModel(embeddingModel)
    .maxResults(4)
    .minScore(0.75)
    .build();

RetrievalAugmentor augmentor = DefaultRetrievalAugmentor.builder()
    .contentRetriever(contentRetriever)
    .queryTransformer(queryTransformer)
    .contentInjector(DefaultContentInjector.builder()
        .metadataKeysToInclude(List.of("Subject"))
        .promptTemplate(PromptTemplate.from("{{userMessage}}\n{{contents}}"))
        .build())
    .build();

return AiServices.builder(HelpdeskAssistant.class)
    .chatModel(getChatModel())
    .retrievalAugmentor(augmentor)
    .chatMemoryProvider(memoryId → MessageWindowChatMemory.withMaxMessages(20))
    .tools(new HelpdeskTicketTool())
    .build();
```



Domino-LangChain4j

```
QueryTransformer queryTransformer = CompressingQueryTransformer.builder()  
    .chatModel(getChatModel())  
    .build();
```

Query Transformation

```
ContentRetriever contentRetriever = EmbeddingStoreContentRetriever.builder()  
    .embeddingStore(embeddingStore)  
    .embeddingModel(embeddingModel)  
    .maxResults(4)  
    .minScore(0.75)  
    .build();
```

```
RetrievalAugmentor augmentor = DefaultRetrievalAugmentor.builder()  
    .contentRetriever(contentRetriever)  
    .queryTransformer(queryTransformer)  
    .build();
```

```
return AiServices.builder(Helper.chatModel(getChatModel()),  
    .retrievalAugmentor(augmentor)  
    .chatMemoryProvider(memoryId → MessageWindowChatMemory.withMaxMessages(20))  
    .tools(new HelpdeskTicketTool())  
    .build();
```

"How do I create a local replica for a database?"

"The database is huge. I don't want all of it."

"User wants to create a smaller local replica for a very large database."



Domino-LangChain4j - Tools

```
@Tool("""
    Create help desk ticket and return the problem id for future reference.
    We need two parameters to use this tool.
    - Severity is either one of "Low", "Medium", "Critical" or "Unknown"
    - Problem Definition is contains detailed information about the problem,
      optionally steps to reproduce and platform.
    """)
public String createHelpdeskTicket(String severity, String problemDefinition) {
    System.out.println("Creating a ticket: " + problemDefinition);

    try {
        Database appDb = ExtLibUtil.getCurrentDatabase();
        Document doc = appDb.createDocument();

        doc.replaceItemValue("Form", "HelpdeskTicket");
        doc.replaceItemValue("Severity", severity);
        doc.replaceItemValue("Problem", problemDefinition);

        doc.computeWithForm(false, false);

        if(doc.save()) {
            System.out.println("Saved!");
            return doc.getItemValueString("ProblemId");
        }
    }
}
```

Tool Prompt

```
return AiServices.builder(HelpdeskAssistant.class)
    .chatModel(getChatModel())
    .retrievalAugmentor(augmentor)
    .chatMemoryProvider(memoryId → MessageWindowChatMemory.with...)
