Copilot Studio Deep Dive: Building, Governing & Deploying AI Agents

Day 1: Start Right, Build right

Who am I?

Christine Adriane Svendsrud

Consultant & Low-code ish developer

Copilot Studio experience: ~2.5 years

Welcome & Introductions

- Who are you?
- Workshop purpose
- What you will be able to do after Day 1 & Day 2
- How we will work: live demo + labs + discussions
- Quick check, have everyone created an agent?

github.com/christineadriane/cevora-copilotstudio-training

Day 1: Agenda Overview

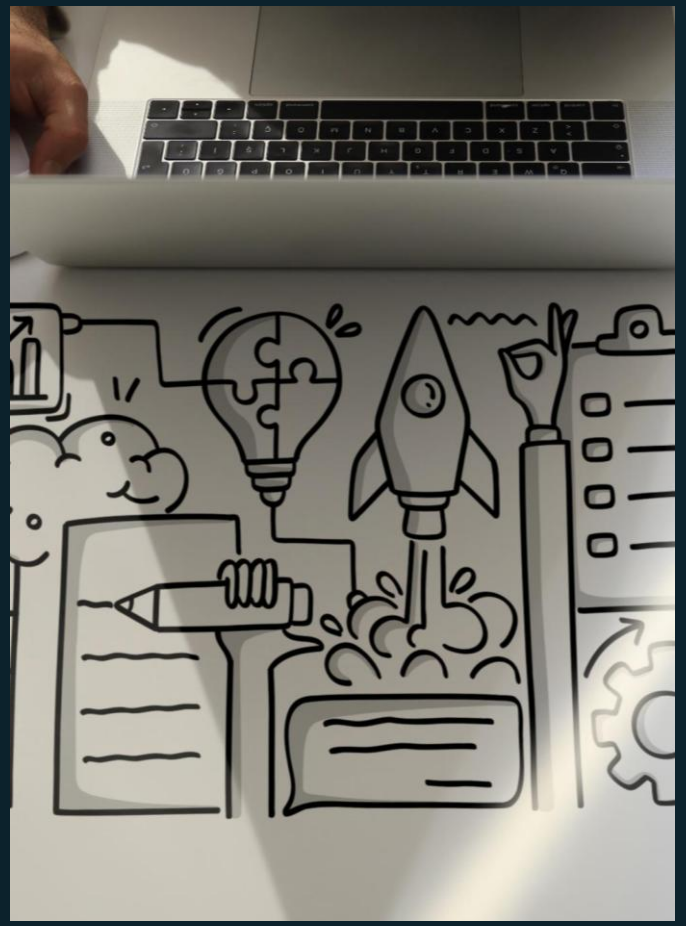
AGENTS, WHO, WHAT, WHEN, WHY?

EXPLORE COPILOT STUDIO

HANDS-ON LABS: BUILD YOUR FIRST(?) AGENT

WRAP-UP AND PREVIEW OF TOMORROW

Overview of Lab Activities: Day 1



AGENT SETUP AND CONFIGURATION

Learn how to create and configure agents, establishing the foundation for further development steps.

INCORPORATING/ EXTENDING KNOWLEDGE SOURCES

Add diverse knowledge sources, improving the agent's ability to understand and respond to user queries.

CUSTOM ACTIONS AND AI FLOWS

Design custom actions with AI prompts and flows, enabling advanced and tailored agent behaviors.

IMPORTANT!

Guidelines for AI Labs

Prioritize Understanding

Focus on grasping AI design principles instead of hurrying to finish all lab activities today.

Embrace Flexibility and Experimentation

Feel free to experiment beyond instructions. Set your own pace and explore new approaches to lab tasks.

Collaborate and Seek Support

Join discussions, ask for help when needed, and share insights to enhance learning.

Levels of AI transformation

Level 1

Humans with assistants

Every person is augmented by AI that understands their job

Level 2

Human-led agents

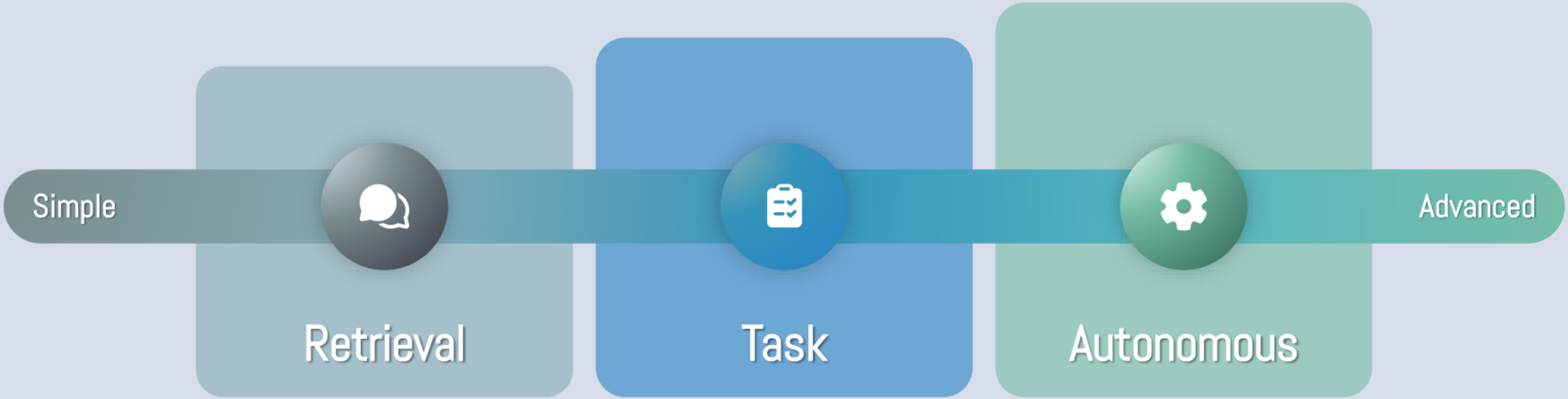
Every team includes AI agents, managed by humans, that complete tasks autonomously

Level 3

Teams of Humans and Agents

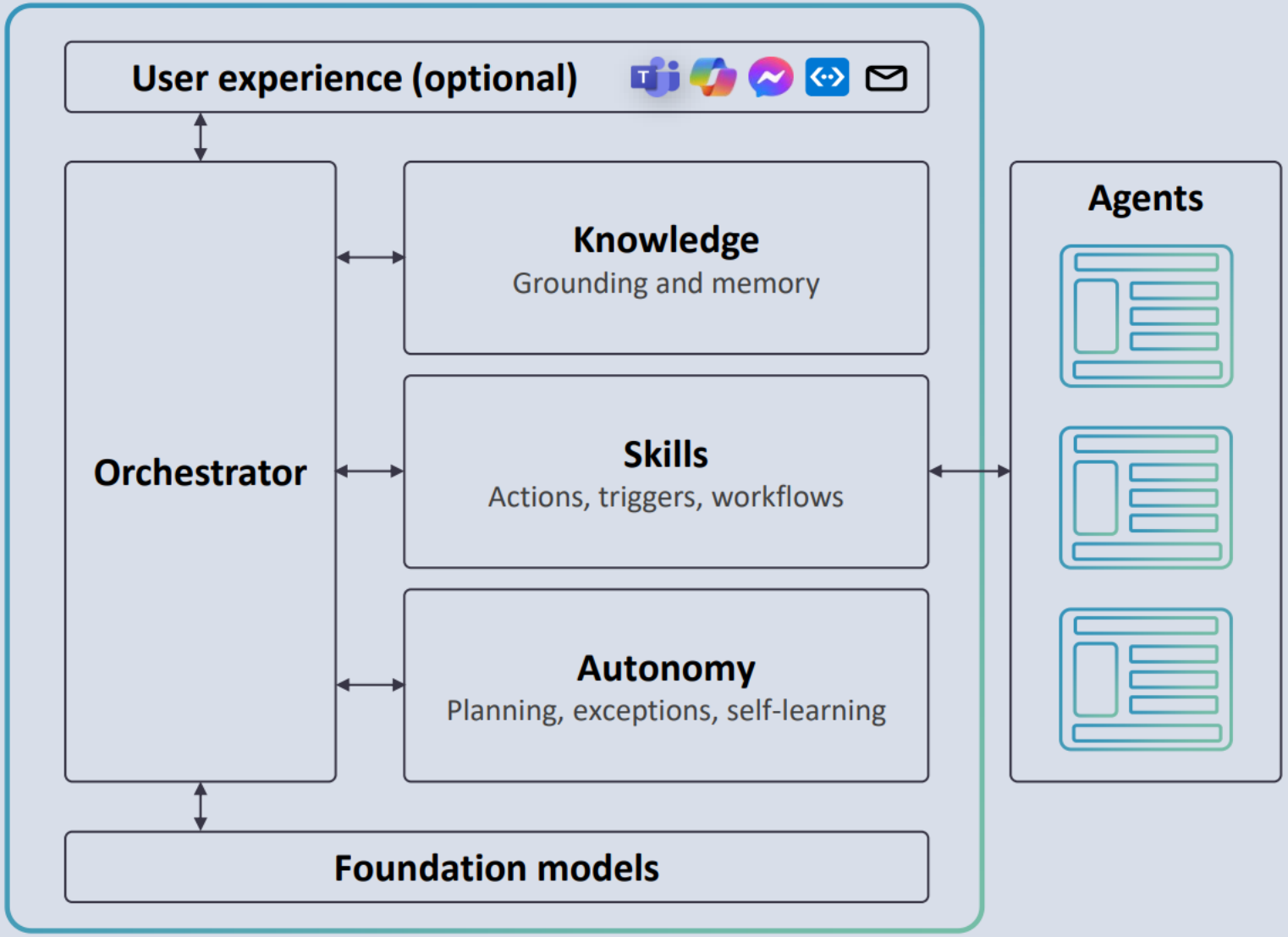
Humans set direction and AI agents run entire teams or departments, checking in as needed

