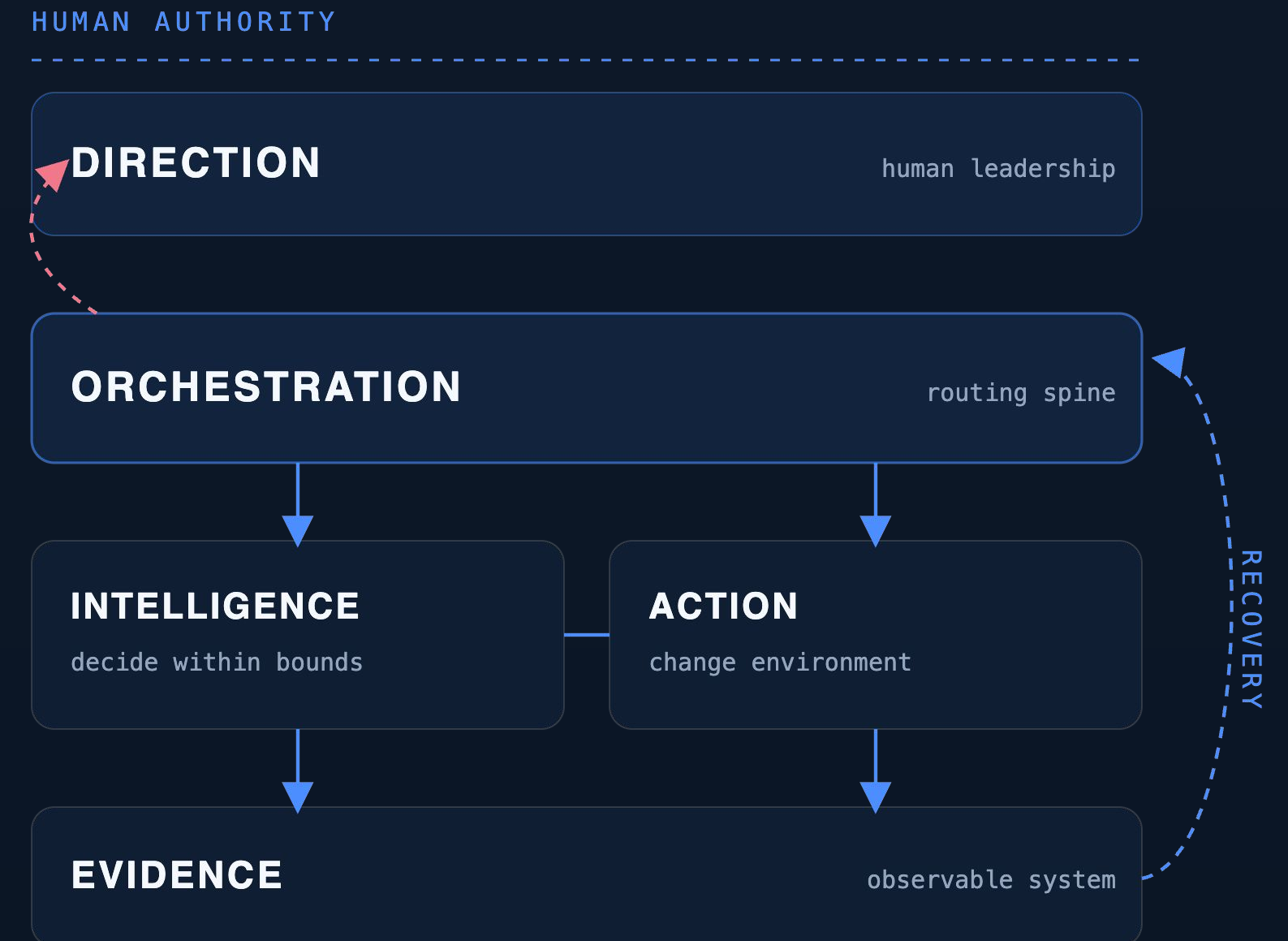


Agentic Operations Architecture

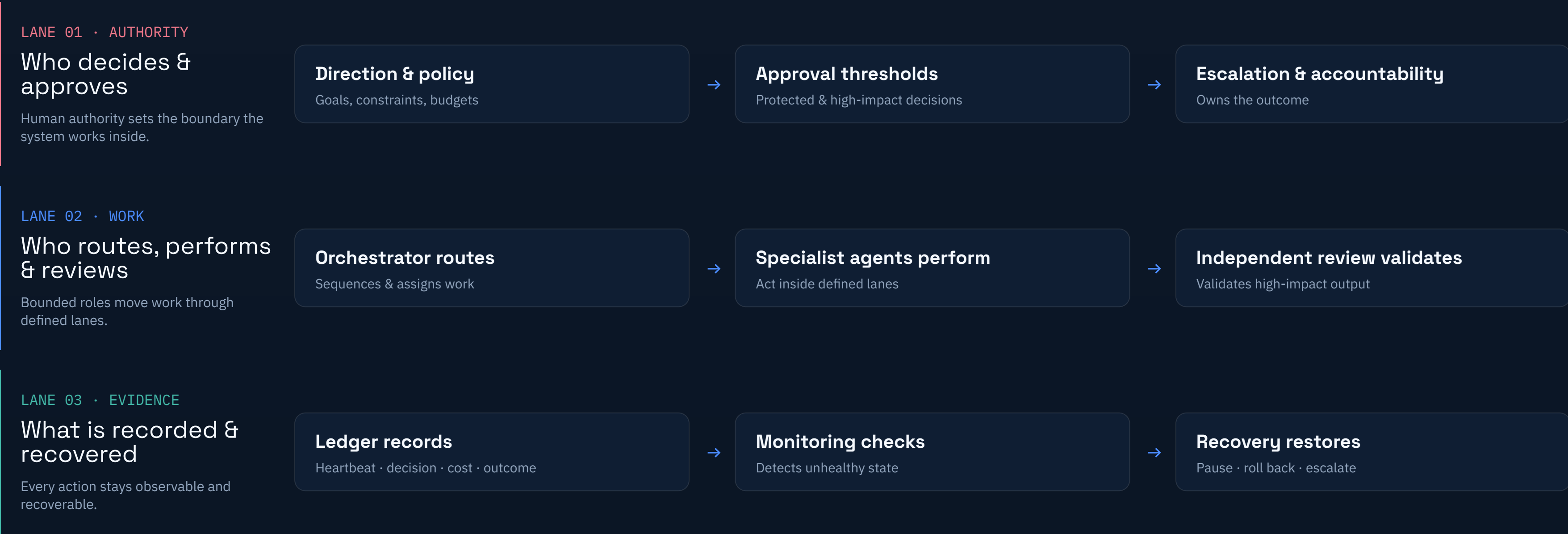
Build autonomy that stays accountable.

A reference architecture for coordinating authority, work, intelligence, action, evidence, and