Microsoft

AI Tech Accelerator Community

Microsoft

Building Agents with Code for Production

3 March 2025

Agenda

	Conference Center 1	Conference Center 2 (next door)
10:00 – 12:00	**Building Agents with Code for Production** Florian Follonier, Sr. Partner Solution Architect for Data & AI	**Coworking Space**
12:00 – 13:30	**Lunch Break**	
13:00 – 17:00	**Hands-on Lab: Building Agents with code for Production** Supported by Florian Follonier, Juan Manuel Servera Bondroit & Martin Abrle	

Reminder: rules of the game

REGISTER AT LEAST 3 BUSINESS DAYS BEFORE THE NEXT APPOINTMENT

(THURSDAY BEFORE THE EVENT)

IF YOU CAN'T MAKE IT – LET US KNOW

PLEASE ALWAYS USE SWISS-SU@MICROSOFT.COM TO CONTACT US

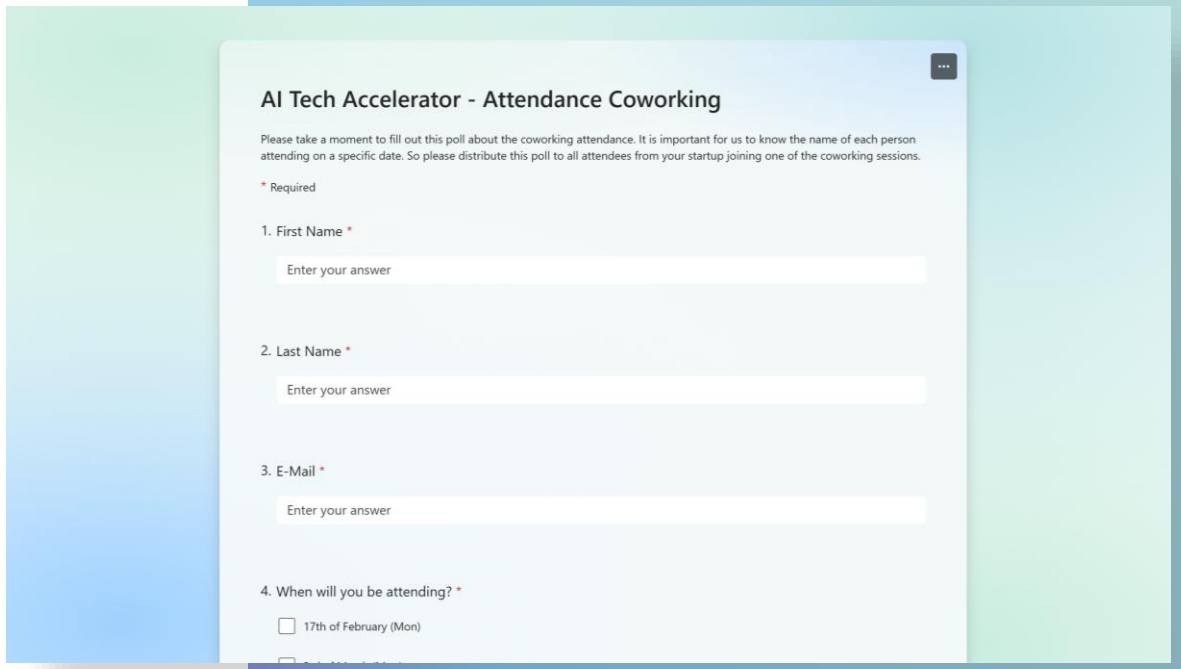
IF SOMETHING DOESN'T WORK, DOESN'T FEEL RIGHT OR COULD BE BETTER – TELL US

MUTUAL RESPECT

BE CURIOUS

Register for Coworking

- Make sure to always register – at least 3 business days in advance: AI Tech Accelerator Attendance
- A maximum of 2 people per startup / company are allowed per day



VOTE FOR THE NEXT 1:MANY SESSION

REQUEST 1:1 EXPERT SESSION

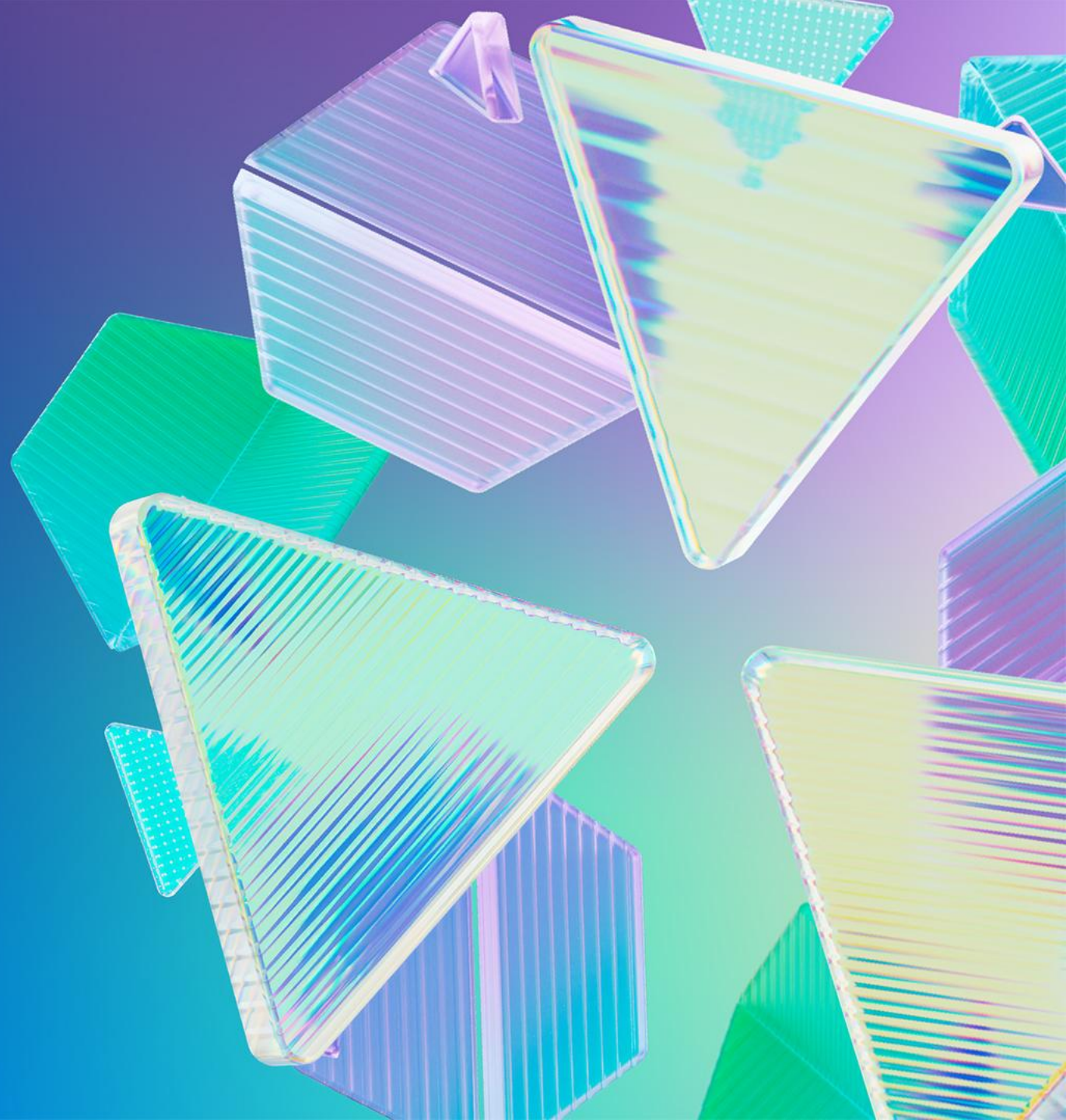
https://aka.ms/AIrepo

Microsoft

Exploring Agentic Systems with Azure AI Agent Service

Florian Follonier

Sr. Partner Solution Architect Data & AI



Agenda

What is an Agent

Agentic building blocks and patterns

Single-Agent demo

Multi-Agent patterns

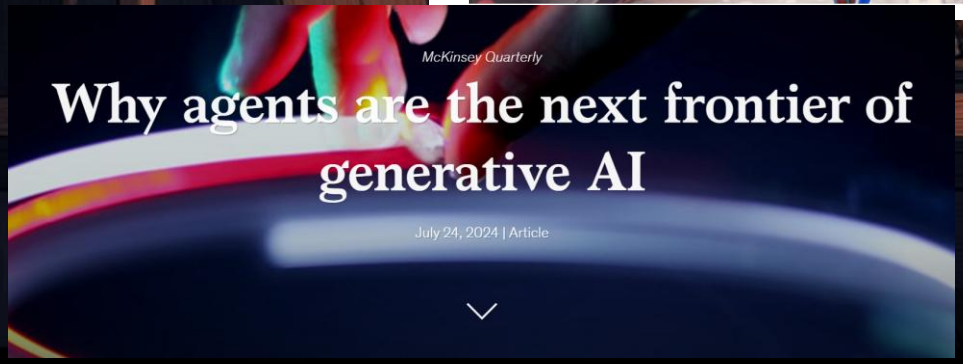
Build Your Own Agent

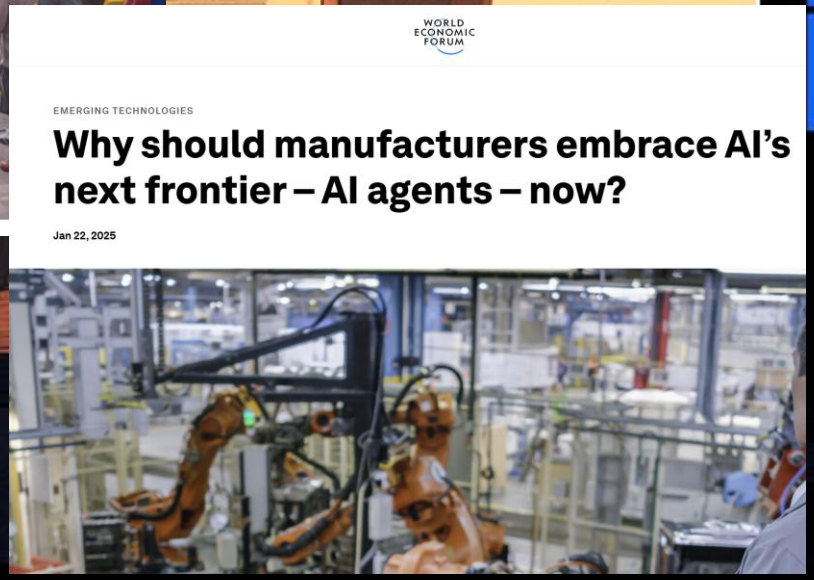

AI AGENTS ARE HERE
Y


2025:
YEAR OF
AGENTS
9:56


Why context-aware AI agents will give us superpowers in 2025


What are
AI Agents?


