

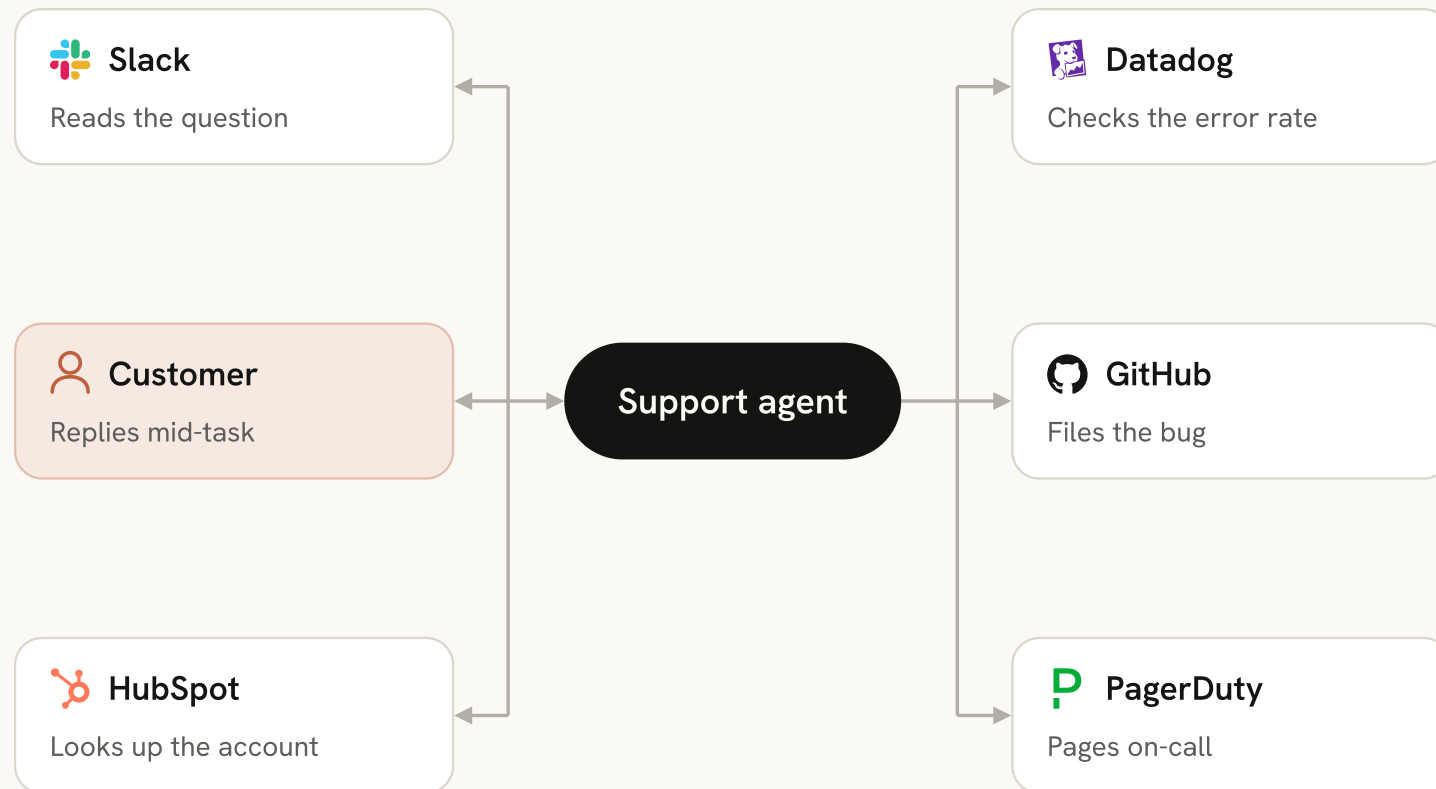
Production, simulated.

Stateful testing playgrounds for AI agents.

We give an agent a working copy of the world it will act in: the tools, the data inside them and the people it talks to. Teams see how it behaves before it ships, and again on every change.



Agents now take real actions inside their customers' software.



Copying this world is a project of its own.