    .tools(new HelpdeskTicketTool())
    .build();
```

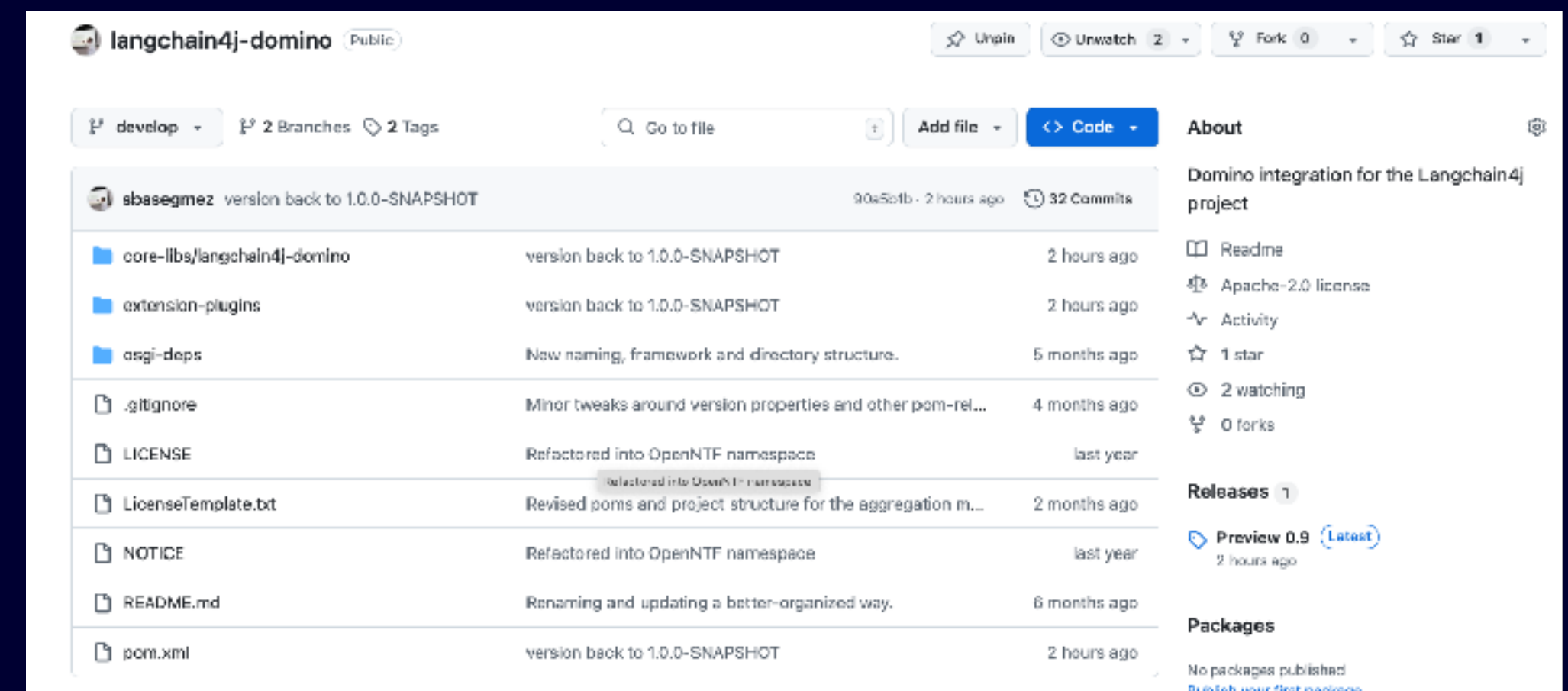
Tool declaration



Domino-LangChain4j: More...

- Server and Designer plugins
 - Requires 14+ (Java 17)
 - Based on Domino JNX
 - Additional plugins for additional LLMs*
- Domino-specific utilities*
 - Java Agent and DOTS support
 - Domino IQ Support (?)
 - Observability / Logging
 - XPages Beans
 - Configuration
- LC4J GA released just last week!
- Feedbacks are welcome!

* In progress





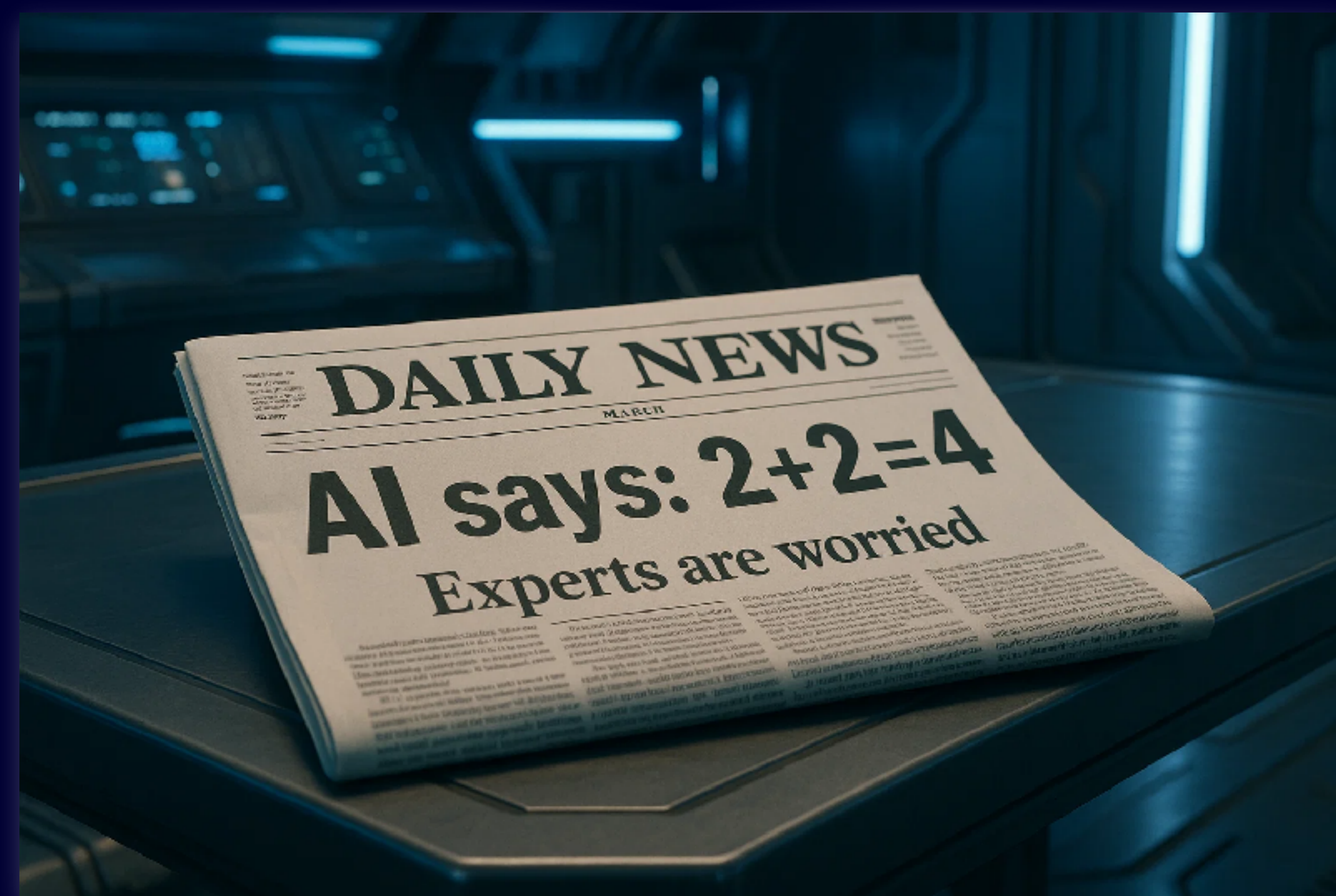
Domino isn't Built for This—Yet

- Domino has the capabilities
- Missing Pieces in the Domino + LLM Pipeline
 - NSF does not **store/index vectors**
 - Options for **background tasks** are limited
 - Programmability is highly **restricted**
 - Outdated parts **limit** integration
- Alternatives?
 - Run your LLM outside of Domino
 - Hybrid Pipelines



Critical Considerations for LLM Projects

- Cybersecurity in LLMs
 - *OWASP Top 10 for LLM Applications*
- Responsible and predictable AI
 - Guardrails, moderation, explainability
- AI Compliance





Emerging Concepts

Keep an eye on these...