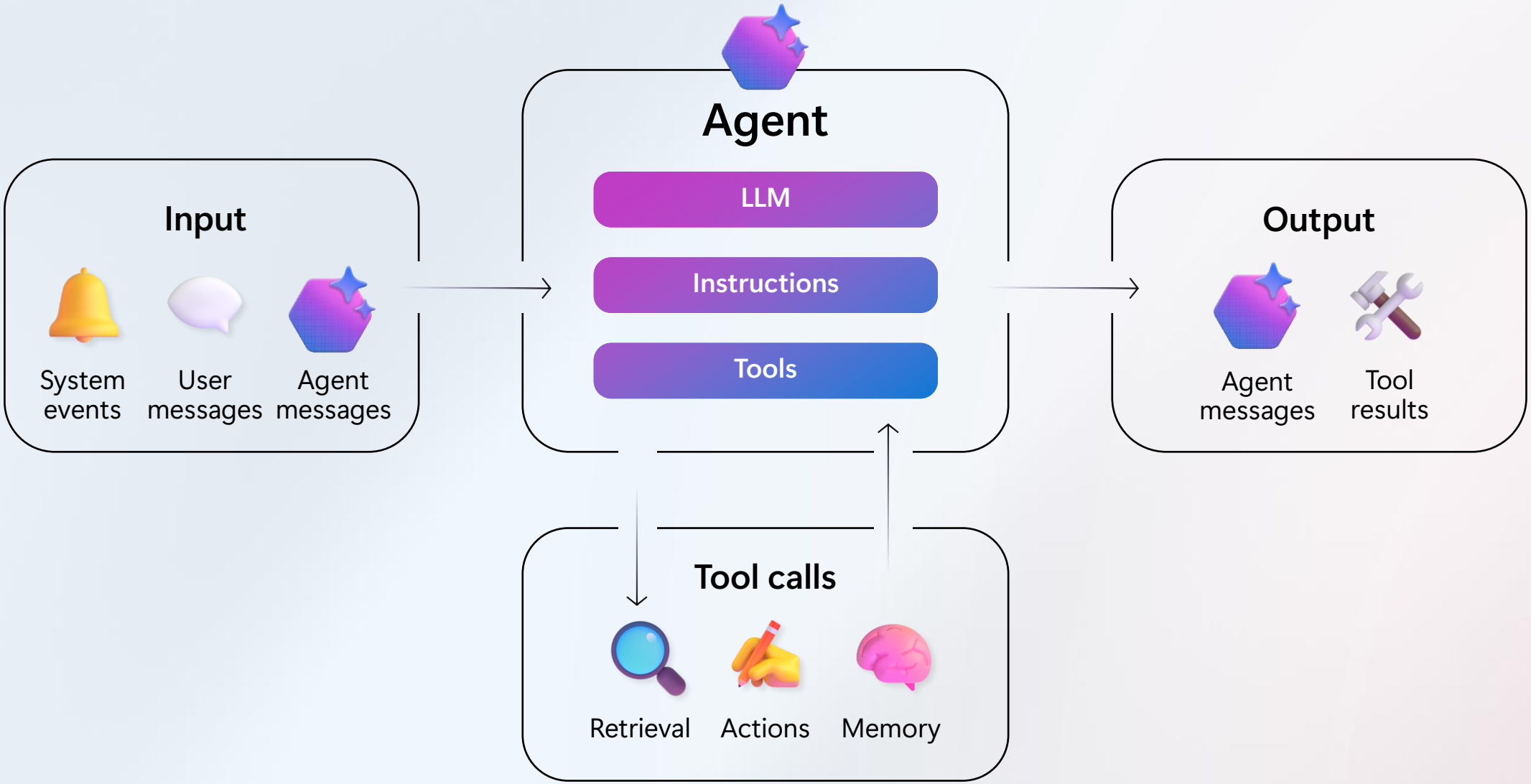
Agents are apps that use AI to reason, plan, connect to systems and complete tasks working alongside or on behalf of a person, team or organization



Agents vary complexity and capabilities depending on your need

Agents anatomy





WHY AGENTS?

Ask Yourself…

- Do you use the same prompt (or parts of it) over & over?

… what about the same instructions?

- Ever reference the same knowledge source?

… or the inverse?

✗ Tell copilot NOT to do something over & over?

