ENTERPRISE STANDARD

AI OPERATIONS

A Specification for the Enterprise Discipline of Running Artificial Intelligence

Daniel S. Wipert

Chorus AI Systems · Version 1.0 · July 2026

Why every company will need an AI Operations function, what it does, how it is measured, and how to build it

About this document. This specification is versioned semantically: major versions change or add normative requirements; minor versions correct, clarify, or extend non-normative content. Conformance claims MUST cite the version claimed against. Version 1.0 was authored by Daniel S. Wipert and published by Chorus AI Systems in July 2026. Feedback, proposed requirements, and errata: wipertds@gmail.com. Current version and changelog: https://danielwipert.github.io/aiops/.

Executive Summary

Enterprises do not have an artificial intelligence problem. They have an *operations* problem that artificial intelligence has exposed.

The capability is everywhere. Models that can read, write, extract, classify, summarize, and reason are available to every company on earth through an API call. Vendors are abundant, tooling is maturing, and employees are already using AI — with or without permission. And yet the dominant enterprise experience of AI is a graveyard of pilots: proofs of concept that impressed a steering committee and then never touched a production workflow, tools purchased and unadopted, spend that no one can attribute to an outcome, and a quiet layer of unsanctioned "shadow AI" usage that carries all of the risk and none of the measurement.

This document argues that the gap between AI capability and AI value is not a technology gap. It is an operations gap — and it will be closed the same way every prior enterprise technology gap was closed: by the emergence of a dedicated operating discipline. IT got ITIL. Software delivery got DevOps and SRE. Cloud spend got FinOps. Data got DataOps. Each discipline arrived roughly three to five years after its underlying technology reached the enterprise, precisely when the chaos of ungoverned adoption became more expensive than the cost of building an operating function.

AI is now at that point in the arc. This specification names the discipline — **AI Operations** — and defines it in full: its theoretical foundation, its operating model, its governance and reliability layer, its economics, its measurement system, its organizational design, its maturity model, and the executive role accountable for it.

Definition. AI Operations is the discipline of deploying, governing, measuring, and continuously optimizing artificial intelligence as an enterprise operating capability, so that AI produces measurable business outcomes rather than technology outputs.

The central claim of this specification can be stated in one sentence: **AI should be run the way a world-class supply chain is run.** A supply chain converts external suppliers, variable

costs, quality risk, and regulatory constraints into reliable delivery at a known cost per unit. An AI portfolio has exactly the same shape: model providers are suppliers; model routing is carrier selection; evaluation is quality control; governance gates are compliance checkpoints; cost per workflow is landed cost per unit; degradation handling is exception management. The mental models, control systems, and management disciplines that made physical operations legible to the enterprise are the same ones that will make AI legible — and the leaders who already carry those disciplines are the natural owners of the function.

The document is organized in seven parts. Part I makes the argument: the operations gap is real, measurable, and follows a historical pattern that predicts its resolution. Part II supplies the theoretical foundation: AI as a supply chain, structured by the Viable System Model, governed by a reliability doctrine that decides when autonomy is acceptable and when it is not. Parts III through V specify the machinery: the portfolio lifecycle, the governance and evaluation layer, the economic model, and the full measurement system. Part VI covers organizational design, the maturity model, and adoption. Part VII is execution: a first-365-days playbook, the complete role specification for a Head of AI Operations, and direct rebuttals to the standard objections.

This is a specification, not an essay: it defines 57 individually numbered, testable requirements (AIOPS-PORT-01 through AIOPS-ROLE-02) using standard conformance language, maps every requirement to a four-level maturity model, and supplies the working artifacts — intake form, tolerance-class rubric, production gate checklist, conformance checklist — required to implement it. An organization can read Part I to be convinced, and Appendix D to know exactly what conforming would take.

A closing note on timing. Disciplines are named by the people who arrive before the org chart does. FinOps existed as a practice inside Adobe, Intuit, and Atlassian years before it had a foundation, a certification, and a conference. The companies that establish AI Operations in the next eighteen months will spend that period compounding measurable advantages — cost per workflow falling, deployment rate rising, audit posture hardening — while their competitors are still asking whose job AI is. The window is the argument.