McKinsey Quarterly
Why agents are the next frontier of generative AI
July 24, 2024 | Article

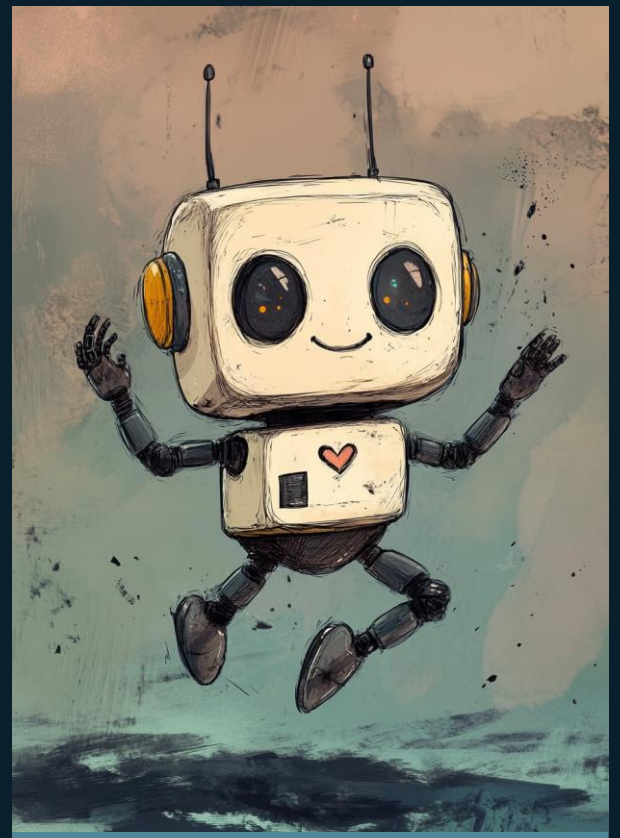

WORLD ECONOMIC FORUM
EMERGING TECHNOLOGIES
Why should manufacturers embrace AI's next frontier – AI agents – now?
Jan 22, 2025

GenAI has come a long way

ML, Deep Learning & BERT, etc.
<2021

LLMs, ChatGPT & Dall-E3
2022

Chat with Your Data (RAG)
Early-2023



Agents

Mid-2024

First wave of generative AI Apps

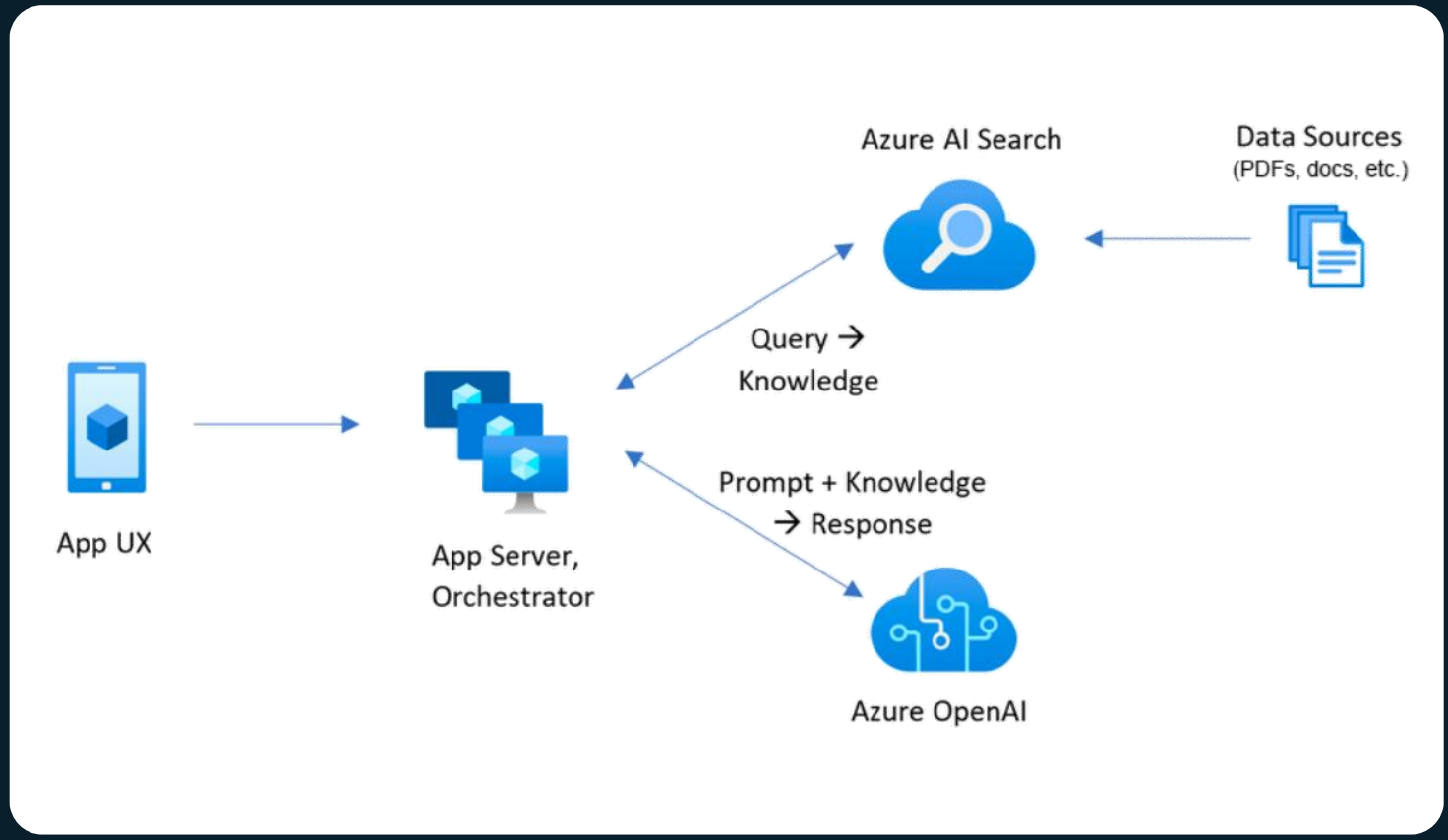
Common use cases:

- Conversational chat on private data
- Text/Document/Audio summarization and classification
- Image description and entity extractions
- Personalized content generation

Prompt engineering

RAG pattern

Application flow is hard-coded



Next wave: Agents

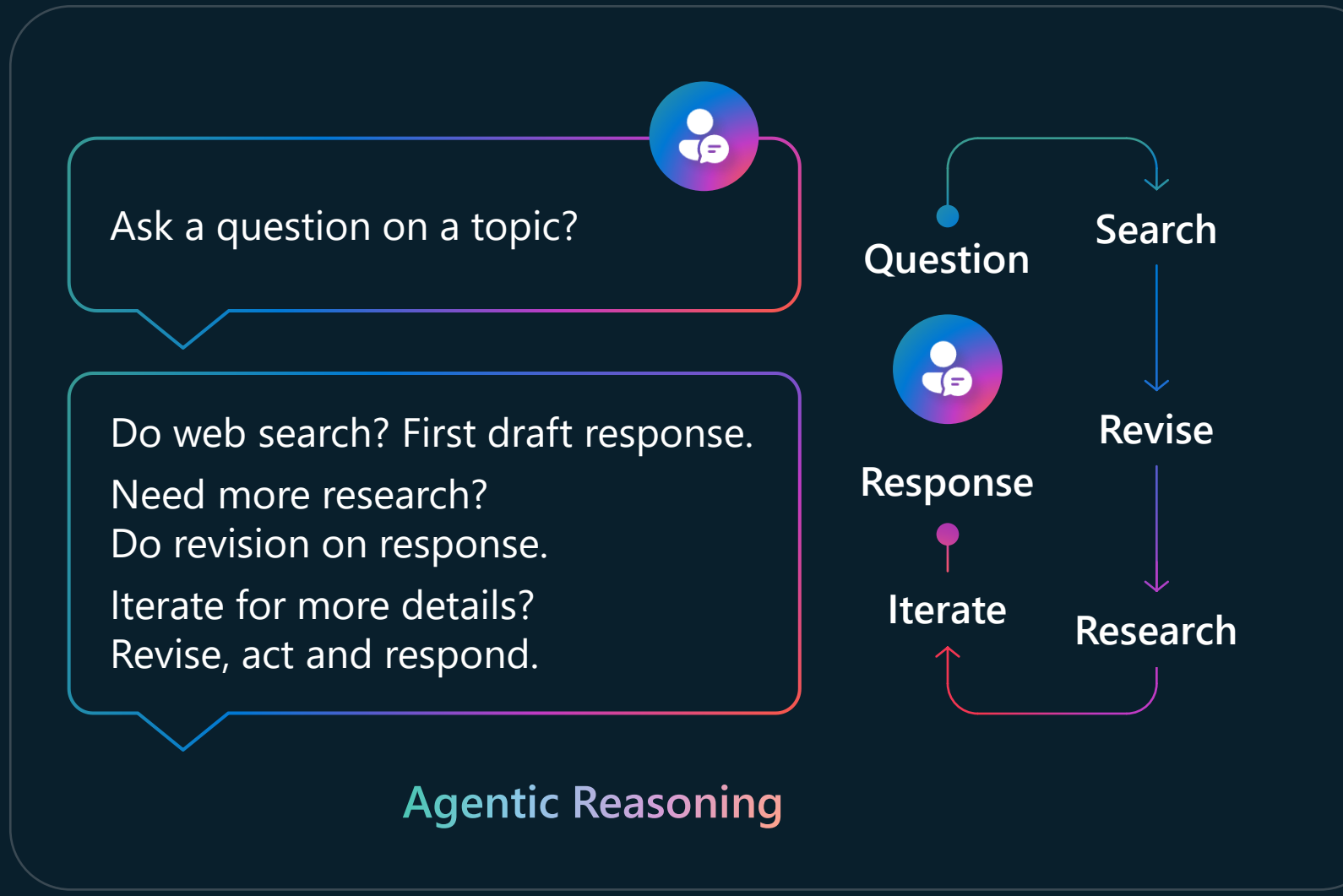
Complex interactions & orchestration

- Virtual assistants
- Customer support
- Intelligent code editors

Tools calling

Many LLM tasks + steps undefined sequence = agentic reasoning

Improve efficiency and accuracy



Agent frameworks and services

Semantic Kernel Agent Framework

Autogen

Langgraph

Azure AI Agent Service

Agents

What is an Agent?

"System that uses a LLM to decide the control flow of an application."