recovery across multi-agent systems.



The whole system reads in three lanes.

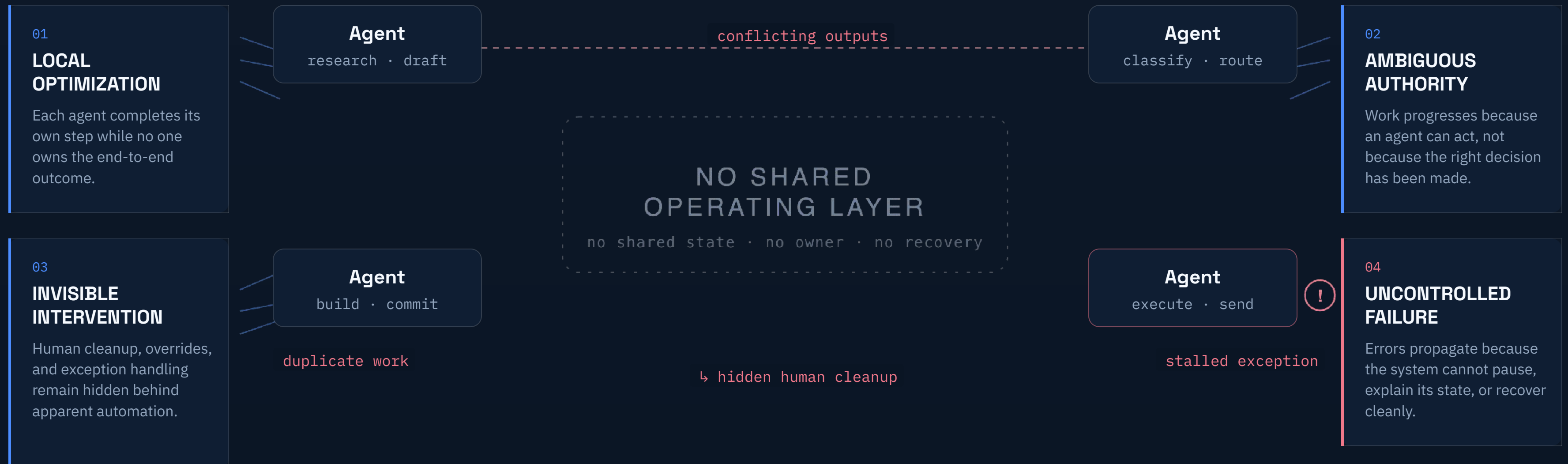
Authority controls the boundaries.
Work moves inside them.
Evidence shows what happened.
Recovery closes the loop.



