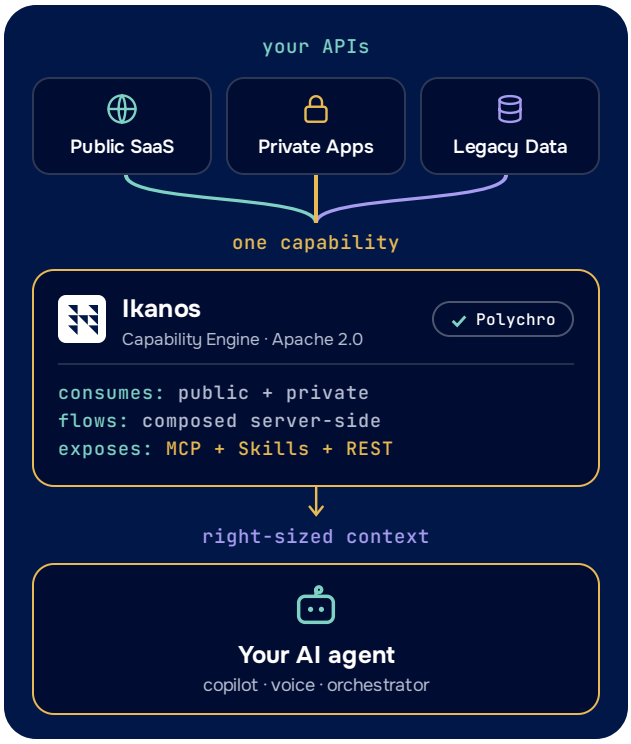
Naftiko

The Agentic Integration Platform

Declare your Agent Capability

You already have the APIs, the gateway, and the identity provider. Naftiko turns what you have into task-shaped capabilities your agents can call, declared in one YAML file and served as **MCP, Skills, and REST** from the same spec. Your OpenAPI or Smithy contracts, catalog, and policy stay authoritative. Naftiko composes on top of them instead of becoming another source of truth.



Approved before it runs, not audited after

Pilots stall at the production gate because nobody can say what the agent is able to reach. The whole capability is in a file security reads first: the operations reachable, the fields returned, and the credential scope.

Sound familiar?

- **Your pilot can't get production access.** It passes in a sandbox and stops at security review.
- **Every team is writing its own MCP server.** You need one reusable pattern before the sprawl becomes debt.
- **Agents are becoming a customer channel.** Your product has to expose real workflows, not raw endpoints.

One Spec, Three Surfaces

Author a capability once. The same YAML serves MCP, Skills, and REST. No hand-written servers and no integration glue.

Right-Sized Context

A task-shaped capability replaces a 200-tool catalog. Agents see the tools and fields they need and nothing else.

Compose Across Systems

A voice agent resolving an order touches CRM, commerce, and identity. One capability does all three in a single call, in parallel.

Your Contracts Stay Authoritative

Not a second source of truth. Naftiko links to your OpenAPI or Smithy contracts, catalog, gateway, and identity rather than replacing them.

Two ways to start

- **Try it out.** Run a capability in the Playground from your browser, with nothing to install. The engine and linter are on GitHub under Apache 2.0.
- **Talk to us about a real use case.** Book a 30-minute capability design session and we'll shape the capability with you.

Scan with your phone

Get started →

Naftiko

APIs weren't designed for agents

Give agents capabilities, not a dump of your API

Generating a 1:1 mirror of your APIs can leave an agent with hundreds of tools and no understanding of the workflows they support. Naftiko exposes a small set of task-shaped capabilities while your APIs, auth, and policies stay authoritative.

For SaaS platform teams: Turn your existing API into an agent-facing product interface, without exposing the entire API surface.

The usual approach	With Naftiko
Hand-write the integration glue: auth, pagination, retries	**Declare a capability in YAML**
Hand-write an MCP server per framework	**Emit MCP, Skills, and REST from one spec**
Generate a 1:1 mirror of 200 endpoints	**Expose task-shaped capabilities**
One generic server per provider, bolted together	**Compose capabilities across systems**
Rewrite when the MCP transport revises	**Bump the engine, the spec is unchanged**

```
capability:
  consumes:                  # APIs you already have
  - import: orders
  - import: crm
  aggregates:                # compose the workflow
  - namespace: support
    flows:
      order-status:
        steps:
          order: { call: orders.get-order }
          customer: { call: crm.get-customer }
  exposes:                   # what agents see
  - type: mcp
    tools: { order-status: { ref: support.order-status } }
```

The whole thing is reviewable

`consumes` Points to the APIs you already have. They stay authoritative.

`aggregates` Composes operations across systems into a task-shaped workflow.

`exposes` Defines the capability the agent sees, not the underlying APIs.

Open source at the core

Ikanos and Polychro are Apache 2.0 and developed in the open. Built on OpenAPI, AsyncAPI, JSON Schema, and MCP. Self-host with Docker or Kubernetes, or use the hosted Playground.

Apache 2.0 · OpenAPI · MCP · Self-host or hosted · No lock-in

Scan to explore