Autonomy Levels:

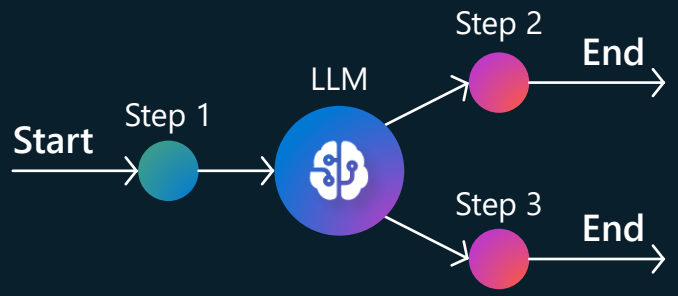
- **No Autonomy:** Traditional RAG
- **Simple:** Paths routing
- **Fully:** Multi-step reasoning & acting

Architectures:

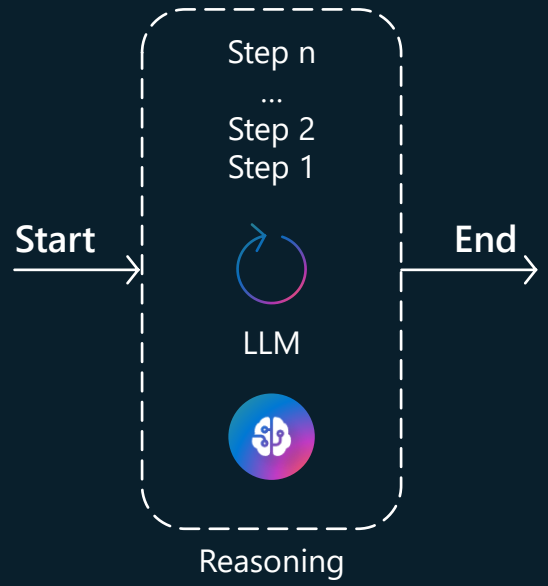
- Single agent
- Multi agent

Wave 1 (2022) -> Wave 2 (2025)

Simple LLM-enabled



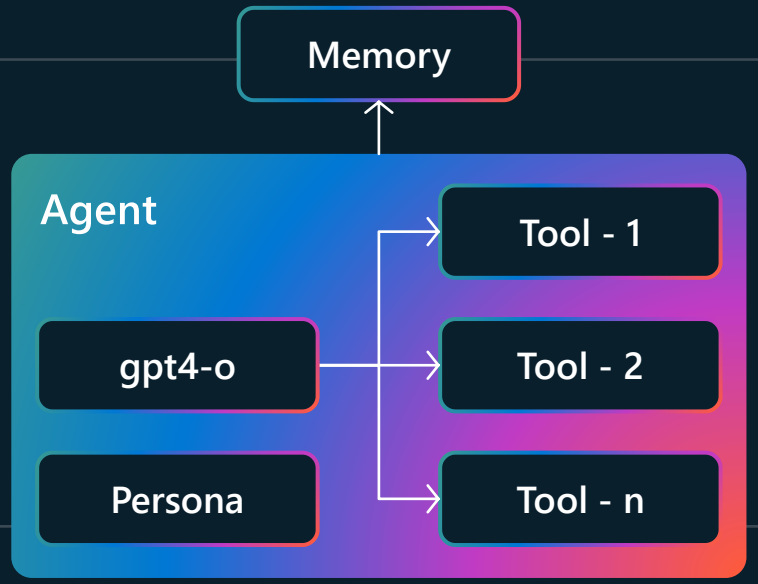
Fully autonomous



Agent Abstractions - Agent First-Class Citizen

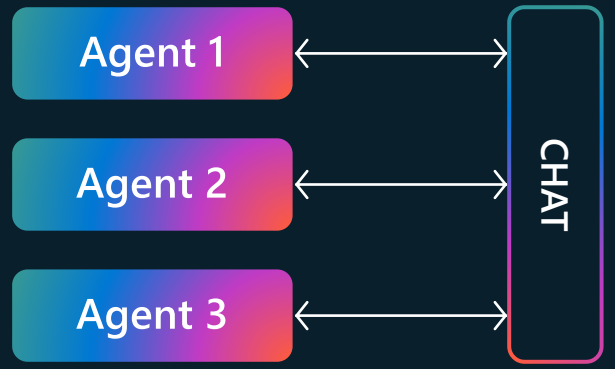
Agent as high-level abstraction

- **LLM** (gpt4-o, o1 etc.)
- **Persona** (system prompt)
- **Tools** (function code calls)

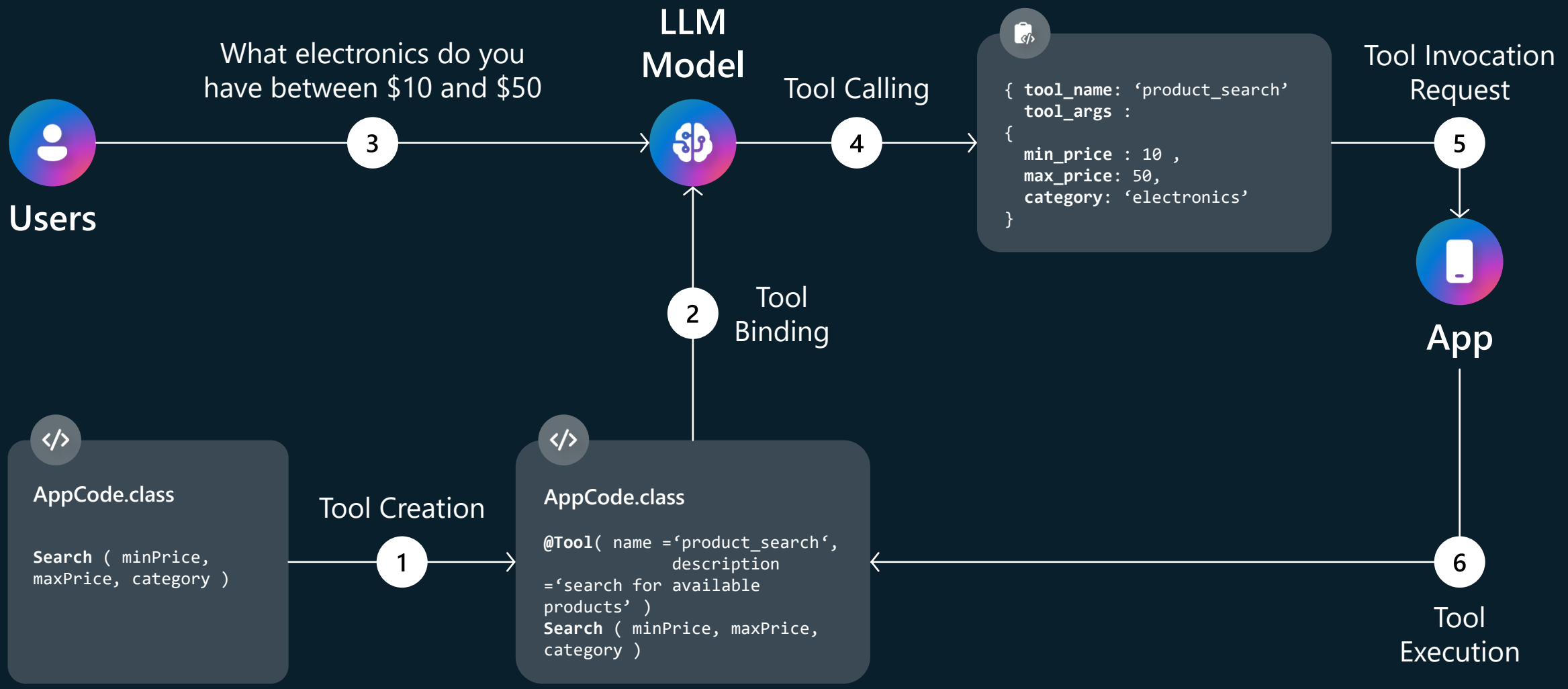


Agent Chat as layer for collaboration

- **Multiple agents** can engage with each other
- Enables multi-turn or parallel execution



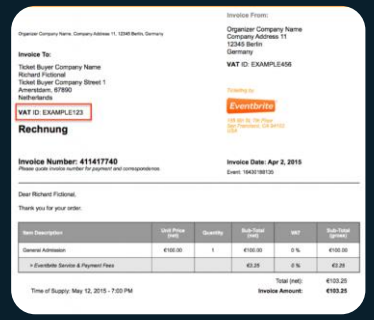
Agentic Pattern - Tools Calling



Agentic Pattern - ReAct Planning with Tools Calling

Users

Pay this bill for me



bill-abc123.jpeg

Bill abc123
successfully paid

Payment Agent

Tools:

- scanImage (filename)
- transactionHistory (billId)
- paymentService (billId, Amount, Payee)

Instructions:

- You are a home banking assistant allowing users to pay the bill uploading a picture
- Always check if a bill has already been paid before submitting a payment
- Confirm the payment result

Planning

while (new tools execution request)

1. **scanImage**
   (bill-abc123.jpeg)
   → abc123, 100$, payeename
2. **transactionHistory**
   (abc123)
   → no results
3. **paymentService**
   (abc123, 100$,payeename)
   → ok

Agentic Pattern - Memory

Short Term

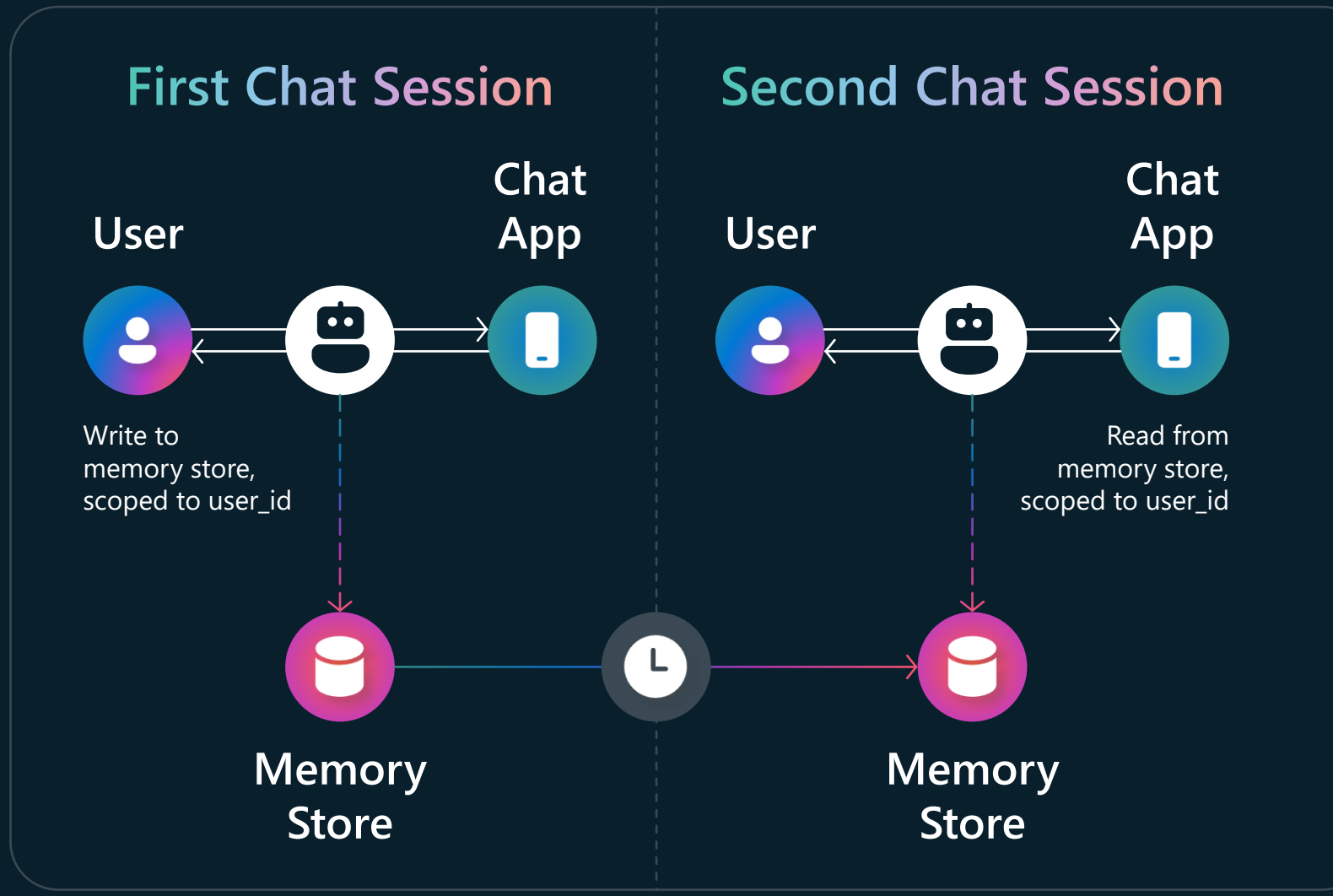
- Access steps info in one loop iteration
- Shared state context
- Chat history