RUNNING EXAMPLE Customer onboarding exception — the owner approves, agents prepare the follow-up, the ledger proves what happened, and recovery closes the loop if it stalls.

More agents do not create an operating system.

Agents can research, draft, classify, route, build, and execute. The system becomes fragile when that work moves without shared context, bounded authority, independent review, or a recovery path.



The unit of design is not the agent. It is the operating loop around the agent.

Six layers turn autonomous activity into accountable execution.

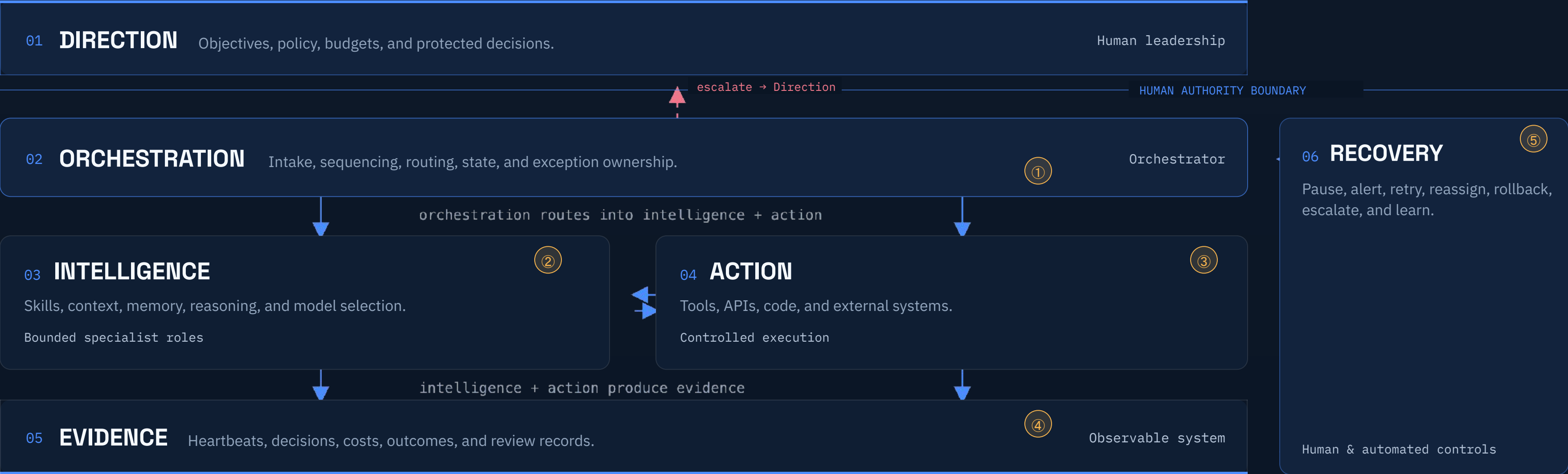
- 01 Direction sets the rules.

02 Orchestration moves the work.

03 Intelligence decides within bounds.
- 04 Action changes the environment.

05 Evidence makes work visible.

06 Recovery restores control.



EXAMPLE · CUSTOMER ONBOARDING EXCEPTION

① incomplete intake routed ② missing requirements identified ③ approved follow-up prepared ④ decision recorded ⑤ pause or escalate if unresolved

Every action should have an owner, a boundary, a visible state, and a recovery path.

Separate initiative from authority.

Accountability comes from distinguishing who identifies work, who performs it, who reviews it, and who retains decision authority.

- work

escalation
- review

authority



Orchestrator routes the incomplete intake · Specialist Agent prepares the follow-up · Independent Reviewer validates high-impact output · Human Owner decides policy exceptions.

