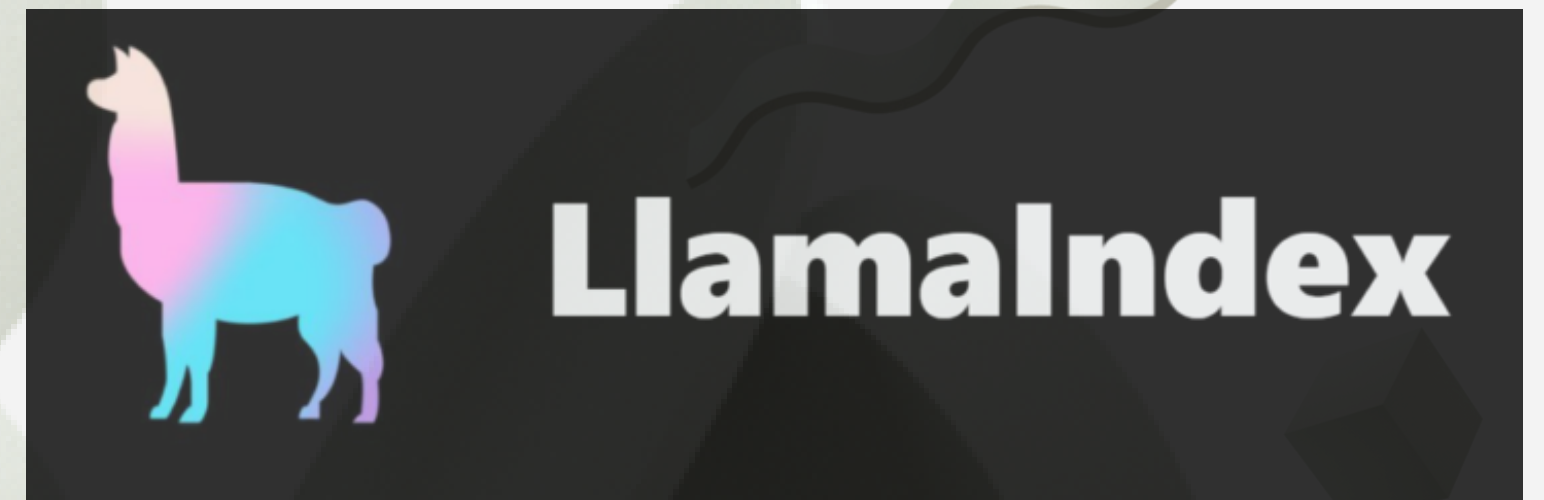


# Building QA Systems with Llamaindex



ft. Ravi Theja  
Data Scientist@Glance(InMobi)  
Open source contributor Llamaindex

BY- AAISHA KHAN


# What are LLMs?

LLMs, or large language models, are a type of artificial intelligence (AI) that are trained on massive datasets of text and code. This allows them to generate text, translate languages, write different kinds of creative content, and answer your questions in an informative way.





