CopilotKit

LIVE SESSION · THU NOV 5 2026

The rise of functional agents

Building Enterprise-Ready Functional Copilots

Nathan Tarbert · Jerel Velarde

BEFORE WE START

Quick question for the group...

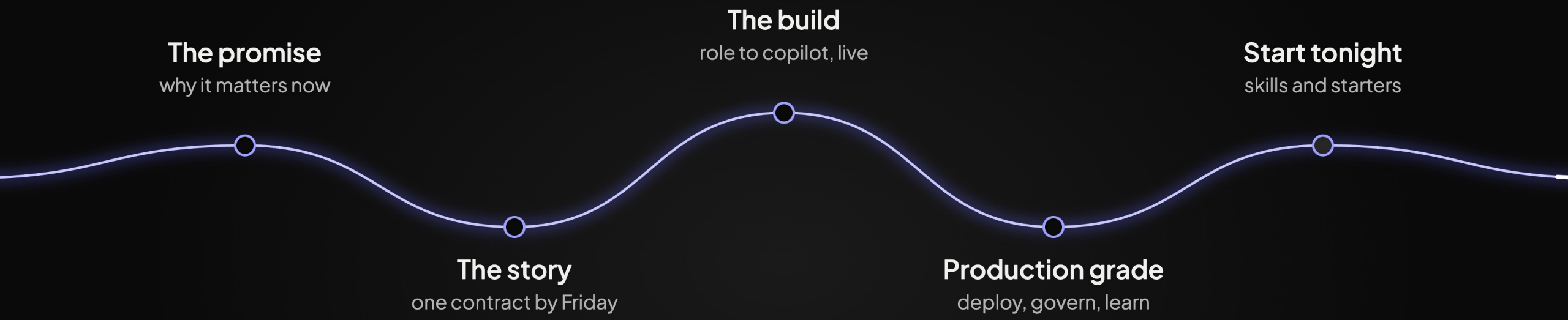
What's your role?



CopilotKit

AGENDA

The next hour



Questions in the chat any time · live Q&A in the last 10 minutes

YOUR HOSTS

Who we are



Nathan Tarbert

Senior Developer Relations Engineer, CopilotKit

INTRO AND Q&A



Jerel Velarde

Staff Applied AI Engineer, CopilotKit

TALK AND LIVE DEMO

CopilotKit | The agentic product stack



GITHUB STARS



DOWNLOADS A MONTH

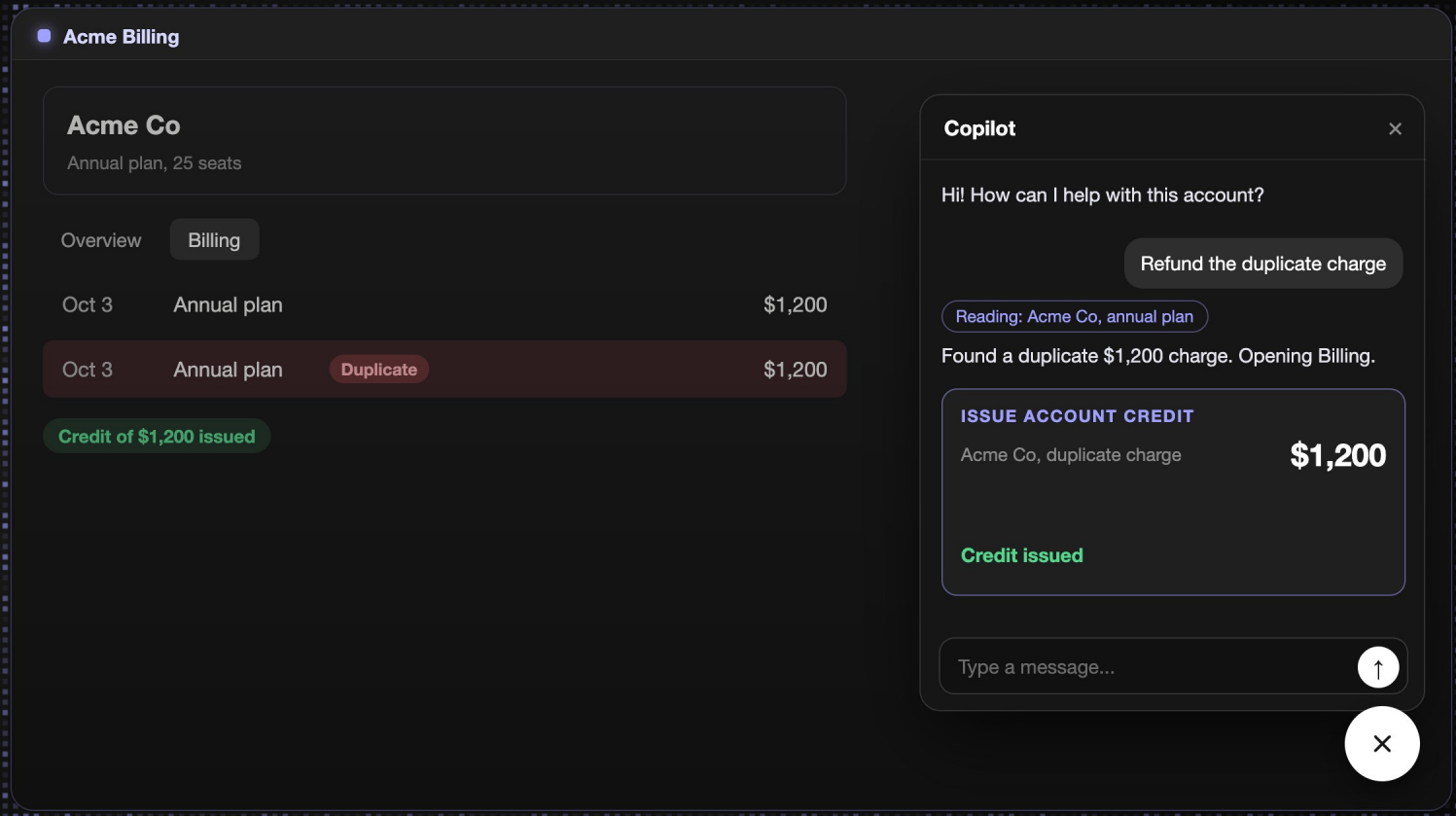
Trusted by most of the Fortune 500

Cisco · Apple · Walmart · Disney · Bank of America · Deutsche Telekom · Tesla · Ford · Toyota · Honda · Boeing · Salesforce · Intel · Hewlett Packard · Dell · AT&T · Comcast · Wells Fargo · Citi · Mastercard · Costco Wholesale · Nike · Shell · General Electric · Booking.com · FedEx · UPS · Tripadvisor · UKG · docusign · McDonald's · SoundCloud

COPILOTKIT

CopilotKit connects your agents to your users

- Chat and headless UI
- Real-time app context
- Controls your UI
- Generative UI
- **Human in the loop**



CopilotKit

AG-UI | We created AG-UI and the industry adopted it



In partnership with

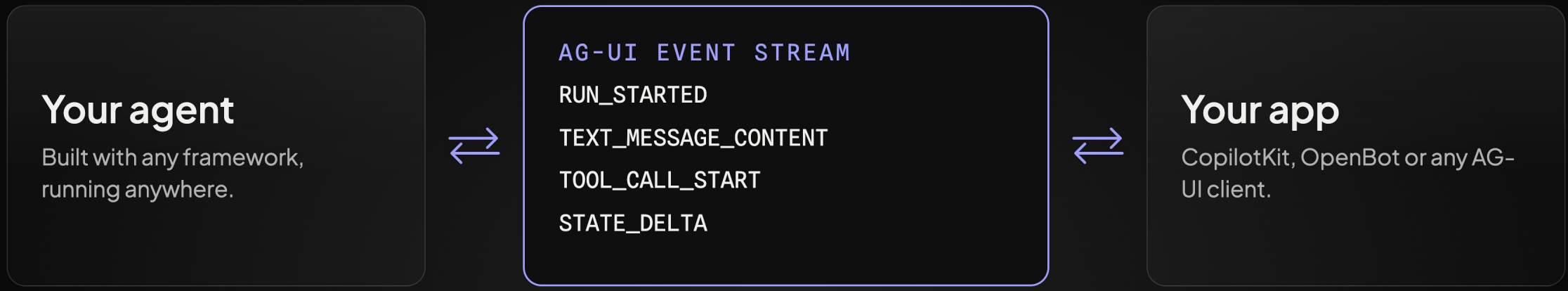
Google | Microsoft | amazon | ANTHROP\C | ORACLE | LangChain | crewai | mastra

LlamaIndex | Pydantic AI | _\gno | A2 | TANSTACK | tavily | Daytona | TAKO

AG-UI

What is AG-UI

The open protocol for how agents talk to the apps people use



Adopted by Google, AWS, Microsoft, LangChain, Mastra and more

LangGraph · CrewAI · Mastra · Pydantic AI · Google ADK · AWS Strands · Microsoft Agent Framework · LlamaIndex · Agno · AG2

Functional Copilots

START WITH THE END IN MIND

Design and build functional agents with CopilotKit, and learn how to improve their performance in production.

