2.2 Building Agentic Systems Today

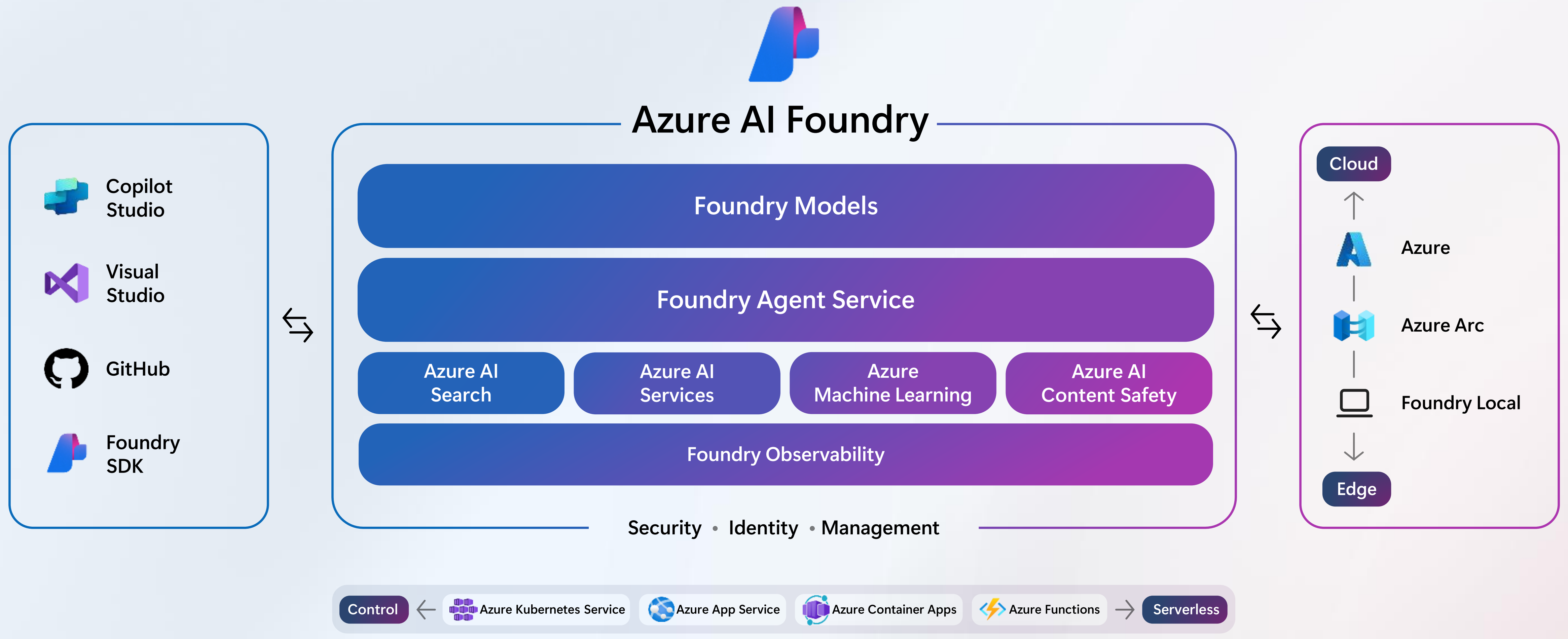
Tobias Unterhauser – Partner Solution Architect

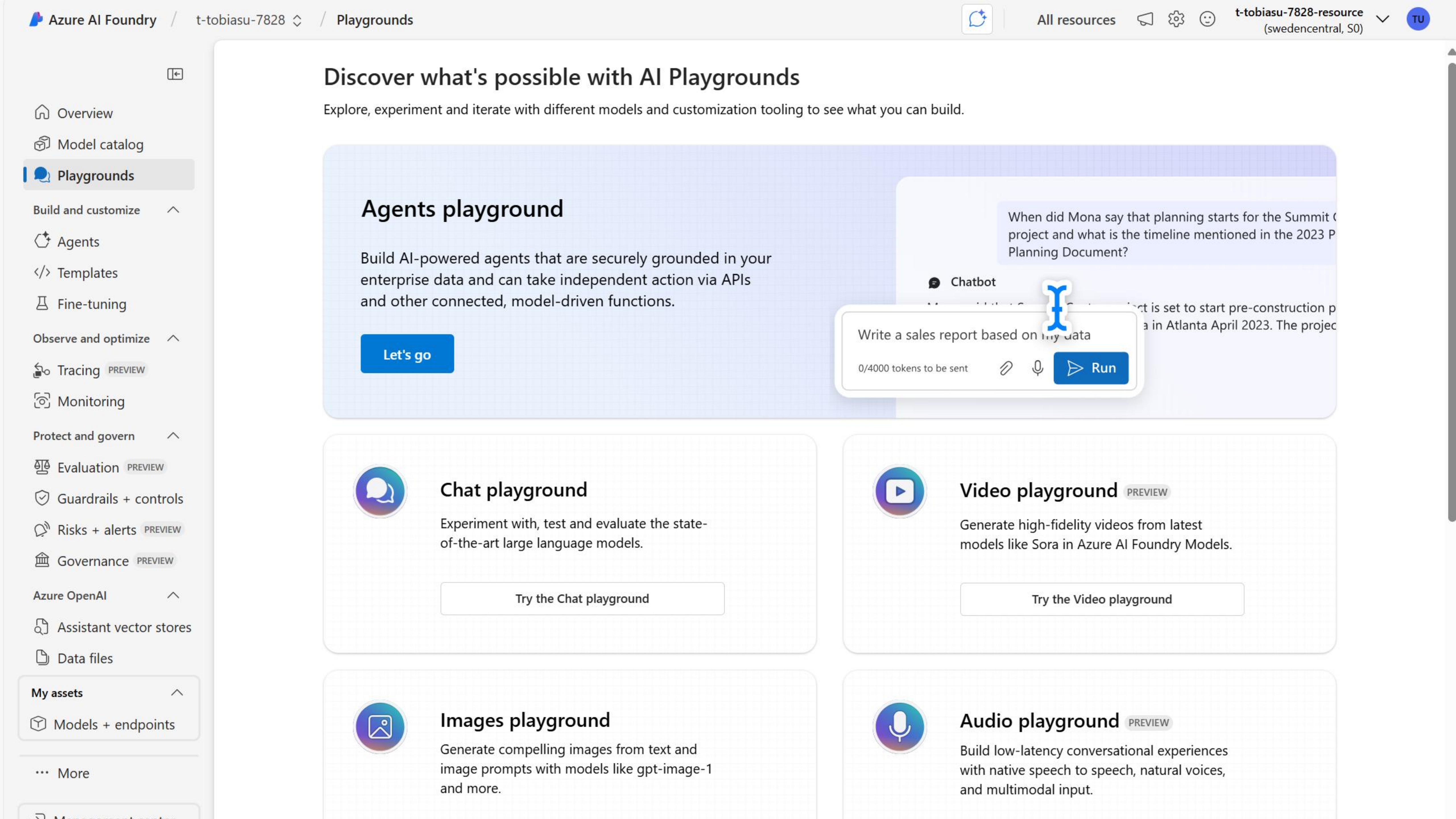
Microsoft has different ways of building agents

