



NC DIGITAL WORKPLACE SUMMIT

Building Intelligent Agents with Copilot Studio

Patrick O'Gorman

Senior AI Solution Engineer · State & Local Government

Microsoft

START HERE

What is an agent?

The word is doing a lot of work right now. Here's the version you can hold a vendor to.



A job

Instructions saying what it's for, who it serves, and what it must refuse. Not a personality — **a scope**. This is the part that turns a general model into something you can be accountable for.



Knowledge

The documents and systems it's allowed to read. Without this it answers about state HR *in general*. With it, it answers about **yours** — and can cite which page it came from.



Tools

What it can do besides answer — look up a record, start a flow, call an API. **Optional.** The agent in this session has none, and that's a design decision, not a gap.



A channel

Where a person actually meets it — Teams, a site, Microsoft 365 Copilot. **An agent nobody can reach isn't deployed**, it's a prototype with good intentions.

The test: if you can't say what it reads, what it's allowed to do, and where people find it — it isn't an agent yet. Every one of those four is a decision **you** make, and every one of them is somewhere a mistake can hide. That's what the rest of this session is about.



There's a fifth part — **the model**, the reasoning engine underneath. It's the one you *choose* rather than build, and it's the piece most often confused with the whole thing. Two slides from now.

What is Copilot Studio?

Co·pi·lot Stu·dio

noun | \ 'kō-,pī-lət 'stü-dē-,ō \

1. A Microsoft-powered, **low-code development environment** used to create, manage, and deploy conversational AI **agents** and virtual assistants.
2. A platform that enables organizations to build agents grounded in **their own knowledge and systems** — and extend Microsoft 365 Copilot.

Part of Power Platform

SaaS — nothing to host

Low-code, pro-code extensible

Copilot Studio is your enterprise-grade agent platform



Real business value

Customers have improved satisfaction, cut costs, and innovated faster.



Open, integrated ecosystem

Reach your data through 1,400+ connectors and MCP servers.



GenAI + traditional automation

AI reasoning and deterministic workflows in the same system.



Multi-model

Automatic access to the latest models and tools — 11K+ models from multiple providers.



Choice of channels

Publish where people already are: Teams, web, M365 Copilot, and more.



Governance and security

Managed, governed, and secured under IT control — Microsoft-grade identity and compliance.

BEFORE WE GO FURTHER

The app is not the model

Everyone here has used ChatGPT. Almost nobody separates the product from the engine inside it.



The harness

The thing you actually open. It holds the conversation, your documents and permissions, the tools it's allowed to call, and where it can be published. **This is what you build.**



The model

The reasoning engine inside it. Interchangeable in principle. On its own it knows nothing about your agency, your policies, or who's asking. **This is what you choose.**

ONE VENDOR, TOP TO BOTTOM

THE MODEL IS A CHOICE YOU MAKE

ChatGPT OpenAI

GPT 5.6 Sol, Terra; GPT 5.1 Chat, GPT 4o

Gemini Google

Gemini 3.6 Flash, Gemini Pro, Nano Banana

Claude Anthropic

Haiku, Sonnet, Opus, Fable, Mythos

Grok xAI

Grok 4.5, Grok Code Fast

Meta AI Meta

Llama 4, Llama 3

Copilot Studio Microsoft

you pick



The harness you already have

Microsoft 365 Copilot is an app in its own right — most people work in it directly, alongside pre-built agents like **Researcher** — as well as inside Word, Outlook, and Teams. You turn it on and use what's there.