Agents help you create repeatable scenario specific conversational experiences

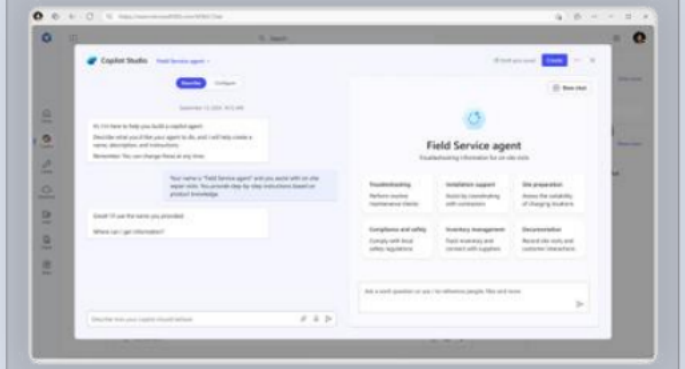
Crayon

A range of tools for agent creation

No code

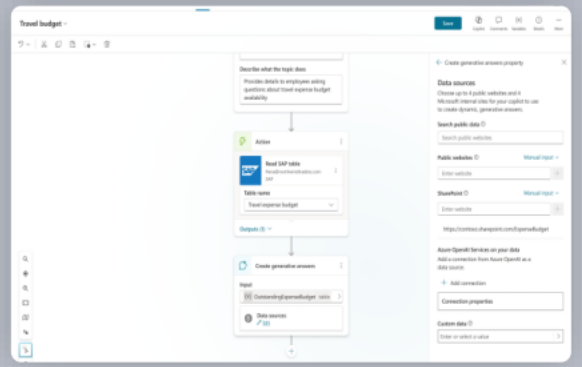
Pro code

For end users



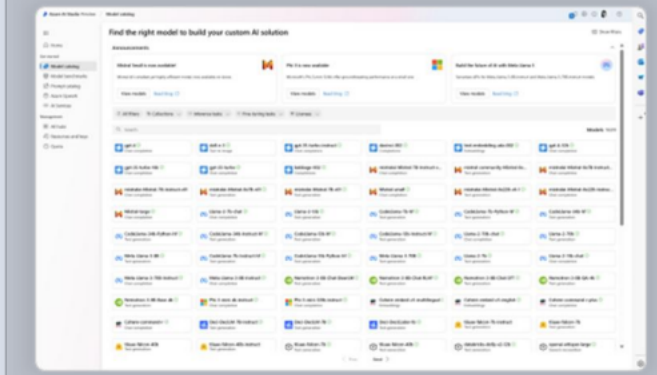
Agent builder

For makers



Copilot Studio

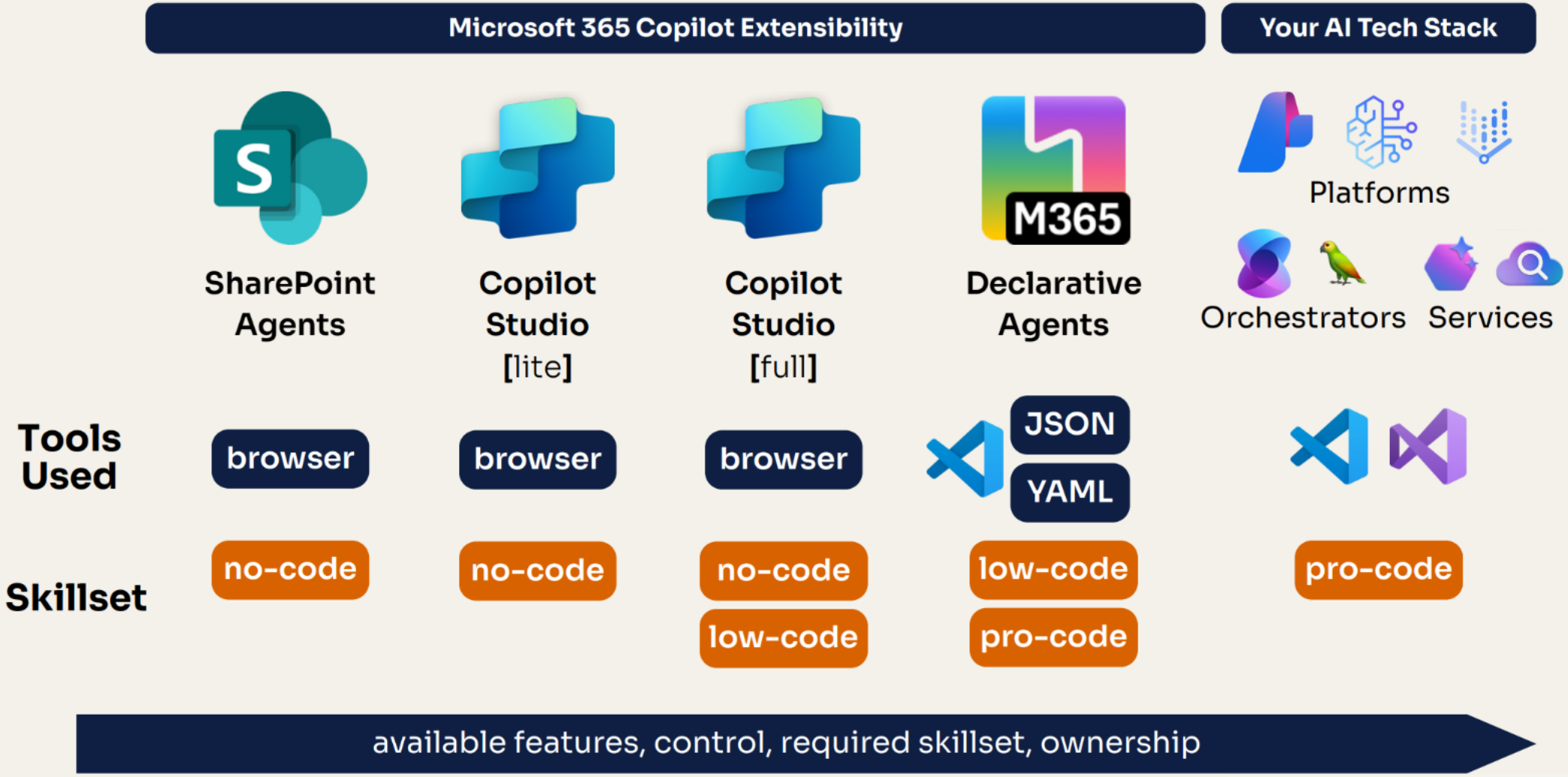
For developers

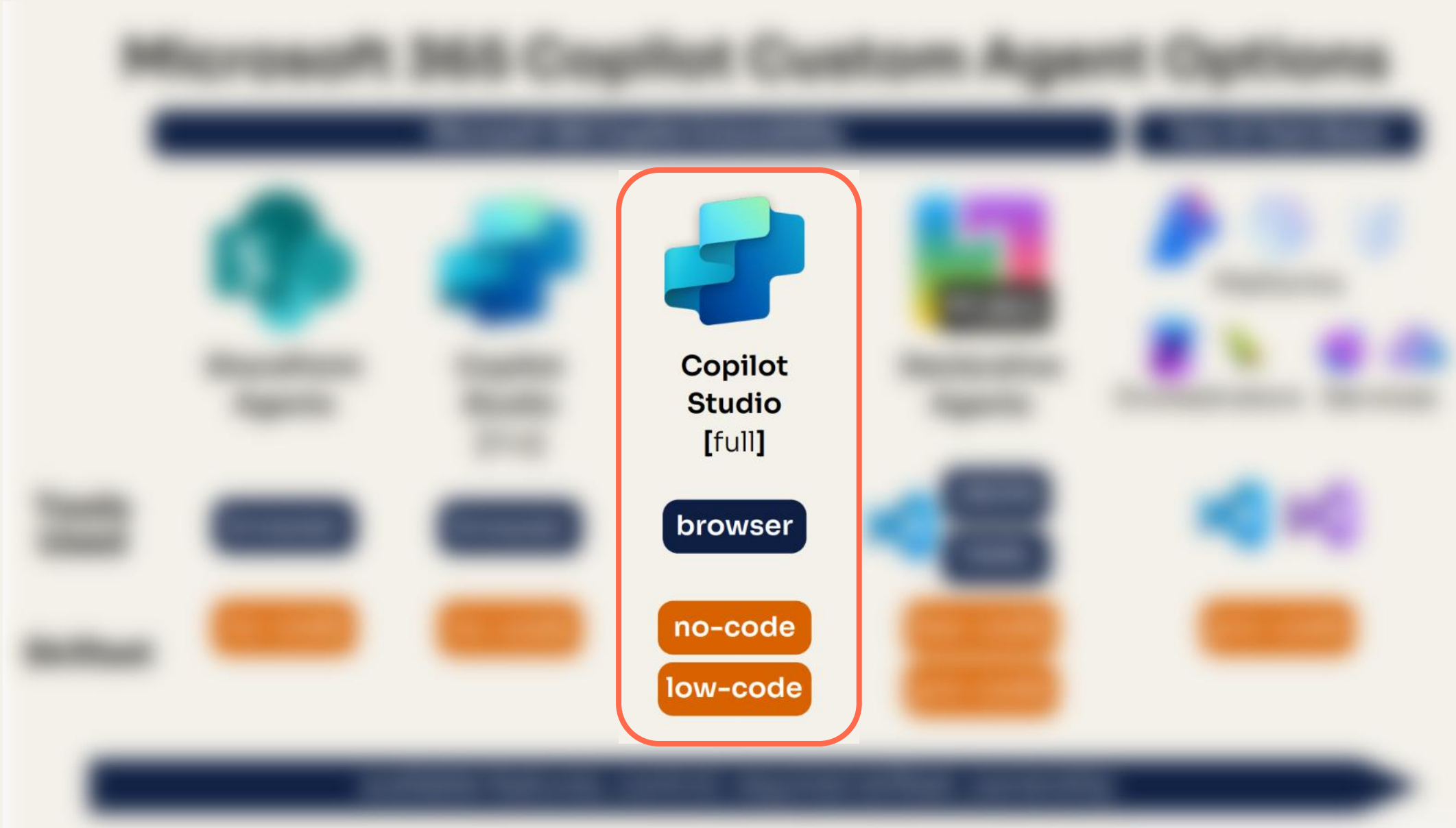


Copilot Studio, Azure AI Foundry, M365 Agents SDK

Data protection, agent sharing & usage limits, and reporting & cost management

Microsoft 365 Copilot Custom Agent Options





Copilot Studio explained

What it is:
Low-code tool to build custom **AI agents (copilots)** for internal & external use

Where it lives:
Part of the **Power Platform**, tightly integrated with **Microsoft 365 ecosystem**

Key benefits:

- Quick to build (no-code/low-code)
- Deep integration with Teams, Outlook, Dynamics, SharePoint
- Governed & secure (Dataverse, DLP, Entra ID)

NO SILVER BULLET.

INTELLIGENT AGENTS
DEMAND INTELLIGENT CHOICE.

A quick reminder on how LLMs work

An LLM can do one thing, and one thing ONLY:
Given a context, it predicts the next word.

Hence the term “completion”.

What users see:

User:

Hi there!

Assistant:

Hi there, how can I help you today?

User:

I'm looking for a new pair of shoes.

What the prompt actually looks like when sent to the LLM:

```
<|im_start|>user
Hi there!<|im_end|>
<|im_start|>assistant
Hi there, how can I help you today?<|im_end|>
<|im_start|>user
I'm looking for a new pair of shoes.<|im_end|>
<|im_start|>assistant
```

What the prompt looks like to the LLM model:

```
{
  'input_ids': [151644, 872, 198, 13048, 1052, 0,
151645, 198, 151644, 77091, 21018, 1052, 11,
1246, 646, 358, 1492, 498, 3351, 30, 151645, 198,
151644, 872, 358, 2776, 3330, 369, 264, 501, 6716,
315, 15294, 13, 151645, 198, 220, 151644, 872,
358, 2776, 3330, 369, 264, 501, 6716, 315, 15294,
13, 151645, 220, 151644, 77091
],
  'attention_mask': [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
}
```


Crayon

Short break: 5 min

https://github.com/christineadriane/cevora-copilotstudio-training/

Lab 1:

In this lab, you will create a new HR agent using Advanced Create, define clear operating instructions, and connect two public websites as knowledge sources. You will also map the agent to your workshop solution and validate its behavior in the Test pane. By the end of the lab, you will have a working HR assistant with proper grounding, tone, scope, behavior rules, and structured instructions.

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-01/README.md

Governance & Security Essentials

«Start right before building»

ENVIRONMENT MANAGEMENT PRACTICES

Environments are separated into Dev, Test, and Prod to ensure safe application development, testing, and deployment.

DATA LOSS PREVENTION CONTROLS

Data Loss Prevention policies help IT manage connectors and enforce restrictions, protecting sensitive business data.

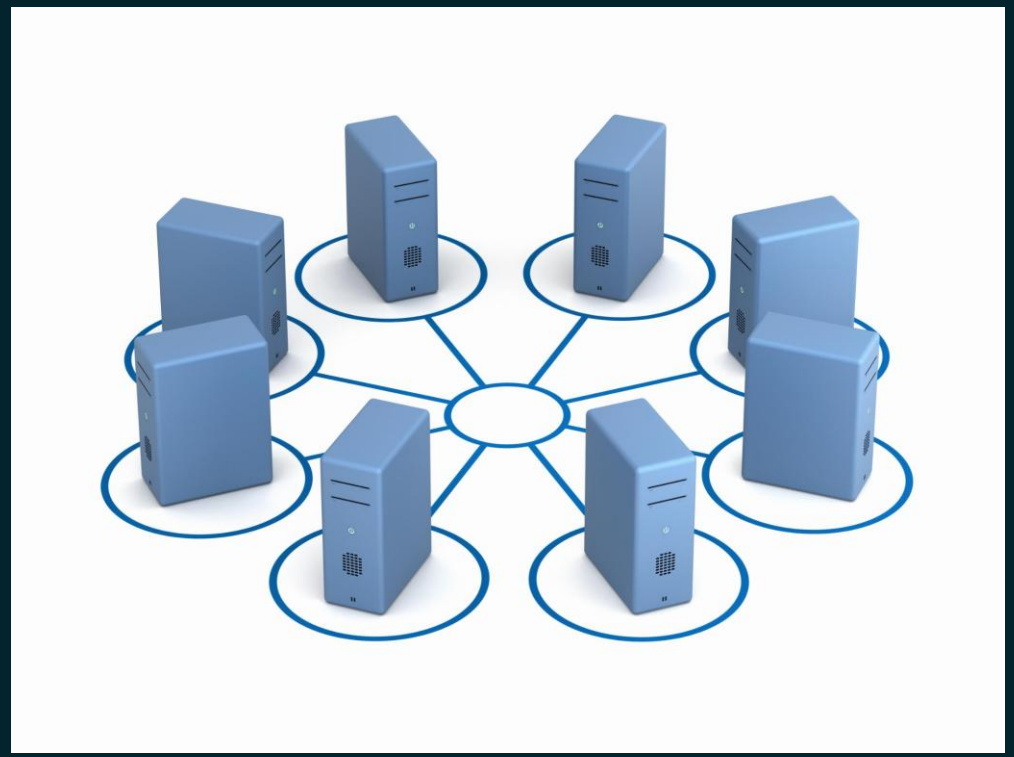
ROLE-BASED ACCESS MANAGEMENT

Access permissions are managed by roles, allowing admins to decide who builds, publishes, or selects communication channels.