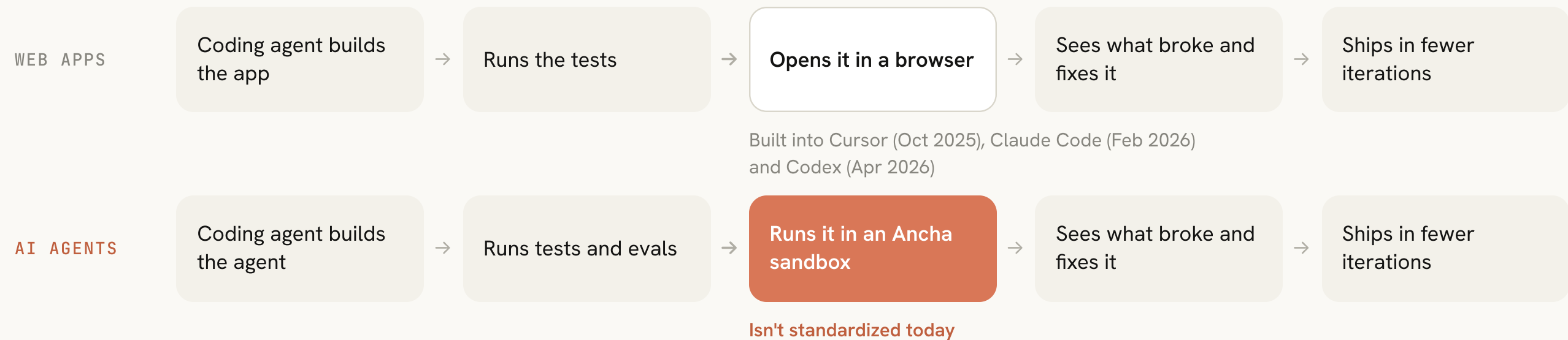
- 🔒 Vendor sandboxes are empty, capped or Enterprise-only
- 🗄️ Every tool, doc and vector DB needs realistic data, mess included
- 🔗 The tools must work together and reset after every run
- 👤 Customers reply in ways nobody scripted

So teams end up testing on clean mock data or on their customers.

**Agent engineering teams
can't see how their agent
will act in a real customer's
environment until it's live.**

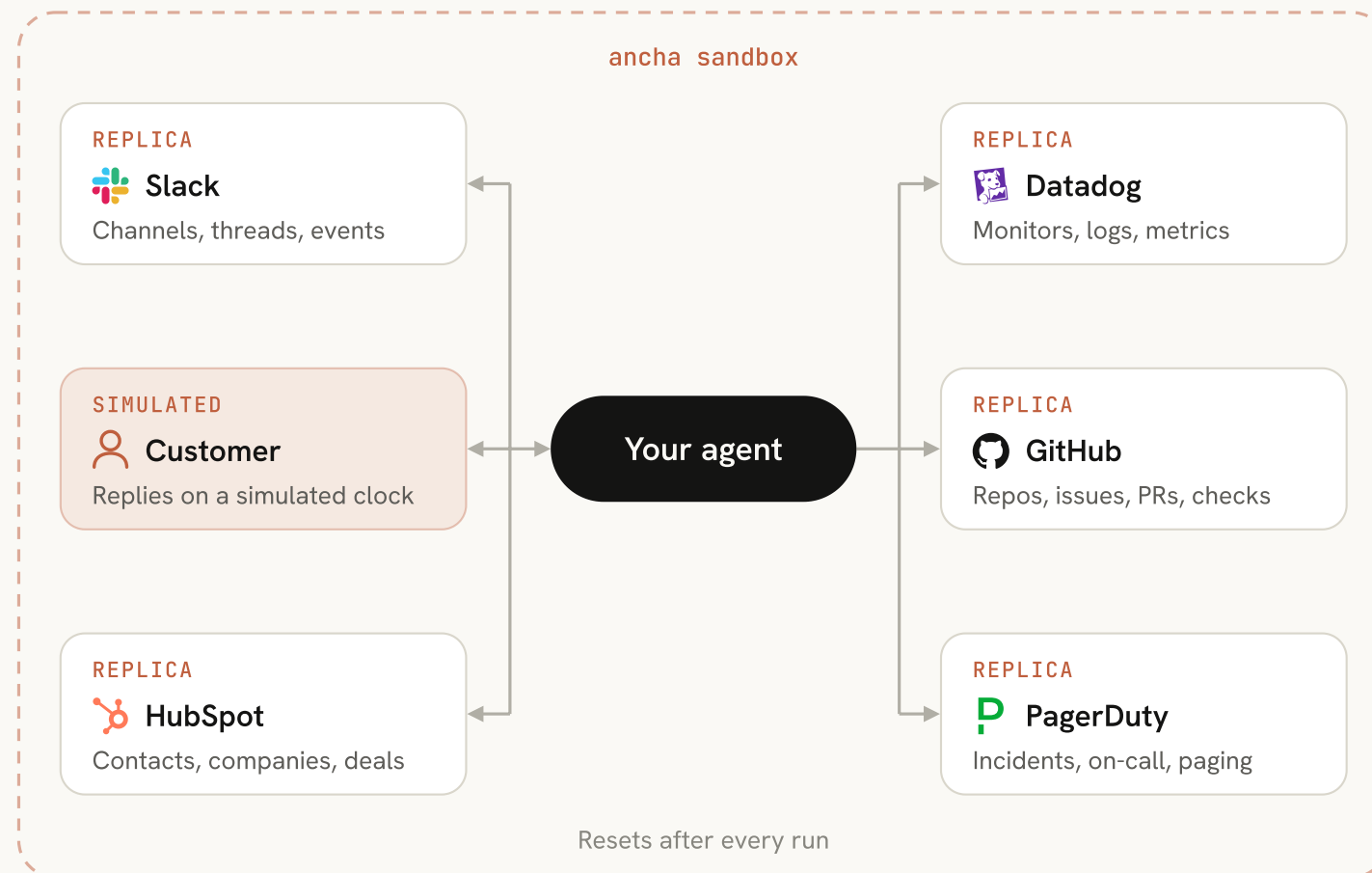
Evals grade what the agent says. Mocks return clean, made-up data. Neither shows what it does with a real customer's tools, data and people.