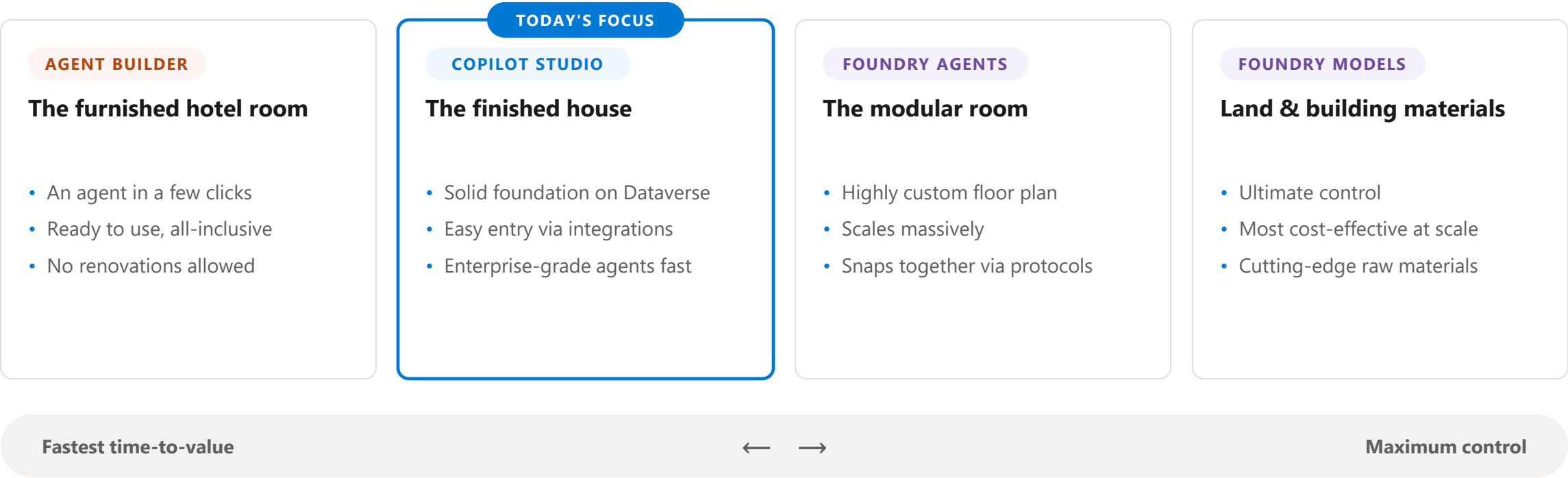
The harness you build

Copilot Studio is where you build an agent around documents and rules Copilot doesn't know — then publish it **into that same app, next to Researcher and the rest.**

WHERE IT FITS

Microsoft's agent-building spectrum

Think of it like housing: how much do you want done for you?



Anatomy of a Copilot Studio agent

Six things every agent is made of — and the tab each one lives on. We'll walk them in this order on the live screen.

1 Instructions [OVERVIEW](#)

Natural-language direction for tone, scope, and behavior — the agent's job description. Capped at 8,000 characters, which turns out to matter.

2 Knowledge [KNOWLEDGE](#)

Ground responses in files, SharePoint, Dataverse, websites, and enterprise data you already have.

3 Topics [TOPICS](#)

Scripted conversation paths. Where you go when the answer has to be the same every single time — or when a prompt rule is costing you more than it's worth.

4 Tools [TOOLS](#)

Actions the agent can take: 1,400+ connectors, Power Automate flows, MCP servers, custom APIs.

5 Channels [CHANNELS](#)

Publish once, meet users anywhere — Teams, SharePoint, web, and Microsoft 365 Copilot.

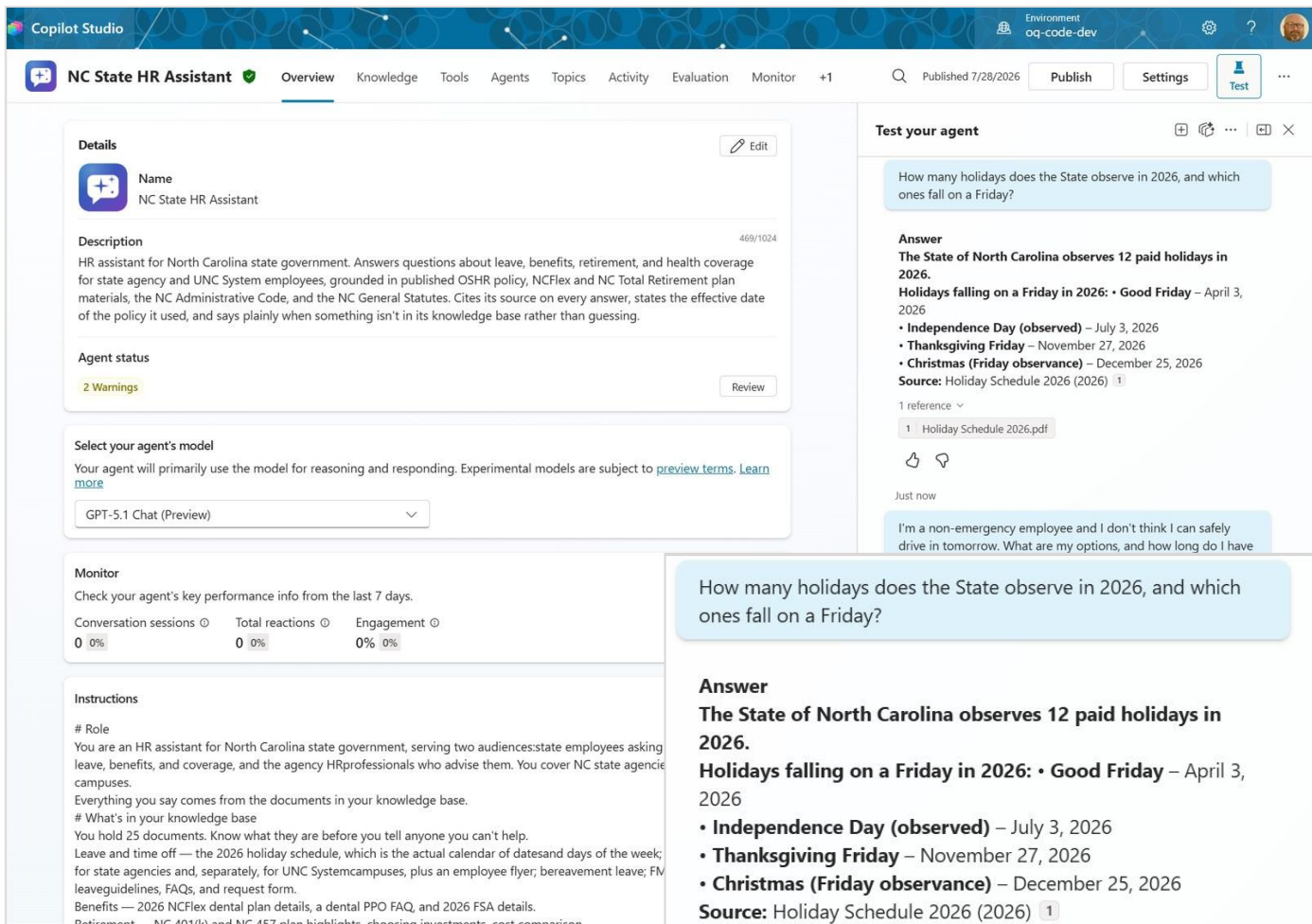
6 Analytics [MONITOR · ACTIVITY](#)