THOUGHTFUL PLANNING REDUCES REWORK

Starting with a clear strategy on use cases and data compliance helps avoid costly future rebuilds.

Build With Environment Strategy



Development Environment Flexibility

The Development environment allows teams to experiment and test new agents quickly without impacting other areas.

Testing and Validation

Test or UAT environments are used to validate business rules, review IT flows, and restrict access to sensitive data.

Stable Production Deployment

Production hosts only published agents and analytics, ensuring stability and preventing accidental changes or data exposure.

Crayon

Lab 2:

In this lab you will add additional internal sources. Enrich the agent with PDF documents, prioritize knowledge from PDFs, and refine instructions for clear sourcing and output quality.

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-02/README.md

Short break: 5 min

https://github.com/christineadriane/cevora-copilotstudio-training/

Choosing Quality Knowledge Sources

PRIORITIZE STRUCTURED INFORMATION

Quality sources are organized, accurate, and current, featuring clear headings and concise content for easy understanding.

AVOID POOR QUALITY MATERIALS

Messy PDFs, unstructured manuals, and outdated or conflicting documents create confusion and hinder knowledge grounding.

START WITH RELIABLE SOURCES

Begin with one or two trustworthy sources and expand as needed for effective knowledge integration.

Lab 3:

In this Lab we will add a Prompt-type tool that summarizes an HR policy into a clear, employee-friendly explanation and wire it into your agent’s conversation flow.

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-03/README.md

Lab 4:

In this Lab we will add a flow-based action that expects basic leave details and registers the request in a SharePoint list, then returns a confirmation (ID + summary) to the agent.

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-04/README.md

Day 1 Recap & Tomorrow's Preview

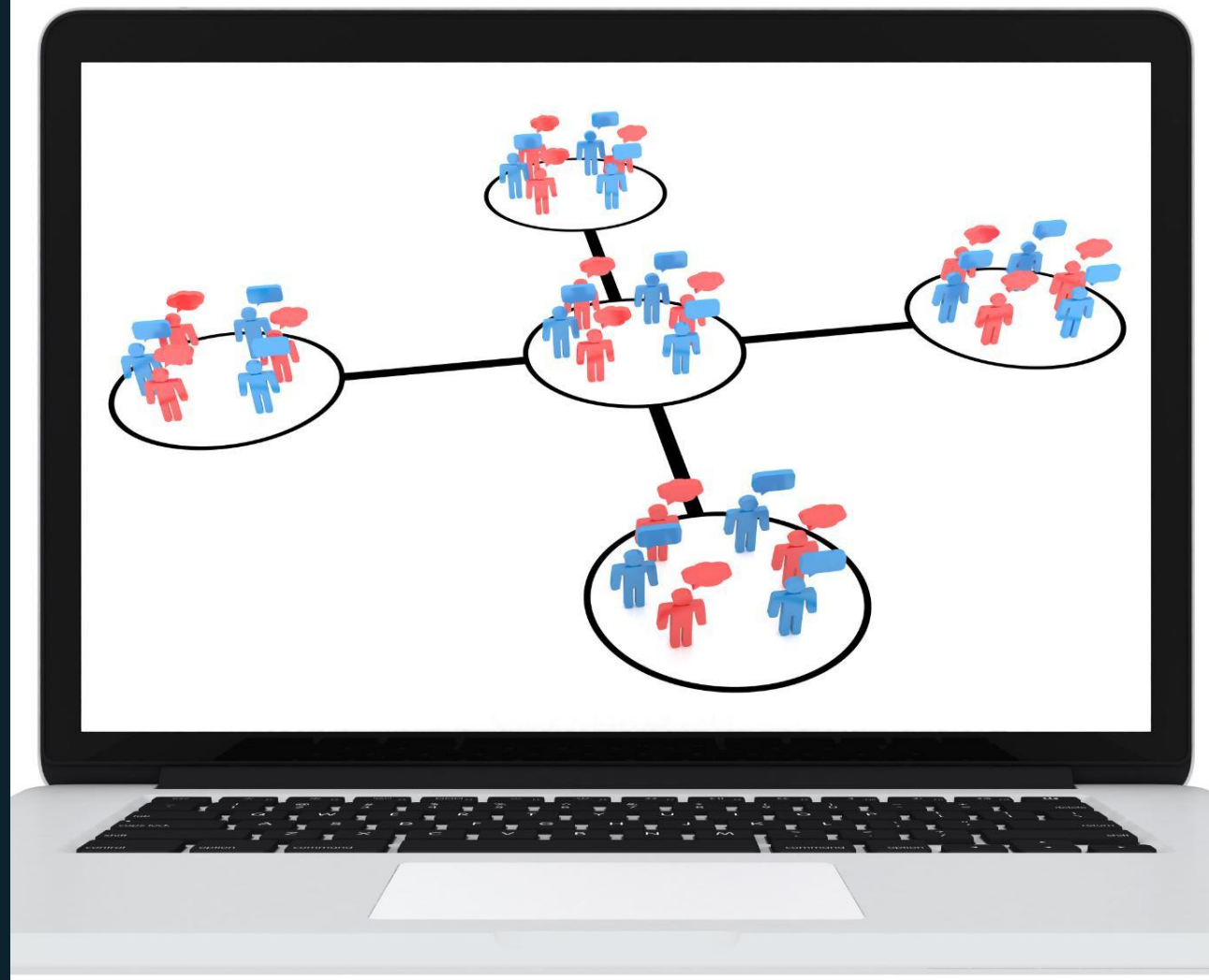
Review of Core Concepts

Day 1 focused on essential concepts and foundational practices that form the basis for deeper understanding.

Tomorrow's Learning Focus

We will cover actions, triggers, deploy and publishing methods, billing and licensing and governance strategies for post-build management.

Anything else?



