

Agentic RAG

Fun interacting with data -

Tarun Jain





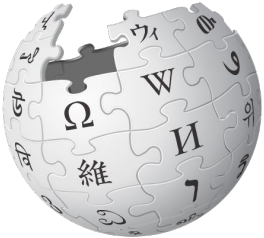
\$whoami 💡

- Data Scientist at AI Planet
- AI with Tarun - YouTube Content Creator
- Google Developer Expert in AI
- Google Summer of Code 2024 Red Hen Lab
- Google Summer of Code 2023 caMicroscope
- Watch Anime and Read Manga in free time.

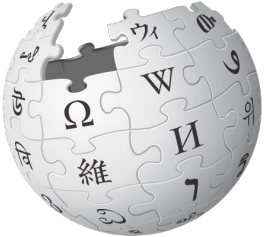
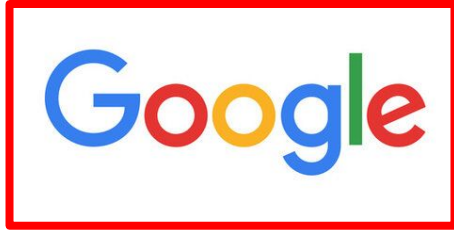
Task



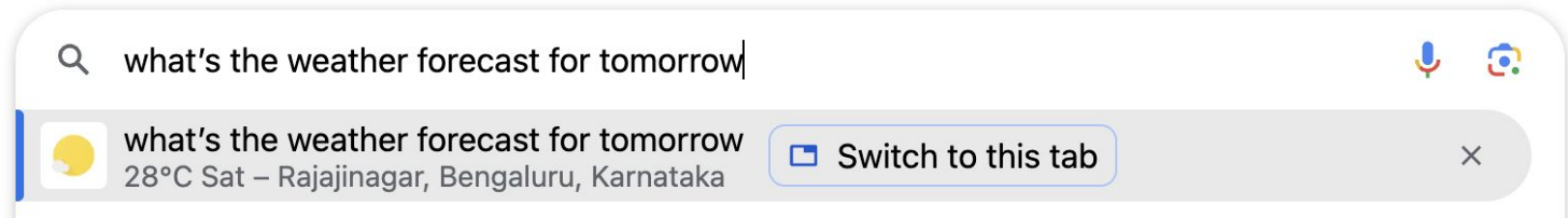
Tools



Action



Decision



***So, tell me: what is your answer:
will you take an umbrella?***

but...

- We have some common questions to address in Agents?
 - Can't the LLM with some context complete the task mentioned in previous slide?
 - What is you showcased is a Function calling + LLM, then why you said its Agent?
 - Function calling vs LLM vs Agent- How is it different?



Time: 19:30

LLM vs Agent



LLM:

Good morning



Good morning! 😊 How can I assist you today?

Agent:

Good morning



Good evening! 😊 It seems like it's still nighttime, but if you're greeting me with "Good morning," maybe you're preparing for a fresh start or a specific context—let me know how I can assist!

Which tool do you think was
used in the previous slide?