Part I — The Argument

1. The Operations Gap

Every enterprise currently exhibits some version of the following four symptoms. Together they define the operations gap — the distance between what AI can do and what AI actually delivers.

1.1 The pilot graveyard

Industry analyses from 2024 and 2025 converged on a striking figure: the overwhelming majority of enterprise generative AI pilots — commonly reported in the 70–95 percent range — never produce measurable P&L impact. Gartner projected that a large share of generative AI projects would be abandoned after proof of concept; MIT research circulated widely in 2025 put the failure-to-impact rate for enterprise GenAI initiatives near 95 percent. The precise number matters less than the shape of the finding, which every operator recognizes: pilots succeed as *demonstrations* and fail as *deployments*.

The reasons are rarely technical. The model worked. What was missing was everything around the model: no owner accountable for the workflow after the demo; no evaluation regime that could tell a risk officer the error rate; no cost attribution that could tell a CFO the unit economics; no change management that could move the affected team from observing the tool to depending on it; no lifecycle plan for what happens when the underlying model is deprecated in nine months. These are not engineering deliverables. They are operations deliverables, and in most organizations no one owns them.

1.2 Shadow AI

While official pilots stall, unofficial adoption accelerates. Employees paste customer data into consumer chatbots, buy departmental AI subscriptions on corporate cards, and build personal automations that touch production data. Surveys through 2025 consistently found that a majority of knowledge workers were using AI tools at work, and that a large fraction of that usage was unsanctioned and invisible to IT.

Shadow AI is the worst of both worlds: the enterprise carries the full risk surface — data leakage, unvetted outputs entering decisions, compliance exposure — while capturing none of the leverage, because nothing is measured, standardized, or scaled. Shadow AI is also the single most useful diagnostic signal an enterprise has: it is a live map of where employees experience enough friction that they will route around policy to remove it. A functioning AI Operations discipline treats shadow AI as demand data, not merely as a violation.

1.3 Unattributed spend

AI spend is fragmenting across at least five ledgers: API consumption billed by model providers, AI features bundled into SaaS renewals, dedicated AI tools bought by departments, infrastructure and vector-database costs inside the cloud bill, and headcount time spent building and babysitting automations. Almost no enterprise can answer the questions a CFO asks of every other cost center: What is our fully loaded AI spend? What does each AI-assisted workflow cost per execution? Which spend is producing return and which is shelfware? Cloud computing produced exactly this condition between roughly

2012 and 2019, and the answer was a named discipline — FinOps — with practitioners, standards, and board-level reporting. AI FinOps does not yet exist in most enterprises. Section 10 of this specification defines it.

1.4 Ungoverned reliability

The final symptom is the most dangerous. AI outputs are probabilistic; business processes are not. When a model-generated answer enters a customer communication, a financial document, a compliance filing, or an operational decision, the enterprise has silently accepted an error rate that no one has measured, no one has approved, and no one can audit after the fact. Most organizations cannot answer the question a regulator, a customer, or a board member will eventually ask: *how do you know that number was right, and who signed off on the process that produced it?* Reliability without measurement is a claim, not a property. Part II of this specification treats this as the core doctrinal problem of enterprise AI.

1.5 The diagnosis

Four symptoms, one disease. Pilots die because no one owns production. Shadow AI grows because no one owns sanctioned capability. Spend is unattributable because no one owns the economics. Reliability is unmeasured because no one owns governance. The common factor is ownership — and ownership of exactly the kind that operations leaders have always held: cross-functional, accountable for outcomes rather than artifacts, fluent in both the technical system and the business process it serves.

The thesis of Part I: Engineering builds AI. Nobody runs it. The gap between building and running is where the value is dying, and closing that gap is a discipline, not a project.

2. The Historical Pattern: Every Wave Produces an Operations Discipline

The claim that AI needs an operations discipline is not speculative. It is the fourth repetition of a pattern that has held for every major enterprise technology wave of the past forty years. The pattern has four phases:

- **Phase 1 — Capability arrives.** The technology becomes broadly available and obviously powerful.
- **Phase 2 — Chaos compounds.** Adoption outruns governance. Spend fragments, quality varies wildly, risk accumulates silently, and heroic individuals hold everything together.
- **Phase 3 — The discipline is named.** Practitioners converge on shared vocabulary, control loops, and metrics. A role appears on org charts.