Coding agents learned to test web apps in a browser. Agents need a sandbox.



Coding agents already run tests. Opening the app in a browser is what cut the iterations. A sandbox will do the same for agents.

Ancha spins up a production-like copy of your agent's world.



Tools

A stateful twin of every tool, wired together over one state, with the same endpoints and credential shapes as the real ones.

Data

Seeded records in every tool from versioned starter datasets. Production-scale synthetic data is next.

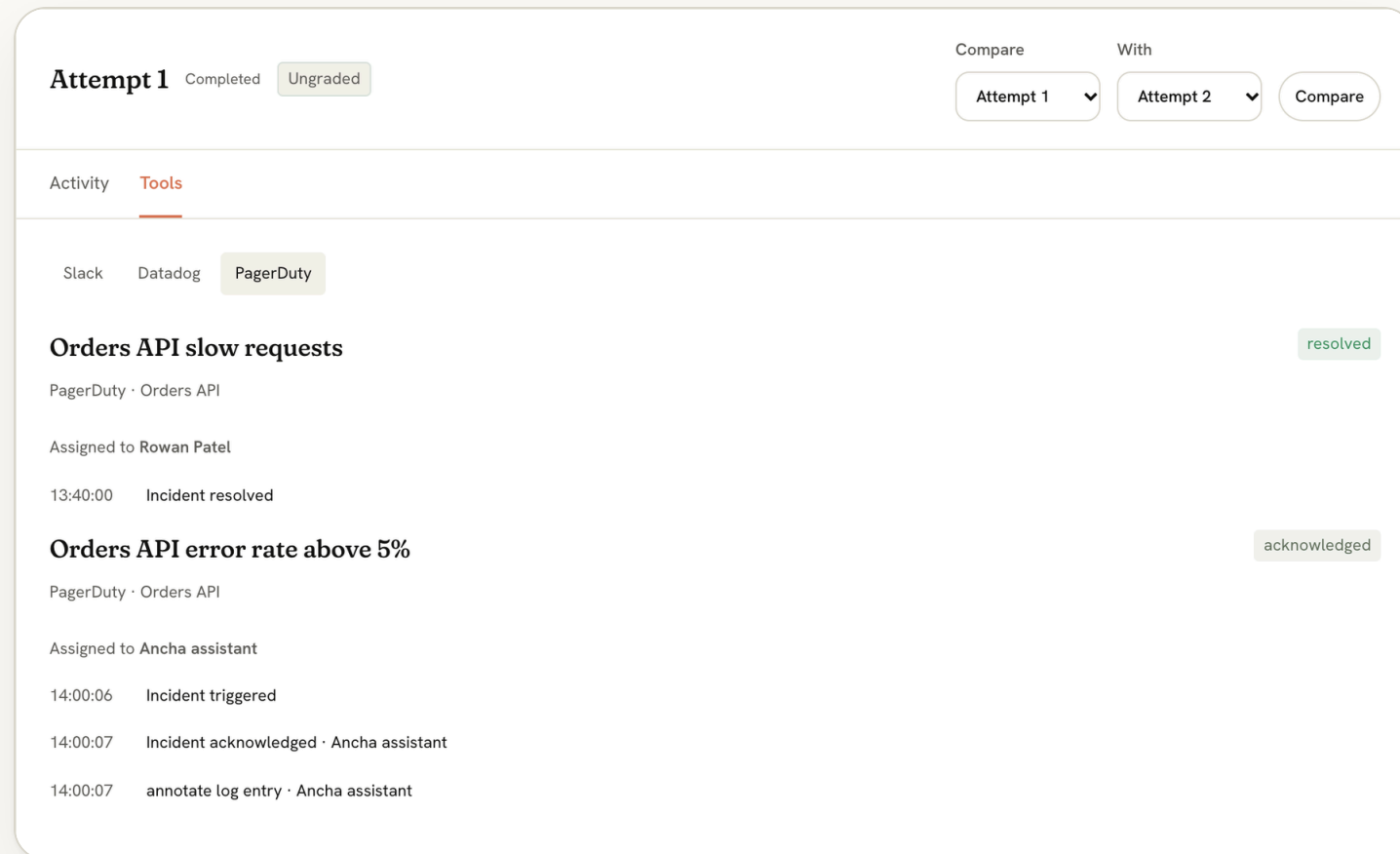
People

Teammates and customers who post and reply on a simulated clock. Off-script replies are next.

CLI-first