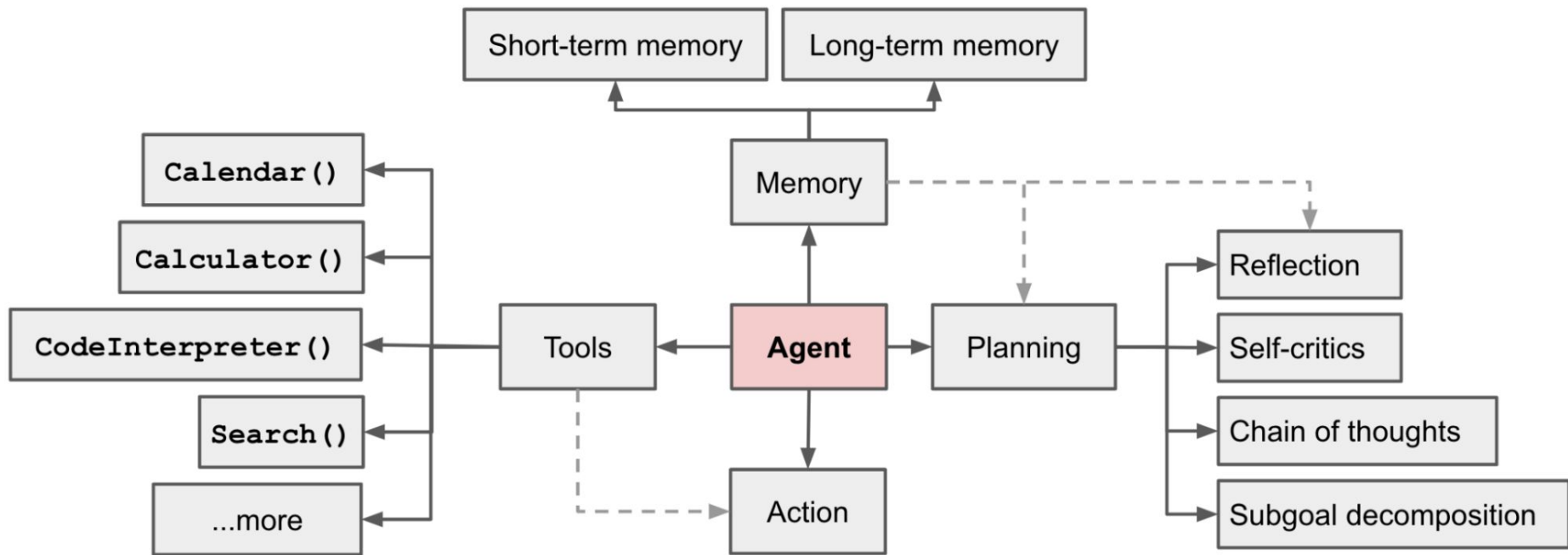


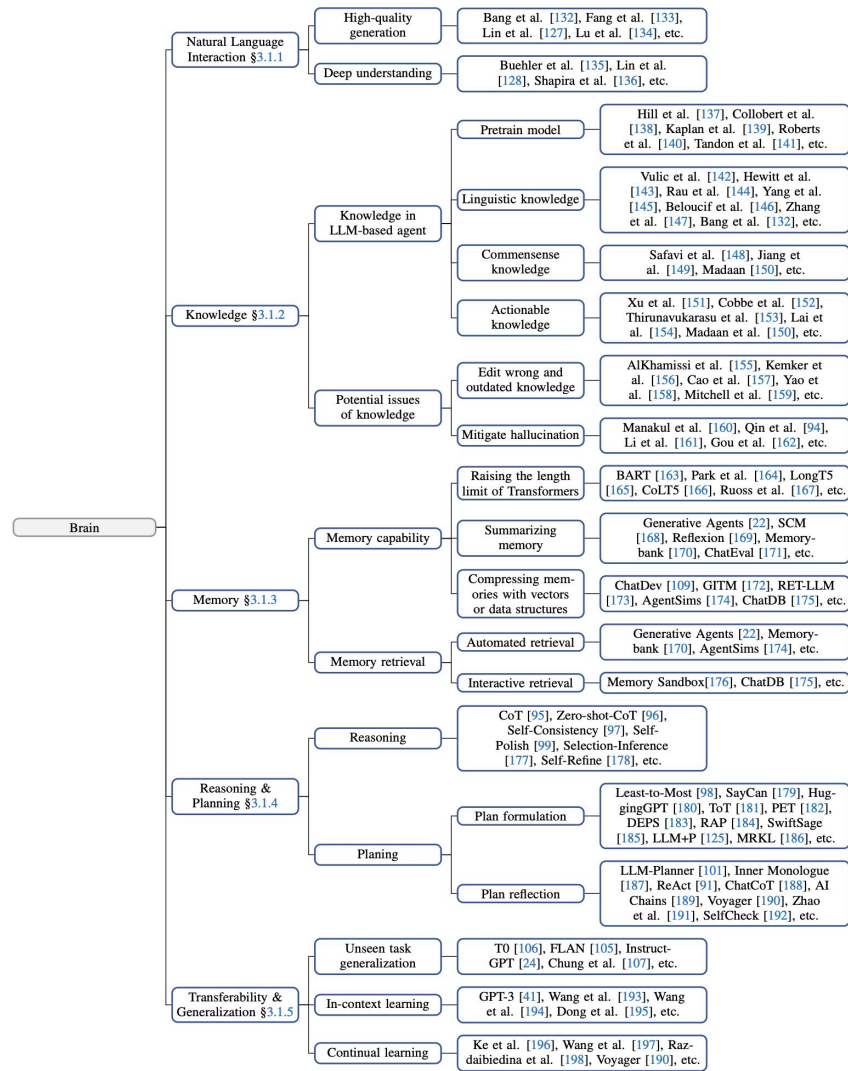
Well, its calendar...