- **Phase 4 — Institutionalization.** Frameworks, certifications, and executive accountability. The discipline becomes table stakes, and *not* having it becomes a due-diligence red flag.

Consider the precedents in sequence.

2.1 IT Operations and ITIL

Enterprise computing in the 1980s was Phase 2 chaos: every department ran systems its own way, outages were normal, and change was uncontrolled. The UK government's response — the IT Infrastructure Library, first published in 1989 — did not invent any technology. It invented *management structure*: incident management, change management, service levels, a named accountability chain. Within fifteen years ITIL vocabulary was universal, and "IT Operations" was a career, an org, and a budget line. The lesson: the discipline that made computing valuable was managerial, not technical, and it arrived years after the technology did.

2.2 DevOps and Site Reliability Engineering

By the mid-2000s, software delivery had split into builders who wanted speed and operators who wanted stability, and the interface between them was where quality died. Two answers emerged. DevOps dissolved the wall culturally and through automation. SRE — Google's formulation, publicized in 2016 — contributed something deeper and directly relevant to AI: the **error budget**. SRE's insight was that perfect reliability is neither achievable nor economically rational; the correct move is to *choose* a reliability target, measure against it continuously, and spend the remaining "budget" of acceptable failure on velocity. This is precisely the intellectual move enterprise AI requires: stop pretending model outputs are either "good" or "bad," and instead set explicit, workflow-specific error tolerances, measure against them, and govern the trade-off. Section 6 imports the error-budget mechanism wholesale.

2.3 FinOps

Cloud computing made infrastructure spend variable, decentralized, and invisible — any engineer could commit the company to cost with an API call. For most of a decade, cloud bills were a monthly surprise. The FinOps discipline (formalized under a foundation in 2019) answered with unit economics, cost attribution, showback and chargeback, and a cultural principle: engineers own the cost consequences of their choices. FinOps is the closest structural precedent to AI Operations because the cost physics are identical — per-call variable pricing, decentralized consumption, and vendor lock-in dynamics — except that AI adds a dimension cloud never had: the *quality* of the output varies with the spend. AI economics is FinOps plus a quality axis. That compound problem is treated in Part IV.

2.4 DataOps and MLOps — and why they are not enough

Data management produced DataOps; classical machine learning produced MLOps — pipelines, model registries, deployment automation, drift monitoring. These are real disciplines and AI Operations sits downstream of both. But neither answers the enterprise question. MLOps is an *engineering* discipline: its customer is the model, and its success condition is that the model is deployed, versioned, and monitored. AI Operations is a *business* discipline: its customer is the workflow, and its success condition is that the enterprise's cost, quality, speed, or revenue measurably improved, with governance a regulator would accept. MLOps ensures the model runs. AI Operations ensures the model running *matters*. Confusing the two is the most common way enterprises convince themselves the gap is already covered. It is not — the org chart shows it: MLOps reports into engineering and holds no P&L accountability, no adoption mandate, no vendor portfolio, and no workflow ownership.

2.5 Where AI sits on the arc

Map generative AI onto the four phases and the position is unambiguous. Capability arrived at enterprise scale in 2023. Chaos — pilot graveyards, shadow AI, unattributed spend — compounded through 2024 and 2025 and is now the consistent finding of every industry survey. The discipline is being named *now*: fragments of it appear in job postings as "AI enablement," "AI transformation," "AI program management," and "Head of AI" roles with operational mandates. Institutionalization has not yet happened. That places AI in early Phase 3 — the exact moment when, historically, the practitioners who could articulate the discipline defined its standards and occupied its senior roles.