Platform Integrations, ease-of-use, and development speed →

IaaS	PaaS	SaaS
Infrastructure-as-a-service	Platform-as-a-service	Software-as-a-service
Semantic Kernel Agent framework & VM, Networking, Containers, and LLMs	Azure AI Foundry Agent Service	Copilot Studio Agents

← Control, visibility, and customization


Azure AI Foundry
Copilot Studio
Visual Studio
GitHub
Foundry SDK
Foundry Models
Foundry Agent Service
Azure AI Search
Azure AI Services
Azure Machine Learning
Azure AI Content Safety
Foundry Observability
Security • Identity • Management
Cloud
Azure
Azure Arc
Foundry Local
Edge
Control
Azure Kubernetes Service
Azure App Service
Azure Container Apps
Azure Functions
Serverless



Azure AI Foundry / t-tobiasu-7828 / Playgrounds

All resources

t-tobiasu-7828-resource (swedencentral, S0)

TU

Overview
Model catalog
Playgrounds

Build and customize
Agents
Templates
Fine-tuning

Observe and optimize
Tracing PREVIEW
Monitoring

Protect and govern
Evaluation PREVIEW
Guardrails + controls
Risks + alerts PREVIEW
Governance PREVIEW

Azure OpenAI
Assistant vector stores
Data files

My assets
Models + endpoints

More

Discover what's possible with AI Playgrounds

Explore, experiment and iterate with different models and customization tooling to see what you can build.

Agents playground

Build AI-powered agents that are securely grounded in your enterprise data and can take independent action via APIs and other connected, model-driven functions.

Let's go

When did Mona say that planning starts for the Summit (project and what is the timeline mentioned in the 2023 P Planning Document?

Chatbot

ct is set to start pre-construction p a in Atlanta April 2023. The projec

Write a sales report based on my data

0/4000 tokens to be sent

Run

Chat playground

Experiment with, test and evaluate the state-of-the-art large language models.

Try the Chat playground

Video playground PREVIEW

Generate high-fidelity videos from latest models like Sora in Azure AI Foundry Models.

Try the Video playground

Images playground

Generate compelling images from text and image prompts with models like gpt-image-1 and more.

Audio playground PREVIEW

Build low-latency conversational experiences with native speech to speech, natural voices, and multimodal input.

Azure AI Foundry Agent Service

Securely customize, orchestrate, and deploy AI agents

Model choice

Model choice and flexibility with the model catalog

Azure OpenAI Service

o1, o3-mini, GPT-4.1, 4o, etc

Models-as-a-Service

Llama 3.1-405B-Instruct

Mistral Large

Cohere-Command-R-Plus

AI tools

Richest set of enterprise connectivity

Knowledge

Actions

Logic Apps

Azure Functions

OpenAPI

MCP

Trust

Customer control over data, networking, and security

- BYO-file storage
- BYO-search index
- BYO-virtual network
- BYO-thread storage
- OBO authentication
- Content filtering