PROTECTED DECISIONS
Human Owner

WORK SEQUENCING
Orchestrator

BOUNDED EXECUTION
Specialist Agents

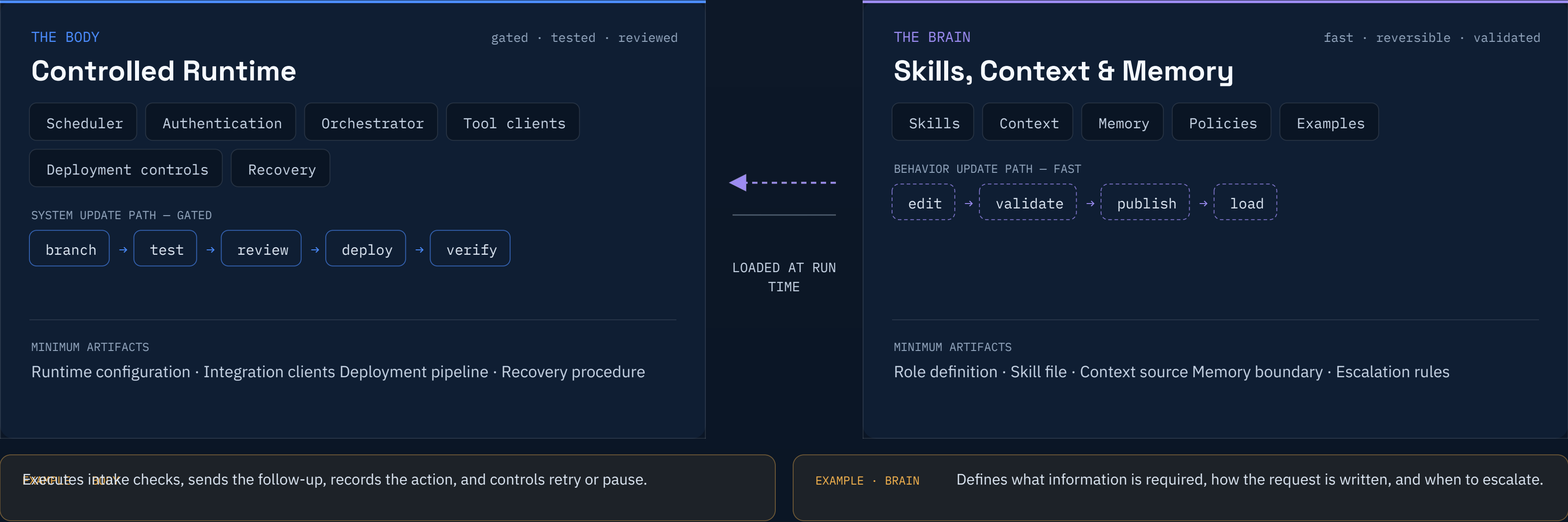
HIGH-IMPACT VALIDATION
Independent Reviewer

PAUSE & RECOVERY
Reliability Control

No single agent should initiate, approve, and verify its own high-impact action.

The runtime is the body. Skills and memory are the brain.

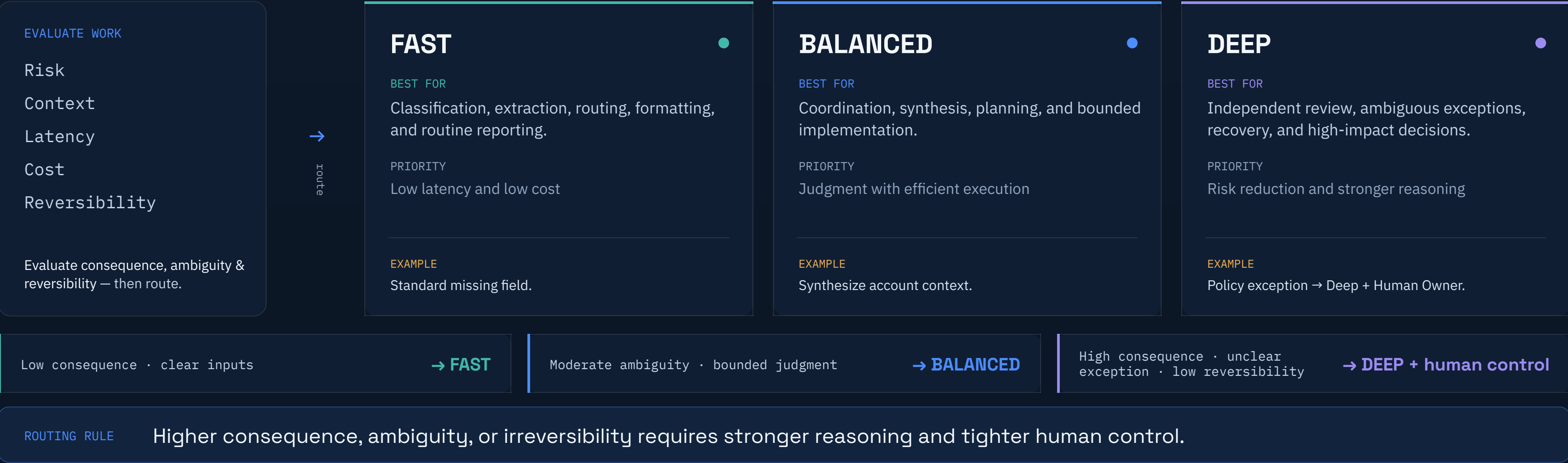
Separate stable system infrastructure from behavior that must evolve quickly — and apply the right control path to each.