A coding agent creates, forks, inspects and resets worlds from the terminal. Nobody needs the UI.

Live today: run an agent, read back every tool.



Attempt 1 of the demo on-call test: the incident the agent raised and acknowledged, read back from the PagerDuty twin. Next: checks and pass rates on the end state, then calibration on production traces.

app.ancha.io

Hosted workspaces, one private machine per account. Sign in with email.

@ancha-labs/cli on npm

Coding agents create, fork and inspect worlds from the terminal; the same key works in CI.

11 tool twins

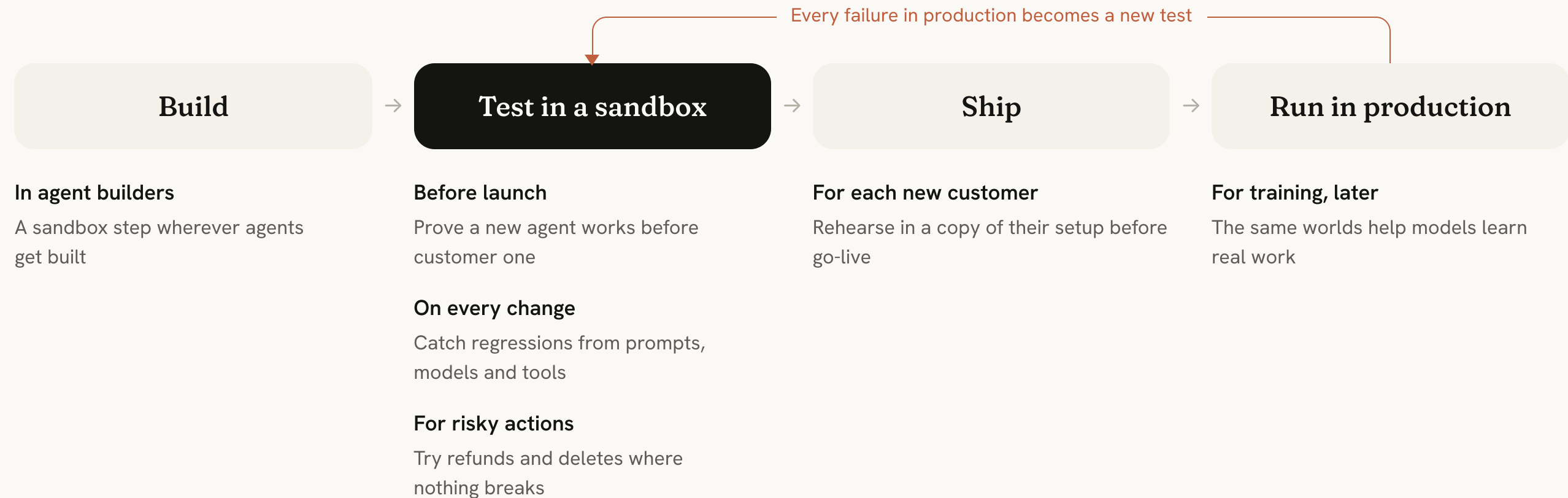
Slack, GitHub, Datadog and PagerDuty verified; seven more in closed alpha.