Microsoft’s six Responsible AI principles are built in AI Foundry

Microsoft’s Responsible AI principles

Fairness

Reliability
& Safety

Privacy
& Security

Inclusiveness

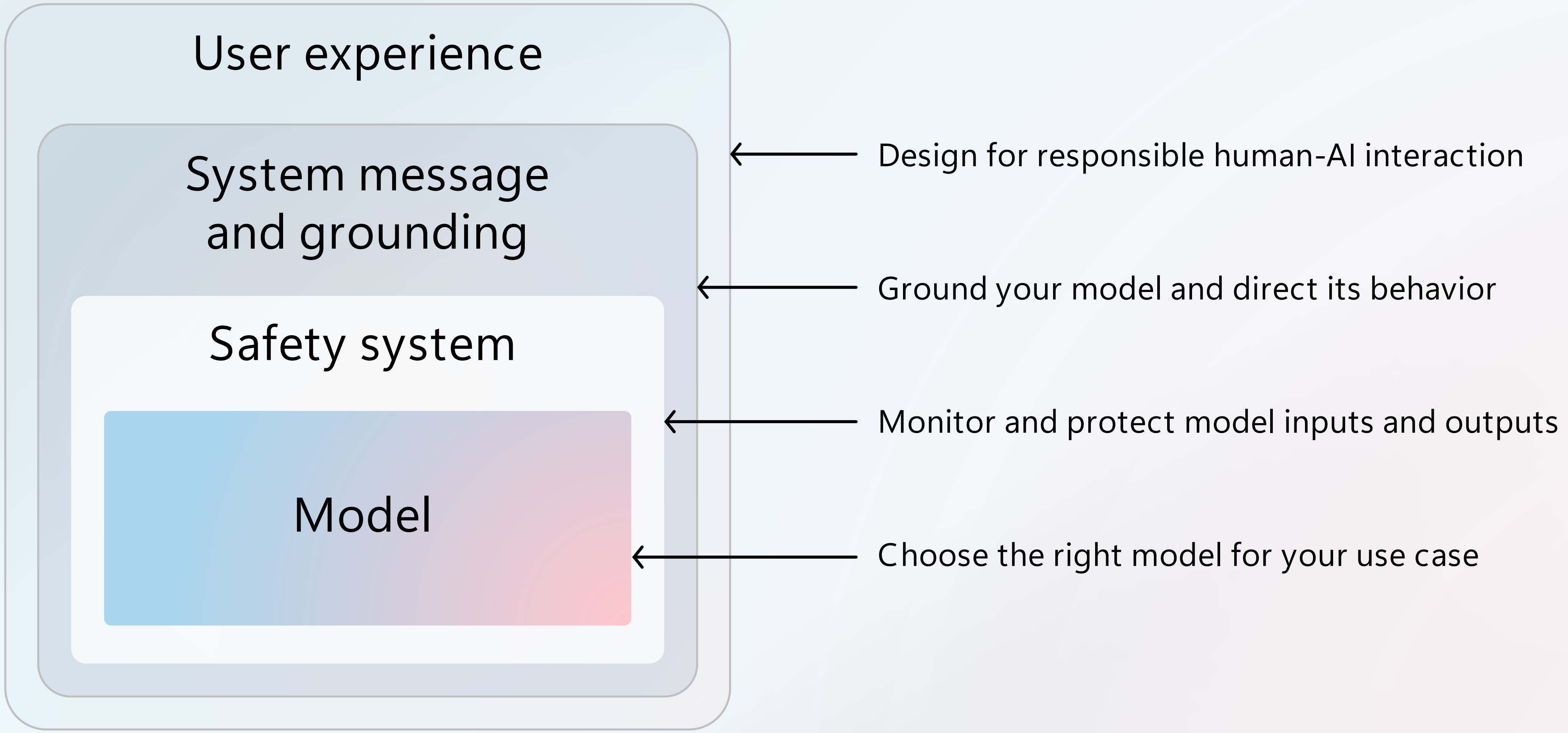
Transparency

Accountability

Microsoft first adopted our six AI principles in 2018, and they continue to drive our policy, research, and engineering investments.

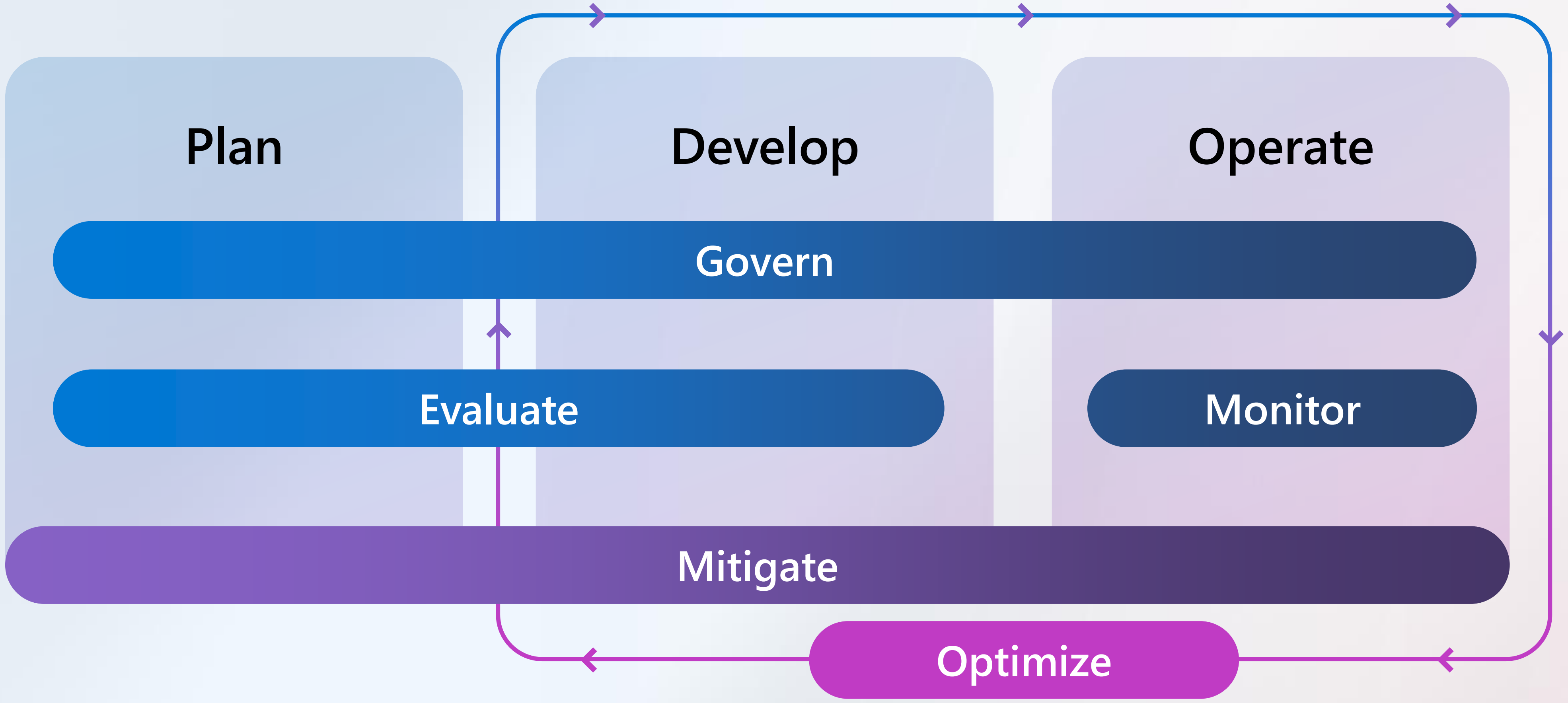
https://www.microsoft.com/en-us/ai/principles-and-approach#responsible-ai-standard