Wave	Chaos period	Discipline	Core contribution	AI Operations inherits
Enterprise IT	1980s	IT Ops / ITIL (1989)	Service management, change control, accountability chain	Lifecycle & incident discipline for AI workflows
Software delivery	2000s	DevOps / SRE (2009/2016)	Error budgets, automation, shared ownership	Explicit error tolerance & reliability targets per workflow
Cloud spend	2012–2019	FinOps (2019)	Unit economics, attribution, engineer cost ownership	Cost per workflow, model routing economics, chargeback
Data / ML	2015–2022	DataOps / MLOps	Pipelines, registries, monitoring	Technical substrate AI Ops builds on — but not the business layer
Enterprise AI	2023–present	AI Operations (now)	This specification	—

One more property of the pattern deserves emphasis because it bears directly on who should hold the role. In every prior wave, the discipline was built by *operators who learned the technology*, not by technologists who learned operations. ITIL came from service managers. FinOps came from finance and procurement people who learned cloud billing. The reason is structural: the discipline's hard problems — accountability, adoption, economics, cross-functional coordination — are operations problems with a technical surface, not technical problems with an operations surface. Part VII returns to this when specifying the ideal background for the role.

3. Definition, Scope, and Boundaries

3.1 Definition

AI Operations is the discipline of deploying, governing, measuring, and continuously optimizing artificial intelligence as an enterprise operating capability to produce measurable business outcomes. Its unit of management is the **AI-assisted workflow**, not the model. Its accountability is to the **P&L and the risk register**, not to the technology roadmap.

Three phrases in that definition carry the weight.

- **"Operating capability"** — AI is treated like manufacturing capacity, logistics, or customer operations: a permanent function with owners, budgets, standards, service levels, and continuous improvement — not a portfolio of projects that end.
- **"The workflow, not the model"** — models are interchangeable components sourced from a volatile supplier market. Workflows are where value is created and where risk lives. Managing at the workflow level is what separates AI Operations from every model-centric discipline.
- **"Measurable business outcomes"** — the function's output is denominated in cycle time, cost per unit, throughput, quality, revenue, and audited risk posture. A deployed model that cannot show up in one of those numbers is inventory, not value.

3.2 What AI Operations owns

- **The portfolio** — the single inventory of every AI initiative, tool, and vendor in the enterprise, from intake through retirement (Section 7).
- **The operating model** — the standard lifecycle, stage gates, and playbooks by which any AI capability moves from idea to governed production (Section 7).
- **Workflow redesign** — identifying where AI creates measurable value and re-architecting the human-plus-machine process around it (Section 8).

- **Governance and reliability** — evaluation regimes, error tolerances, human-review policy, audit traceability, incident response, and lifecycle risk management (Section 9).
- **Economics** — budgets, vendor and contract management, model routing policy, unit-cost measurement, ROI reporting, and FinOps partnership (Section 10).
- **Adoption** — change management, enablement, training, and the conversion of shadow AI into sanctioned capability (Section 15).
- **Measurement** — the KPI system and executive dashboards through which the board sees AI as a managed capability (Sections 11–12).

3.3 What AI Operations explicitly does not own

A discipline is defined as much by its boundaries as by its mandate. AI Operations does **not** own:

- **Model research or training** — that is data science and ML engineering.
- **Software engineering of AI systems** — engineering builds; AI Operations specifies requirements, gates releases, and runs what is built. The relationship is the same as between a supply chain organization and the engineering team that builds its WMS.
- **Data platform and pipelines** — owned by data engineering; AI Operations is a demanding customer with contractually explicit quality requirements.
- **Security policy** — owned by the CISO; AI Operations implements and evidences compliance within AI workflows.
- **Legal and regulatory interpretation** — owned by counsel; AI Operations translates interpretation into enforceable gates inside workflows.
- **Enterprise strategy** — owned by the CEO and the line; AI Operations makes strategy executable and measurable, and supplies the intelligence about what is becoming possible.