Putting everything together





Agno: <https://github.com/agno-agi/agno> (previously: Phidata) ★



```
import os

os.environ['AZURE_OPENAI_API_KEY'] = "****"
os.environ['AZURE_OPENAI_ENDPOINT'] = "https://<resource-name>.openai.azure.com"
os.environ['AZURE_API_VERSION'] = "2024-10-21"
os.environ['AZURE_DEPLOYMENT'] = "****"

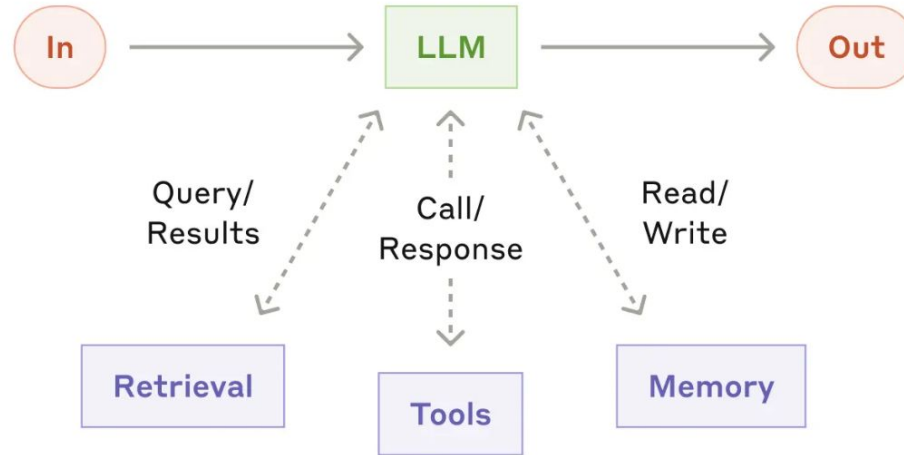
from agno.agent import Agent
from agno.models.azure import AzureOpenAI
from agno.tools.duckduckgo import DuckDuckGoTools

agent = Agent(
    model=AzureOpenAI(id="gpt-4o-mini"),
    tools=[DuckDuckGoTools()],
    show_tool_calls=True,
    markdown=True,
)

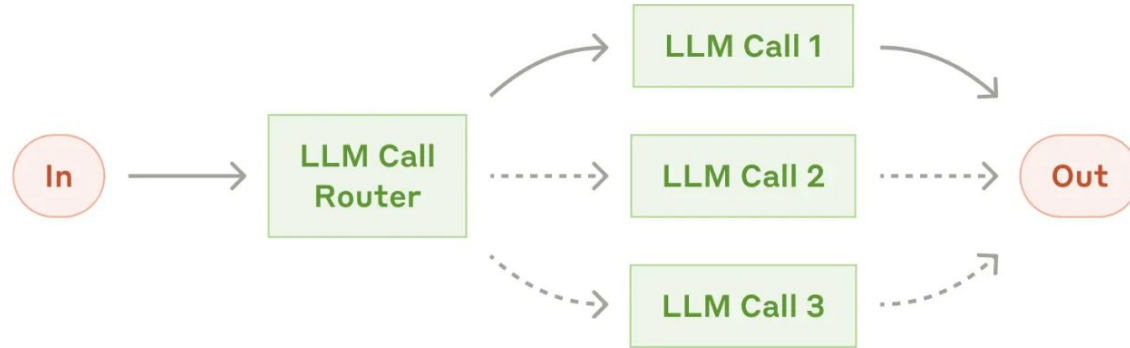
agent.print_response("Whats happening in Chennai today?", stream=True)
```