PLUS a head start: the skills and two starters, in one public repo

CopilotKit

THE LANDSCAPE · NOV 2026

Functional copilots are already big business

FUNCTION	COMPANY · LIVES IN	VALUATION	LATEST RAISE	REPORTED ARR
Coding	**Cursor** Its own editor, a fork of VS Code	$60B	Acquired · Aug '26	>$4B
Legal	**Harvey** Inside Microsoft Word	$15.5B	$550M · Sep '26	>$400M
	Legora Inside Word and Outlook	$8.8B	$550M · Oct '26	>$200M
Support	**Sierra** In your chat, voice and email	>$15B	$950M · May '26	$400M+
Healthcare	**Abridge** Inside Epic	$5.3B	$316M · Apr '26	Not reported
Finance	**Basis** Inside the client's ledger	$1.15B	$100M · Feb '26	Not reported
Video	**Kino AI** An editor shared with agents	Early · YC S23	Undisclosed	Not reported

As reported by the companies, filings or the press, Nov 5, 2026; not audited. Bars are latest valuations, to scale (Cursor: value at its acquisition).

CopilotKit

WHY WE'RE ALL HERE

In company after company, developers are being asked to build functional agents

THE UPSIDE

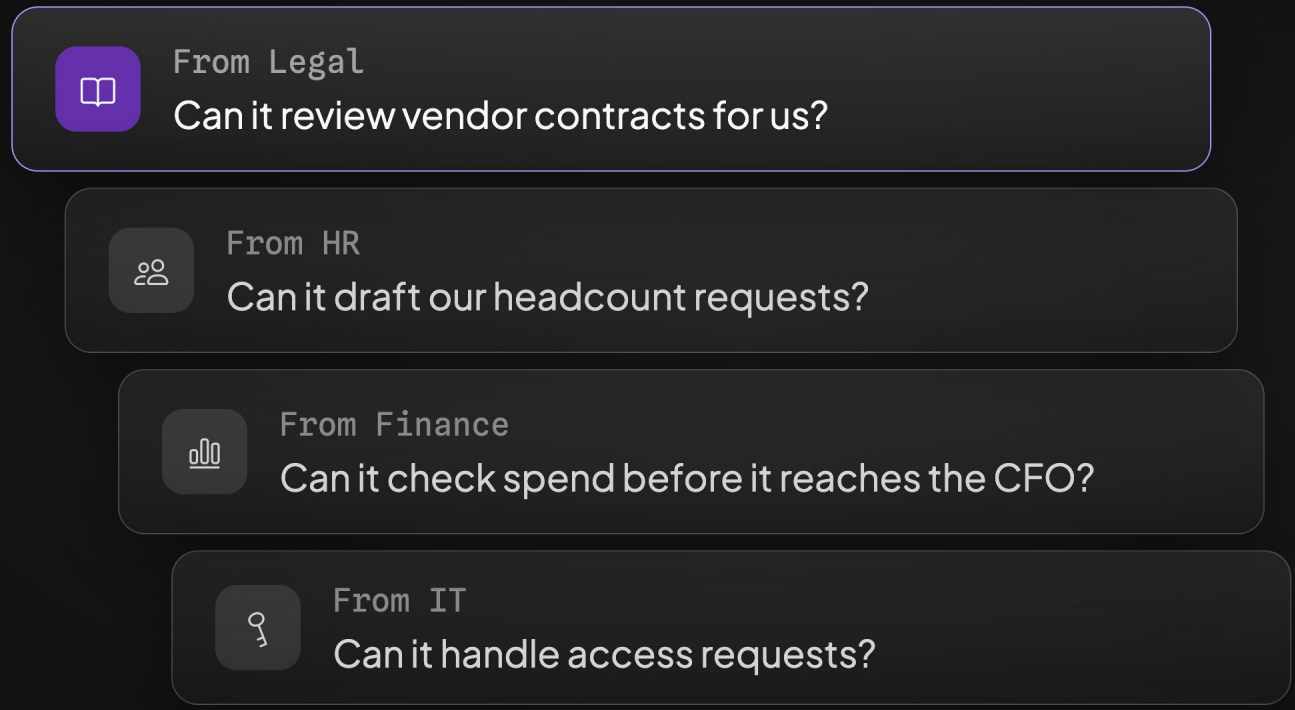
Hours back on work that repeats every week

THE CATCH

It can't go off the rails

TODAY

Build one live, then make it safe to ship



THE SAME CONTRACT, TWO WAYS

From answering questions to owning the work

	CHAT ASSISTANT	FUNCTIONAL COPILOT	
Read the contract	You paste clauses in	Reads the open matter itself	
Check the playbook	You ask, one rule at a time	Checks every rule, cites each	
Draft the redlines	You copy its text into Word	Drafts them in the review table	You edit
Route for approval	You chase three approvers	Routes by rule, waits for each	You send
Learn from send-backs	Forgotten by next week	Proposes a rule for next time	GC publishes

A functional copilot does the work inside the team's software, and a person makes every call.

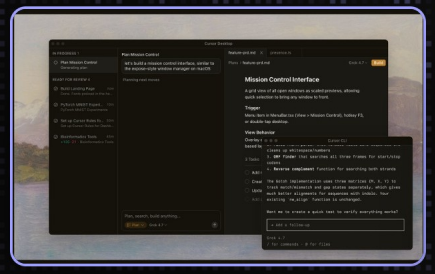
Poll 1 is open

WHERE IT LIVES

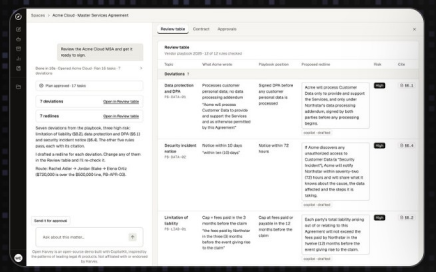
It lives where the work happens

WENT FIRST

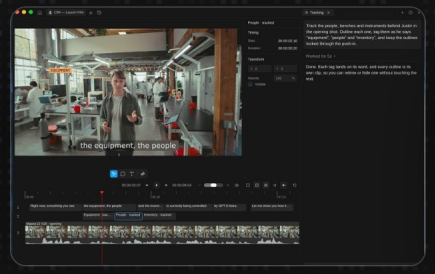
NOW EVERY FUNCTION



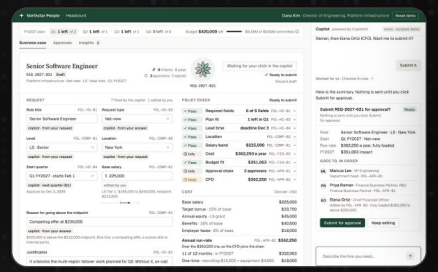
Coding
Cursor



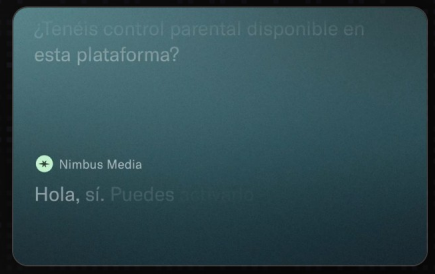
Legal
Open Harvey · ours



Video
Kino AI



HR
Headcount starter · ours



Support
Sierra

Coding went first. Every function is next.

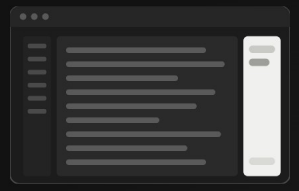
Cursor forked an IDE and put the agent inside it, where developers already work.

CopilotKit

Cursor, Kino and Sierra screenshots from their public websites, for illustration; not affiliated with or endorsed by the companies shown. Open Harvey: an open-source CopilotKit demo, not affiliated with or endorsed by Harvey.

THE 2026 SHIFT

Plug in, or rebuild it AI-first

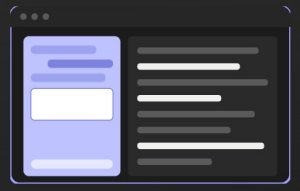


PLUG IN

Add the agent to the tool people already use

Legal	Harvey and Legora ship Word add-ins
Healthcare	Abridge works inside Epic

Fast to reach people. You live inside someone else's UI.



REBUILD AI-FIRST

Build the workspace around the agent

Coding	Cursor forked VS Code into its own editor
Video	Kino AI built its own editor, not a plugin
Creative	Higgsfield built its own AI studio first

More to build. You own the UI, the data and the approvals.