Risk mitigation layers


User experience
System message and grounding
Safety system
Model
Design for responsible human-AI interaction
Ground your model and direct its behavior
Monitor and protect model inputs and outputs
Choose the right model for your use case

Aligned with your end-to-end workflow

Visibility, monitoring and optimization across the entire AI development lifecycle.



Two possible options for building Agentic Systems

A

Need fully managed, secure, scalable deployment

B

Need flexibility, openness, and bleeding-edge features

Low Code

Azure AI Foundry Agent Service

"Connected Agents" Orchestration via Foundry UI and SDK

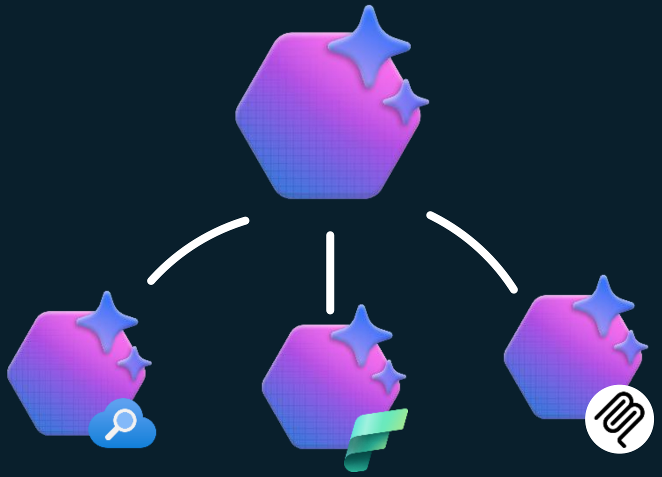
Pro Code

Semantic Kernel or other Orchestration Frameworks

Advanced Orchestration Patterns via Semantic Kernel SDK

Multi-agent orchestration in Foundry Agent Service

Connected agents



Give one agent the abilities of another.

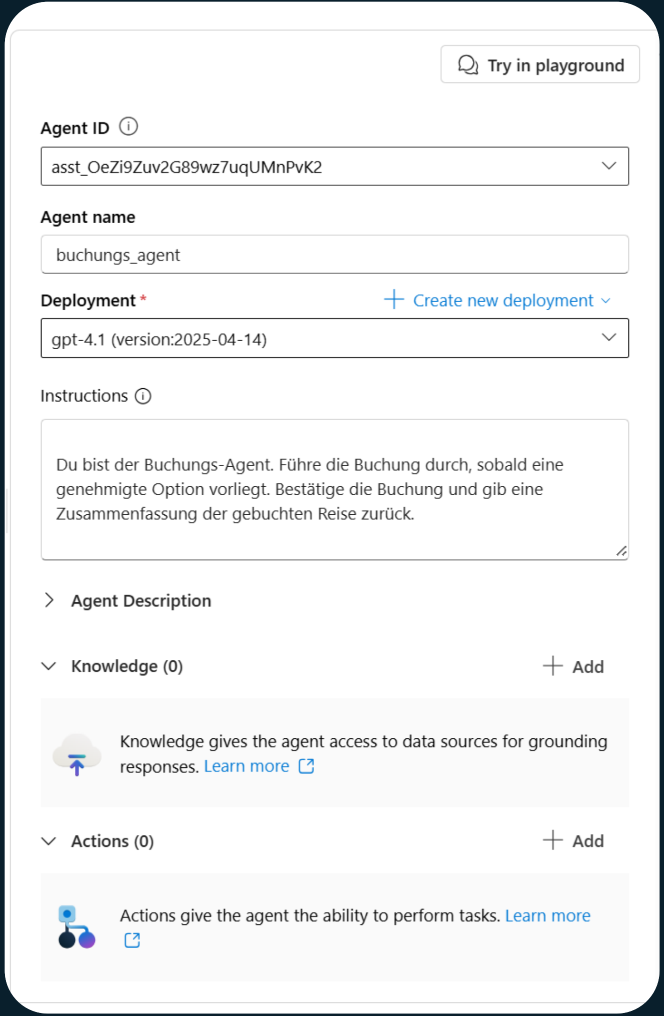
```
# Create the Booking Agent
buchungs_agent = agents_client.create_agent(
    model=model_deployment,
    name=buchungs_agent_name,
    instructions=buchungs_agent_instructions
)

# Define the Booking Agent as a tool
buchungs_agent_tool = ConnectedAgentTool(
    id=buchungs_agent.id,
    name=buchungs_agent_name,
    description="Bucht genehmigte Reiseoptionen."
)

# Create the Orchestrator Agent
orchestrator_agent = agents_client.create_agent(
    model=model_deployment,
    name=orchestration_agent_name,
    instructions=orchestration_instructions,
    tools=[
        buchungs_agent_tool.definitions[0]
    ]
)
```

