AZOTH OS

The runtime layer for production AI.

Governance, observability, and cost control for the AI systems that are already running in production — not the ones in the demo.

AI shipped. Now it has to *run*.

Every serious company now has AI in production — answering customers, valuing assets, moving money.

The model was the easy part. Operating it reliably, every day, at cost, is the part nobody solved.

MODELS PER APP, TODAY

3–7 providers

Apps fan out across OpenAI, Anthropic, open weights and fine-tunes — each with its own price, latency and failure mode.

OF AI PROJECTS THAT STALL

70%

%

Most never clear the gap between a working prototype and a system the business can run, audit and afford.

afford.

TAKEAWAY

The bottleneck has moved from **building** AI to **operating** it. That operational layer is missing — and it is where the durable infrastructure value sits.

AI applications are hard to operate at scale.

A prototype calls one model in a notebook. A business runs thousands of thousands of calls a minute across providers, versions and tenants — and tenants — and every one of them can fail, drift or overspend silently. silently.

01

Calls fail in ways nothing catches

Timeouts, truncations and malformed output surface as a bad customer experience, not an alert. not an alert.

02

Quality drifts between model versions

A provider ships an update and output silently degrades — with no baseline to measure against. measure against.

03

Every team rebuilds the same plumbing

Retries, fallbacks, routing and logging get hand-rolled per app, then rot.

TAKEAWAY

Operating AI is an **infrastructure problem**, not a modelling one — and infrastructure problems are won by a platform, not a feature.

Production AI is missing three layers it can't ship without.

GOVERNANCE

Who can call what, and on whose authority?

No policy, no approvals, no audit trail. In regulated work that is a blocker, blocker, not a backlog item.

OBSERVABILITY

What actually happened on that call?

Tokens, latency, cost and quality per request — traced end to end, not end, not guessed from a bill.

COST CONTROL

Are we paying for the right model?

Spend scales linearly with usage and lands as one opaque invoice nobody invoice nobody can attribute.

TAKEAWAY

These aren't three products. They're three faces of one missing system: a **runtime** that sits between the app and the models.

WHAT THE MISSING RUNTIME COSTS, PER APP, PER YEAR

30–

OF MODEL SPEND IS AVOIDABLE

Premium models answer requests a cheaper one would have handled identically.

0 traces

INTO MOST PRODUCTION CALLS

When a call goes wrong, teams reconstruct it from logs — if they kept any.

weeks

TO PASS A SINGLE AUDIT

Without a policy and audit layer, every compliance review starts from scratch.

TAKEAWAY

The gap is already being paid for — in wasted spend, blind incidents and stalled deals. **Azoth OS turns that recurring loss into the budget for the runtime.**

RUNTIME OPTIMIZATION LAYER

One runtime that observes every call and routes every request.

Azoth OS sits between your application and every model it uses. It watches each LLM call across the stack, routes it to the optimal model, enforces policy, and settles the bill — keeping latency, quality and cost in equilibrium.

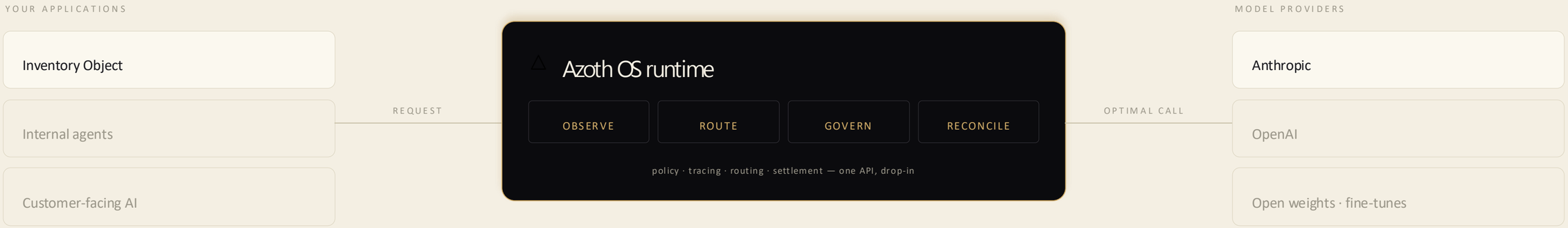
OBSERVE
Every call traced — tokens, latency, cost, quality. quality.