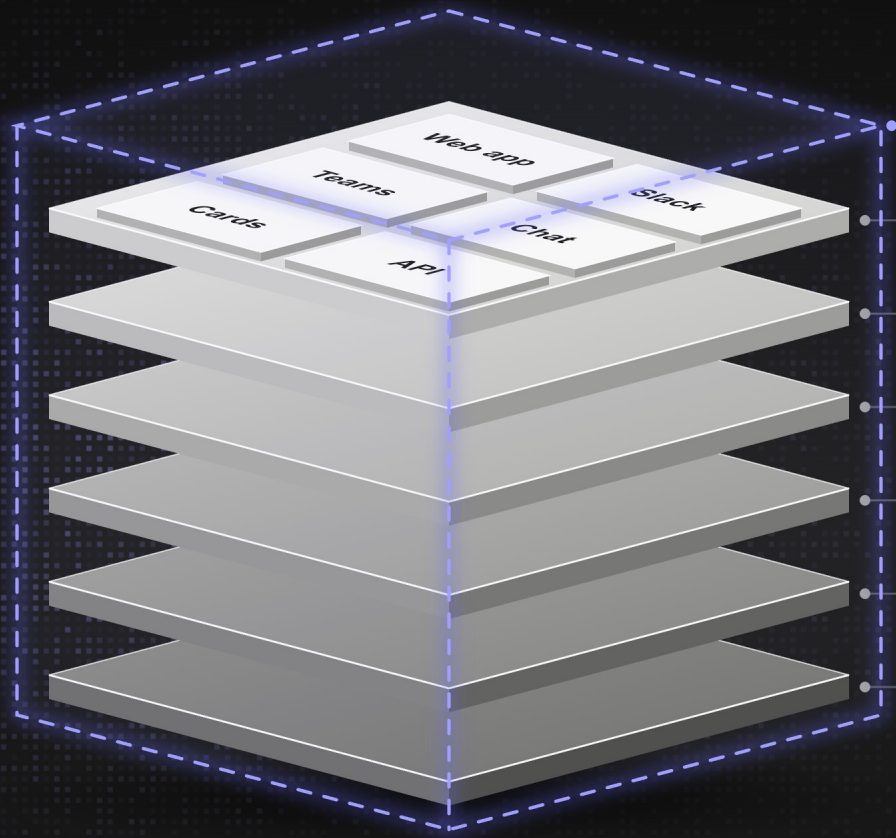
Today: build a function-specific, AI-first copilot with CopilotKit.

Open Harvey is ours: a legal review workspace built around the agent.

CopilotKit

THE 2026 STACK

The functional agent stack



Security and governance
Around every layer: SSO, roles, audit; OpenBot ships it

Products and interfaces
Your app with CopilotKit, or Slack and Teams via Channels

Domain capabilities
The jobs of one function: review, draft, check, route

Context and knowledge
Policies, history and what Intelligence learns

Data and integrations
Systems of record, through MCP or direct APIs

Agent harness
CopilotKit Runtime over AG-UI, any agent framework

Models
Any model you choose; the key stays on the server

CopilotKit

THE STORY · THURSDAY, 9:12 AM

Acme’s contract has to be signed by Friday

Northstar Software is buying from Acme Cloud · $240,000 a year for three years · both fictional



Dana Kim
Platform Infrastructure
Needs it signed by Friday

Maya Chen
Senior Counsel
Reviews it, with the copilot

Rachel Adler
General Counsel
Approves the deviations

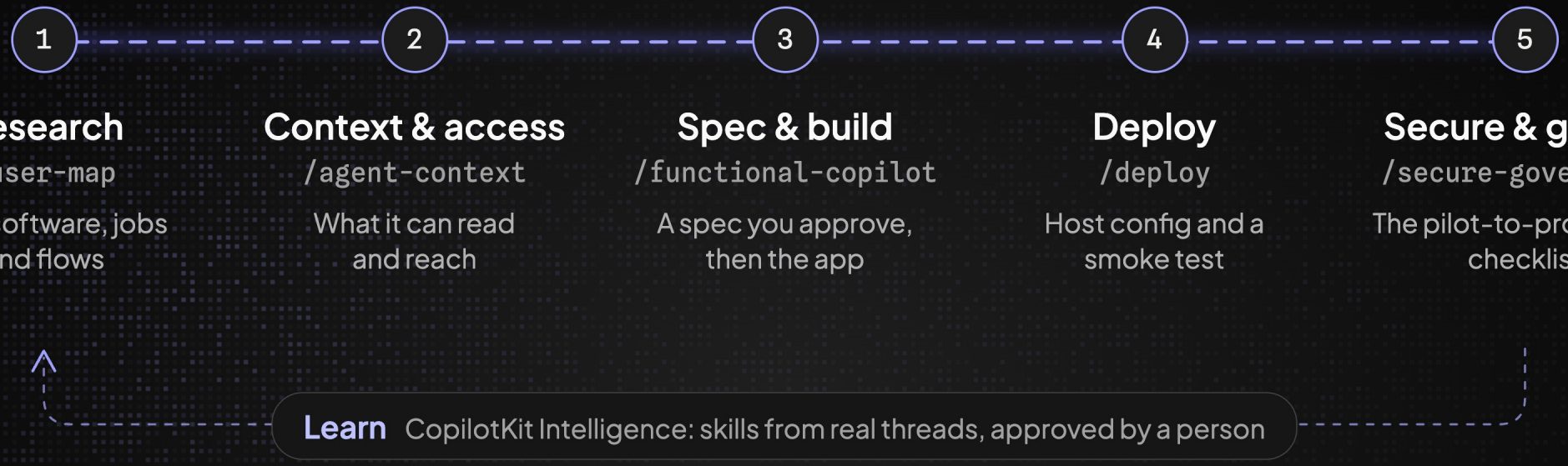
Jordan Blake
Head of Security
Sends it back

Elena Ortiz
CFO
Approves the spend

One contract, four teams, three approvers.

THE METHOD

From a role to a copilot, in five steps



Run /functional-copilot. It walks you through every step.

Each step ends in a page you keep open

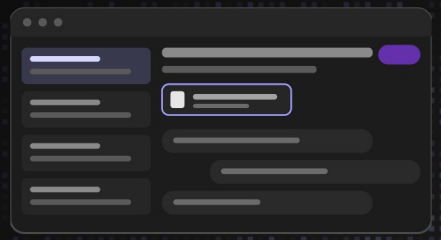
CopilotKit

Step 1 · Research · /user-map

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

Let's say you're tasked to build a legal copilot.

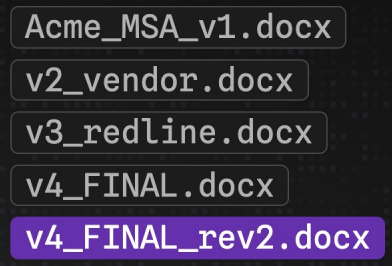
First, map how the work happens today. /user-map · senior counsel, 41 open vendor contracts



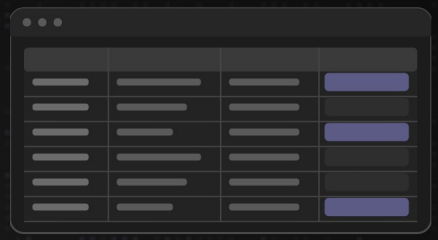
Email
Deadline, approvals



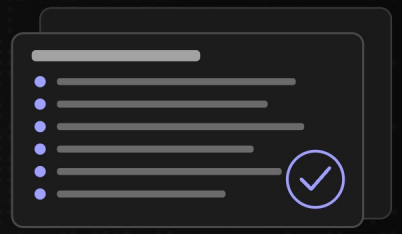
The document
Word with Track Changes



Many versions
A new file every round



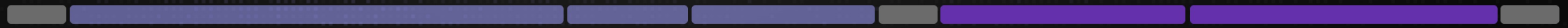
A spreadsheet
Deviations, by hand



The playbook
Rules, clause templates

WHERE TIME GOES

the study's fictional figures: 10.5-day median, a third approved first time



Received, then about two days reading the playbook

Spreadsheet, then redlines

Three approvers, days waiting

SOC 2 send-back: a week lost

To vendor

CopilotKit

Step 1 · Research · /user-map

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

The jobs to be done

From the same study: what senior counsel needs done, in their words.

J1

When a vendor contract lands in my queue,

I want to have it checked against our playbook, with every deviation cited and a redline drafted.

J2

When a position falls below the playbook,

I want to route it to the right approvers by rule, with my reason attached.

J3

When an approver sends a review back,

I want to see what they asked and whether it has happened before.

J4 assumed

When the business asks what our contracts say about something,

I want to get one answer across the portfolio, with each contract's words quoted.

J5

When the same unwritten rule keeps coming up,

I want to add it to the playbook once the General Counsel agrees.