…. but don’t worry, you can also do it via the UI

1 Create the Booking Agent



Try in playground

Agent ID: asst_OeZi9Zuv2G89wz7uqUMnPvK2

Agent name: buchungs_agent

Deployment *: gpt-4.1 (version:2025-04-14)

Create new deployment

Instructions:

Du bist der Buchungs-Agent. Führe die Buchung durch, sobald eine genehmigte Option vorliegt. Bestätige die Buchung und gib eine Zusammenfassung der gebuchten Reise zurück.

Agent Description

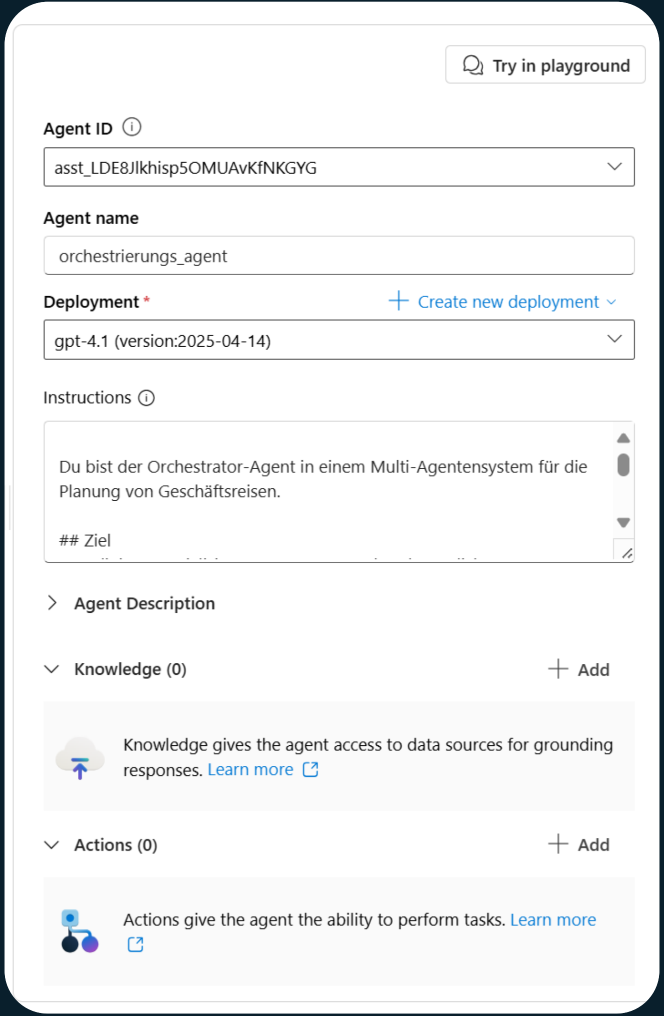
Knowledge (0) — Add

Knowledge gives the agent access to data sources for grounding responses. Learn more

Actions (0) — Add

Actions give the agent the ability to perform tasks. Learn more

2 Create the Orchestration Agent



Try in playground

Agent ID: asst_LDE8JlkhispSOMUAvKfNKGYG

Agent name: orchestrierungs_agent

Deployment *: gpt-4.1 (version:2025-04-14)

Create new deployment

Instructions:

Du bist der Orchestrator-Agent in einem Multi-Agentensystem für die Planung von Geschäftsreisen.