Different ways to build Agents (not-really)
ONE WORKFLOW AT A TIME

Simple LLM workflow



Routing



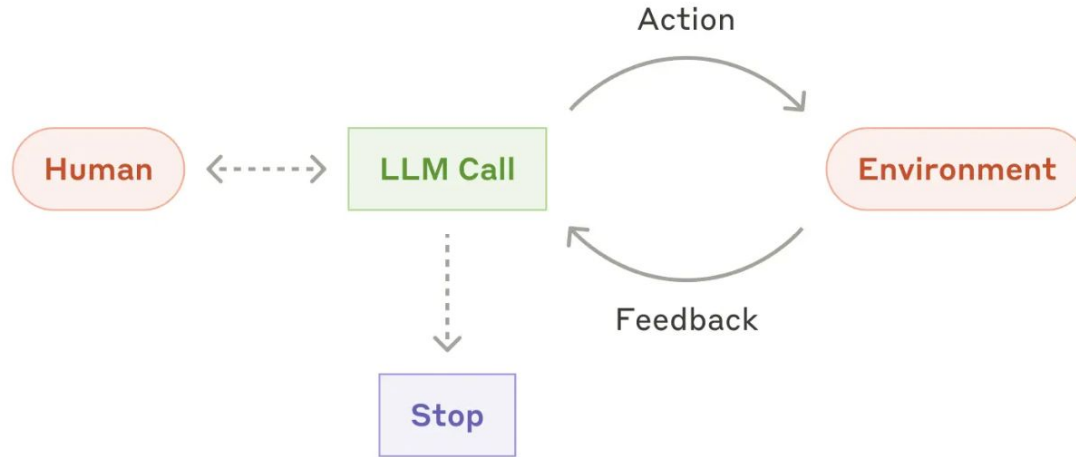
Parallelization



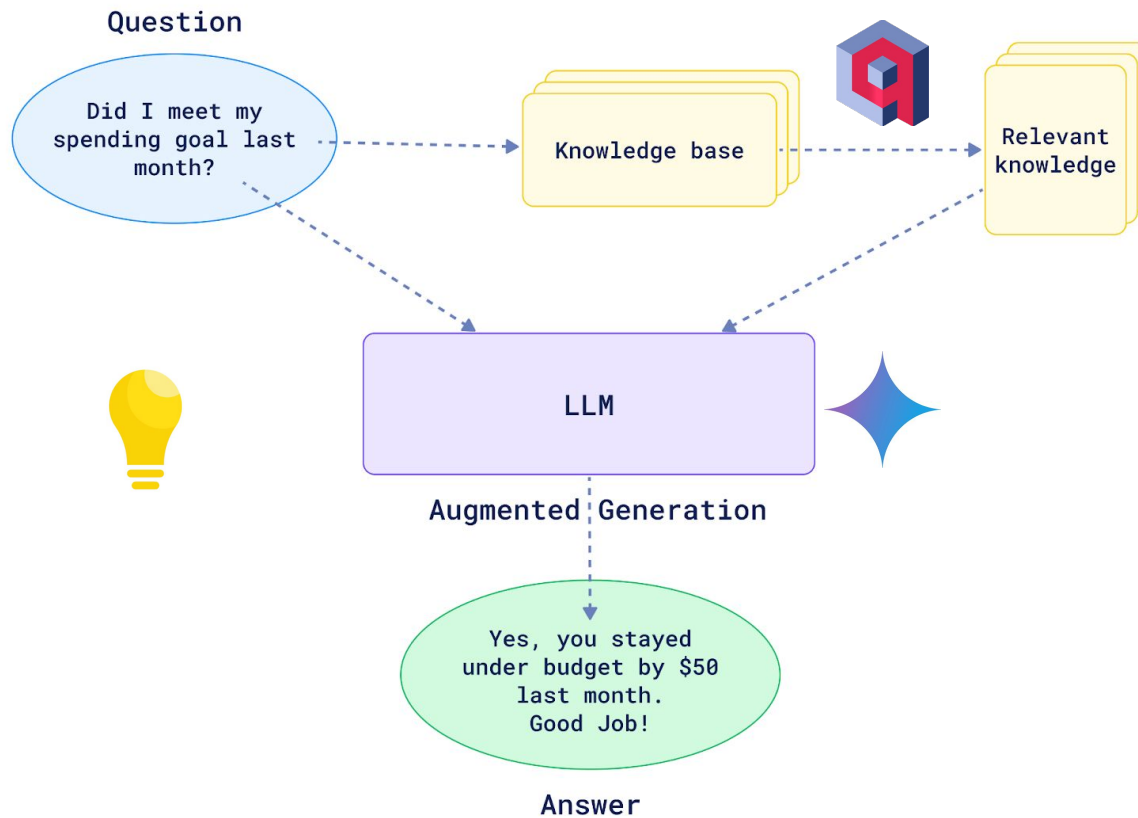
Multi-Agent Orchestrator

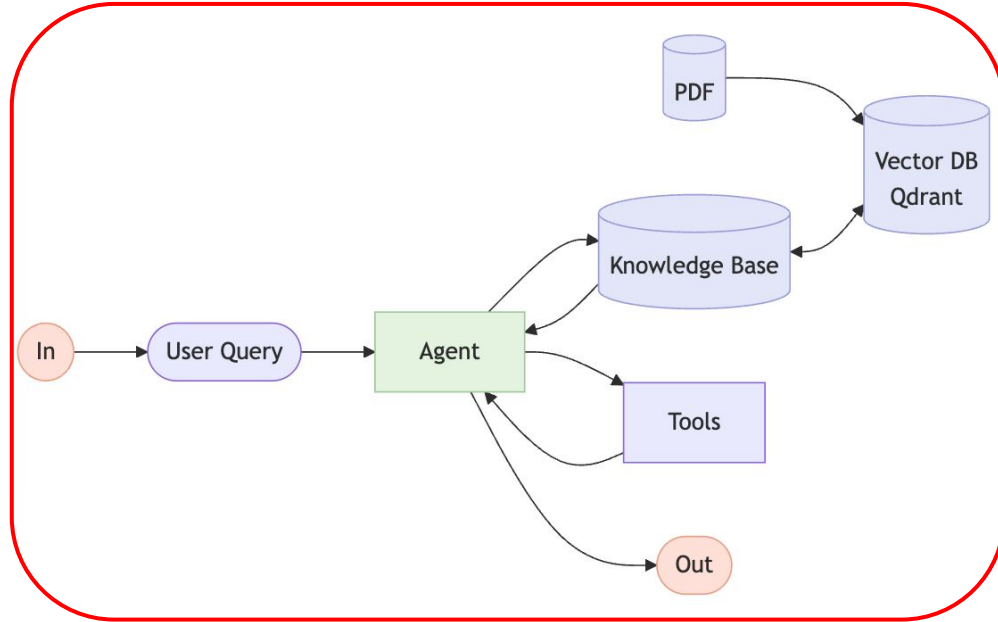


Agents



RAG– First time?





Qdrant & Gemini

First, let's set up our retriever with vector database and embedding model.

- cloud.qdrant.io
- aistudio.google.com/

```
import os  
from agno.vectordb.qdrant import Qdrant  
from agno.embedder.google import GeminiEmbedder
```

```
os.environ['GOOGLE_API_KEY'] = "<your-api-key>"  
embed_model = GeminiEmbedder()
```

```
COLLECTION_NAME = "agentic-rag"  
QDRANT_URL = "<your-endpoint-url>"  
QDRANT_API_KEY = "<your-api-key>"
```

```
vector_db = Qdrant(  
    collection=COLLECTION_NAME,  
    url=QDRANT_URL,  
    api_key=QDRANT_API_KEY,  
    embedder= embed_model)
```

Knowledge Base

Retrieval pipeline is ready with our own data (i.e., PDF).

```
from agno.knowledge.pdf import PDFKnowledgeBase
```

```
knowledge_base = PDFKnowledgeBase(  
    path="Indian-Recipes.pdf",  
    vector_db = vector_db,  
    num_documents=3  
)
```

```
knowledge_base.load(recreate=False) # only once
```

Agentic Workflow

It's time to combine our knowledge base to the Agentic Workflow along with LLM.

```
from agno.agent import Agent
from agno.models.google import Gemini
from agno.tools.duckduckgo import DuckDuckGoTools

agent = Agent(name="agentic-rag",