J6 assumed

When the business owner asks for a date,

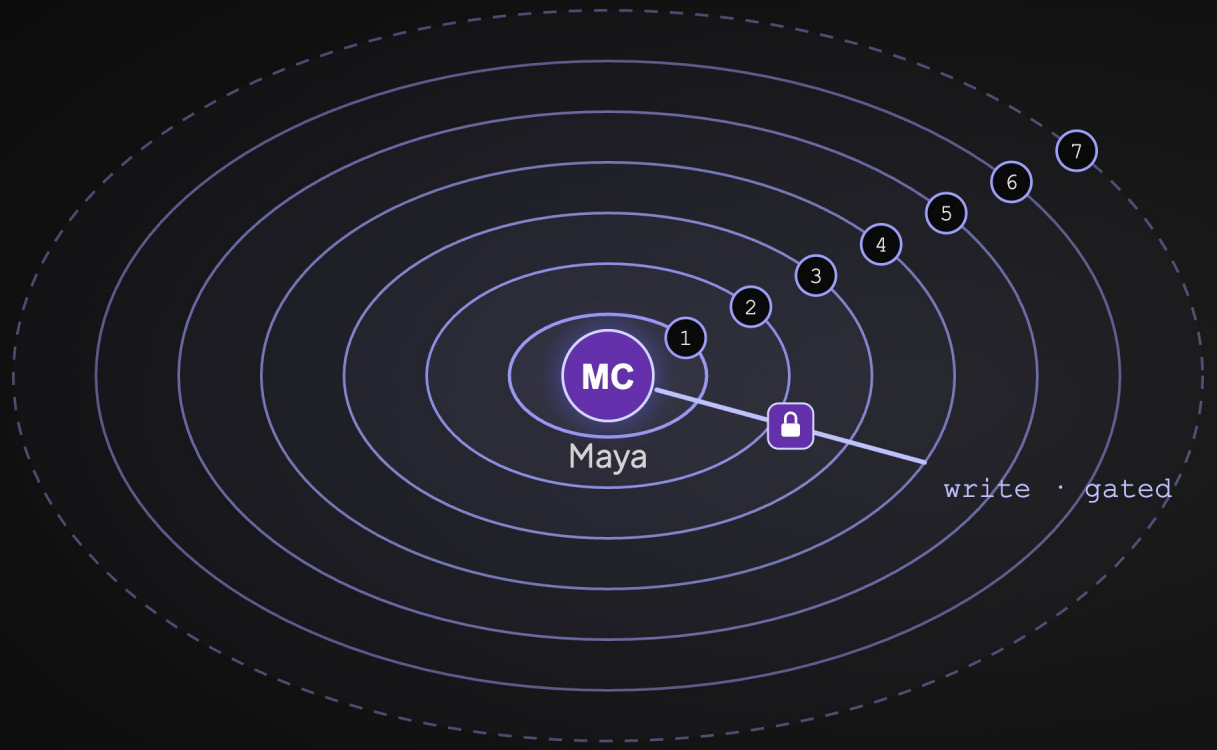
I want to see where the contract stands and who it waits on.

So the copilot lives in the review, the redlines and the approvals.

Step 2 · Context & access

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

Map what it can reach



1. **In the app** `useAgentContext`
   The open matter and the plan
2. **Your computer** `local tool or MCP`
   The files on her laptop
3. **Your server** `defineTool`
   The playbook and the terms
4. **Org systems** `MCP`
   The contract repository
5. **People** `Channels`
   Approvers in Slack or Teams
6. **Coworkers** `OpenBot`
   The Contract Copilot
7. **Memory** `Intelligence`
   Threads and learning

Use the closest ring that works. Gate every write.

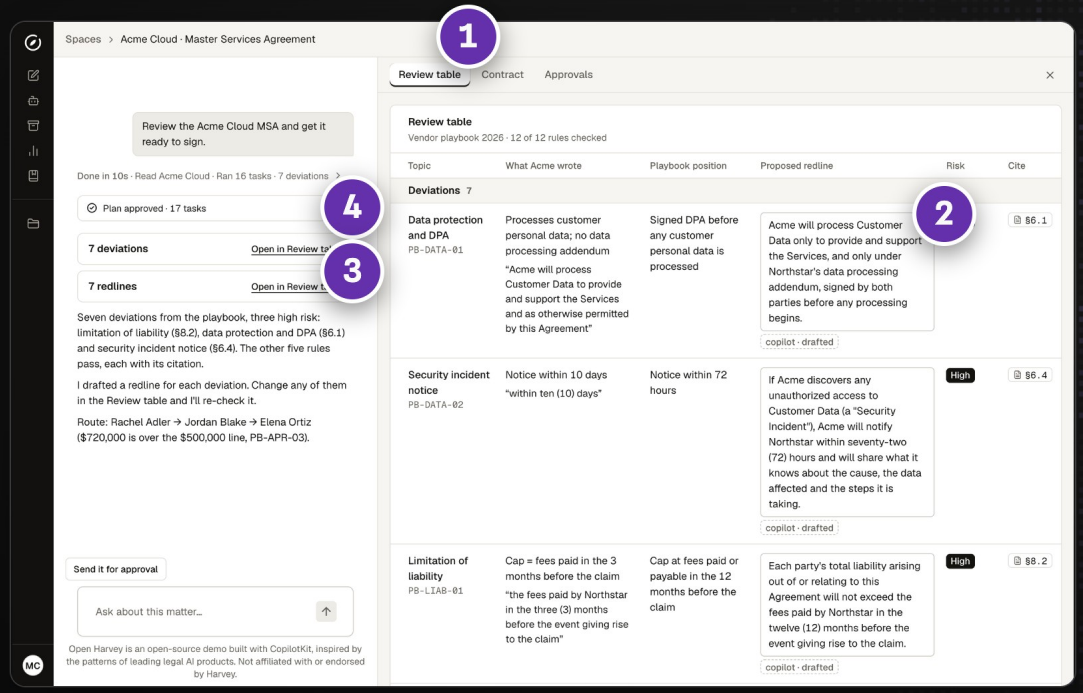
CopilotKit

Step 3 · Spec & build

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

Built with CopilotKit: four hooks connect your app to any agent.

Over AG-UI, the same hooks work with CopilotKit's built-in agent, LangGraph, Mastra, Google ADK and others.



1. **What it sees**
   `useAgentContext`
   the open matter, the plan, the findings
2. **What it can do in the app**
   `useFrontendTool`
   run a task, draft a redline, fill a cell
3. **What it shows**
   `useComponent`
   the plan card, cited findings, redline diffs
4. **What waits for a person**
   `useHumanInTheLoop`
   approve the plan, send for approval