Copilot Studio Deep Dive: Building, Governing & Deploying AI Agents

Day 2: Build better, Publish Safely

Day 2: Agenda Overview

TOPICS, TRIGGERS AND TOOLS IN COPILOT STUDIO

HANDS-ON LABS: CONTINUE BUILDING AGENTS

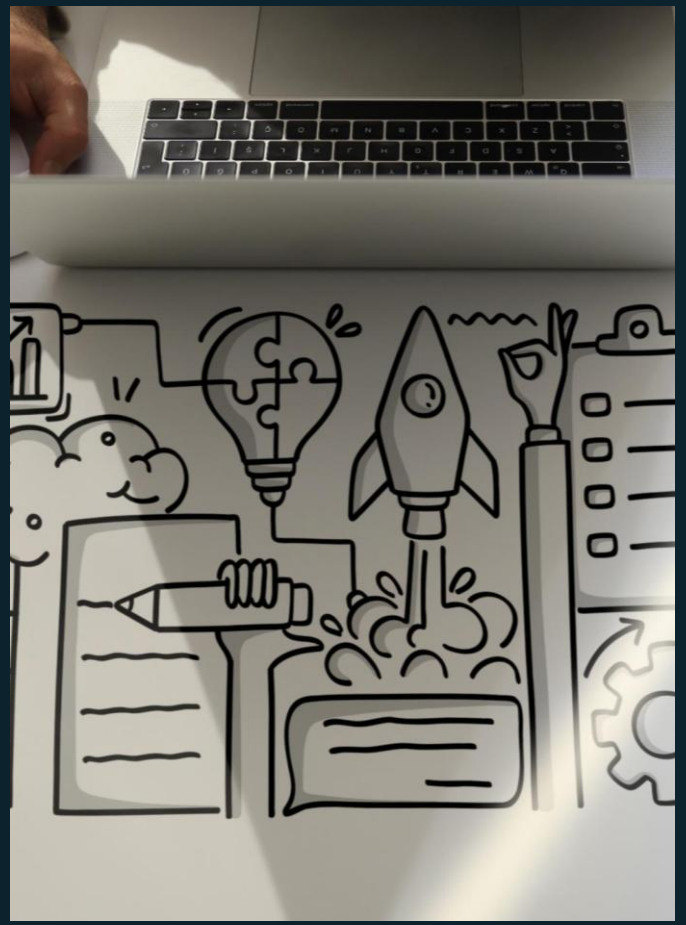
WORK ON THE HR-AGENT & BECONNECTED

SECURITY

DEPLOYMENT AND BILLING

WRAP-UP AND THE PATH FORWARD

Overview of Lab Activities: Day 2



TRIGGERS AND TOPICS

Create custom triggers, and set up topics (BONUS: use adaptive cards),

EVALUATION

Set up ways to constantly improve and evaluate the performance of the agent

FIRST: DEMO

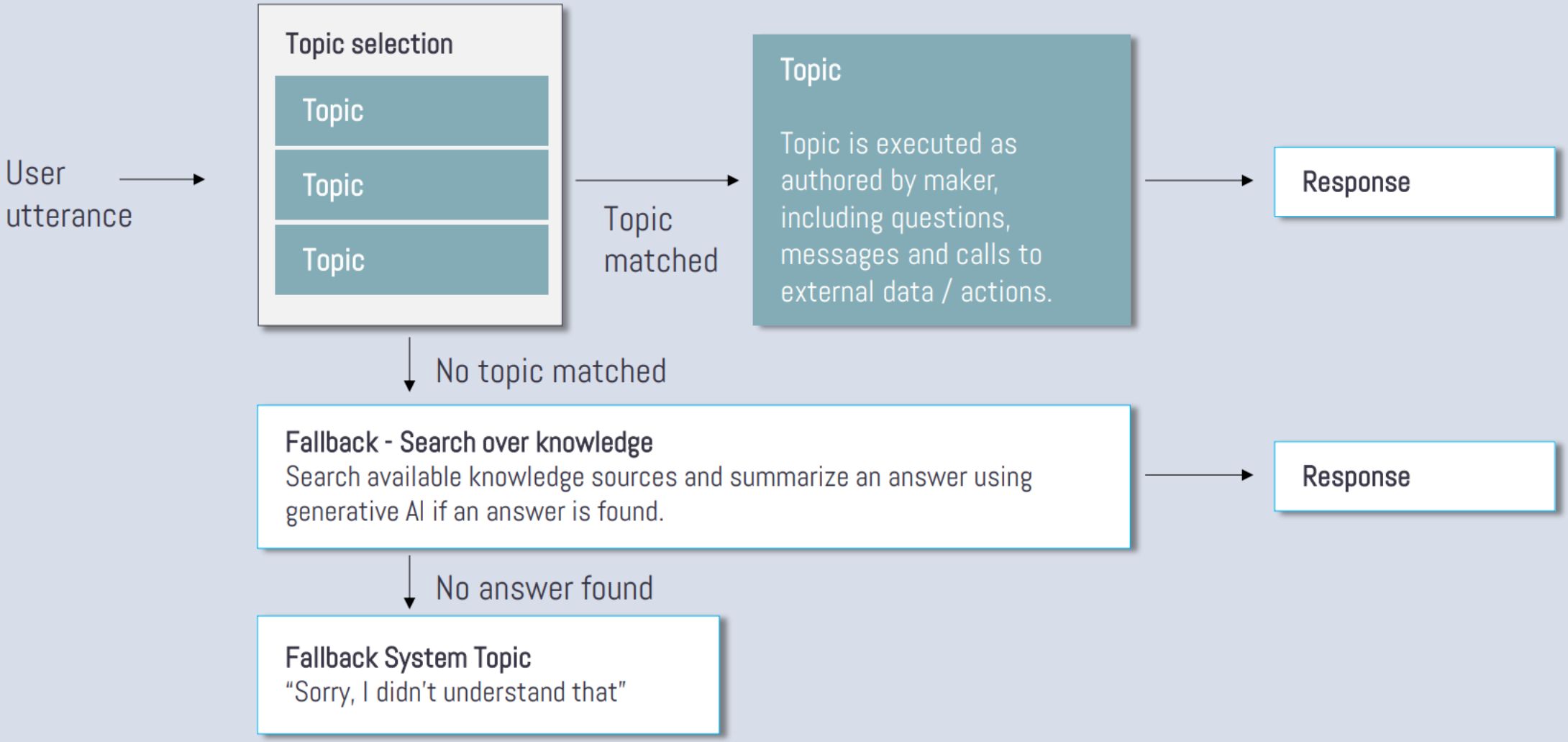
Topics in Copilot Studio

The structured building blocks that define how your copilot handles specific user intents.

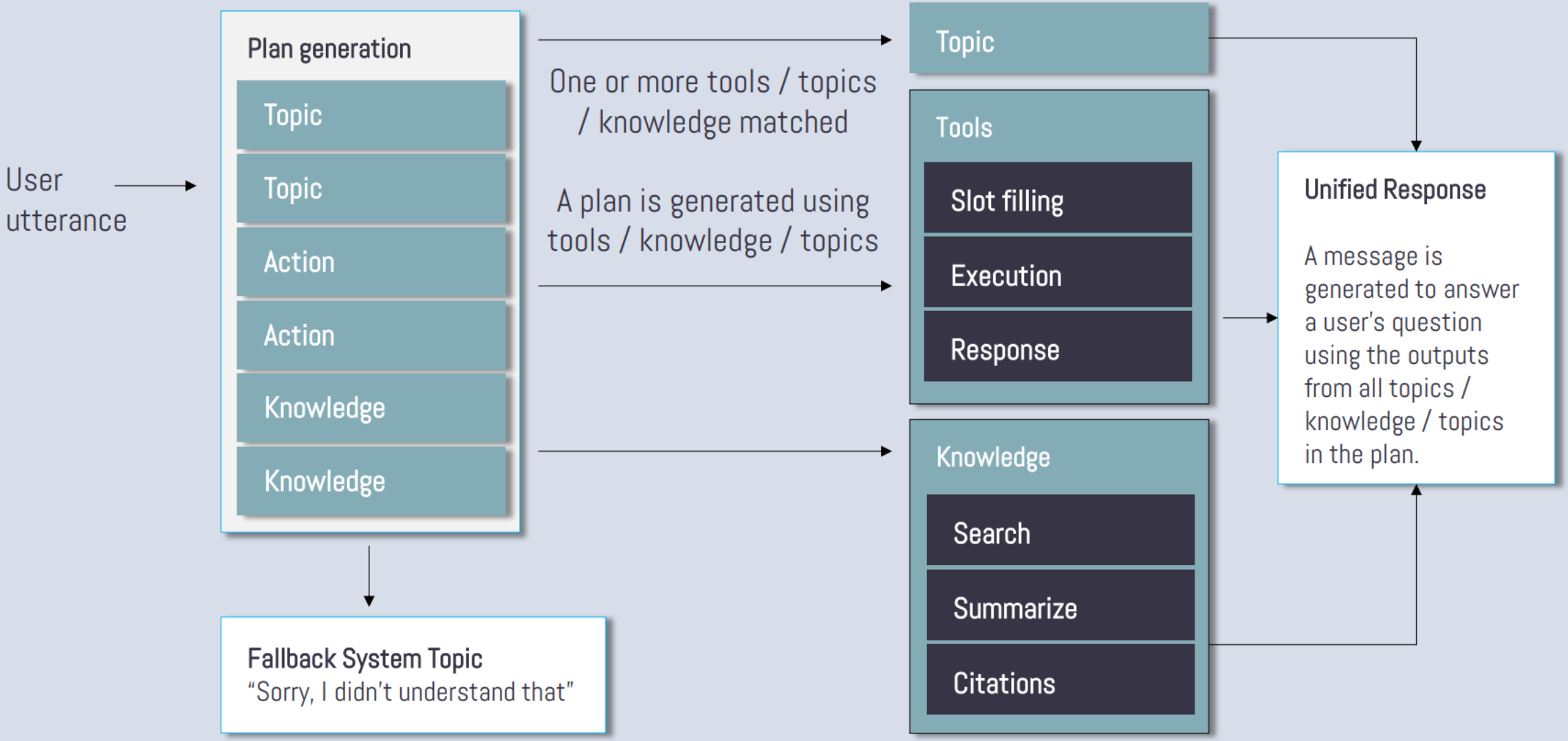
- Understand user intent based on trigger phrases or generative interpretation
- Run step-by-step guided conversations
- Ask questions, collect information, branch logic
- Call actions (APIs, flows, etc)
- Redirect to other topics for reusable logic
- Combine AI-generated responses with deterministic flows

📌 Topics = predictable, reusable, safe dialog logic for your copilot.

Classic orchestration

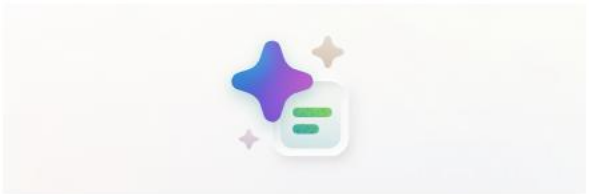


Generative orchestration



Add tool

Let your agent do more. Learn more



Prompt
Analyze and transform text, documents, images, and data, with natural language and AI reasoning.

Agent flow
These predictable automations run the same way each time, giving you more control when you need it.

Custom connector
External services and data sources.

REST API
Flexible and scalable way for your agent to connect with and use data.

Model Context Protocol
Open standard for connecting your agent to data, designed with AI in mind.