ROUTE
Each request sent to the optimal model, automatically. automatically.

GOVERN
Policy, approvals and a full audit trail by tenant. tenant.

RECONCILE
Spend attributed and settled, request by request.

A thin layer in the request path — not another another framework to adopt.



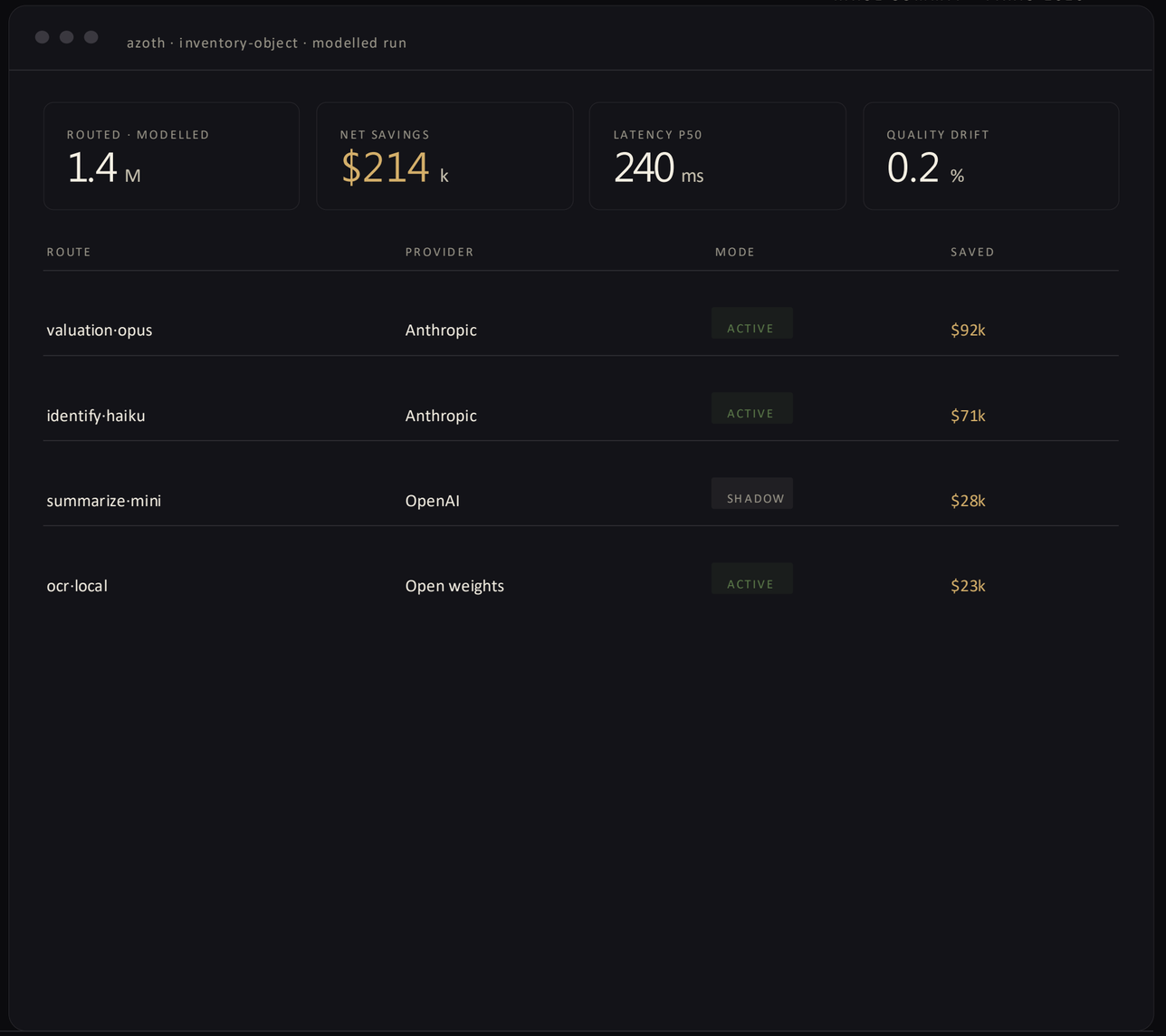
TAKEAWAY

Integration is a single endpoint swap. Azoth OS earns its place in the request path on day one — and compounds with every call it sees.