3.4 Boundary map

Adjacent function	Their question	AI Operations' question	Interface
ML / AI Engineering	Does the system work as designed?	Does the workflow hit its error, cost, and cycle-time targets in production?	Requirements in, release gates out
MLOps / Platform	Is the model deployed, versioned, monitored?	Is the workflow delivering measured business value?	AI Ops consumes platform telemetry; sets SLOs

Adjacent function	Their question	AI Operations' question	Interface
IT / CIO	Are systems available, secure, supported?	Is AI capability adopted, governed, economically sound?	Shared service catalog; shadow-AI conversion
CISO / Security	Is the attack and leakage surface controlled?	Are controls embedded in workflows without killing adoption?	Control requirements in, evidence out
Finance / FinOps	Is spend visible, attributed, in budget?	Is spend productive — cost per workflow trending down, ROI proven?	Joint unit-economics model (Section 10)
Legal / Compliance	Are we defensible?	Can every output be traced to evidence and policy?	Audit-traceability requirements (Section 9)
CAIO / AI strategy	What should AI do for the enterprise?	Is it actually doing it, at what cost, at what risk?	Strategy in, verified outcomes out (see 19.1)

The pattern across every row is the same: adjacent functions ask whether their *artifact* is sound — the model, the system, the policy, the budget. AI Operations asks whether the *outcome* occurred. That is the definitional divide, and it is why the function cannot be assembled by adding responsibilities to any existing role. Outcome accountability that spans engineering, finance, risk, and the business line is an operations mandate, and it requires an operator holding it.

3.5 Conformance language

This specification contains normative requirements. The key words **MUST**, **MUST NOT**, **SHOULD**, **SHOULD NOT**, and **MAY** are to be interpreted consistent with the convention established in IETF RFC 2119: MUST denotes an absolute requirement for conformance; SHOULD denotes an expected practice from which deviation is permitted only with documented justification; MAY denotes a genuinely optional practice.

Requirements are individually numbered in the form **AIOPS-<AREA>-<NN>** (for example, AIOPS-GOV-05) so that they can be cited in conformance claims, audit findings, and contracts. Requirements appear in a "Normative requirements" block at the end of the section whose argument motivates them; the surrounding prose is explanatory and non-normative. Where a requirement is scoped to a tolerance class (Section 6), the scope is stated in the requirement itself. Appendix D maps every requirement to the maturity level (Section 14) at which it must be satisfied; per AIOPS-MAT-01, a claim at any level requires satisfying all requirements at that level and every level below it.

Part II — Theoretical Foundation

A discipline needs more than a mandate; it needs a theory — a set of mental models that tell practitioners what to measure, where failure hides, and which trade-offs are real. AI Operations rests on three foundations: a *structural metaphor* (AI as supply chain), a *formal architecture* (the Viable System Model), and a *reliability doctrine* (governed workflows versus autonomous agents, decided by error tolerance). Together they turn "managing AI" from vibes into engineering.

4. AI as Supply Chain: The Structural Metaphor

The most useful way to understand an enterprise AI portfolio is to notice that it has exactly the shape of a supply chain. This is not a decorative analogy. It is a structural isomorphism — the same objects, the same flows, the same failure modes, and therefore the same management disciplines apply.

A supply chain converts a volatile external supplier market into reliable internal delivery at a known landed cost per unit, under quality control and regulatory constraint. Every clause of that sentence maps onto AI:

Supply chain concept	AI Operations equivalent	The shared management problem
Suppliers	Model providers (Anthropic, OpenAI, Google, open-weight vendors)	Concentration risk, qualification, dual-sourcing, price volatility
Procurement & contracts	API agreements, enterprise AI licenses, SaaS AI features	Rate negotiation, SLAs, exit clauses, renewal leverage
Carrier / route selection	Model routing (which model serves which task)	Cost-quality-latency optimization per lane / per task
Incoming QC	Model qualification & benchmark evals before adoption	Accept/reject criteria; supplier scorecards
In-process QC & SPC	Continuous evaluation, faithfulness checks, verification gates	Sampling regimes, control limits, drift detection
SLA adherence	Latency, availability, and error-rate targets per workflow	Measurement, penalty structures, escalation
Landed cost per unit	Cost per workflow execution (tokens + infra + review labor)	Unit economics; make the true cost visible

Supply chain concept	AI Operations equivalent	The shared management problem
Inventory & BOM	Prompt assets, retrieval corpora, context templates, eval sets	Versioning, freshness, obsolescence management
Exception management	Degradation handling, fallbacks, human escalation	Detect, contain, disclose, resolve, learn
Trade compliance	AI governance gates, data-handling rules, audit trails	Controls embedded in flow, evidenced on demand
S&OP	AI portfolio planning (demand for AI capability vs. capacity to deploy well)	Prioritization under constraint; forecast vs. actual
Network redesign	Workflow redesign around AI capability	Re-architecting flow when the cost structure changes

Three consequences follow from taking the isomorphism seriously.