LIVE DEMO · OPEN HARVEY · STEPS 1–4

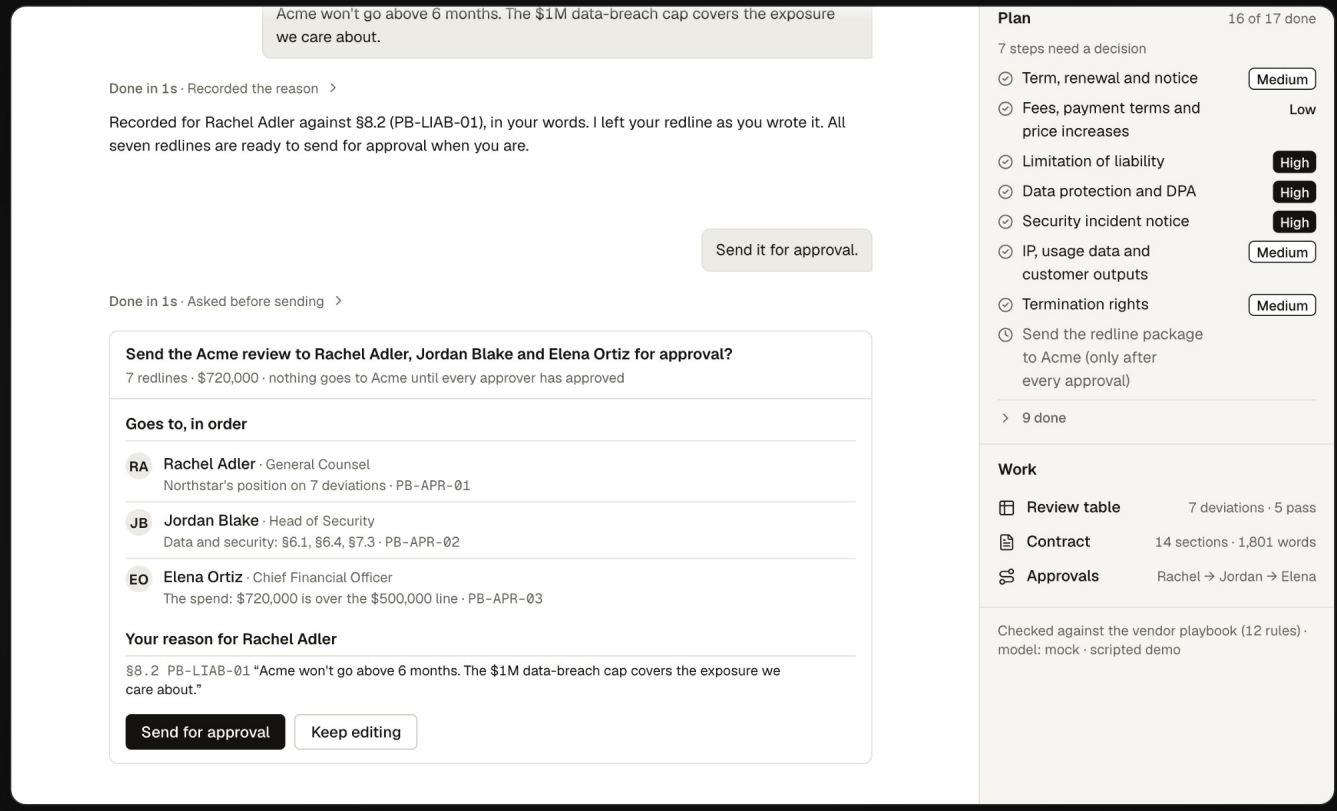
From the ask to the approval gate

01 **Plan**
Maya approves the 17-task plan

02 **Run**
Every finding cites its passage

03 **Edit**
She rewrites §8.2; it re-checks

04 **Gate**
Nothing leaves before her click



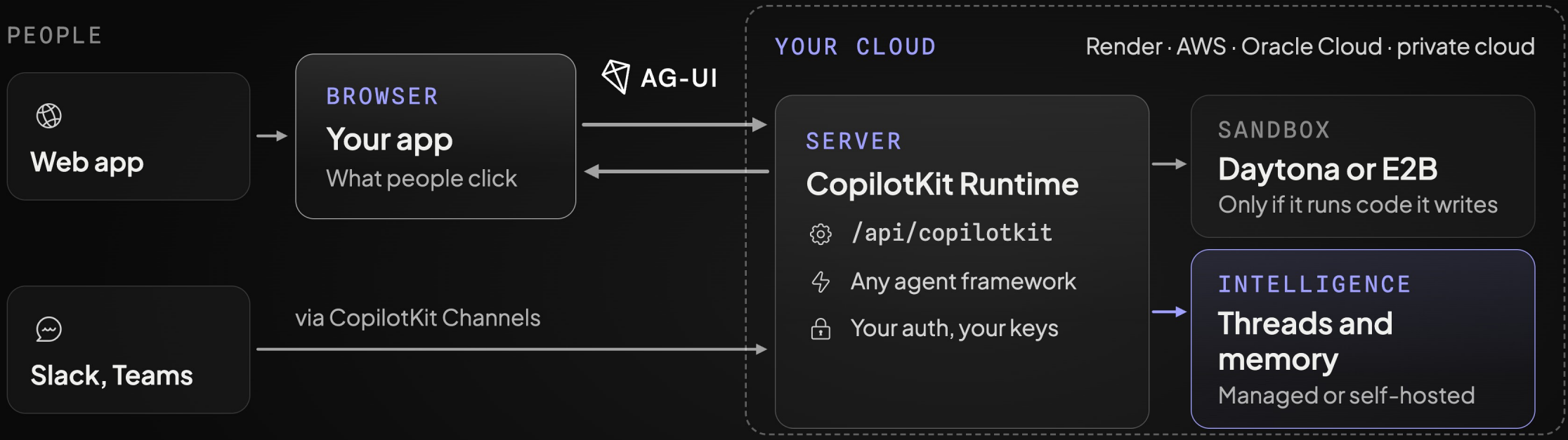
Contract terms come from a tested file; your extractor and its evals go here.

CopilotKit

Open Harvey: an open-source CopilotKit demo, not affiliated with or endorsed by Harvey.

DEPLOY · WHERE IT RUNS

Where each piece runs



The starter is one Next.js app. Any Node host runs it.

Bind to 0.0.0.0 and $PORT, and put /api/copilotkit behind your auth.

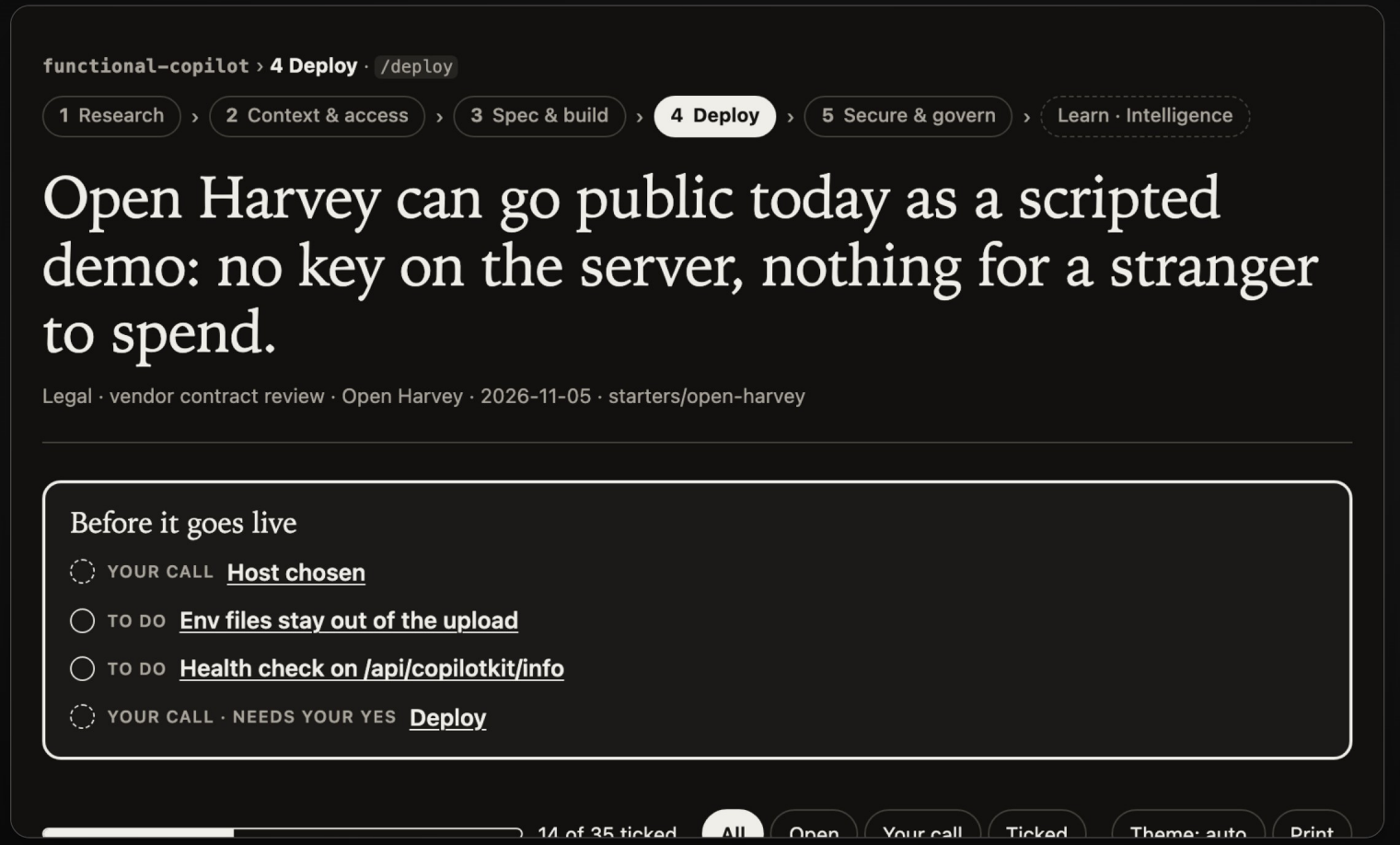
Step 4 · Deploy

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

Put it online with /deploy

/deploy

It prepares everything, then stops at “Needs your yes”.

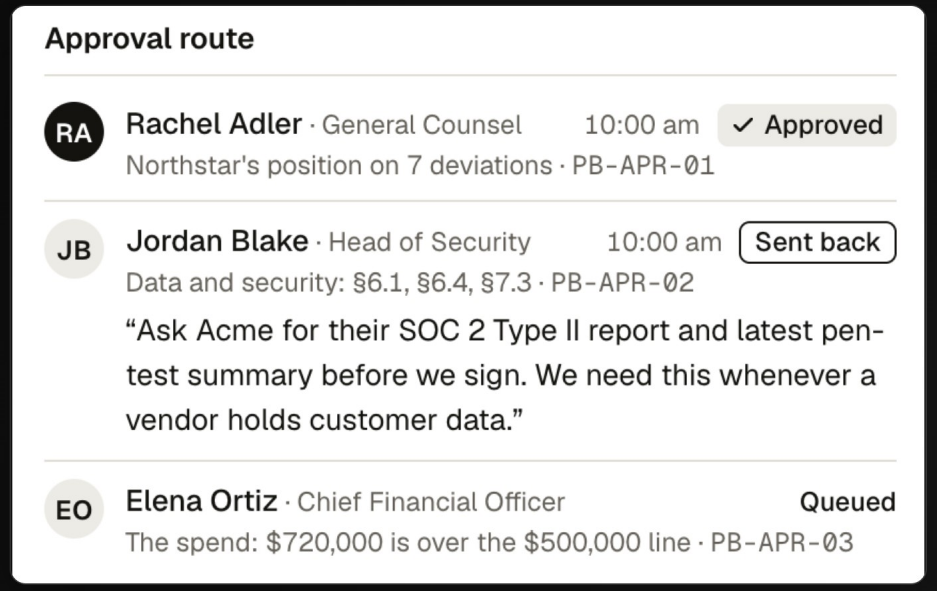


CopilotKit

GOVERN · DEMO STEP 5

Keep it on the rails

Poll 2 · What's blocking your agent from production?