Ziel

Agent Description

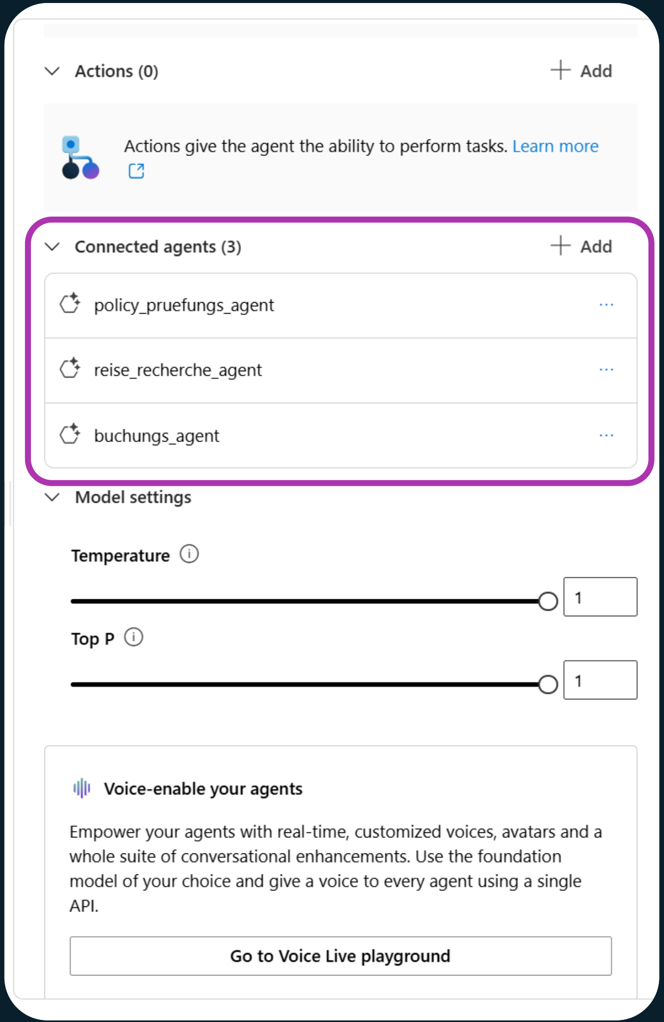
Knowledge (0) — Add

Knowledge gives the agent access to data sources for grounding responses. Learn more

Actions (0) — Add

Actions give the agent the ability to perform tasks. Learn more

3 Add Agents as Tools



Actions (0) — Add

Actions give the agent the ability to perform tasks. Learn more

Connected agents (3) — Add

- policy_pruefungs_agent
- reise_recherche_agent
- buchungs_agent

Model settings

Temperature: 1

Top P: 1

Voice-enable your agents

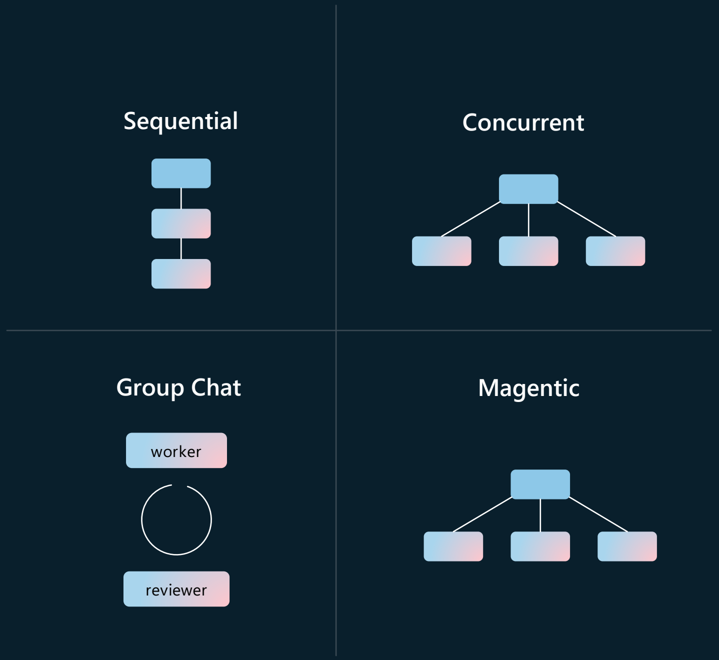
Empower your agents with real-time, customized voices, avatars and a whole suite of conversational enhancements. Use the foundation model of your choice and give a voice to every agent using a single API.