# Use cases for LLMs

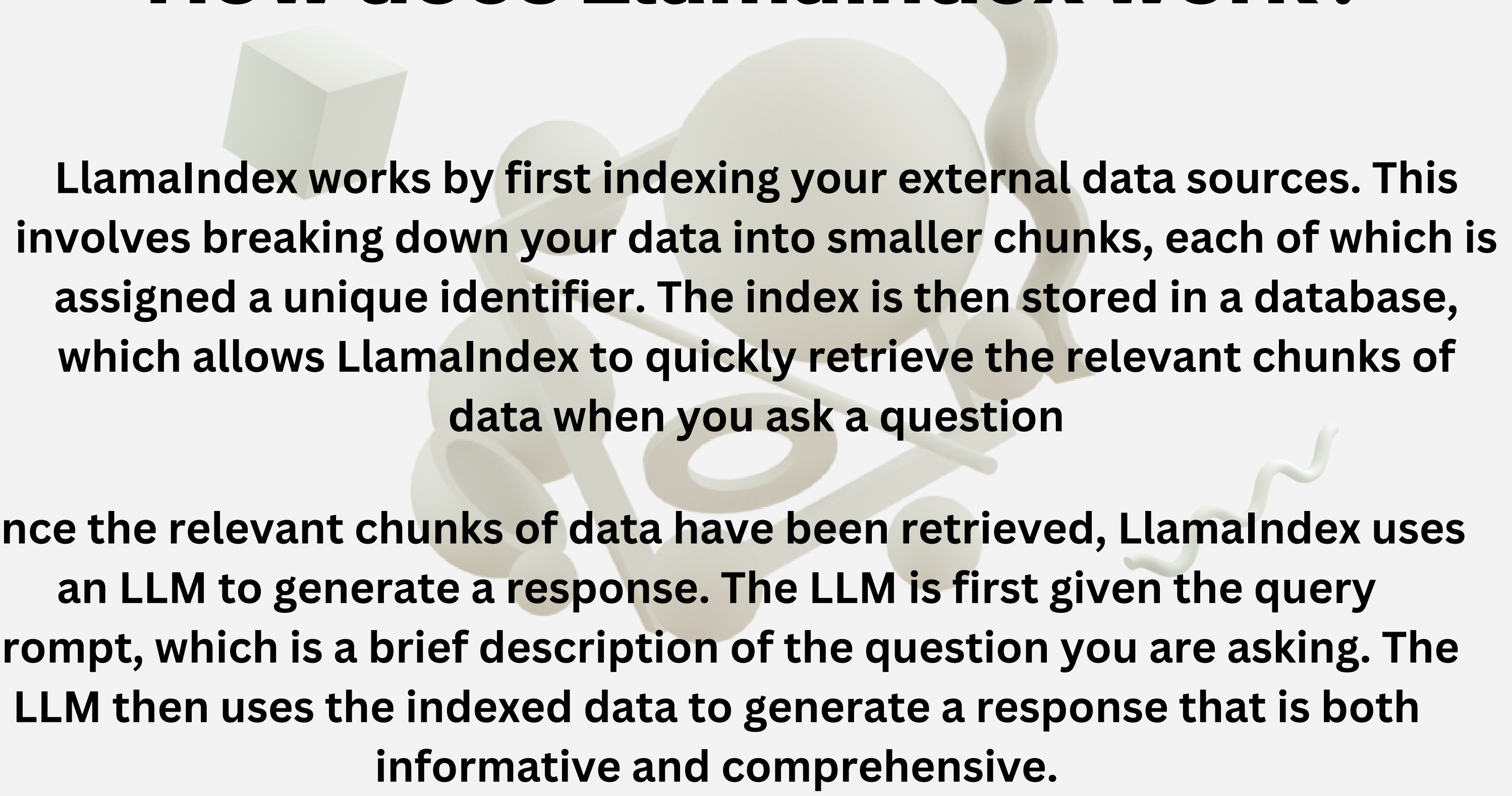
- Question answering: LLMs can be used to answer questions in a comprehensive and informative way, even if they are open ended, challenging, or strange.
  - Text generation: LLMs can be used to generate text, such as poems, code, scripts, musical pieces, email, letters, etc.
  - Summarization: LLMs can be used to summarize long documents into a more concise format.
  - Planning: LLMs can be used to help with planning tasks, such as scheduling appointments or creating travel itineraries.
- 



# **What is LlamaIndex?**

**LlamaIndex is a data framework that makes it easy to connect LLMs to external data sources. This allows you to build more powerful and informative applications.**

# How does LlamaIndex work?



LlamaIndex works by first indexing your external data sources. This involves breaking down your data into smaller chunks, each of which is assigned a unique identifier. The index is then stored in a database, which allows LlamaIndex to quickly retrieve the relevant chunks of data when you ask a question

Once the relevant chunks of data have been retrieved, LlamaIndex uses an LLM to generate a response. The LLM is first given the query prompt, which is a brief description of the question you are asking. The LLM then uses the indexed data to generate a response that is both informative and comprehensive.