TOPICS INSIDE COPILOT STUDIO

Lab 5:

In this lab we will leverage Topics to intercept the AI-generated response before it is sent, extract the employee-facing action items using a parsing prompt, and return a single message back to the agent, gaining more control over conversation flow

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-05/README.md

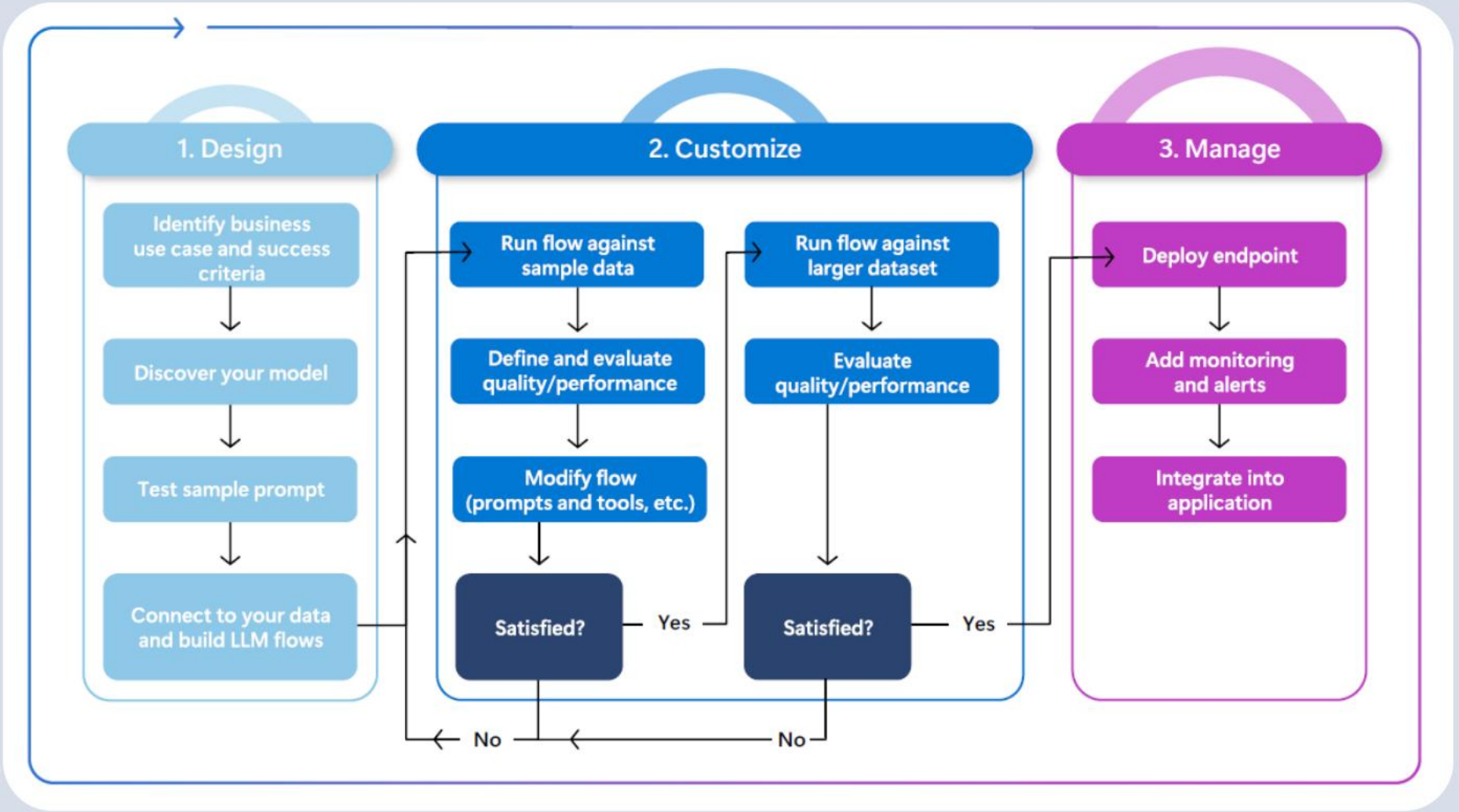
TRIGGERS INSIDE COPILOT STUDIO

Lab 6:

In this lab we will add an email-based trigger that captures incoming HR communications, extracts the key employee action points from the message body, and routes the structured output back to the agent for automated processing.

https://github.com/christineadriane/cevora-copilotstudio-training/blob/main/labs/lab-06/README.md

Evaluation loop



Billing in Copilot Studio

Objective: Establish a strategy for licensing and cost management enabelig boad access to AI-capabilities while keeping expenses optimized

Scope: Focus on how to deliver Agent functionality for employees with most cost efficient way by leveraging: *M365 Copilot user licenses*, ***Pay-As-You-Og (PAYG)*** billing for metered agent usage and *Prepaid message capacity packs*.

Controlling cost across the agent lifecycle

1. Estimate before you deploy
2. Bill with the right model (PAYG, Prepaid, User License)
3. Track consumption continuously
4. Govern with policies, alerts, and guardrails

4 key stages

Choosing the Right Billing Model

Pay-as-you-Go (PAYG)
(best for trials/departments, price from $0.01 per message)

Prepaid Packs
(predictable monthly, $200 monthly for 25 000 messages)

User License
(enterprise-wide, power users, $30 monthly per user for Enterprise plans and $20 for Business plans)

Explore
PAYG & Prepaid for pilot use

Scale
Track patterns, adjust billing mix

Optimize
User License for predictable per-user cost

Utilization rates depend on type of agent and prompt

	Orchestration Mode	M365 Copilot Users	Copilot Chat Users	Use of Other Agents Built w/ Copilot Studio
Web-grounded answers Dynamically-generated responses based on the web as a knowledge source.	Classic & Generative	0	0	2 Copilot Credits
Classic answers Predefined responses manually authored by makers through topics (includes messages, connectors, flows etc.) that are static unless manually updated in Classic Orchestration mode. Used when a precise or controlled response is desired output. Each action (not each topic) counts as an answer. Not available in agent builder.	Classic only	0	1 Copilot Credits	1 Copilot Credits
Generative answers [1,2] Dynamically-generated responses based on knowledge sources and context that provide flexible and natural interactions.	Classic & Generative	0	2 Copilot Credits	2 Copilot Credits
Tenant graph grounding for messages [1,2] Grounding to enhance AI agents with up-to-date, context-aware knowledge from Microsoft 365 and external data, offering built-in security and inheriting data access governance policies.	Classic & Generative	0	10 Copilot Credits*	10 Copilot Credits*
Agent actions [1,2] AI-led orchestration for triggers, topics, agent flows, text & generative AI tools, Power Platform premium connectors and custom connectors to automate complex business processes. Not available in agent builder.	Generative only	0[4]	5 Copilot Credits*	5 Copilot Credits*
Text & generative AI tools Specialized tools that extend agents capabilities by teaching them to perform specific tasks, leveraging a combination of AI prompt engineering, model configuration, code execution, and knowledge retrieval	-	-	-	-
Basic (Message rate per 10 responses[3])	Classic & Generative	1 Copilot Credits*	1 Copilot Credits*	1 Copilot Credits*
Standard (Message rate per 10 responses[3])	Classic & Generative	15 Copilot Credits*	15 Copilot Credits*	15 Copilot Credits*
Premium (Message rate per 10 responses[3]) For deep reasoning prompts	Classic & Generative	100 Copilot Credits*	100 Copilot Credits*	100 Copilot Credits*
Agent flow actions (Message rate per 100 agent flow actions) Agent flow actions are used to create agent flows. Agent flows are rules-based automations in Copilot Studio that follow a predefined sequence of agent flow actions to perform repetitive tasks.	Classic & Generative	13 Copilot Credits*	13 Copilot Credits*	13 Copilot Credits*

Notes

1. Each interaction with an agent could utilize multiple utilization rates simultaneously i.e., an agent grounded in Tenant graph could use 12 Copilot Credits (10 for the graph grounding and 2 for Generative Answer) to respond to a single complex prompt from the user. Most agents built natively in SharePoint or Copilot Chat will have tenant graph grounding enabled by default.
2. Generative answers, tenant graph grounding for messages, web-grounded answers and agent actions apply to both declarative agents and custom engine agents.
3. 1 response = 1,000 tokens for LLM models, 1 image for image processing, 1,000 characters for text processing and 1 row when processing rows for prediction. Billing will be prorated to exact number of responses.
4. Agent actions are included at no additional cost for interactive use only. Autonomous use will incur a 5 Copilot Credits charge

Basic credit consumption examples

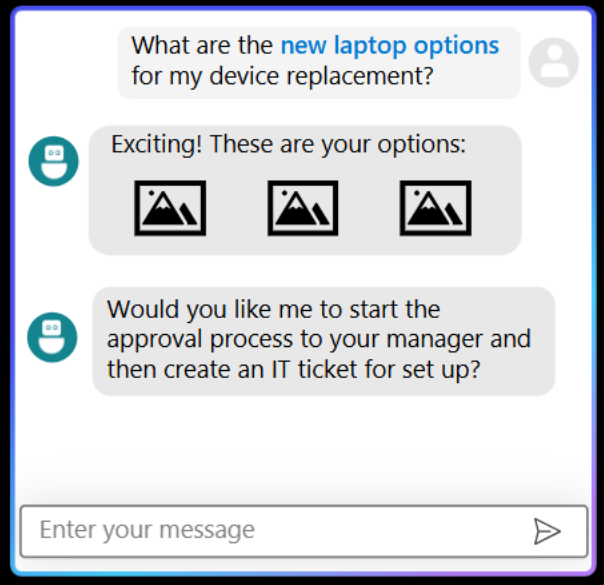
Rule: 1 message sent by the user = 1 credit consumed (includes unspoken messages like "conversation start" or "inactivity" trigger).

Exception: using different AI features will trigger additional credit consumption to offset the cost of LLM.