Long Term

- Access steps info in long running conversation
- State persistence

Conversation History Truncation

- Trim by tokens
- Trim by message count
- Trim + summary (LLM call required)



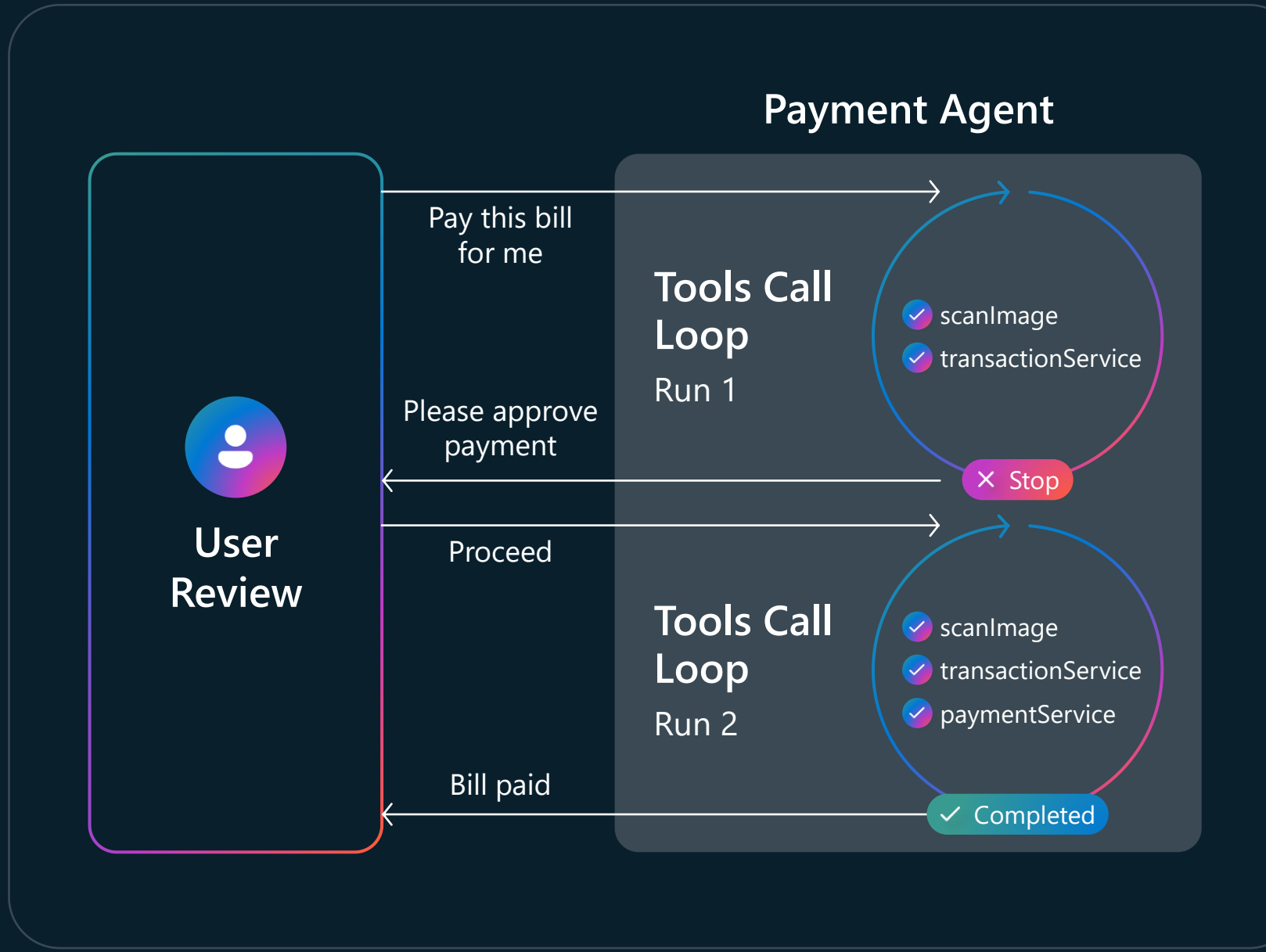
Agentic Pattern - Flow control

Looping Termination

- MaxIterations
- Message termination
- Human step /Human in the loop

Human in the loop

- Action execution approval
- Escalation
- Data review

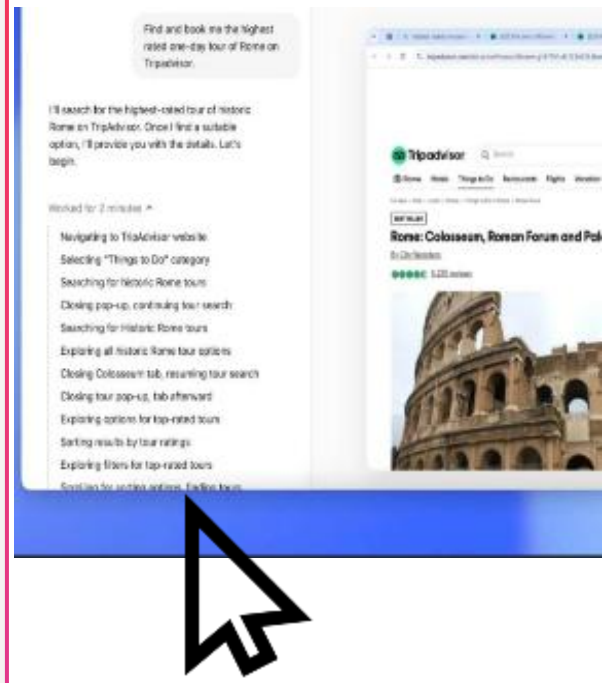


Agent Examples

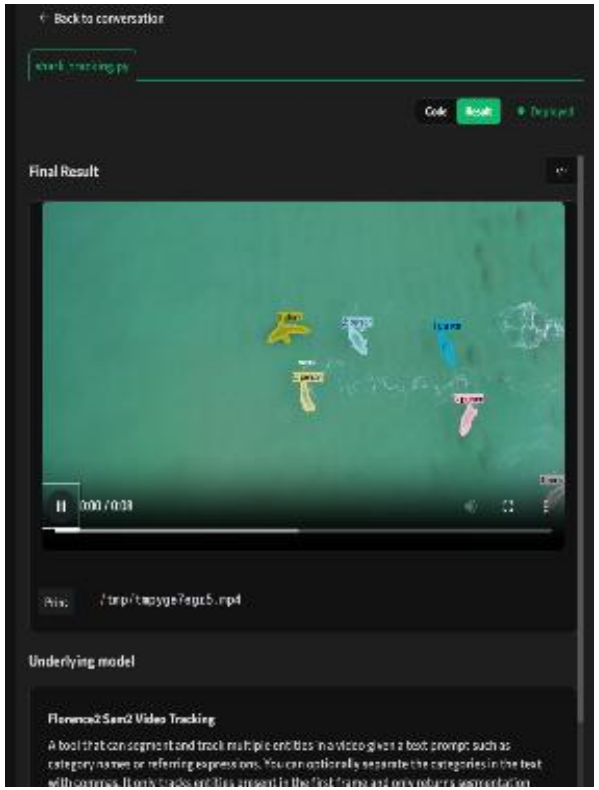
Search Agents
like Copilot in Teams

Coding Agents
like GitHub Copilot (Chat)