Named approvers

General Counsel, Security, CFO

Routing by rule

PB-APR-01 to 03, in code, not the prompt

An audit trail

Every decision, in the approver's words

Next: the whole production checklist, run by /secure-governance.

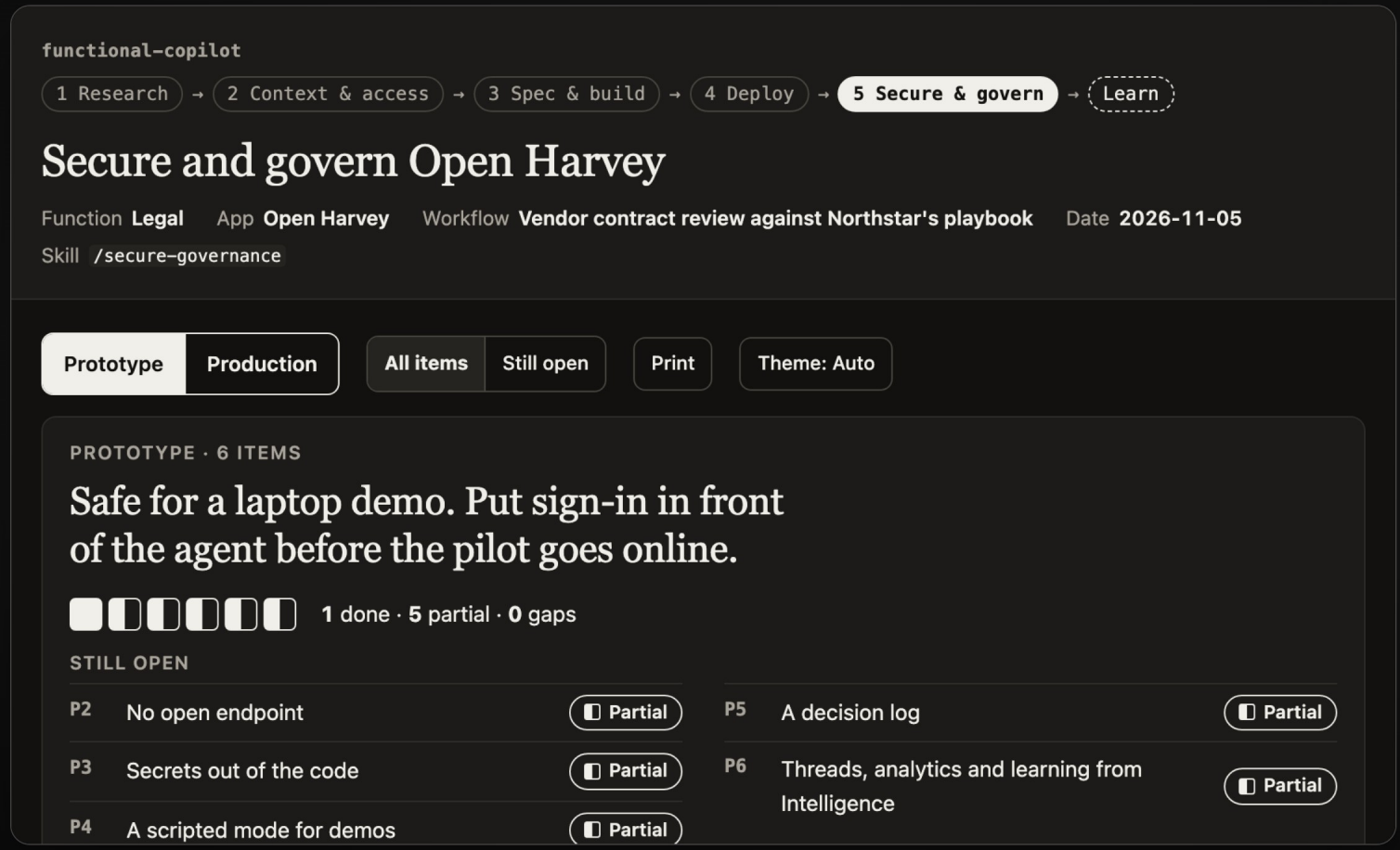
Step 5 · Secure & govern

1 Research | 2 Context & access | 3 Spec & build | 4 Deploy | 5 Secure & govern

From MVP to enterprise-ready

/secure-governance

Six items to a safe pilot. Sixteen to production.

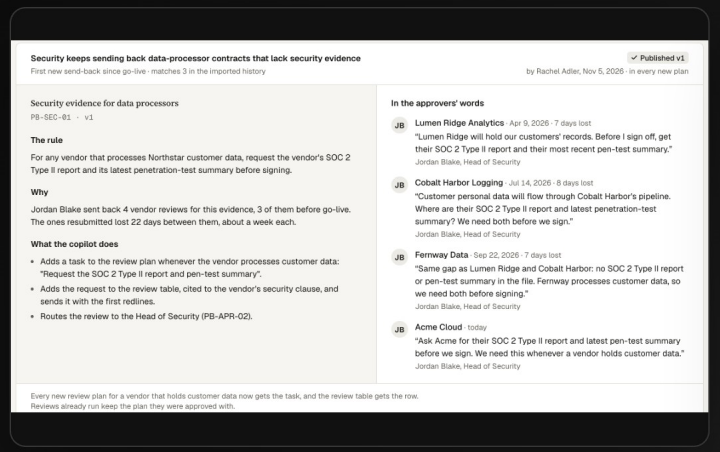


CopilotKit

LEARN · DEMO STEPS 6–7

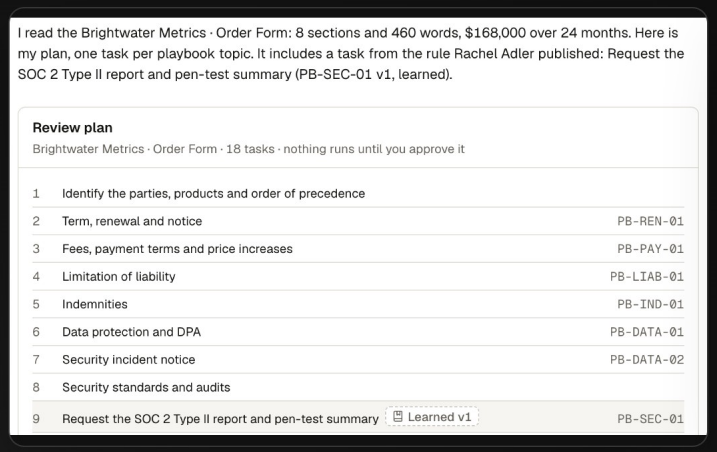
The rule nobody wrote down

06 · Rachel publishes it



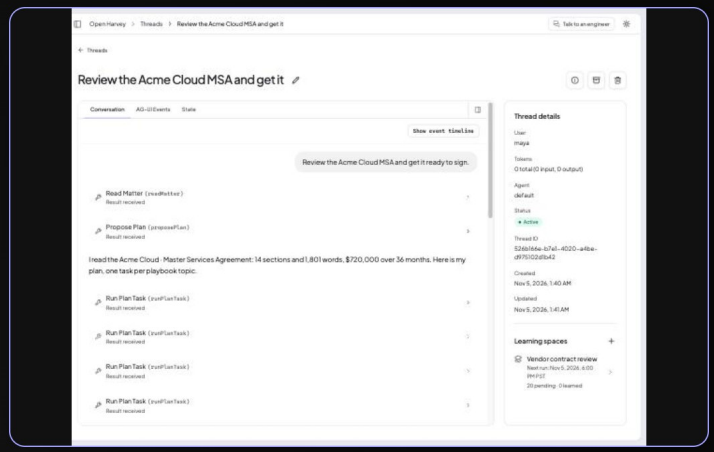
PB-SEC-01, beside four send-backs in Security's own words.

07 · Brightwater, that afternoon



Task 9 came from the rule Rachel published. Nobody had to ask for it.