4.1 Supplier management, not vendor worship

Enterprises currently treat model providers the way naive buyers treat a sole-source supplier: they standardize on one, absorb every price and deprecation decision, and confuse the supplier's roadmap with their own strategy. A supply chain organization would never run a critical input this way. The AI Operations posture is classical supplier management: qualify multiple providers against explicit criteria; dual-source every critical task category; maintain switching capability as a tested, living asset (routing abstraction, provider-agnostic prompts, portable eval sets); score suppliers quarterly on quality, cost, reliability, and roadmap risk; and negotiate from the credible ability to move volume. Model markets are more volatile than any physical supplier market — capabilities shift quarterly, prices move by an order of magnitude, and deprecations are unilateral. Volatility is precisely the condition under which supplier management earns its keep.

4.2 Routing is the new carrier selection

No operator ships every parcel by the fastest, most expensive carrier. Yet enterprises routinely route every AI task — trivial classification and high-stakes analysis alike — to a single frontier model at frontier prices. Model routing is the carrier-selection problem restated: for each task lane, choose the option that clears the *quality bar* at the lowest cost and acceptable latency. The discipline is identical: define lanes (task categories with explicit quality thresholds), qualify carriers per lane (eval every candidate model against the lane's test set), set routing policy in code, and re-bid lanes periodically as the market moves. Mature routing routinely cuts model spend by 40–80 percent with zero quality loss, because most enterprise task volume is boring and does not need a frontier model. This single practice usually pays for the AI Operations function by itself.

4.3 Cost per unit is the sovereign metric

Supply chains matured when they learned to compute true landed cost — not the invoice price, but everything: freight, duty, handling, quality failure, carrying cost. AI has the same hidden-cost structure. The invoice shows tokens; the true **cost per workflow execution** includes tokens across all calls and retries, retrieval and infrastructure, the amortized cost of prompt and eval-set maintenance, the labor cost of human review at the current review rate, and the downstream cost of the errors that slip through. Until that number exists per workflow, every ROI claim is fiction and every optimization is guesswork. Section 10 specifies its construction; Section 11 makes it a headline KPI.

The metaphor's payoff: every practice in this specification — supplier scorecards, routing policy, QC sampling, exception management, S&OP-style portfolio planning — is a proven supply chain discipline transposed onto a new class of supplier. Nothing here is exotic. That is exactly why it will work, and why operators are the natural owners.

5. The Cybernetic Frame: A Viable System Model for Enterprise AI

The supply chain metaphor supplies the objects; the Viable System Model supplies the architecture. Stafford Beer's VSM, developed in the 1970s as a cybernetic theory of organization, answers a question every new function faces: *what set of subsystems must exist for an operating capability to remain viable — adaptive, self-regulating, and coherent — inside a changing environment?* Beer's answer is five interacting systems, and mapping them onto enterprise AI produces, almost mechanically, the org design and control loops of an AI Operations function.

VSM system	Function in Beer's model	AI Operations instantiation
System 1 — Operations	The primary activities that do the work	The portfolio of production AI workflows, each embedded in a business unit and each individually accountable for its outcomes
System 2 — Coordination	Anti-oscillation: shared standards that stop units colliding	Common platform standards, shared prompt/eval libraries, routing policy, naming and logging conventions, intake process — the boring glue that prevents fifty teams solving the same problem fifty ways
System 3 — Control	Day-to-day resource allocation, synergy, accountability	Portfolio management: budgets, prioritization, stage gates, SLO enforcement, cross-workflow optimization
System 3* — Audit	Sporadic direct inspection, bypassing management filters	The evaluation and audit regime: sampled output review, eval runs against golden sets, surprise deep-dives into any workflow's actual behavior — evidence gathered independently of the team's self-report

VSM system	Function in Beer's model	AI Operations instantiation
System 4 — Intelligence	Outward- and forward-looking adaptation	Model-market scanning, capability roadmapping, pilot pipeline, regulatory horizon watching — the function that notices the environment changed before the P&L does
System 5 — Policy	Identity, values, ultimate constraint-setting	The AI governance constitution: risk appetite, error tolerances by workflow class, prohibited uses, escalation rights — the closure that makes everything below it legitimate

Two properties of the VSM matter enormously in practice.