Change behavior quickly. Change the system through controlled engineering.

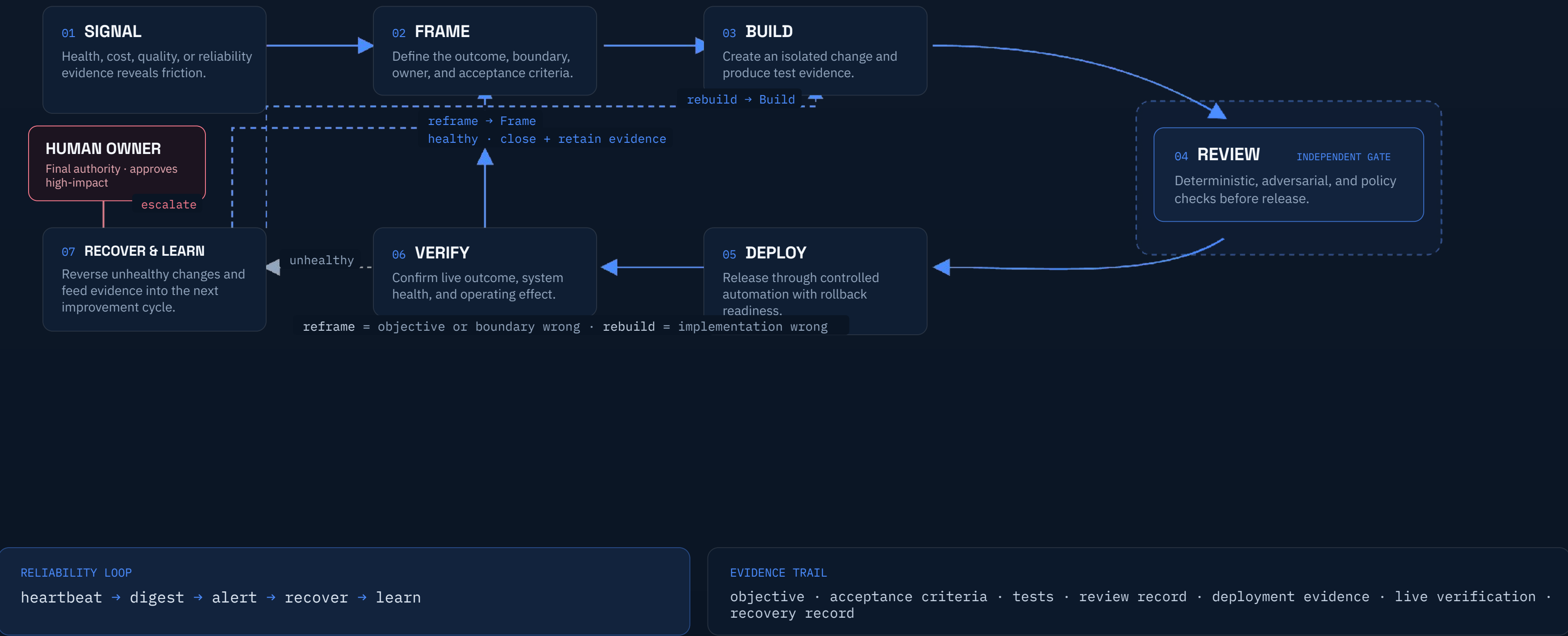
Use the lightest intelligence that can safely complete the work.