In CopilotKit Intelligence



Thread stored · 0 learned yet. The first learning run needs 15 threads; PB-SEC-01 here is a prepared check.

Open Harvey: an open-source CopilotKit demo, not affiliated with or endorsed by Harvey.

IN PRODUCTION

It’s live, and your agent isn’t working as well as you want.

How do you improve it?

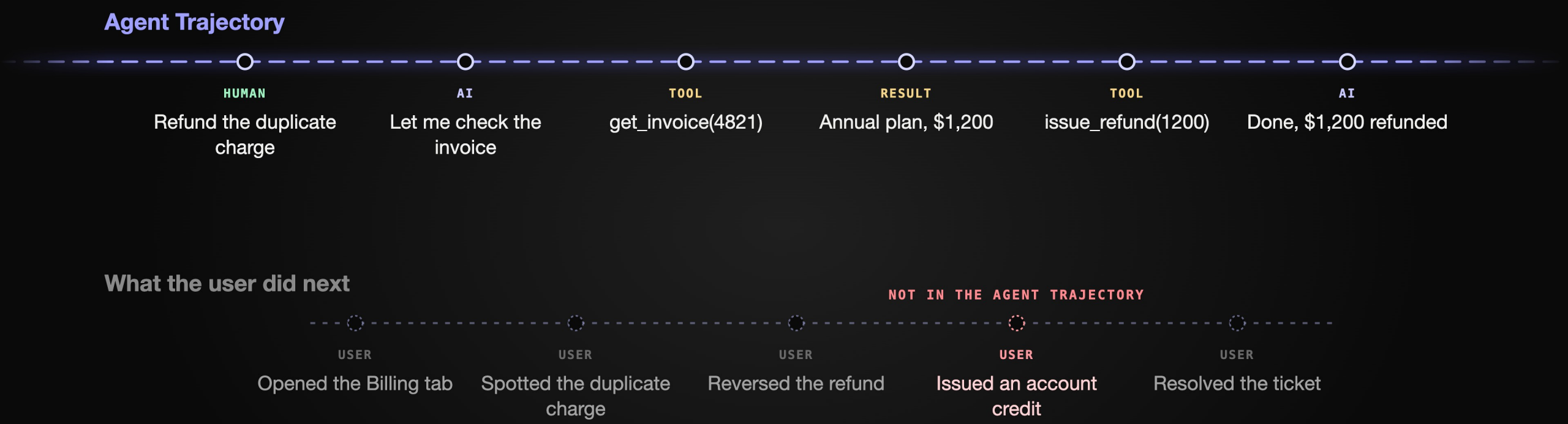
INTRODUCING

CLIF

Continual Learning from Full-Stack Trajectories

LEARN · CLIF

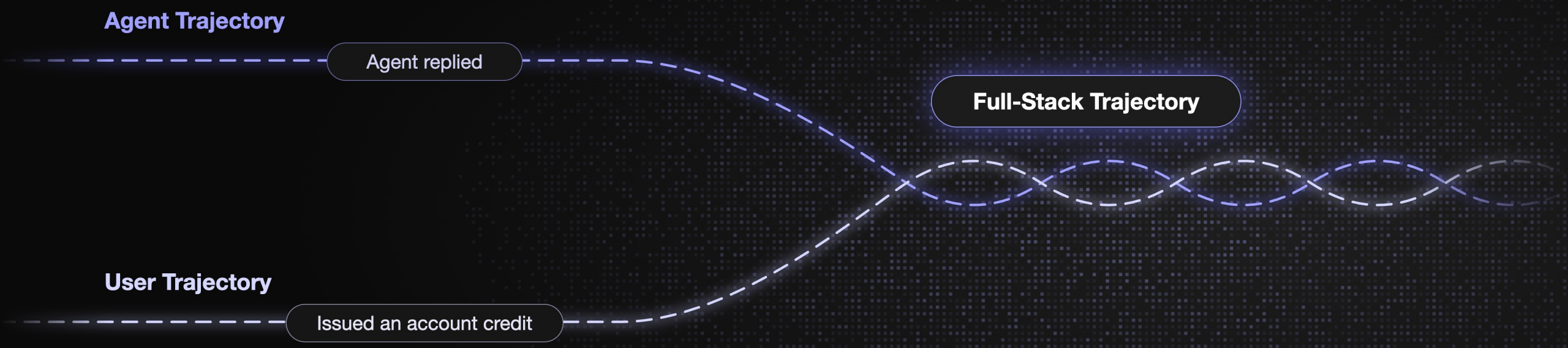
The agent trajectory stops at the chat window



CopilotKit

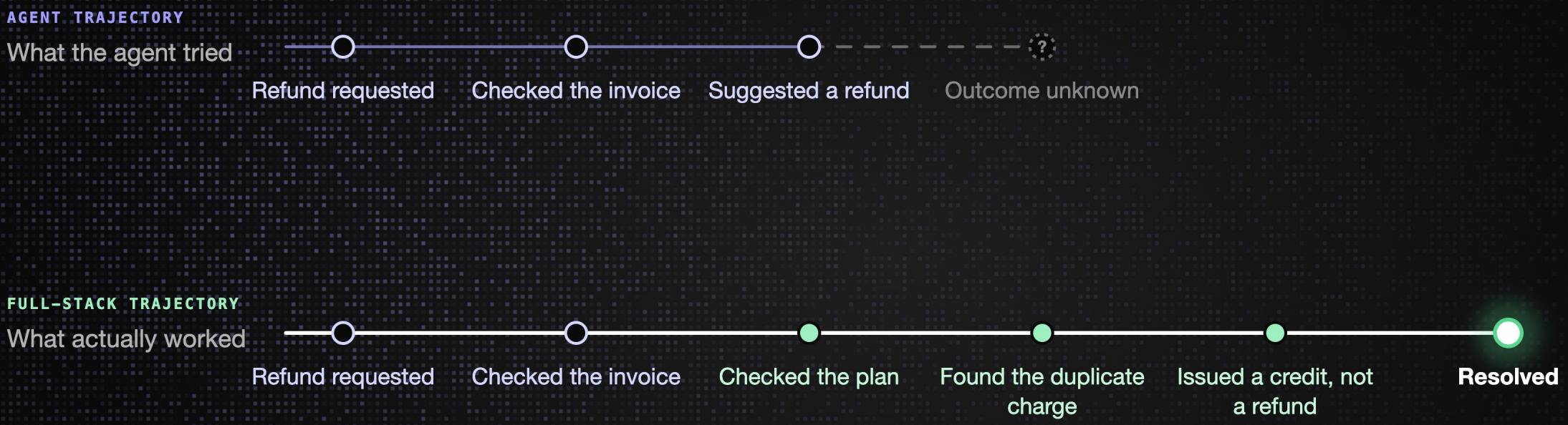
LEARN · CLIF

We add the user's side: the full-stack trajectory



LEARN · CLIF

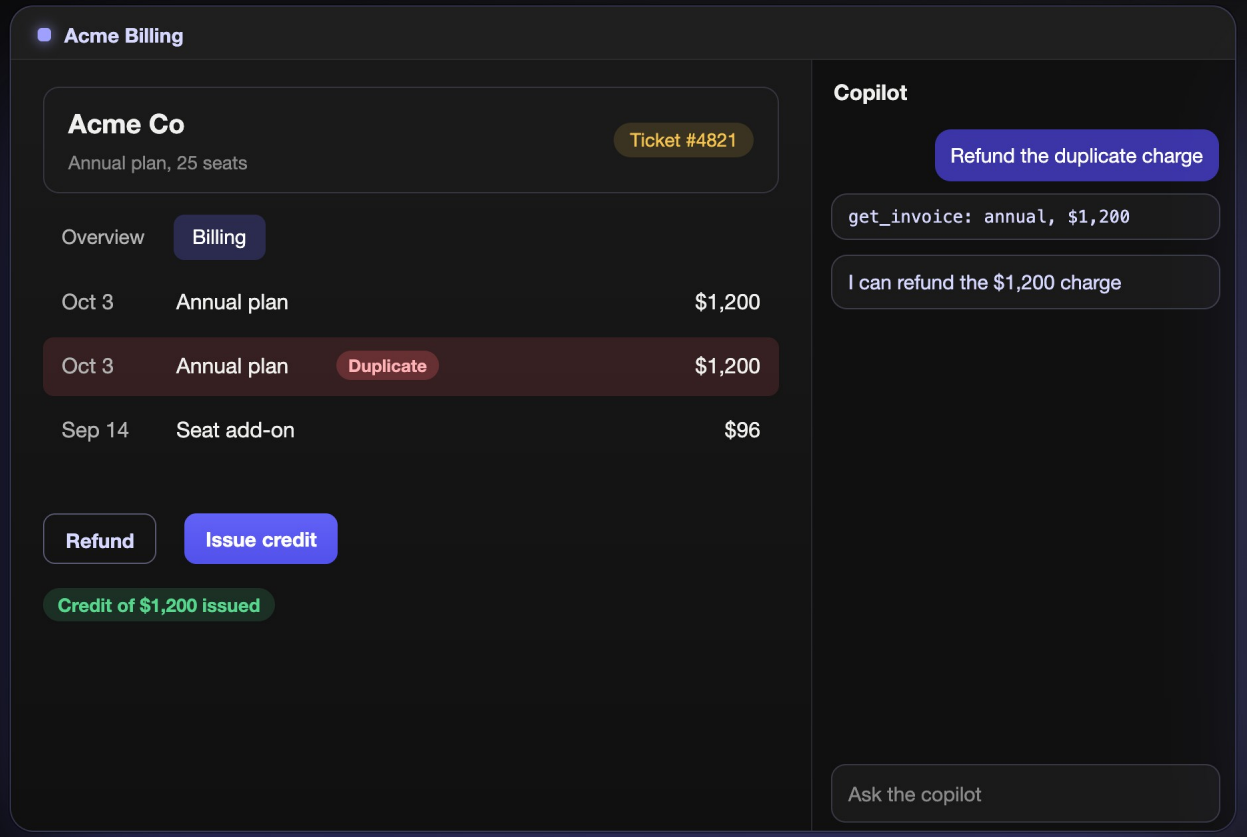
Only the full stack shows what actually worked