Every call, visible. Every route, under control.

The Azoth OS console gives operators one place to watch live traces, manage traces, manage routes, replay runs and reconcile spend — across providers providers and tenants.

ROUTES	auto-balance cost, latency & quality per request
LIVE TRACES	inspect any call end to end as it happens
RUNS	replay and compare model versions before rollout
RECONCILIATION	spend attributed by tenant, route and model

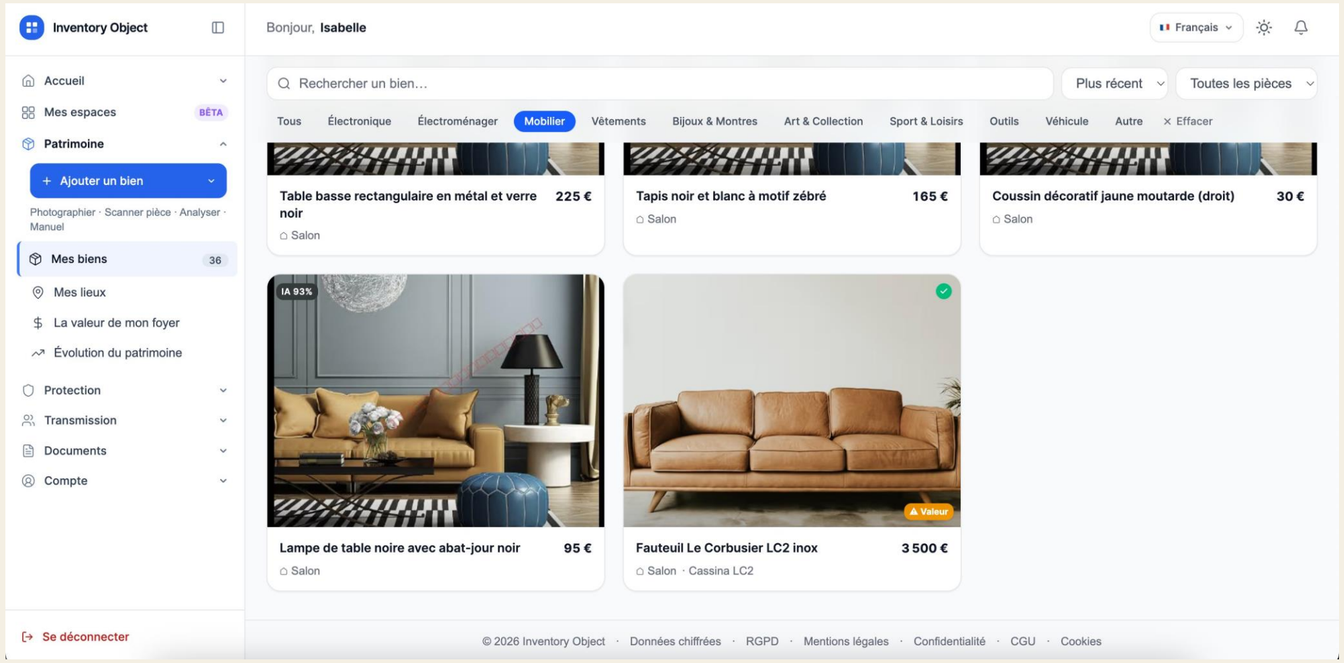


FLAGSHIP WORKLOAD · BUILT ON AZOTH OS

Inventory Object is built on Azoth OS.

A French *coffre patrimonial numérique* — households photograph their belongings, Claude identifies and values each piece, and the notarial-grade dossiers for insurance and succession.

It is a real, regulated AI workload — high call volume, strict audit needs, unforgiving cost math. Exactly the system Azoth OS is built to run.



One inventory scan is dozens of AI calls. Azoth OS is built to run every one.

Each photographed object fans out into a pipeline of model calls — and Azoth is designed to route, trace and price each step without Inventory Object writing a line of model-ops code.