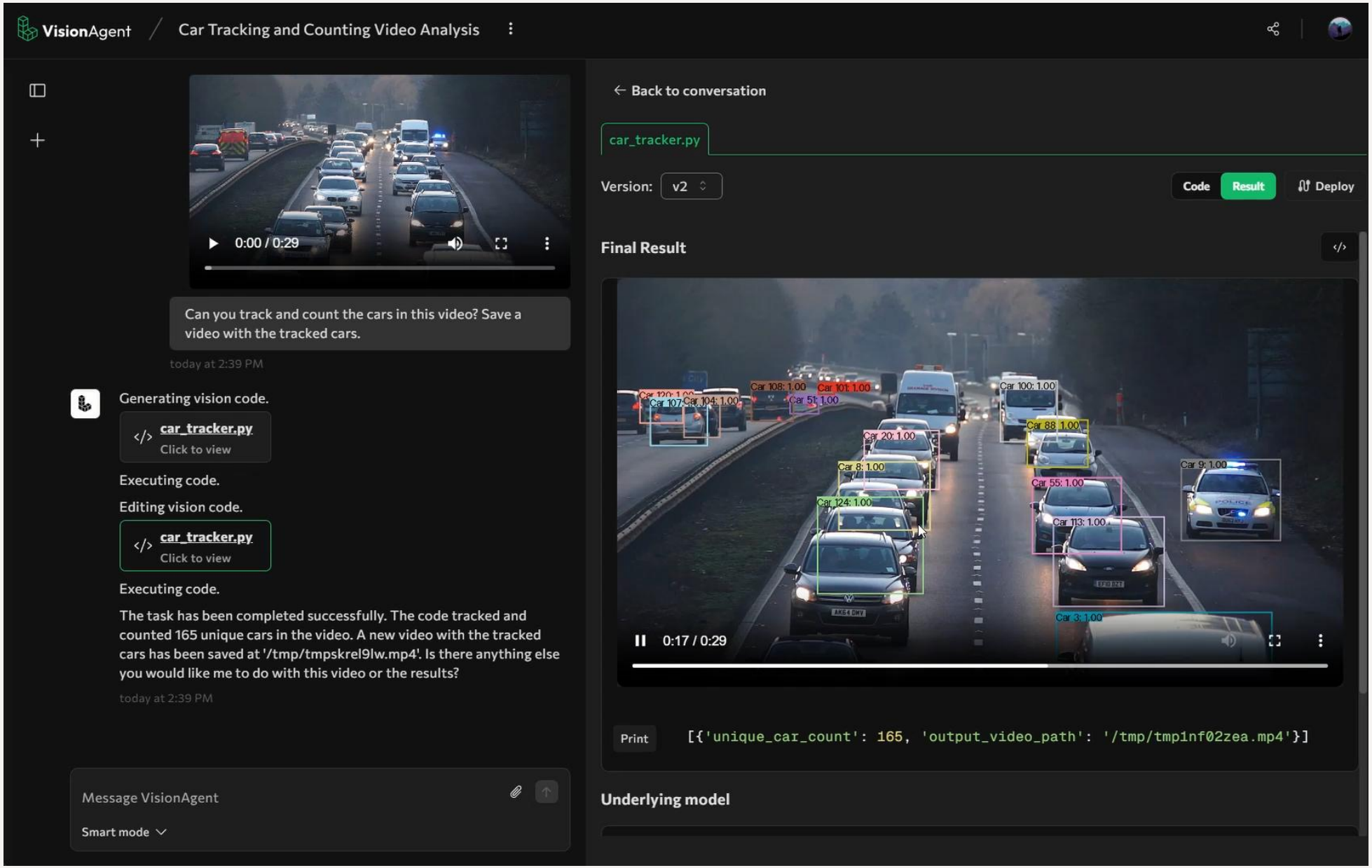
UI Task Agents
like OpenAI Operators



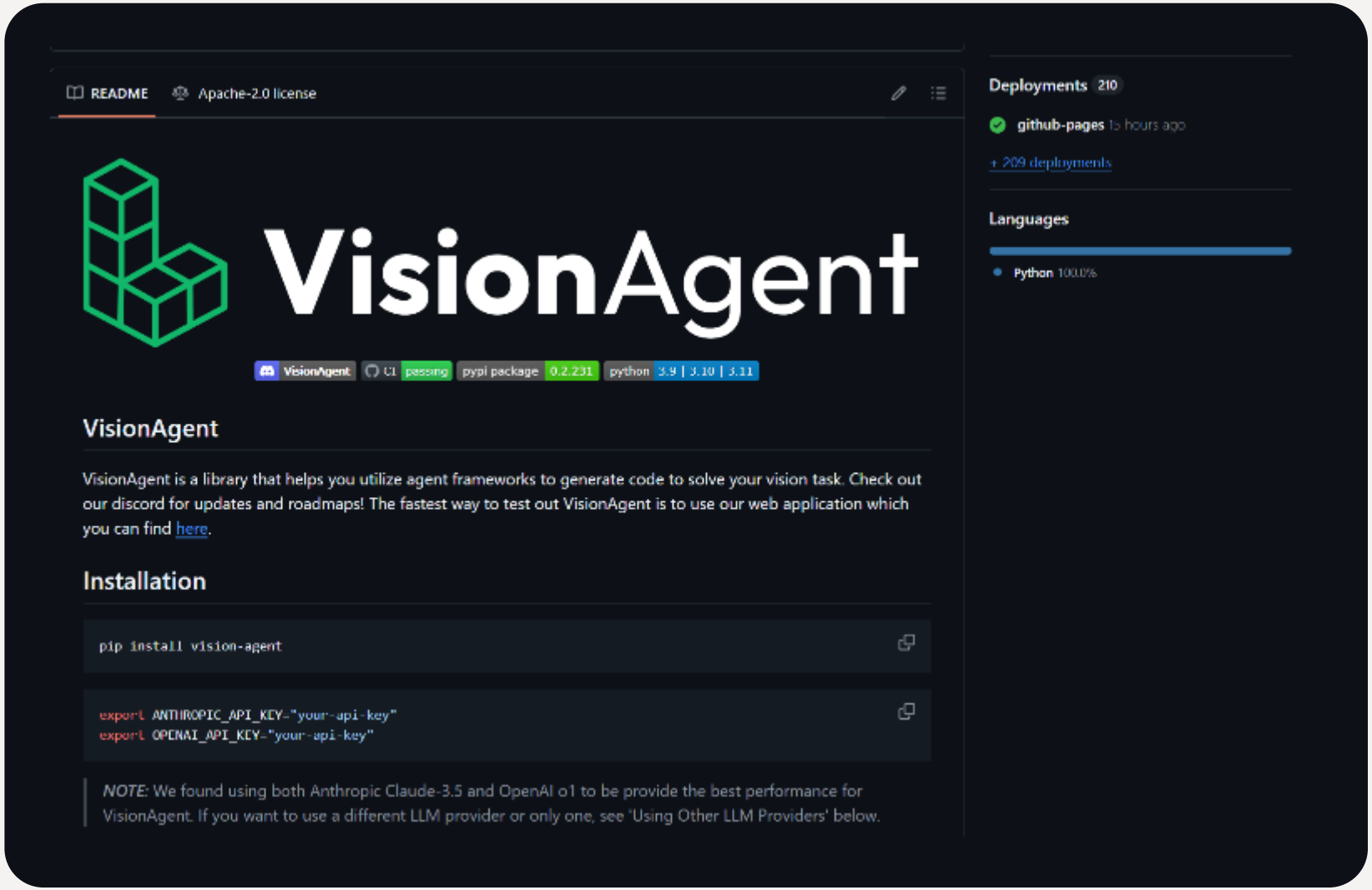
Data Science
like Open Source Vision Agent



Example for An Advanced Agent: Vision Agent



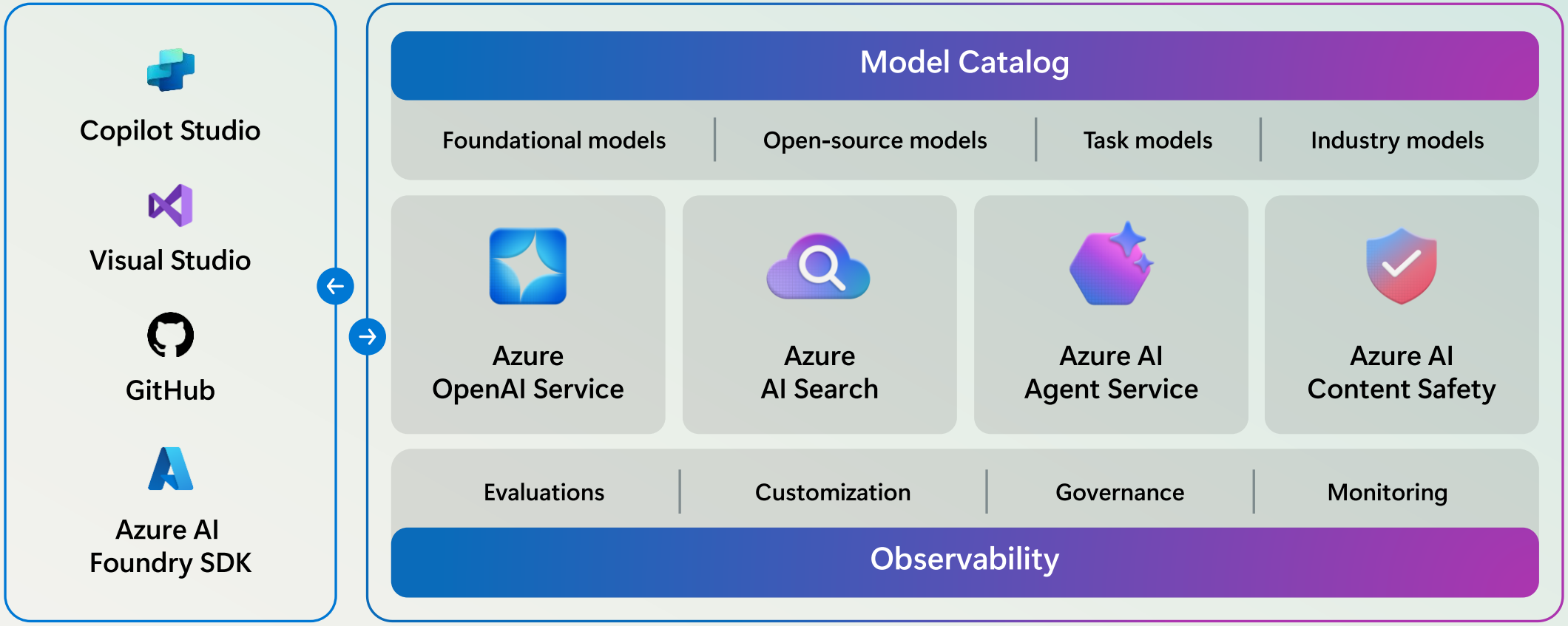
Example for An Advanced Agent: Vision Agent



Azure AI Agent Service

Announcing

Azure AI Foundry

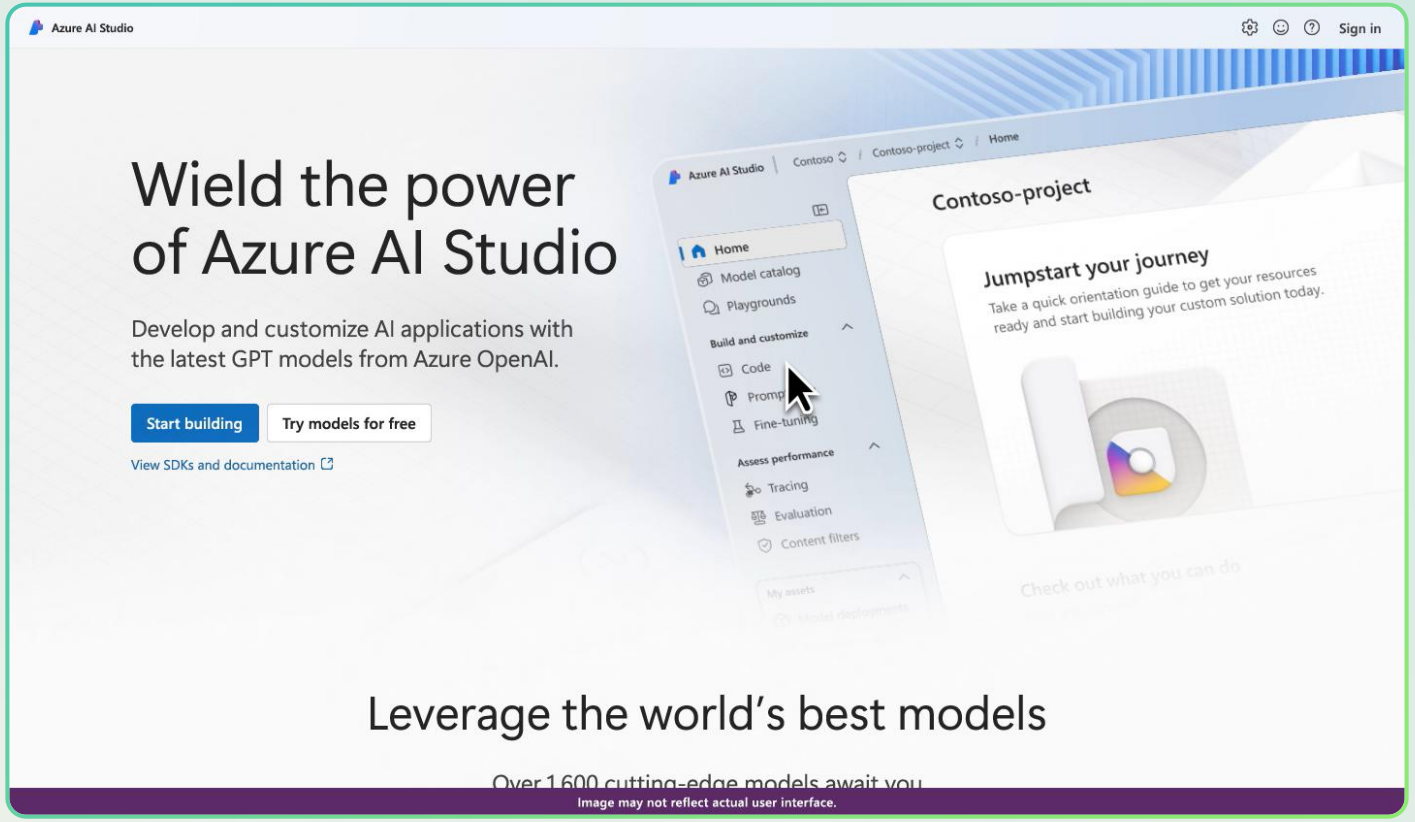


Azure AI Foundry SDK

The Azure AI Foundry SDK is a comprehensive toolkit for developers, offering pre-built modules and resources to integrate AI functionalities seamlessly into applications.