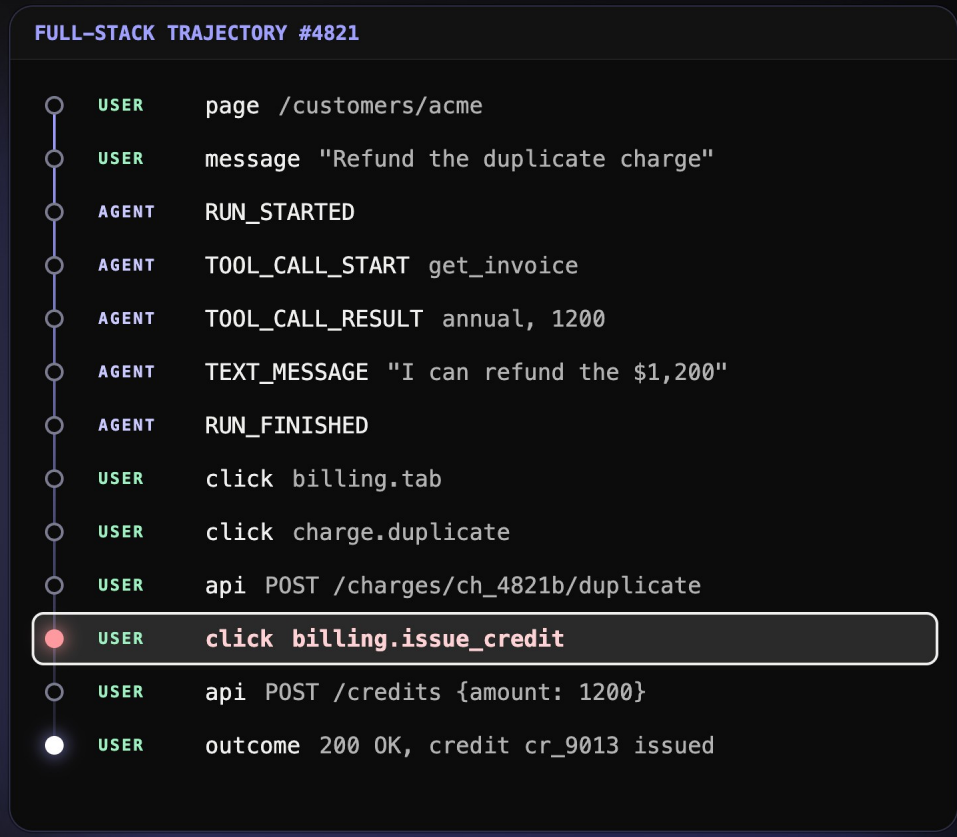


CopilotKit

LEARN · CLIF

One full-stack trajectory, event by event





CopilotKit

LEARN · CLIF

Built on AG-UI, the open agent-user protocol

Google | Microsoft | Amazon | Anthropic | LangChain | Mastra

AG-UI

Web app | Mobile | Slack | Teams

CopilotKit

LEARN · CLIF

Label what's worth learning from

TRAJECTORY #4821 · APPROVED

Duplicate charge on an annual plan



NOTE Annual plans get a credit, not a refund

TRAJECTORY #4907 · EXCLUDED

Agent looped, user gave up



NOTE Agent looped on get_invoice

TRAJECTORY #5013 · APPROVED

Seat change mid-contract



NOTE Prorate first, then confirm

CopilotKit

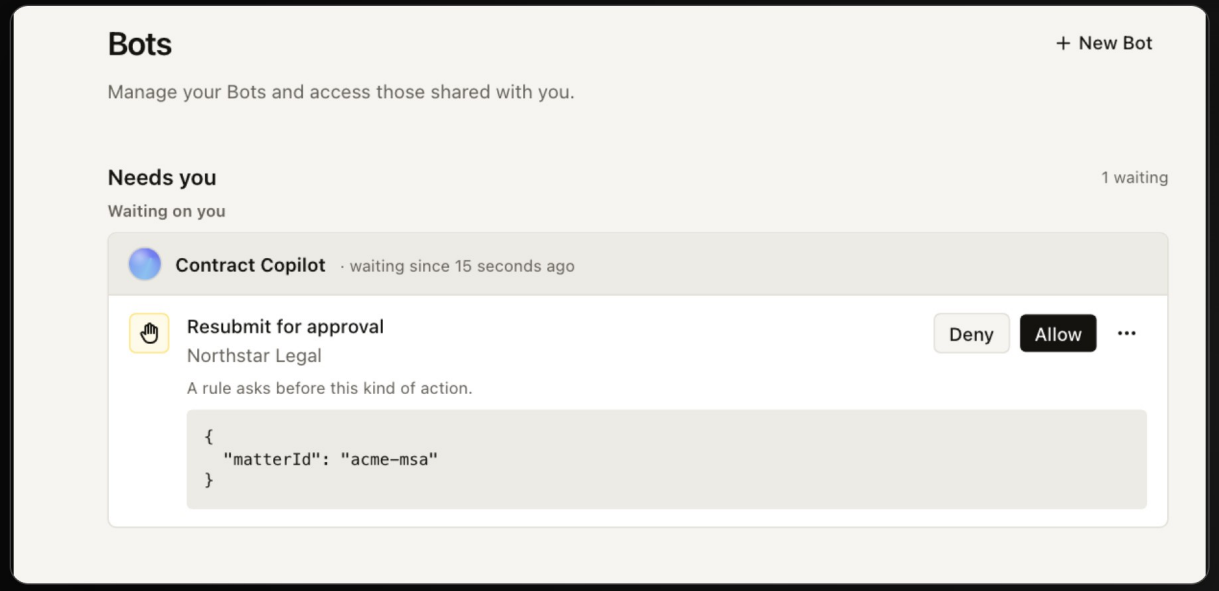
LEARN · CLIF

Then we put full-stack trajectories to work



PRODUCTION GRADE

OpenBot built for Enterprise



Evidence arrives. It checks, then asks before it resubmits.

WHAT OPENBOT SHIPS

Sign-in	SSO: Google, Microsoft, Okta, SAML, OIDC
Roles	Admins and members, sharing by group
Guardrails	One gateway, fails closed
Approvals	Ask-first rules
Audit	Every action, with its rule
Logs	To your SIEM over OpenTelemetry
Secrets	Encrypted, never in the transcript
Hosting	Your infra and database

Open source, in beta: a template you host.

START TONIGHT

Everything you saw today, ready to clone

`/functional-copilot` one entry point, five steps

1	Research	`/user-map`
2	Context and access	`/agent-context`
3	Spec and build	`in the root`
4	Deploy	`/deploy`
5	Secure and govern	`/secure-governance`

Copy skills/ into .claude/skills/, then run /functional-copilot.

`github.com/jerelvelarde/functional-copilots`

</> Open Harvey

Today's demo, as a web app

`starters/open-harvey`

Open Harvey on OpenBot

The same copilot as a coworker

`starters/openbot-legal`

Start with one workflow that happens every week and ends in a person's approval.

IF YOU REMEMBER ONE THING

Building a functional copilot means building the environment where your users and agents work the task together.

Build it with CopilotKit, and you learn from both: your users' actions and your agent's traces.

github.com/jerelvelarde/functional-copilots

Open Harvey: an open-source CopilotKit demo, not affiliated with or endorsed by Harvey.

LIVE Q&A · 10 MINUTES

Questions

Ask in the chat. Nathan will read them out.

Repo: github.com/jerelvelarde/functional-copilots



Teach your agents how work gets done.

go.copilotkit.ai/get-clif

CopilotKit