01

Identify

Recognise the object from the photo.

route · identify·haiku

▸

02

Value

Estimate market value with reasoning.

route · valuation·opus

▸

03

Verify

Flag inconsistencies & possible fraud.

route · verify·mini

▸

04

Dossier

Compose the notarial PDF record.

route · summarize·local

TAKEAWAY

Azoth sends the cheap step to a cheap model and the hard step to a strong one — **same output, a fraction of the cost**, fully audited.

INVENTORY OBJECT WORKLOAD · MODELLED · REPRODUCIBLE VIA npm run benchmark

1.4
CALLS SIMULATED

Across four routes and three providers, auto-balanced for cost.

$21
NET SPEND SAVED

A modelled 41% reduction versus running every call on the premium model.

240m
LATENCY P50

Modelled p50, held within target as routing shifts across providers.

0.2%
QUALITY DRIFT

Modelled output quality, flat against the tracked baseline.

4k

TAKEAWAY

Modelled on Inventory Object's real regulated workload, Azoth's routing cuts model spend **41%** with no modelled loss of quality or latency — deterministic, reproducible from the repo, not slideware.

Every production AI system needs a runtime. None of them have one yet.

Model spend is compounding while the layer that governs and optimises it barely it barely exists. Azoth OS sells into that gap — and the gap grows with every model a every model a company adopts.

FIGURES ARE DIRECTIONAL ESTIMATES

TAM
AI infrastructure & ops software
$150B

SAM
Runtime, observability & cost layer for production AI
$25B

WEDGE
Multi-model, regulated AI apps in the EU
$2B

By 2030 · bottom-up from per-call routed spend across providers.

Others observe, or route, or live inside one cloud. None do all three, neutrally.

OBSERVABILITY TOOLS

Datadog, LangSmith, Helicone

They show you what happened after the spend. Azoth sits in the request request path and acts — routing and enforcing budget before the call is the call is made. Observe vs control.

ROUTERS & GATEWAYS

OpenRouter, Portkey

Routing is the commodity. Azoth wraps it with policy, audit, risk-gated gated caching and per-tenant reconciliation — the governance a regulated regulated workload can't ship without.

HYPERSCALER NATIVE

Bedrock, Azure AI Foundry

They optimise to keep traffic inside their own cloud — a structural structural conflict. Azoth is model- and cloud-neutral, and can route out of route out of a provider for cost or sovereignty.

TAKEAWAY

The defensible position is the neutral runtime in the path: observe, route **and govern** across every model, owned by no provider.

We earn a share of the spend we save.

Usage-based pricing that scales with value delivered — adoption is a single endpoint swap, so it lands swap, so it lands bottom-up and expands with volume.

PLATFORM FEE	a few % of routed model spend
PER CALL	metered tracing & routing, no seats
ENTERPRISE	governance, SSO & audit, annual

The moat is the runtime position.

Sitting in the request path, Azoth OS sees what nobody else does — and that data compounds into routing no competitor can match.

01 **Data flywheel.** Every routed call sharpens cost-quality routing for the next one.

02 **Switching cost.** Once governance, audit and spend run through Azoth, it becomes load-bearing.

03 **Model-neutral.** We win as the layer above the model market — not a bet on any one provider.

Builders who have run AI in production — and and felt this gap first-hand.

MS

Malaika Samuel

CHIEF EXECUTIVE OFFICER · CHIEF TECHNOLOGY OFFICER

Sets the runtime architecture and the company direction — bridging deep systems systems engineering with the commercial story investors and enterprises need to hear. need to hear.

SB

Soraya Bengrine

CHIEF PRODUCT OFFICER

Owns the operator experience — turning traces, routes and reconciliation into a console teams trust to run regulated AI every day.

TAKEAWAY

A focused founding team spanning **runtime engineering and product** — and already building a regulated AI workload, Inventory Object, on Azoth OS.

RAISING · SEED · €1.5M–2.5M

Become the layer every production AI app runs on.

Inventory Object proved the value in a real, regulated workload. This round takes Azoth OS from invite-only alpha to the default runtime for multi-model AI — starting in Europe.

USE OF FUNDS

Runtime & routing engineering	50%
Governance & compliance	% 25%
Design partners & GTM	% 25%

18-MONTH GOAL — 3–5 production design partners