CAG: Cache-Augmented Generation

- An alternative model for RAG

Retrieval Inefficiencies

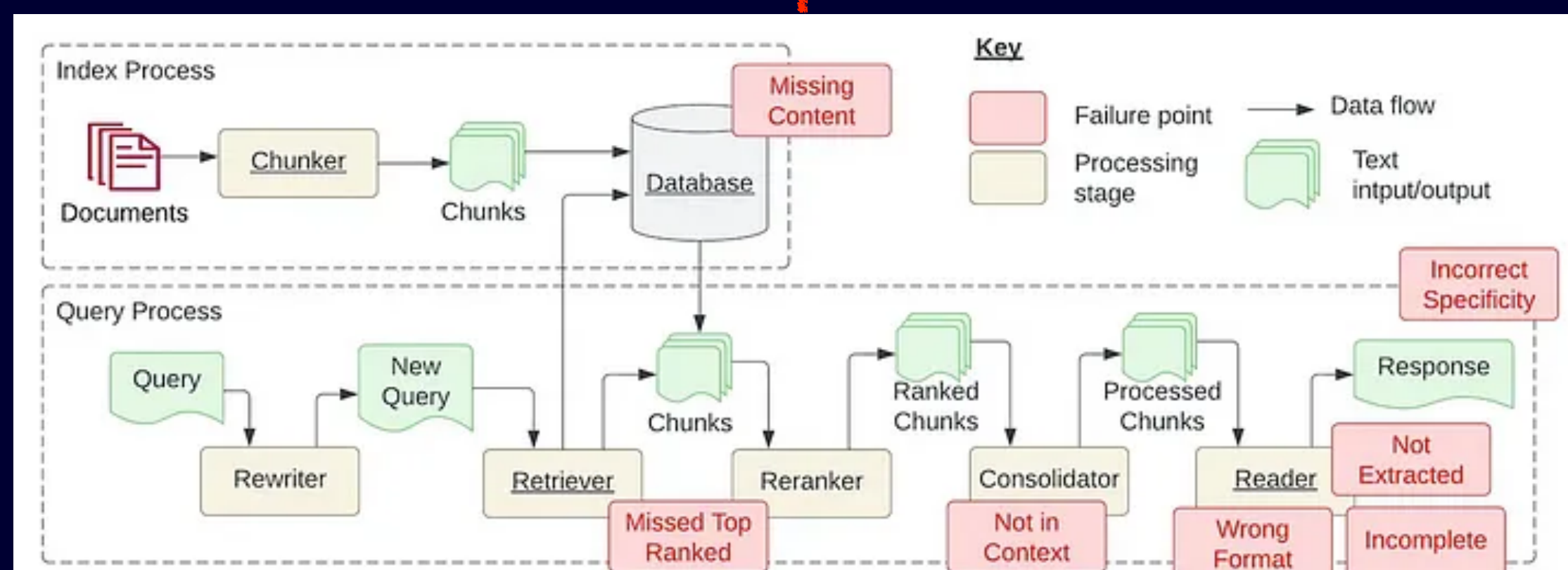
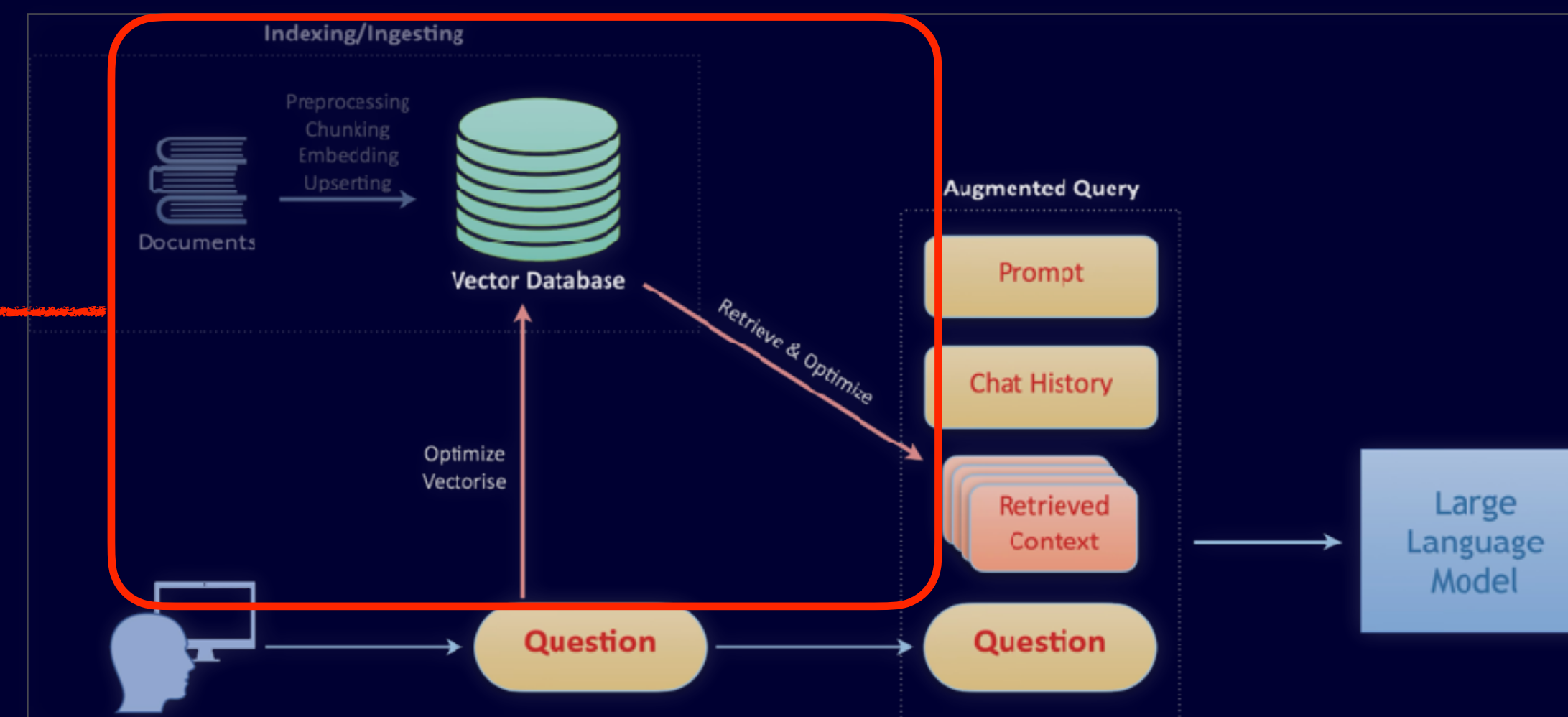
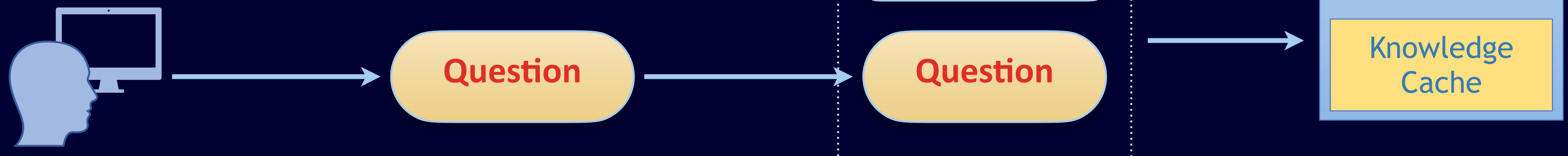


Figure 1: Indexing and Query processes required for creating a Retrieval Augmented Generation (RAG) system. The indexing process is typically done at development time and queries at runtime. Failure points identified in this study are shown in red boxes. All required stages are underlined. Figure expanded from [19].

CAG: Cache-Augmented Generation

- ✓ No Retrieval Overhead
- ✓ Simplicity
- ✓ Consistency in Behaviour
- ✓ Efficient and Accurate Response
- ✗ Large Context-size Needed
- ✗ Limited Knowledge-base
- ✗ Better for Stable Knowledge
- ✗ Not Suitable for Multiple Sources





MCP: Model Context Protocol

```
@Tool("""
    Create help desk ticket and return the problem id for future reference.
    We need two parameters to use this tool.
    - Severity is either one of "Low", "Medium", "Critical" or "Unknown"
    - Problem Definition is contains detailed information about the problem,
      optionally steps to reproduce and platform.