Isolated playground runs

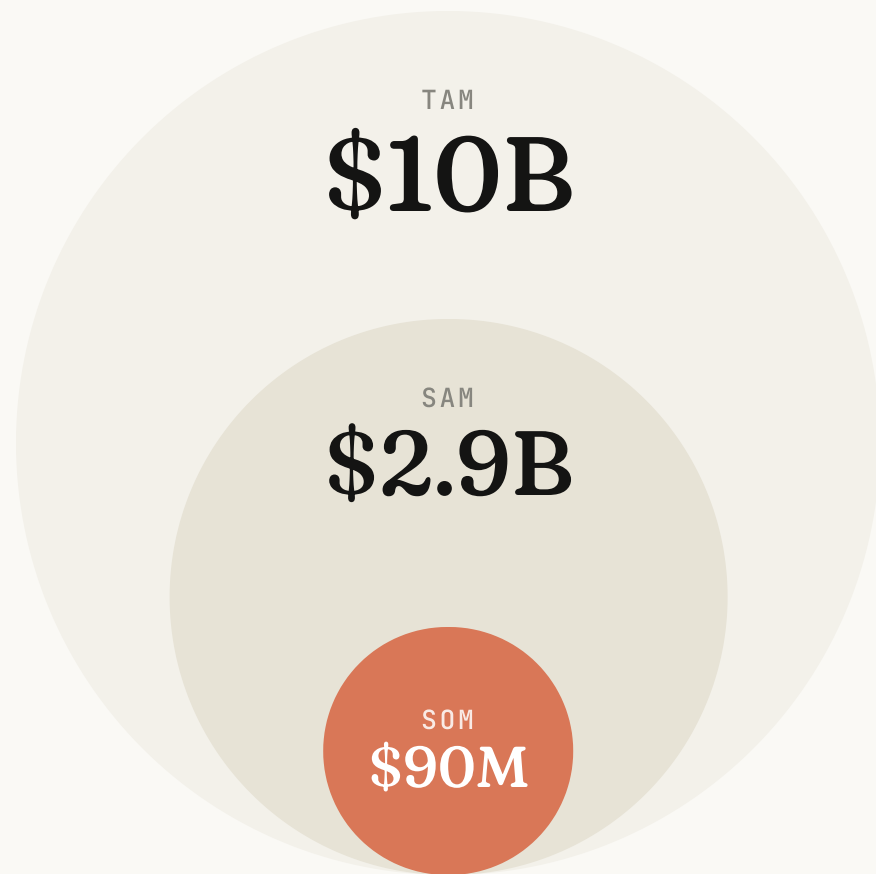
Your agent in its own container, N attempts, every call and end state kept.

Sandboxing becomes a core step in how every agent ships.

We believe it is a necessary part of building agents, the way nobody ships code without CI.



A \$10B-a-year market by 2030.



TAM • \$10B A YEAR

Testing every AI agent. What teams could spend on it each year by 2030.

SAM • \$2.9B A YEAR

Business agents. Agents that work inside business software, built by enterprises and agent startups.