Go to Voice Live playground

Semantic Kernel Agent Framework – Core Concepts

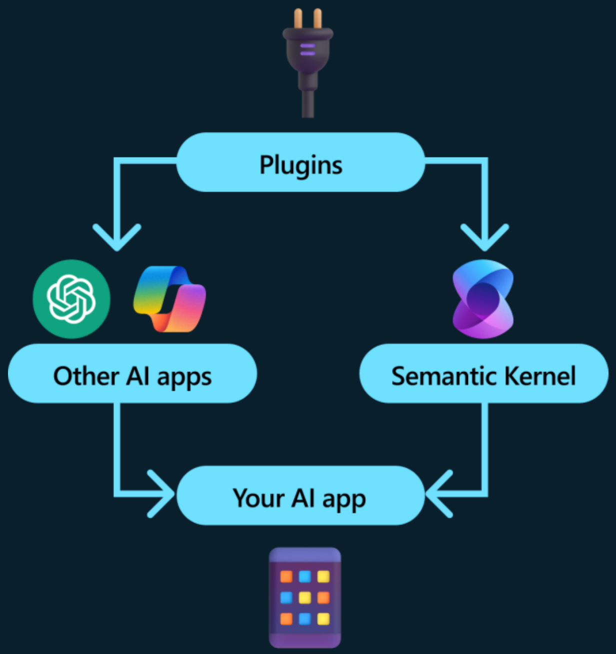
1

Built-in advanced orchestration patterns



2

Plugins for Function Calling, OpenAPI & MCP



3

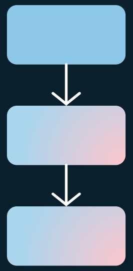
Native Support of several Agent Types

Azure AI Agents Service

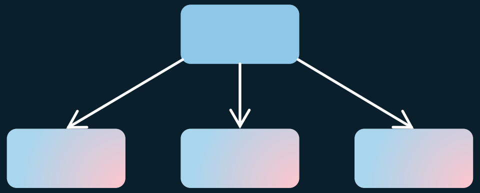
Bedrock AI

Multi-Agent Orchestration Patterns in Semantic Kernel

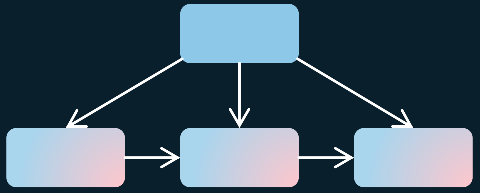
Sequential



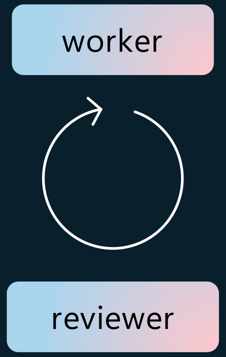
Concurrent



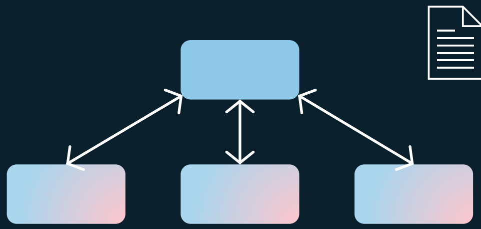
Handoff



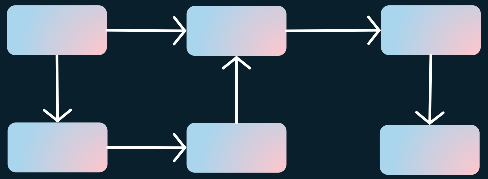
Group Chat



Magentic



Workflow Process



Full Documentation

Invocation of Orchestration Frameworks

All orchestration patterns share a unified interface for construction and invocation. No matter which orchestration you choose, you:

1. Define your agents and their capabilities
2. Create an orchestration by passing the agents (and, if needed, a manager).
3. Optionally provide callbacks or transforms for custom input/output handling.
4. Start a runtime and invoke the orchestration with a task.
5. Await the result in a consistent, asynchronous manner.

```
# Create the Agents
agent_a = ChatCompletionAgent(
        name="agent_a",
        instructions="You are agent a",
        service=AzureChatCompletion(),
    )

(…)

# Choose an orchestration pattern with your agents
orchestration = SequentialOrchestration(members=[agent_a,
agent_b])
# or ConcurrentOrchestration, GroupChatOrchestration,
HandoffOrchestration, MagenticOrchestration, ...

# Start the runtime
runtime = InProcessRuntime()
runtime.start()

# Invoke the orchestration
result = await orchestration.invoke(task="Your task here",
runtime=runtime)

# Get the result
final_output = await result.get()

await runtime.stop_when_idle()
```

Full Documentation

2 Kernel Plugins

Plugins are a key component of Semantic Kernel

Native Code Plugins

```python
agent = ChatCompletionAgent(
        service=AzureChatCompletion(),
        name="Host",
        instructions="Answer questions about the menu.",
        plugins=[MenuPlugin()],
    )


class MenuPlugin:
    @kernel_function
    def get_specials(self):
        return """
        Special Soup: Clam Chowder
        Special Salad: Cobb Salad
        Special Drink: Chai Tea
        """

    @kernel_function
    def get_item_price(self, menu_item):
        return "$9.99"
```

External Connectors

Logic Apps

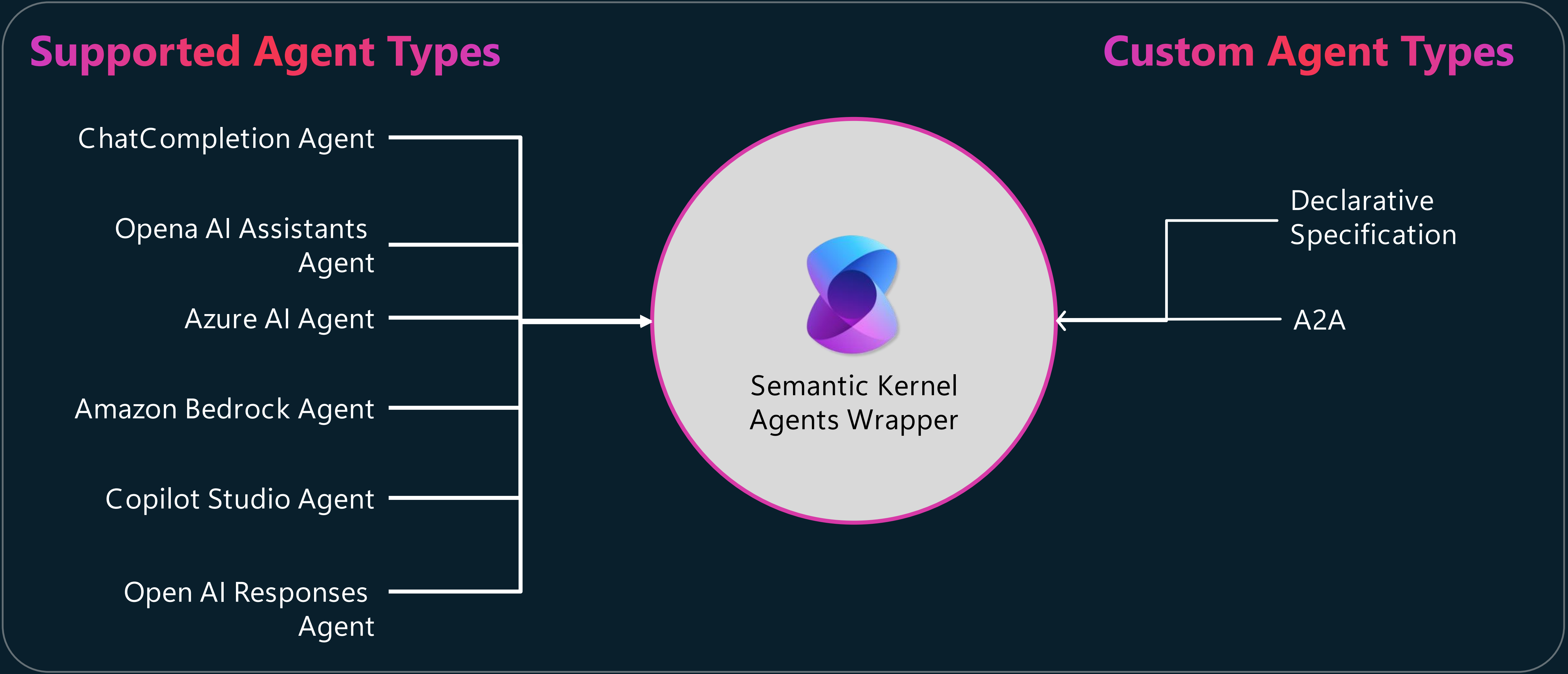
MCP Server

OpenAPI Spec.