    """)
public String createHelpdeskTicket(String severity, String problemDefinition) {
    System.out.println("Creating a ticket: " + problemDefinition);
}
```

Tool Prompt

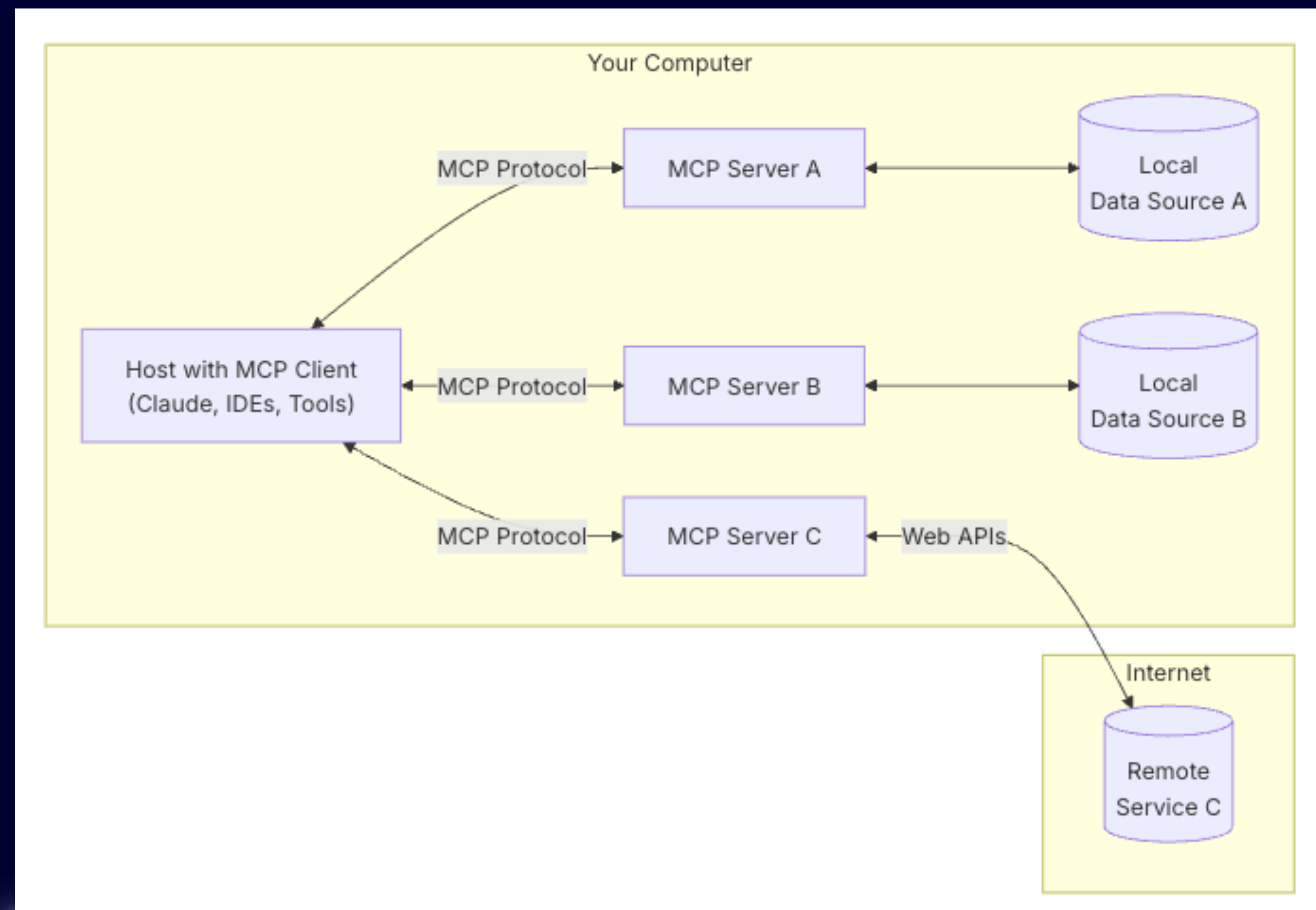
```
return AiServices.builder(HelpdeskAssistant.class)
    .chatModel(getChatModel())
    .retrievalAugmentor(augmentor)
    .chatMemoryProvider(memoryId -> MessageWindowChatMemory.withMaxMessages(20))
    .tools(new HelpdeskTicketTool())
    .build();
```

Tool declaration

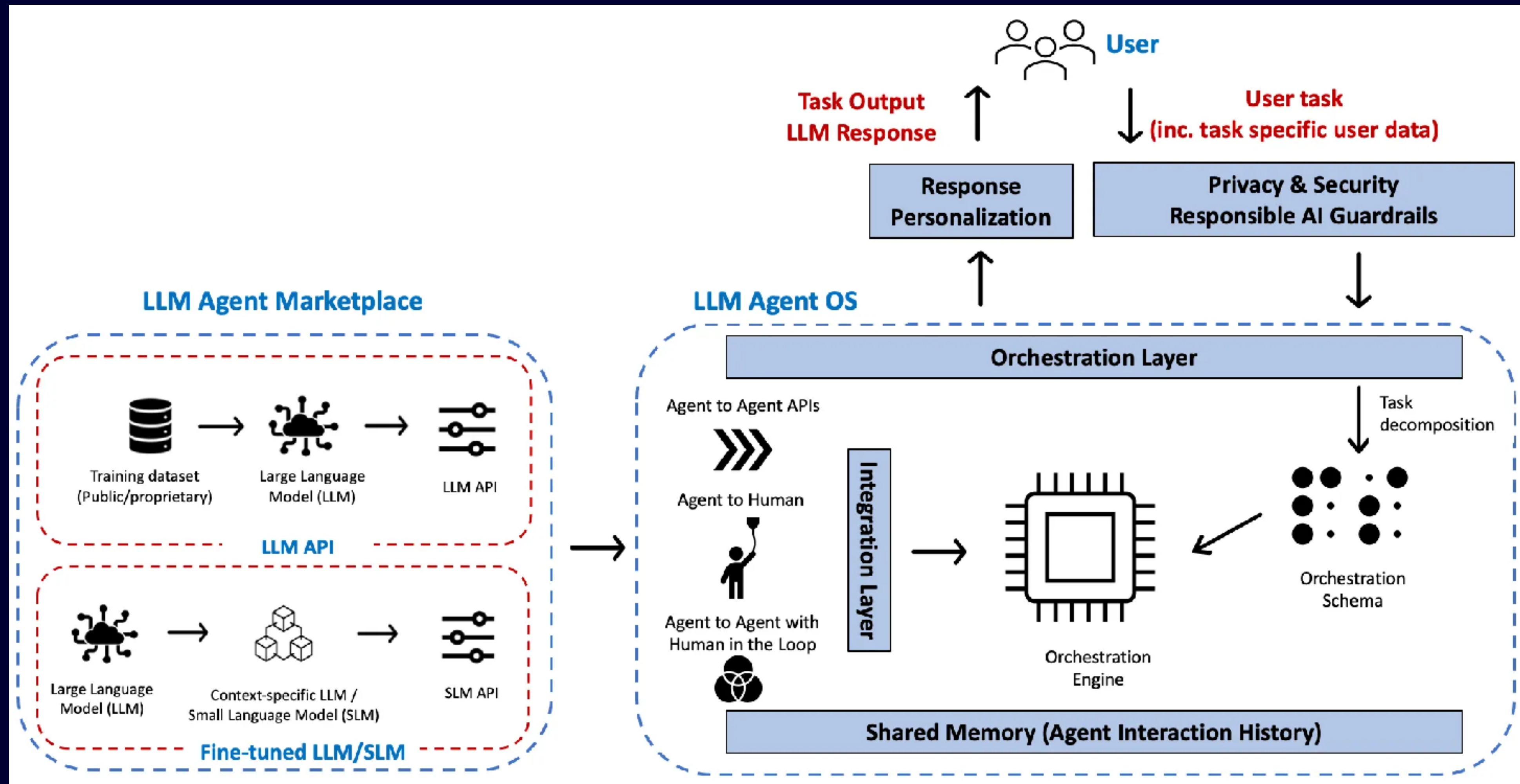