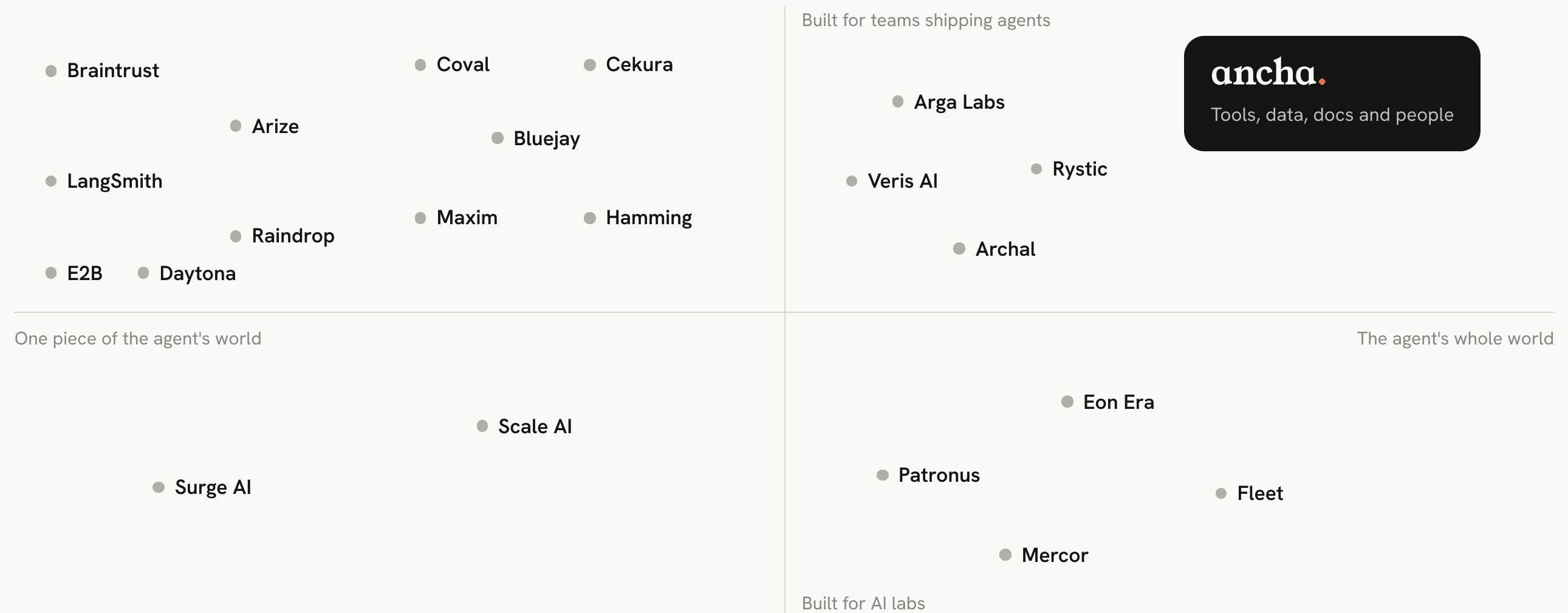
SOM • \$90M A YEAR

Our share. What we can realistically win by 2030.

Not counted: AI labs, who buy the same kind of worlds to train models.

TAM: 20% of a \$52.6B AI agent market in 2030 (MarketsandMarkets); testing took 26–35% of IT budgets in 2015–17 (World Quality Report). SAM: 20,000 enterprises at \$125K and 10,000 agent startups at \$40K a year. SOM: 400 of those enterprises and 1,000 of those startups.

Every other layer of agent testing is already funded and growing.



We win on realism, proof and coverage.

REALISM

The whole world, not one piece

Agents act on tools, search docs and talk to people in the same task. We copy all of it, together.

PROOF

Calibrated on production

Production traces tune the replicas and become new scenarios. What we sell is how well a sandbox pass predicts production.