               model=Gemini(id="gemini-2.0-flash"),
               tools=[DuckDuckGoTools()],
               instructions = "You have the knowledge base and the
                               search tool, based on the user question, choose either one of
                               them. If user question, not in knowledge base use search",
               knowledge=knowledge_base,
               markdown=True,
               search_knowledge=True
               )

agent.print_response("<user-query>",stream=True)
```




AI With Tarun

@AIwithTarun · 2.72K subscribers · 47 videos

🚀 "Learn AI for FREE with Tarun!" 🚀 ...more

twitter.com/TRJ_0751 and 5 more links

Customise channel

Manage videos

Home Videos Shorts Live Podcasts **Playlists** Posts 🔍

Created playlists



Innate Dojo

Public

[View full podcast](#)



Agents

Public

[View full playlist](#)



Open Source

Public

[View full playlist](#)



LlamaIndex

Public

[View full playlist](#)



Langchain - Open Source

LLMs

Public

[View full playlist](#)

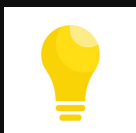


Large Language Models

(LLMs)

Public

[View full playlist](#)



★ More examples ★



- Blog to Podcast Generator Agent
- Different Multimodal Examples:
 - Image to Image
 - Audio to Text
 - Text to Video
 - and many more combinations
- Coding Agent

and more... 50+ cookbooks available on GitHub ★

- Tarun R Jain - LinkedIn
- @TRJ_0751- Twitter
- AI with Tarun - YouTube
- AlwithTarun.ai - Instagram

bit.ly/agent-ic-rag-chennai