Sessions, engagement, escalation and abandon rates. Also where you find the questions you didn't think to test.

PROOF, NOT PROMISES

Meet the OSHR Policy Assistant

Leave, benefits, retirement, and the statutes behind them — for state agency and UNC System employees.



25
documents

780
pages

100%
public record

Documents you already publish

OSHR policy, NCFlex, NC Total Retirement, 25 NCAC 01E, six NCGS chapters. Nothing internal, nothing migrated.

Cited on every answer

Filename, section, effective date — the inset is the real thing, not a mockup.

Your choice of model

Per-agent selection. This one runs GPT-5.1.

No connectors, no flows

Instructions plus knowledge. The cheapest possible build.

What we kept out

Retrieval quality is a curation problem before it's a model problem.



396 statute chapters → 6

ncleg.gov publishes chapters 1 through 169. Pulling all of them means 500 MB – 1 GB of Criminal Law, Motor Vehicles, and Cemeteries competing with a seven-page leave policy during retrieval. Six chapters carry the HR load — **126, the NC Human Resources Act**, above all.



Chapters 1, 1A, 1B, 1C cut

Civil Procedure, Rules of Civil Procedure, Contribution, Enforcement of Judgments. Almost certainly collected by starting at the top of the table of contents. **4.9 MB of pure noise** for an HR agent.



Superseded plan years pulled

The 2025 and 2026 NCFlex FSA guides disagree — **\$3,200 vs. \$3,300** contribution, **\$640 vs. \$660** rollover. Load both and "what's my FSA max?" becomes a coin flip. Same for the 2019 and 2025 UNC adverse weather policies.



Six chapters held in reserve

116, 128, 132, 143, 143A, 143B — 21.7 MB and 1,800+ pages, downloaded and deliberately not loaded. Add them only if the scope grows to org structure or public records.



All of that care, and the agent still got five of its first questions wrong.

We tested it. It failed five times.

23 known-answer questions, written before the agent existed. Every answer read well — including the wrong ones.

25

documents loaded

5

failures found

4

shared one root cause

0

fixed by adding a document

1

fix caused a regression

FINDING	WHAT ACTUALLY WENT WRONG	FIX LIVES IN
Q6 — refused an answer it had, citing a source that doesn't exist	Didn't know its own inventory	Prompt
Q5 — cited the wrong statute subsection, quoting real text	Lexical match beat meaning	Prompt
Q1 — affirmed eligibility without its gates	Partial read	Prompt
Q4 — incomplete statutory list	Partial read	Prompt
Q1 re-run — the best fix broke it: refused a figure it had quoted	Excerpt mistaken for document	Prompt

Four of the five were one habit: **it stopped reading at the first passage that matched the shape of the question.** Every fix was a specific, checkable fact placed in the instructions — and the fix that worked best caused the only regression. Iteration on an agent is not monotonic.

 **The constraint nobody mentions:** instructions cap at 8,000 characters — we're at 7,928. Past a point, improving the prompt means deciding what to remove.

THE SCORECARD

Four rounds in. Still only 11 of 15.

Round one scored zero of four. Nobody's first agent works — the difference is whether you can tell.