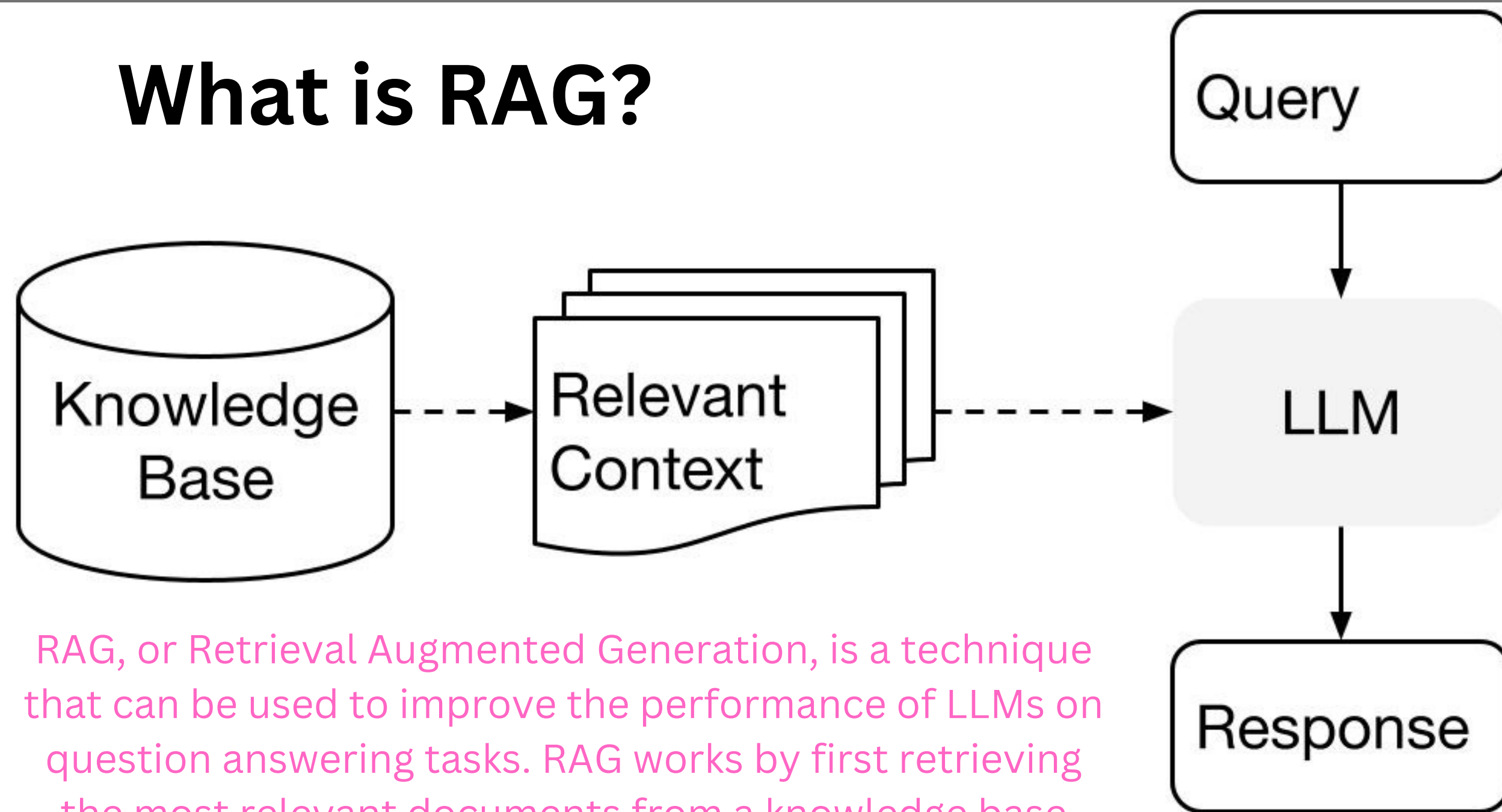


# Components of LlamaIndex

- Index: The index is a database that stores the indexed data.
- Retriever: The retriever is responsible for retrieving the relevant chunks of data from the index.
- Response synthesizer: The response synthesizer is responsible for generating the response from the LLM.
- Query engine: The query engine is responsible for coordinating the different components of LlamaIndex to generate a response to a query.