Scenario 1

1 question triggering a regular topic

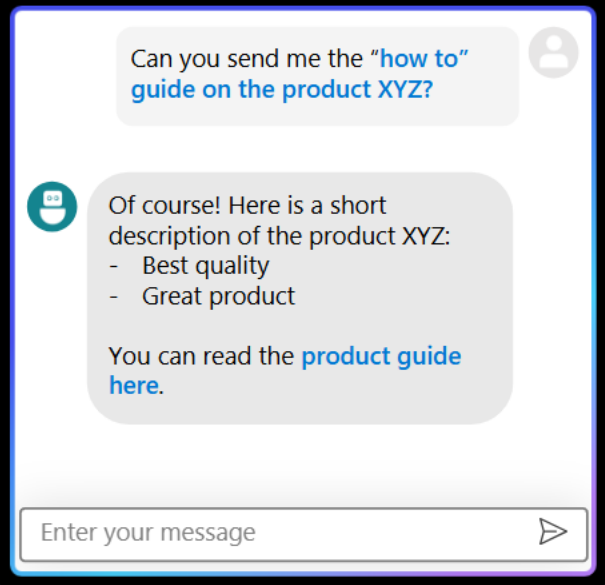
Consume 1 credit (despite the answer being 2 messages)



Scenario 2

1 question triggering Generative Answer

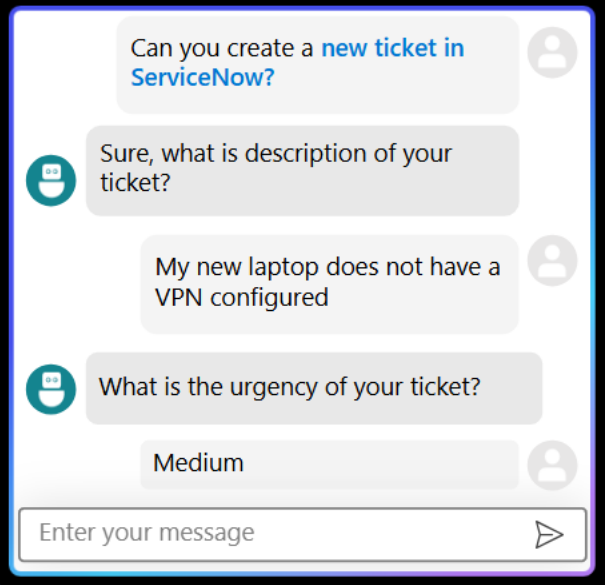
Consume 2 credits (despite the answer being 1 message)



Scenario 3

1 question triggering a tool via Generative Orchestration

Consume 5 credits (whatever the back and forth to collect inputs)



Estimate Agent Message Consumption

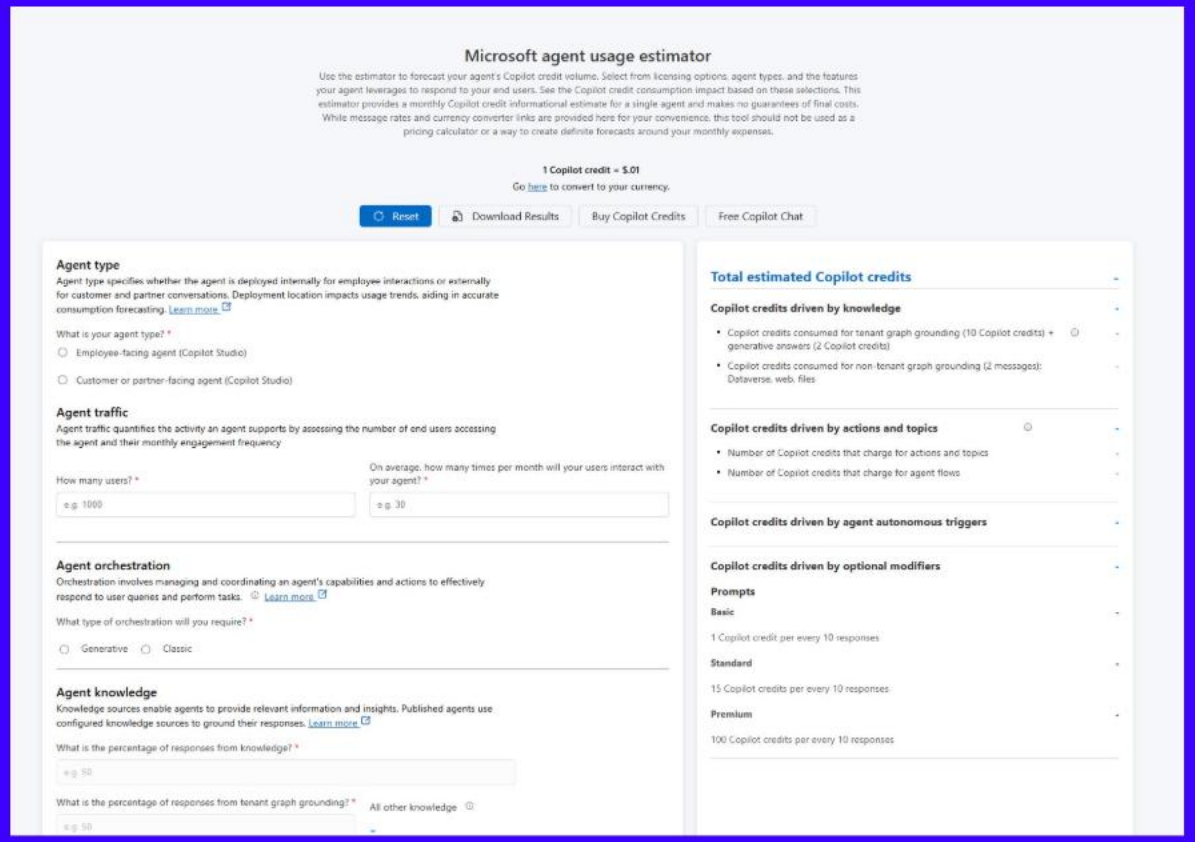
General Available!

Forecasts message consumption using data driven trends and assumptions

Customizes estimates based on licensing options and feature usage

Accessible from within Copilot Studio, Power Platform admin center, and public website

aka.ms/copilotstudioestimator



Capacity management

Add pay-as-you-go to specific environments

- **Flexible Cost Management** - assigning PAYG billing to specific environments allows payment only for the actual agent usage in those environments, ideal for pilots, seasonal projects, or departments with unpredictable consumption.
- **Seamless Integration with Prepaid Capacity**: You can combine PAYG with prepaid Copilot credits packs—using PAYG as a backup for overages—so environments or agents never run out of capacity, and costs remain under control.

Assign capacity to specific environments & agents

- **Targeted Resource Allocation** - Assign prepaid Copilot credits capacity to particular environments (such as a team, project, or department) and to individual agents. TIP: Keep some unassigned capacity to be pooled at the tenant level or assign dynamically to handle growth of specific agents.

Cost Alerts & Enforcement

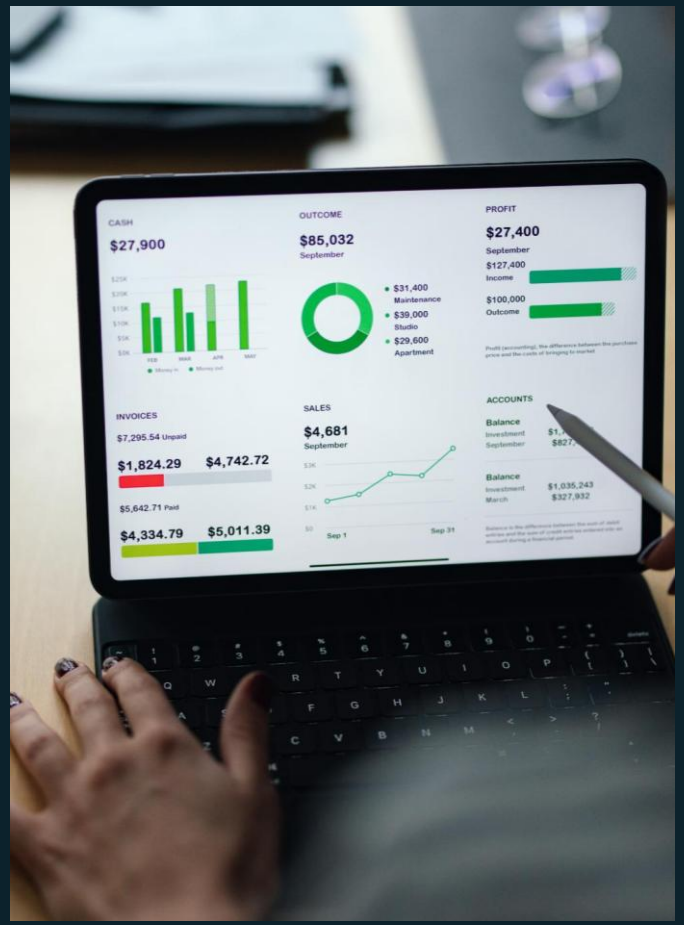
- **Multi-level alerts -** notify admins and stakeholders as usage approaches defined thresholds (e.g., 80%, 90%, 100% of quota).
- **Enforcement actions -** prepaid agents are automatically disabled after exceeding 125% of quota, while PAYG relies on alerts and manual or custom automatizations to control spend.

GET YOUR AGENTS READY FOR DEPLOYMENT

Set up Pay-As-You-Go

Manage Prepaid message capacity

Smart Cost Management Tips



IDENTIFY CONSUMPTION TRIGGERS

Recognize which features, like premium connectors and specific API calls, increase consumption and drive up costs.

LEVERAGE FREE FEATURES

Utilize free tools such as basic flows and standard connectors to minimize expenses in your workflows.

MONITOR AND SET ALERTS

Use built-in dashboards to track usage and set alerts to prevent overspending or unexpected charges.

REVIEW AND EDUCATE ON COSTS

Regularly review cost reports and inform users about pay-as-you-go scenarios to avoid surprise billing.

Governance After Publishing

Clear Agent Ownership

Assigning clear ownership to agents maintains accountability and enables proper change management after publishing.