COVERAGE

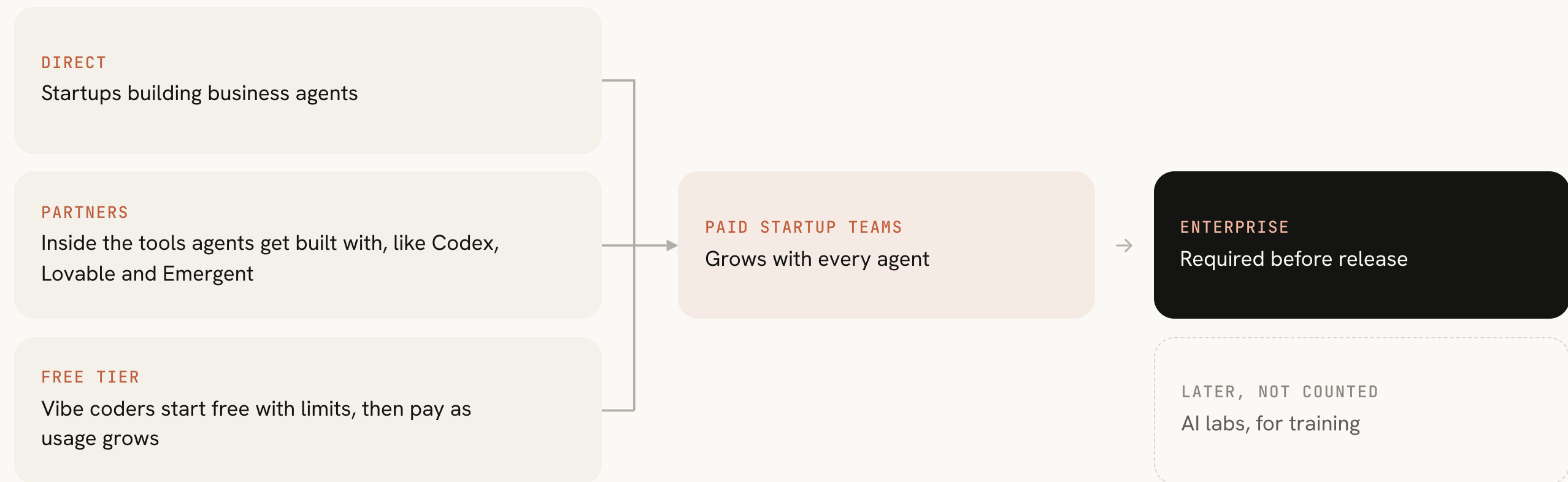
Every customer's setup, on every change

One agent ships into many companies, each set up differently. Every change is tested against a copy shaped like each one.

This is the self-evolving part: the sandbox learns from production and gets closer to the real world with every run.



Land through startups and builders, then grow with usage.



A free tier with limits is how Vercel and Composio built awareness and usage. We follow the same playbook.

A live product, and early signals from agent teams.

Shipped

- ✓ app.ancha.io · hosted workspaces, one private machine per account
- ✓ @ancha-labs/cli on npm · sandboxes and secrets from the terminal
- ✓ 11 tool twins · 4 verified, 7 in closed alpha
- ✓ Isolated playground runs · two containers per attempt, every call and end state kept
- ✓ Demo workspace · 5 agents and 6 tests across support, on-call, CRM and release notes

Signals



agent teams
interviewed



design partners



teams on the waitlist

“What an agent builder told us about testing their agent today”

Name, role, company

Jayanth and Patrick are building Ancha.



Jayanth Putta

Co-founder & CEO

Built IBM's first LLM and agent observability product from the ground up, now in production at enterprises worldwide.

Built Kubernetes observability from 0 to 1 at OpsRamp, about \$2M in new revenue; OpsRamp was acquired by HPE. Before that, product and solutions engineering at Datadog and firewall VPN engineering at Cisco.

Founded TourByte, an AI agent that answered renters and booked tours for NYC brokers. Open-sourced LoopEval, an eval framework for agents.



HPE OpsRamp



Datadog



NIT Karnataka



Patrick Liu

Co-founder & CTO

Founded SimpleFunctions: agents that researched events and traded live on Polymarket and Kalshi.

Built SparkCo Harness, an open-source CLI where a local and a Cloudflare agent runtime negotiate, share state and change their own environments. Shipped a phone agent that ran live at a skilled nursing facility.

MA in Physics, Brown. BA in Physics and Math at UW-Madison in two years. Hand-wrote a neural network to classify particles in CERN collider data.



SimpleFunctions

BROWN

UW-Madison

A pre-seed round to take Ancha from design partners to paying teams.

RAISING



Pre-seed, months of runway

WHERE IT GOES

%

Engineering

Tool replicas, the synthetic data engine and evals

%

Go to market

Design partners and builder integrations

%

Infrastructure

Compute for sandboxes and test runs

Every agent should be tested in a copy of its world before it ships.