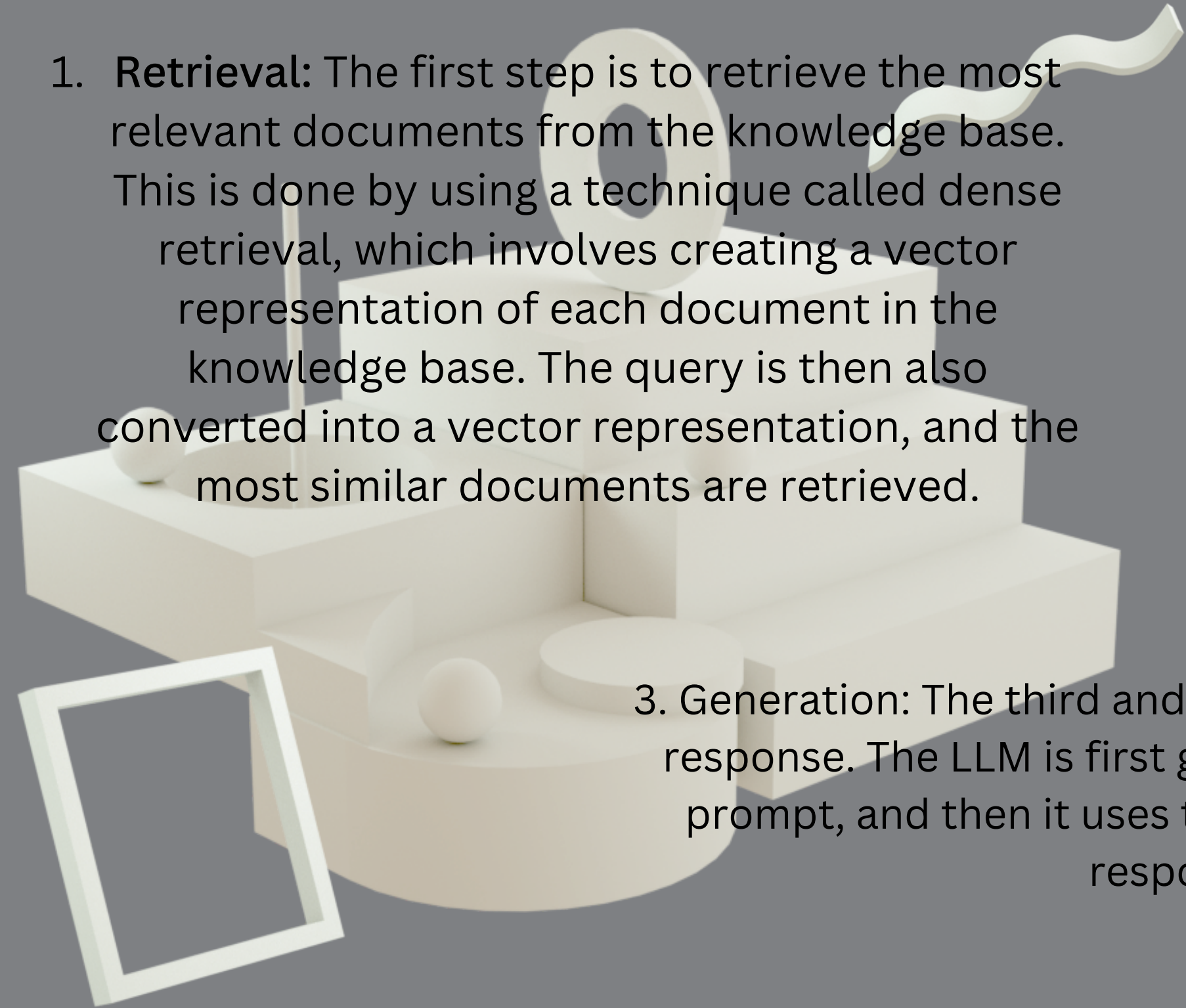


# What is RAG?



RAG, or Retrieval Augmented Generation, is a technique that can be used to improve the performance of LLMs on question answering tasks. RAG works by first retrieving the most relevant documents from a knowledge base that are related to the query. These documents are then used to augment the input prompt for the LLM, which helps the LLM to generate a more accurate and informative response.

# How does RAG work?



1. **Retrieval:** The first step is to retrieve the most relevant documents from the knowledge base. This is done by using a technique called dense retrieval, which involves creating a vector representation of each document in the knowledge base. The query is then also converted into a vector representation, and the most similar documents are retrieved.

2. **Augmentation:** The second step is to augment the input prompt for the LLM. This is done by adding the text of the retrieved documents to the input prompt. This helps the LLM to have a better understanding of the context of the query, which can lead to more accurate and informative responses.

3. **Generation:** The third and final step is to generate the response. The LLM is first given the augmented input prompt, and then it uses this prompt to generate a response.



# Applications of LlamaIndex

- Simple QA systems:  
LlamaIndex can be used to build simple QA systems that can answer questions about a specific domain. For example, you could use LlamaIndex to build a QA system that can answer questions about the weather, sports, or news.
- Chat engines: LlamaIndex can also be used to build chat engines that can hold conversations with users. Chat engines can be used for a variety of purposes, such as customer service, marketing, or education.
- Agents: LlamaIndex can also be used to build agents that can automate tasks. For example, you could use LlamaIndex to build an agent that can book appointments, send emails, or manage finances.



# Challenges of In-Context Learning

Retrieving the right context for the prompt:  
One of the challenges of in-context learning is retrieving the right context for the prompt. This can be difficult because the context of a prompt can be ambiguous or incomplete.

Dealing with long text: Another challenge of in-context learning is dealing with long text. This is because large language models (LLMs) are trained on massive datasets of text, and this text can be very long.

Dealing with potentially large data: In-context learning can also be challenging when dealing with potentially large data. This is because LLMs need to be able to access and process this data quickly in order to generate responses in a timely manner



The background is a solid bright pink. It is decorated with various white 3D geometric shapes. At the top, there are two sets of semi-circles. In the center, there is a large semi-circle with a circular cutout, flanked by two monstera leaves. Below this, there are two cones. At the bottom left, there is a complex structure of intersecting lines and rings. At the bottom right, there are several circles, some with smaller circles inside them, and a ring.

**HAPPY LEARNING**

**THANK YOU**