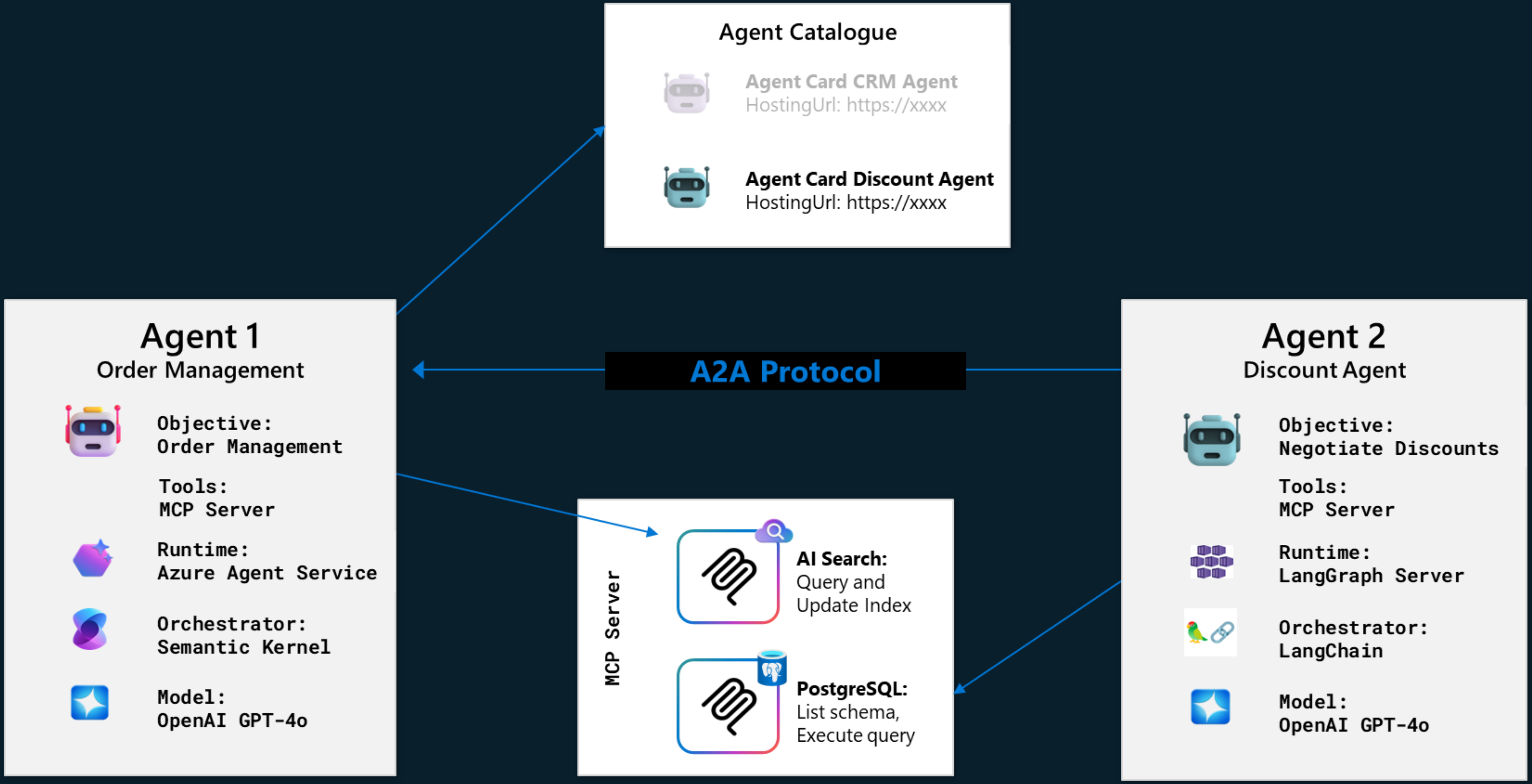
A2A Protocol

Integrating various Agent Types into Semantic Kernel

Multiple agents of different types can collaborate within a **single conversation**, be integrated in **orchestration frameworks** and utilize **kernel plugins**



Taking a look in the future - Open Agentic Web


Agent Catalogue
Agent Card CRM Agent
HostingUrl: https://xxxx
Agent Card Discount Agent
HostingUrl: https://xxxx
Agent 1
Order Management
Objective:
Order Management
Tools:
MCP Server
Runtime:
Azure Agent Service
Orchestrator:
Semantic Kernel
Model:
OpenAI GPT-4o
A2A Protocol
Agent 2
Discount Agent
Objective:
Negotiate Discounts
Tools:
MCP Server
Runtime:
LangGraph Server
Orchestrator:
LangChain
Model:
OpenAI GPT-4o
MCP Server
AI Search:
Query and
Update Index
PostgreSQL:
List schema,
Execute query