- Access our most popular models through a single interface
- Easily Azure AI capabilities into your application with a single project client
- Unlock another level of intelligence with Azure AI Agents
- Integrated tracing enables you to log back to AI Studio projects
- Evaluate your apps locally, in the cloud, and in production using state-of-the art safety and quality evaluators
- Incremental Azure Building Block app templates beyond SDKs, including templates for copilot scenarios, hosted in web, container, function app, and more

Move Seamlessly between UI and Code



The Azure AI Foundry SDK provides a local developer experience that reduces the complexity of using multiple resources together in code when building AI apps and agents.

Azure AI Agent Service SDK

Azure AI Foundry SDK – Agent Service

Azure OpenAI Assistants API

- File Search
- Code Interpreter

+

Model Catalog

Azure OpenAI Service
(GPT-4o, GPT-4o mini)

Models-as-a-Service

- Llama 3.1-405B-Instruct
- Mistral Large
- Cohere-Command-R-Plus

+

Extensive Ecosystem of Tools

Knowledge

- Microsoft Fabric (coming soon)
- SharePoint (coming soon)
- Grounding with Bing Search
- Azure AI Search
- Your own licensed data (coming soon)
- Files (local or Azure Blob)

Actions

- Azure Logic Apps (coming soon)
- OpenAPI 3.0 Specified Tools
- Azure Functions

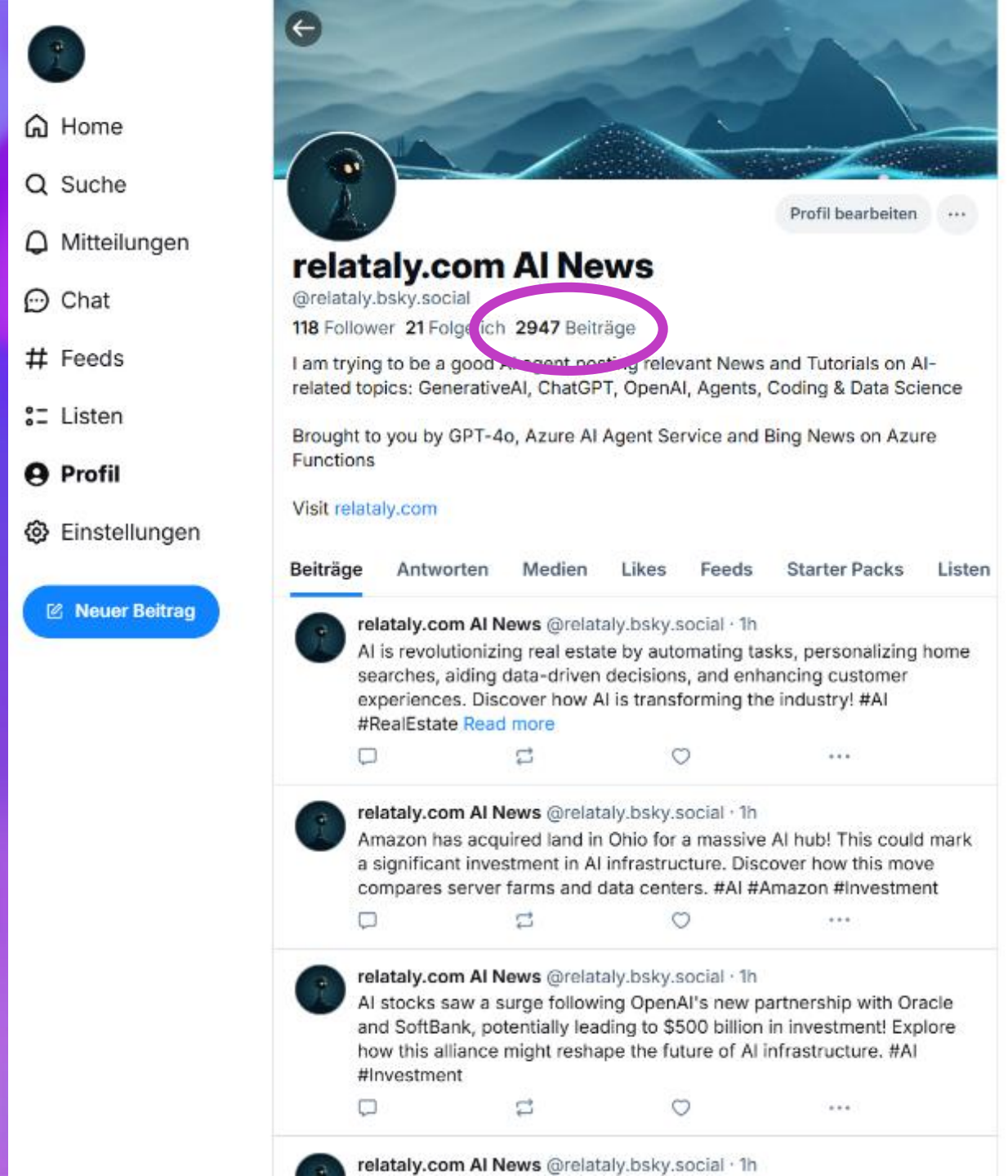
Built-In Enterprise Readiness

BYO-file storage (coming soon) | BYO-search index | BYO-thread storage | BYO-virtual network (coming soon) | OBO Authorization Support | Enhanced Observability

Practical Example
Social Media Agent

Microsoft

Bluesky
Social Media Agent



A “Classic” GenAI-infused Process

Wave 1 (2022)

LLM decision freedom is narrowed to three separate specific tasks

Time Trigger (every 4h)

Fetch News from Bing Search

Evaluate news for relevance

Check for duplicates with previous posts

Select best rated news and create post

Publish Post

simplified

Structured Responses: Curse and blessing

Wave 2 (2025)

Evaluating News

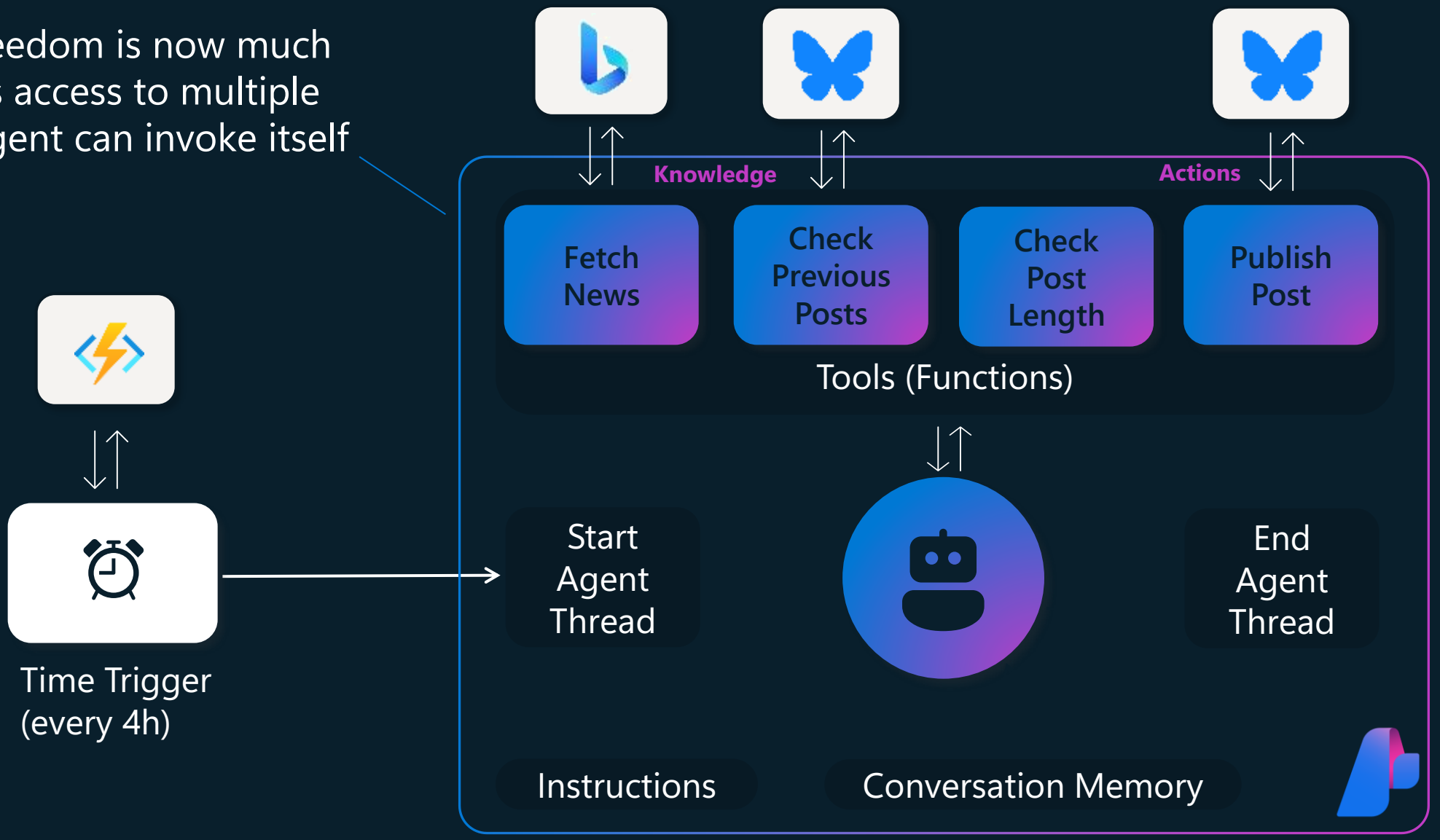
```python
#### Define OpenAI Prompt for news Relevance
def select_relevant_news_prompt(news_articles, topics, n):
    instructions = f'Your task is to examine a list of News Titles and return a list of boolean values that indicate which of the News Titles are in scope of a list of topics. \
    Return a list of True or False values that indicate the relevance of the News Titles.'
    task =  f"Which of the following news titles: {news_articles} are within the scope of these topics: {topics}?"
    sample = [
        {"role": "user", "content": f"Which of the following {n} News Titles: [new AI model available from Nvidia, We Exploded the AMD Ryzen 7, XGBoost 3.0 Making Decision Forest A
              are within the scope of these topics: {topics}?"},
        {"role": "assistant", "content": "[True, False, True, False, False]"},
        {"role": "user", "content": f"Which of the following {n} News Titles: [new AI model available from Nvidia, We Exploded the AMD Ryzen 7, XGBoost 3.0 Making Decision Forest A
              are within the scope of these topics: {topics}?"},
        {"role": "assistant", "content": "[True, False, True]"}]
    return instructions, task, sample
```