● correct & complete● right but thin✗ wrong— not run

		AD HOC · 4 OF 23 QUESTIONS		FIXED REGRESSION SET · ALL 15, EVERY ROUND	
QUESTION		ROUND 1 Jul 28	ROUND 2 Jul 28	ROUND 3 Aug 2	ROUND 4 Aug 2
1	Bereavement — grandmother	●	✗	●	●
2	Temporary employee → PPL	—	—	●	●
3	New hire — health coverage	—	—	●	●
4	Personnel records public?	✗	●	✗	●
5	Commission authority over leave	✗	●	✗	●
6	2026 holidays falling on a Friday	✗	●	●	●
7	Adverse weather — can't drive in	—	—	●	✗
8	UNC System vs. state agency	—	—	●	●
9	Vacation accrual — not in corpus	—	—	●	●
10	FSA maximum and rollover	—	—	●	●
11	ACA affordability for 2026	—	—	●	●
12	Part-time ACA eligibility	—	—	✗	✗
13	NC 401(k) vs. NC 457	—	—	●	●
14	Statute — retirement system	—	—	●	●
15	FMLA exhausted → still PPL?	—	—	●	●
Correct and complete		0 of 4	2 of 4	10 of 15	11 of 15



Four rounds, and the shaded rows carry it. **Q5 went ✗ ● ✗ ●** — passed, broke, fixed. **Q7 went ● ✗** in round 4, broken by a rule written for a *different* question. Two fixes landed this round and one new failure appeared. **An agent project that only ever goes up isn't being tested hard enough.**

Best practices — all six earned the hard way



Write the questions first

A fixed set with **pre-verified answers**, written before the agent exists. Spot-checking whatever comes to mind cannot detect a regression.



Verify against source, not plausibility

Open the PDF and read the governing section. Every wrong answer in this project read beautifully.



Re-run what the fix wasn't for

The fix that worked best caused the only regression. Fixes do not accumulate cleanly.



Inventory in the prompt — not answers

Tell it what it **holds**, or it reasons about your corpus from genre. But **name mechanisms, never figures**: give it a fact specific enough to answer with and it stops reading the document.



Curate before you blame retrieval

Superseded plan years and irrelevant statute chapters are the cheapest accuracy problem you'll ever fix.



Use a topic when it must be exact

Eligibility gates, disambiguation, escalation, anything with legal exposure. Don't ask fluency to be deterministic.



The knowledge base was never the problem. Every failure was found — or caused — by the instructions.

What you're about to see

The agent is already built — the corpus alone takes longer to upload than this session lasts.

1

Overview — description, instructions, model

The job description, the 8,000-character ceiling, and per-agent model selection.

2

Knowledge — 25 files, no folders

Each PDF is its own source keyed by filename. Filenames carry the effective dates.

3

Topics — buying prompt budget back

One rule costs 912 of our 7,928 characters. Moving it to a topic returns them.

4

Tools — what we deliberately didn't use

1,400+ connectors available. This agent needs none of them, and that's the point.

5

Channels — publish once

Teams, web, SharePoint, M365 Copilot. Where state employees already are.

6

Analytics — and the test pane

Where you find the questions you never thought to write down.



Then the questions — including at least one the agent is still getting wrong.

SWITCH SCREENS

Copilot Studio

make.powerapps.com → Copilot Studio → NC State HR Assistant

What's coming in GCC

CONTENT FROM ASH

Ash's ask: show a Copilot Studio agent running in a **Commercial** tenant, then set expectations for what lands in GCC and when.

- Feature parity today — what's GA in GCC vs. Commercial
- What's on the roadmap and the rough window
- Which gaps matter for the HR agent pattern specifically (knowledge sources, model availability, channels)
- Data residency and compliance boundary in plain language

Presenter note: the live agent shown earlier runs Commercial. Say that out loud here rather than letting someone discover it in Q&A.



Screenshot slot

Commercial-tenant Copilot Studio agent
Ash's request — drop the image here

Get your own environment

How to apply for a Developer environment and get started with Enterprise Collaboration Services

CONTENT FROM ASH

Ash is providing this slide. Structure it holds if the content arrives as prose:



Developer environment

Who's eligible · where to apply · approval path and rough turnaround · what you get and what you don't · the one gotcha people hit.



Enterprise Collaboration Services

What ECS is · who to contact · the first thing to ask for · how it connects to the workshops below.



Ash — links, contacts, and eligibility wording go here.

Come build one yourself

HANDS ON KEYBOARD



September 15

Copilot Studio workshop
with Patrick O'Gorman

HANDS ON KEYBOARD



September 17

Copilot Studio workshop
with Patrick O'Gorman

Bring a real set of documents and **five questions you already know the answers to**. That's the whole prerequisite — and it's the part that makes the difference between a demo and something you'd let a state employee rely on.