Model selection is an operating policy — matched to risk, context, latency, cost, and reversibility.



From operating signal to verified production.

No single agent should identify, implement, approve, and validate its own high-impact change. Discovery, review, deployment, and recovery are separate responsibilities.



Autonomous work can move quickly without becoming opaque or self-authorizing.

The architecture is portable. This build shows one working implementation.

The reference architecture has been implemented as a working agent fleet. The rows below show one possible mapping from the operating model to a deployed system — not the only valid architecture.

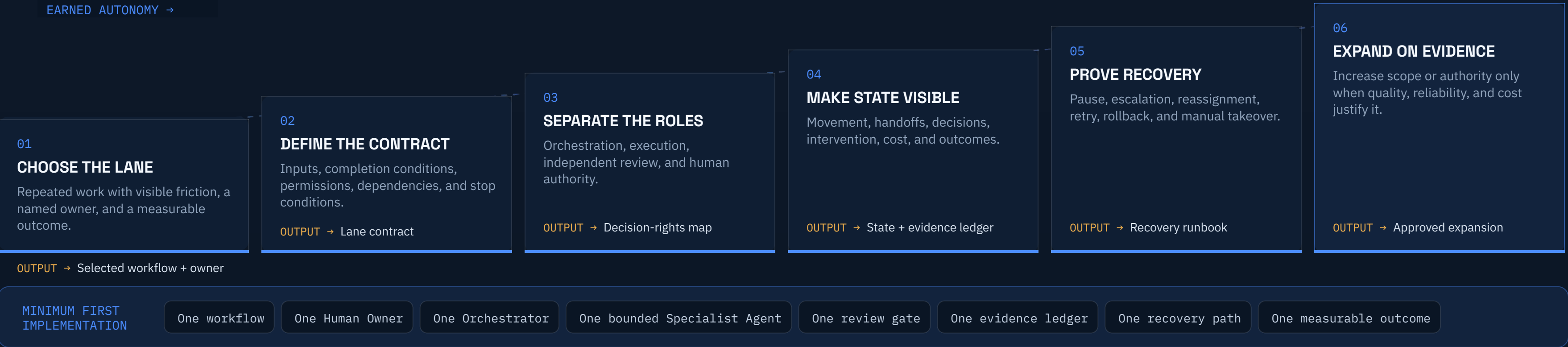
ARCHITECTURE LAYER	BUILT COMPONENT	WHAT IT DOES	STATUS
Human Owner	Roman — founder	Sets priority, approves outbound & spend, merges protected paths	● BUILT
Orchestrator / Director	Director agent	Fleet-health, active check-ins, the human front door, deploys, cross-session memory	● BUILT
Specialist Agents	Project Manager · Chief of Staff · Continuous Improvement Engineer · Staff Engineer	Bounded lanes — Tracker, knowledge work, backlog ranking, code review	● BUILT
Tool & Runtime	Agent loop · scheduled runtime · Slack, Drive & GitHub clients	Acts inside scoped permissions; code writes are branch + PR only — never main	● BUILT
Intelligence Routing	Model router	Haiku for mechanical work, Sonnet for judgment, Opus for adversarial review	● BUILT · routed + independent review
Memory / Context	Company Brain · fleet state	Behavior & context load at runtime; edit Drive → new behavior, no redeploy	● BUILT
Evidence Ledger	Heartbeat records · cost monitoring · cache-savings records	Durable proof each job ran; spend and caching savings recorded	● BUILT
Recovery & Change Control	Fleet health · independent engineering review · automatic rollback	Detects unhealthy state, escalates protected paths to a human, rolls back bad deploys	● BUILT

SAFETY INVARIANTS

Cron code writes are branch + PR only, never main · the reviewer is never the author · merge needs a deterministic gate and adversarial review · protected paths always escalate to a human · failed deploys auto-revert.

Start with one lane. Earn the right to expand.

Begin with one meaningful workflow, explicit accountability, and a tested recovery path — then widen the lane only as evidence justifies it.



THE RESULT Faster execution with clearer ownership, lower hidden intervention, and a system that can recover when conditions change.

The objective is not maximum autonomy.

It is an operating system that can increase autonomy without losing clarity, accountability, or control.