{ "relevant_news": [true, false, true] }

The Agentic Approach

Wave 2 (2025)

LLM decision freedom is now much wider and spans access to multiple tools that the agent can invoke itself

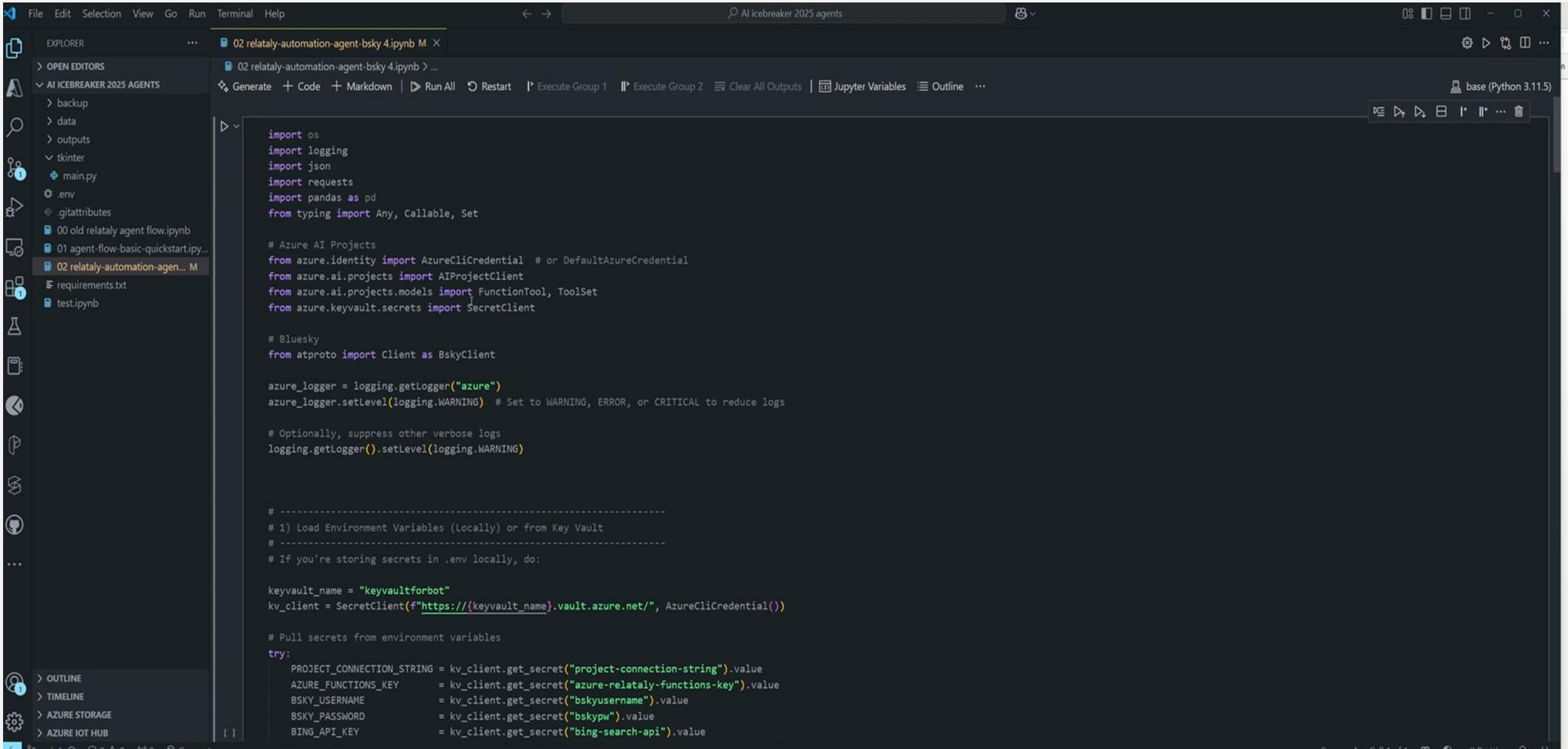


Agents allow for more robust workflows

```
instructions =

You are a helpful assistant with the goal to post about relevant AI news on Bluesky social media.
When a user requests, you will fetch the latest AI news and create a post on Bluesky with a tweet and a
link to the news article.
Follow these steps to ensure accurate and concise responses:

1. **Fetch News from Bing Search**: Always use the 'search_for_relevant_news_via_bingsearch' function
to retrieve any AI related news.

2. **Get Your Recent Bluesky Feed**: Always use the 'receive_previous_posts_from_bluesky_social_media'
function to retrieve the latest posts to avoid redundant posts.

3. **Evaluate the News**: Identify the most interest news article from Bing search results considering
previous posts to avoid posting about the same topic twice.

4. **Create a Tweet**: Create a tweet text about the selected news (avoid topics from previous posts).
Always use the 'check_tweet_length' function to ensure the tweet is within the 280-character limit.
Avoid adding the url into the text and instead provide the url as part of the
call_function_to_post_on_bluesky_social_media function.

5. **Error Handling**: If there are issues, inform the user about the problem and end the process.
```

Demo: Social Media News Agent

Wave 2 (2025)



Lessons Learned

Lessons Learned

Tooling

Tool design significantly impacts performance

Wrap APIs. Encapsulate functions into tool components.

Tools to overcome LLM limitations (e.g., for math).

Testing

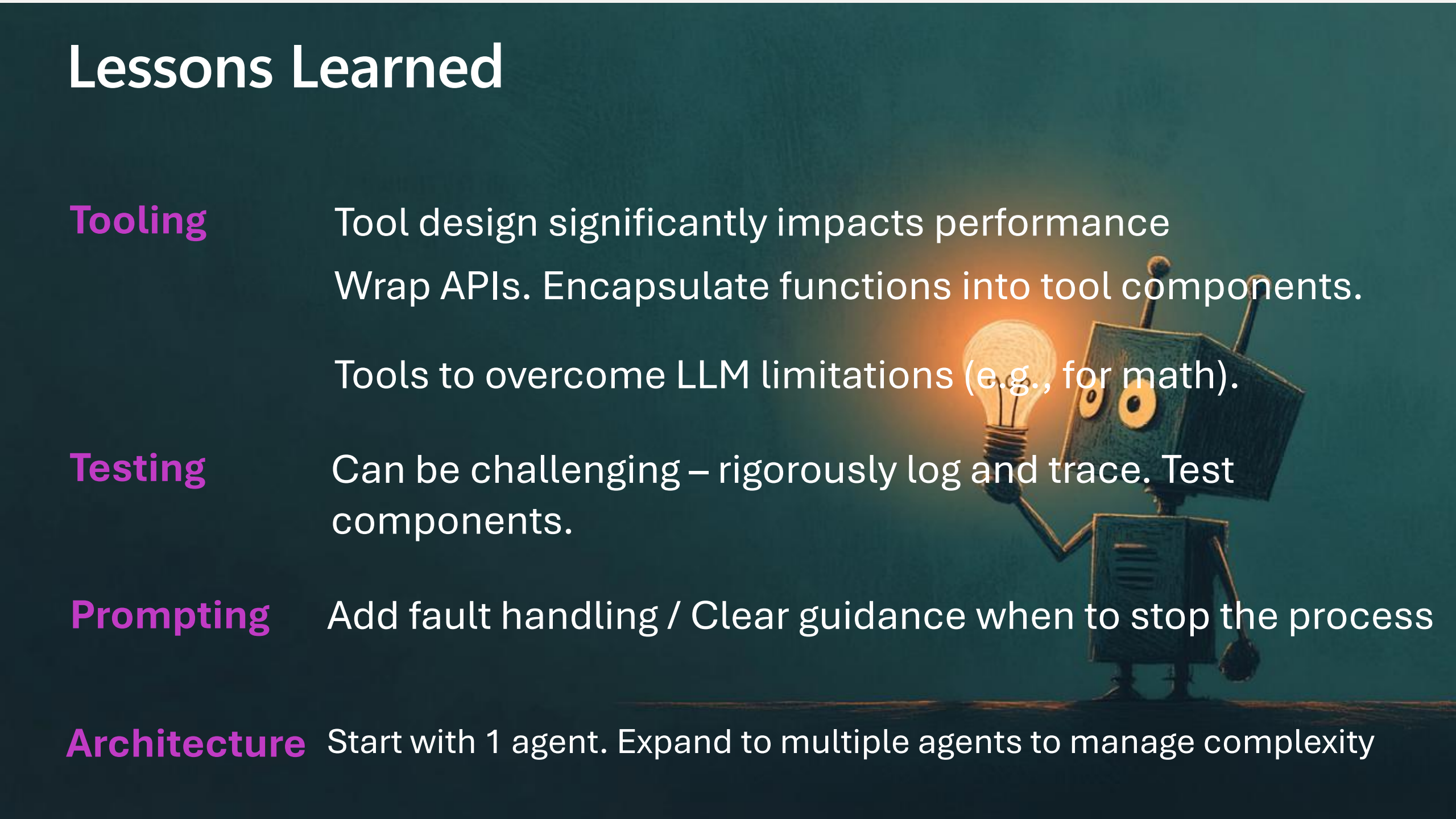
Can be challenging – rigorously log and trace. Test components.