MCP: Model Context Protocol

- Universal tool/capability discovery
- Agentic AI



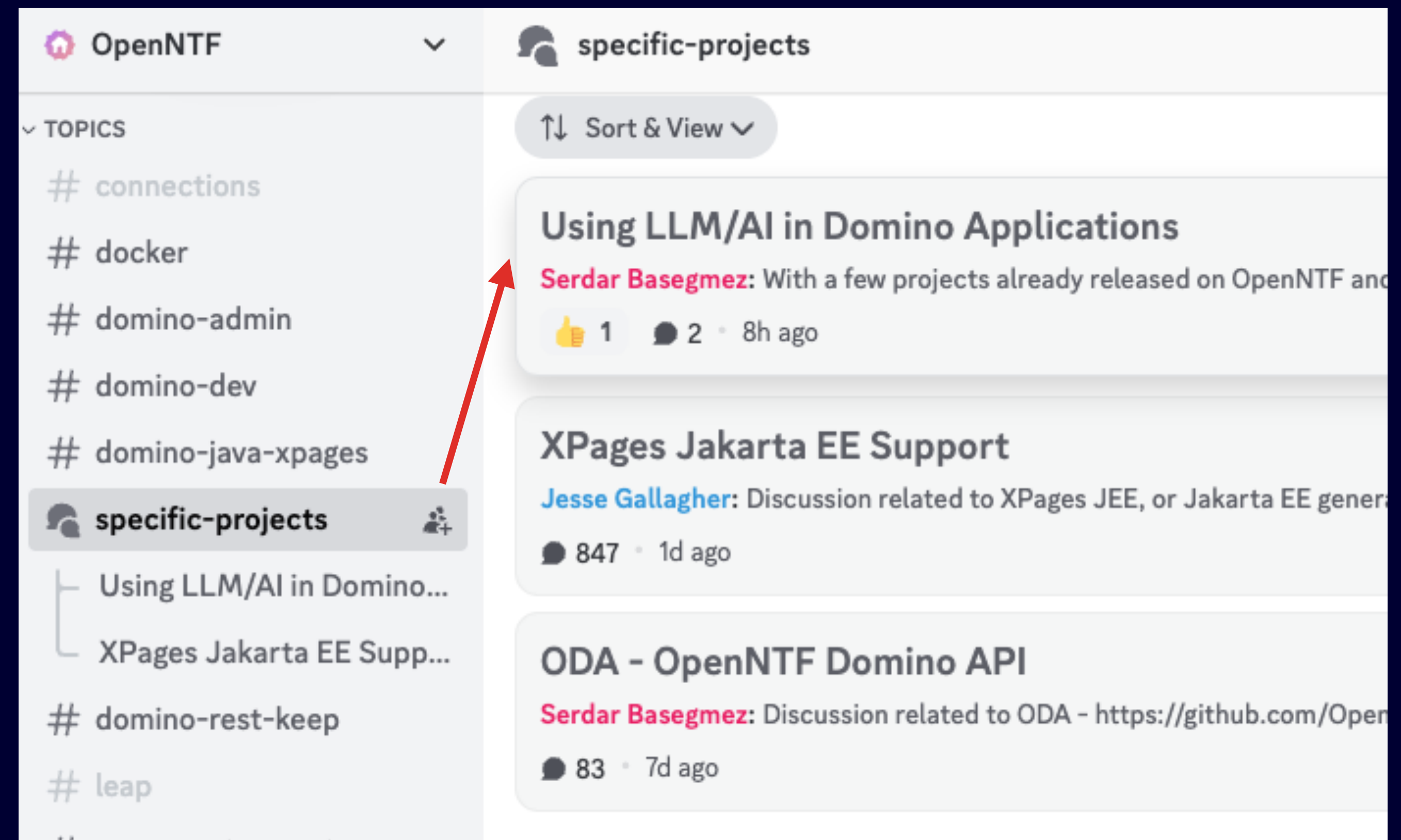
Agentic AI





Feedbacks and Discussions

- Q&A now, or...
- OpenNTF Discord Server
 - Specific Projects —> Using LLM/AI in Domino Applications





Resources

➔ All the demo materials:

- <https://github.com/sbasegmez/LLM-Demos>

➔ OpenNTF Projects Metadata:

- [https://www.openntf.org/main.nsf/project.xsp?r=project/OpenNTF Projects Dataset](https://www.openntf.org/main.nsf/project.xsp?r=project/OpenNTF%20Projects%20Dataset)

➔ Domino-Langchain4j experimental version:

- <https://github.com/sbasegmez/domino-langchain4j>



More Good Stuff: Odds and Ends

- Further reading...
 - Huggingface blogs
 - RAG - Retrieval Augmented Generation
 - Multimodal approaches
 - Prompt Engineering
- Courses, guides
 - Quick Start Guide to Large Language Models (LLMs)
Course by Sinan Ozdemir
 - Large Language Models: Application through Production
Databricks
 - Large Language Model Ebooks
NVidia