Ongoing Data Monitoring

Regular scanning identifies sensitive data, tracks oversharing, etc. to safeguard information.

Monitoring and Retirement Practices

Utilize analytics for insights and responsibly retire agents to ensure ongoing security and compliance.

Closing & Next Steps

THANK YOU! ❤️

EXPLORE FURTHER RESOURCES

You are encouraged to use additional resources and tools to reinforce your workshop learning and skills.

SHARE AND INTEGRATE INSIGHTS

Share the key insights with your teams and integrate the new methods into your daily workflow for better outcomes.

RESOURCES

ALM Golden Rules

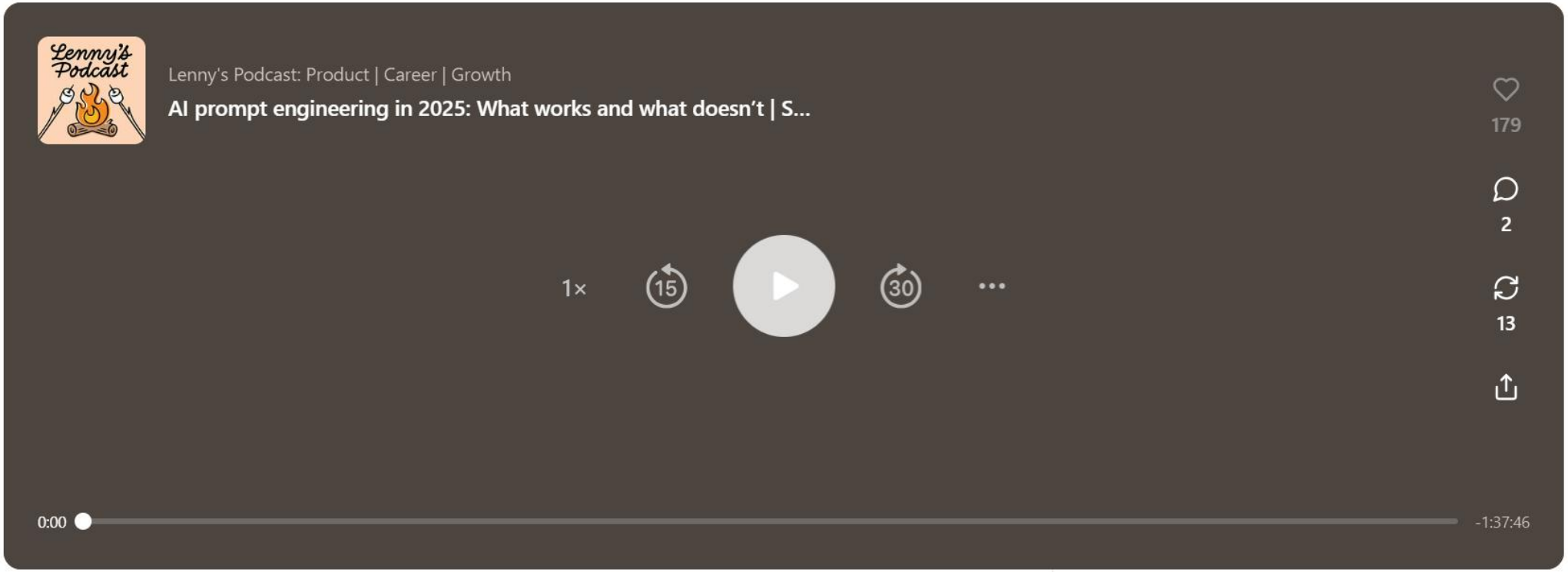
✓ Work in the context of solutions
✓ Create separate solutions only if you need to deploy components independently.
✓ Use a custom publisher and prefix.
✓ Use environment variables for settings and secrets that change across those.
✓ Export and deploy solutions as managed, unless setting up a dev environment.
✓ Don’t do customizations outside of dev.
✓ Consider automating ALM for source control and automated deployments.

Small mistakes early turn into big cleanup later

- Everyone

✓ Start small and build iteratively
✓ Test after every small change
✓ Use clear naming conventions
✓ Keep knowledge sources focused
✓ Validate knowledge early
✓ Split actions logically
- (1 flow = 1 purpose)

✓ Organize topics by function
✓ Think about future growth
✓ Avoid Over-engineering
✓ Have a clear idea of which channels to publish to
✓ Write effective instructions
✓ Leverage user feedback to
✓ Focus on Conversational Flow
- Design natural, engaging flows rather than rigid, menu-driven interactions for better user experience.



AI prompt engineering in 2025: What works and what doesn't | Sander Schulhoff (Learn Prompting, HackAPrompt)

How to get better answers from AI, avoid common prompt-engineering myths, and keep your AI tools safe from bad actors

LENNY RACHITSKY
JUN 19, 2025

Lenny's Podcast: Product | Career | Growth

Interviews with world-class product leaders and growth experts to uncover concrete, actionable, and tactical advice to help you build, launch, and grow your own product.

Copilot Studio Kit

The **Copilot Studio Kit** is a comprehensive set of capabilities built for Microsoft Copilot Studio. The kit includes features such as

📊 **Conversation KPIs** – Gain deeper insights into long-term agent performance with structured analytics.

🧪 **Test Automation** – Batch test custom agents with multiple validation methods.

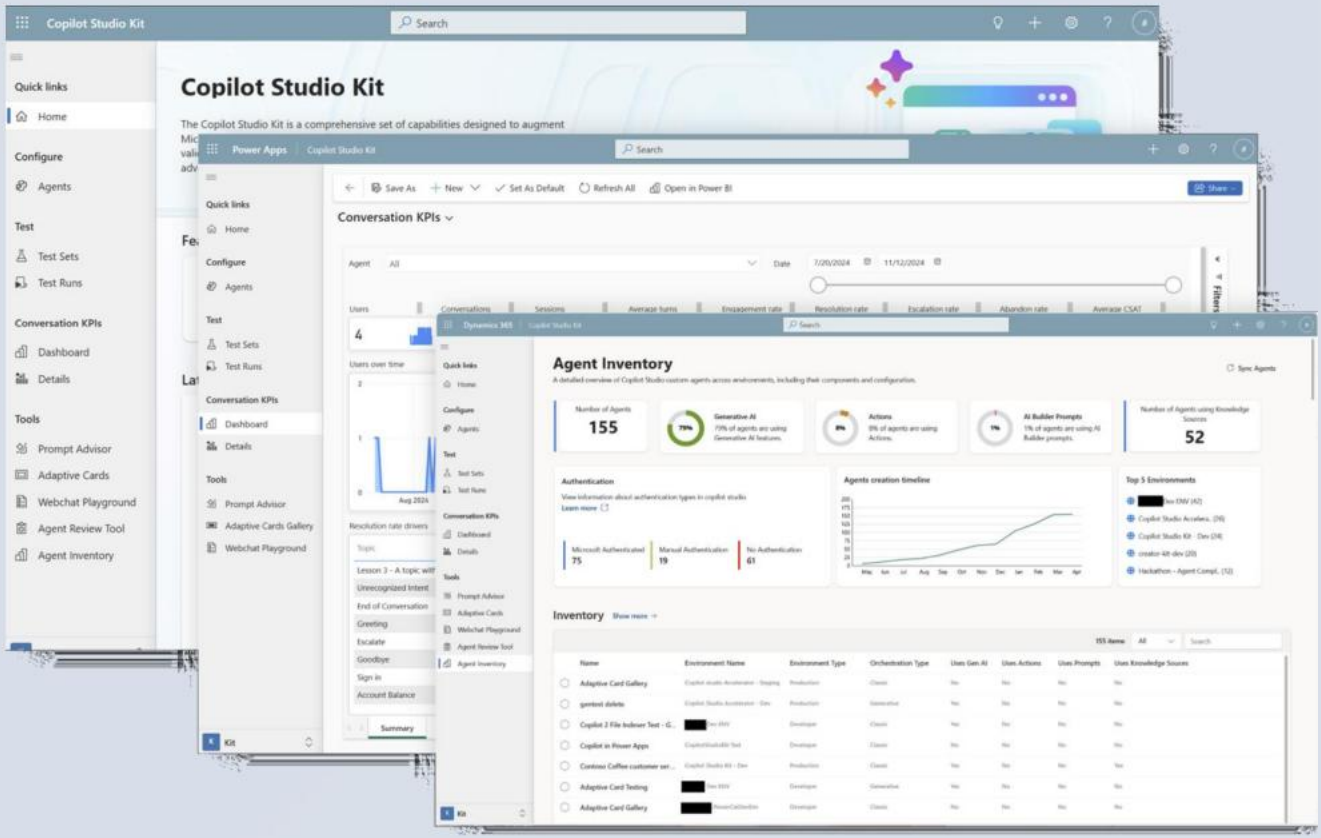
📈 **Agent Inventory** – Get tenant-wide visibility to all the Copilot Studio custom agents in the organization, across environments.

🛠️ **Agent Review Tool** - Static analysis of copilot studio agents from solution. Detecting anti-patterns and suggesting mitigation.

🗂️ **SharePoint Synchronization -** Selectively synchronize knowledge from a SharePoint site as local knowledge of your custom agent.

💬 **Webchat Playground** – Customize the look and feel of your webchat with an easy-to-use interface.

🧩 **Adaptive Cards Gallery** – Access a set of pre-built Adaptive Card templates for custom agents



➡️ https://aka.ms/DownloadCopilotStudioKit (AppSource)
➡️ https://aka.ms/CopilotStudioKit (GitHub + Docs)

INSPIRE

A Framework for Successful Implementations