Prompting

Add fault handling / Clear guidance when to stop the process

Architecture

Start with 1 agent. Expand to multiple agents to manage complexity



Multi-Agent Architectures

Single Agent Architecture - Scaling

As the system grows you might run into scaling challenges

Too many tools. Tool hallucinations

Agent context (a.k.a. prompt) grows too much and it fails to follow instructions

Handling complex and dynamics tasks spanning different business domains

Multi agent architecture opportunities

Manageability – Modular agents reduce development and testing complexity

Predictability – More control over application flow using structured agents communication

Flexibility – Ease to incorporate new agents as solution domains increase

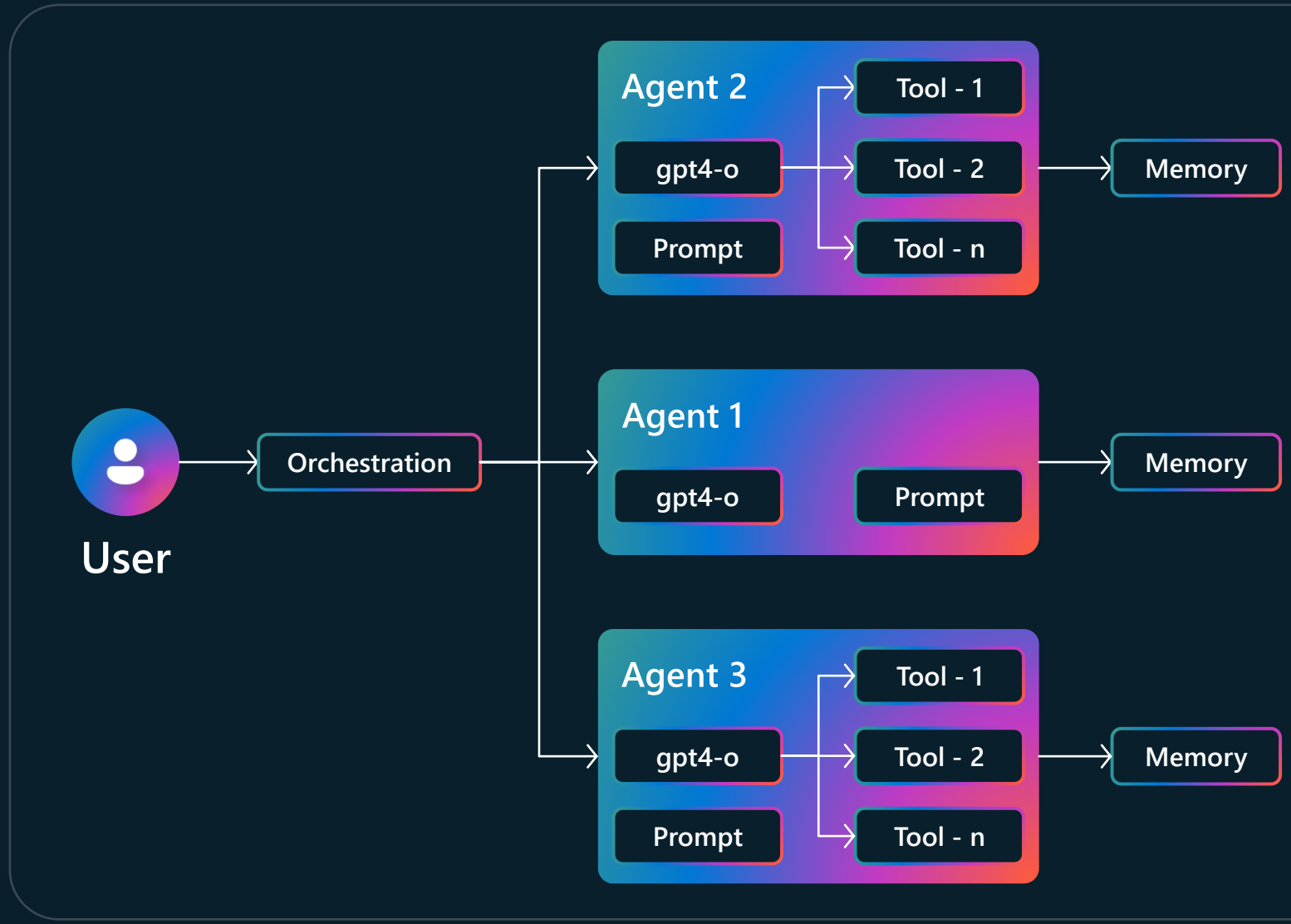
Multi Agent Logical Architecture

Each agent is specialized in different tasks or aspects of a problem

Agents can communicate and coordinate with each other. Structured orchestration is crucial

2 primary categories based on orchestration types

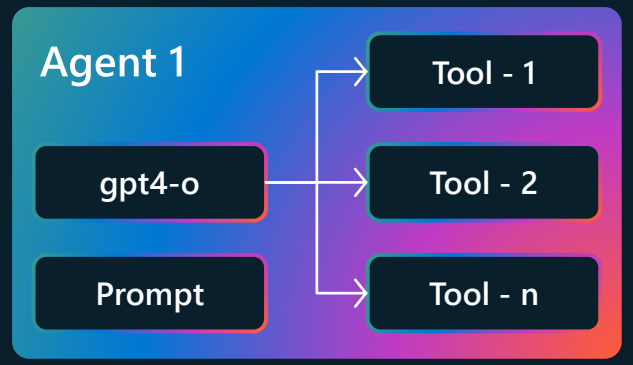
- Vertical Architecture
- Horizontal Architecture



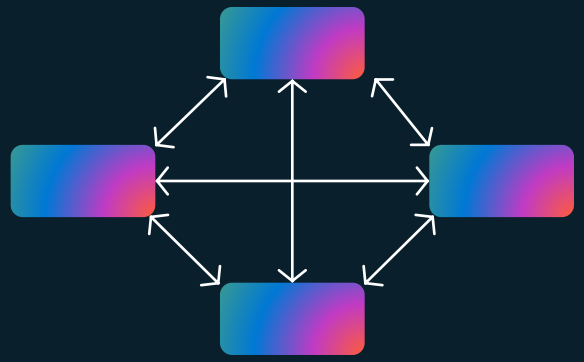
Agents orchestration and communication styles

Start here

Single Agent

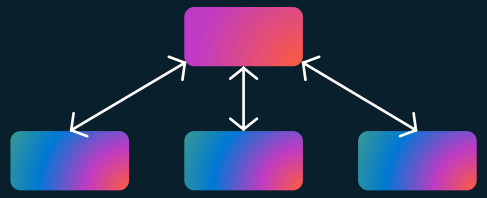


Network

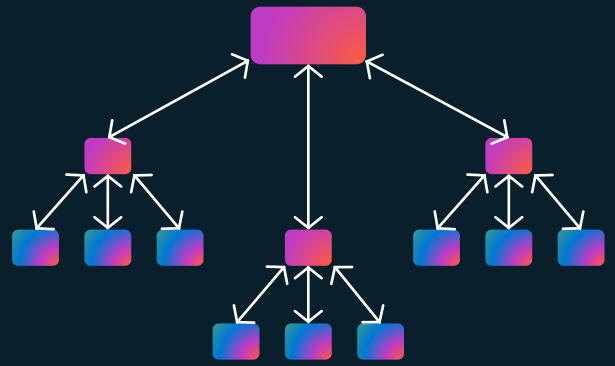


Reflection

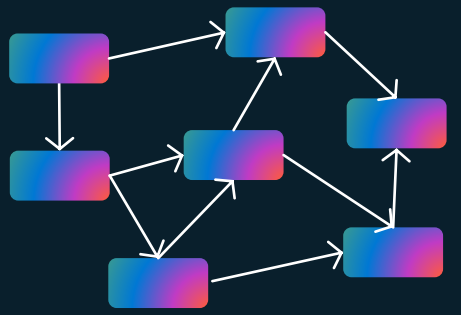
Supervisor



Hierarchical



Custom

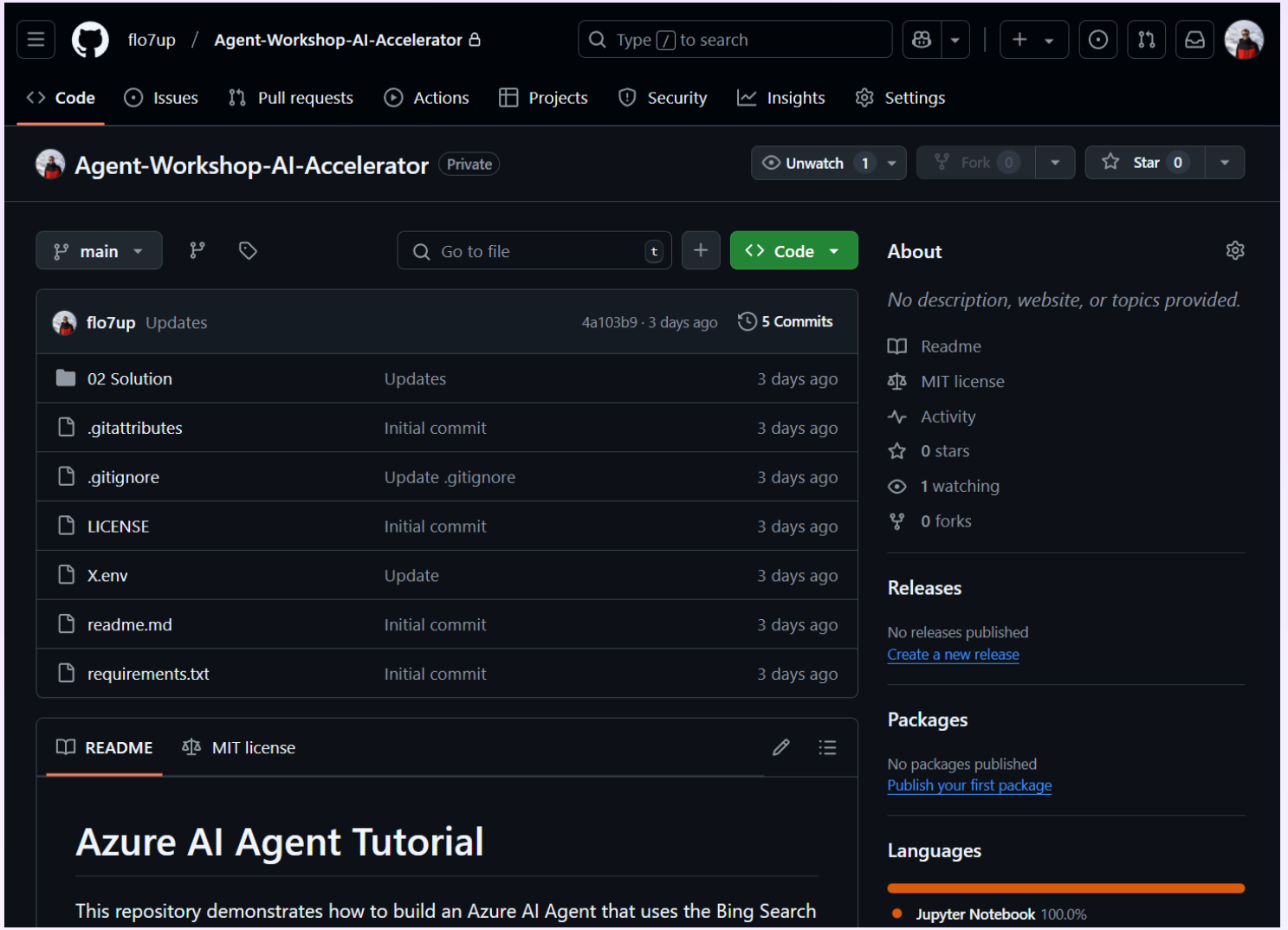


Build Your Own Agent!

GitHub Repo

[flo7up/Agent-Workshop-AI-Accelerator]

3bebe
313a6
604e0
4976f
01c7b
504dbb3



Thank You

Florian Follonier

Sr. Partner Solution
Architect Data & AI

Lunch

RESUME AT 13:30

Microsoft

Afternoon – Building Agents with Code for Production, Hands-on Lab