5.1 Recursion

Beer's model is recursive: every viable system contains, and is contained in, other viable systems with the same five-part structure. Applied here: the enterprise AI function is a viable system, and *each production workflow inside it is also a viable system* — with its own operations (the pipeline stages), coordination (schemas and contracts between stages), control (orchestration logic), audit (per-stage verification), intelligence (monitoring for drift and input change), and policy (the workflow's own error tolerance and governance gates). This recursion is what makes governance scale. The board does not audit prompts; it audits that System 5 constraints exist and that System 3* evidence flows upward. Each level regulates the level below by the same pattern, so adding the hundredth workflow costs no new governance theory — only a new instance.

5.2 The audit channel is non-negotiable

System 3* — direct, sporadic inspection that bypasses reporting lines — deserves special emphasis because it is the piece enterprises most consistently omit. Dashboards report what teams instrument; teams instrument what makes them look good. In probabilistic systems, self-reported health is worth little. The AI Operations function must retain the standing right and the technical means to pull raw production outputs from any workflow and evaluate them independently against golden sets. Every mature quality regime in physical operations — incoming inspection, line audits, mystery shopping — exists because of this same insight. Section 9 builds the mechanism.

The VSM also supplies the correct answer to a design question this specification will face in Part VI: is AI Operations centralized or federated? Beer's answer is *both, by construction* — System 1 workflows live in the business units, close to the work; Systems 2 through 5 are the thin central function that coordinates, resources, audits, adapts, and constrains. Hub-and-spoke is not a compromise between centralization and federation; it is the shape viability requires.

6. The Reliability Doctrine: Error Tolerance Decides Architecture

The third foundation is doctrinal: a principled answer to the question that sits under every AI deployment decision — *how much autonomy should the machine have in this workflow?* The current industry discourse frames this as a technology race (agents are the future; workflows are legacy). AI Operations reframes it as a risk-allocation decision, made per workflow, from a single variable: **the cost of an undetected error**.

6.1 The axis that matters

The meaningful distinction is not "automated versus manual" — both governed pipelines and autonomous agents run without a human clicking between steps. The axis is **who decides what happens next: deterministic code, or the model at runtime.**

- In a **governed workflow (compound AI system)**, control flow lives in code. Each model call does one bounded job — extract, classify, draft, verify — and hands the result back; code decides the next step, enforces schemas, runs verification, and gates release. The system can be replayed, audited, and reasoned about, because its path is fixed.
- In an **agent**, the model holds the control loop. Given a goal and tools, it chooses which tool to call, interprets results, decides whether to loop, and declares itself done. The path is chosen at runtime and can differ between two identical inputs.

6.2 The decision rule

Agents where a wrong action is cheap and recoverable; governed pipelines where it is not. Autonomy is purchased with error tolerance. Where an undetected error costs little (internal drafts, research triage, brainstorming), agentic flexibility is a bargain. Where an undetected error is expensive, regulated, or public (financial reporting, customer commitments, compliance filings, clinical or legal content), determinism, verification gates, and audit traceability are non-negotiable — because non-determinism is an *auditability* liability before it is a quality liability.

This rule converts a religious argument into an engineering table. Every workflow entering the portfolio is classified by error tolerance, and the tolerance class dictates the minimum control architecture:

Class	Undetected-error cost	Example workflows	Minimum architecture
T1 — Tolerant	Trivial; human catches downstream	Ideation, internal search, draft outlines	Agentic patterns acceptable; light logging

Class	Undetected-error cost	Example workflows	Minimum architecture
T2 — Moderate	Rework, minor confusion	Internal summaries, ticket triage, research assists	Structured outputs, spot-check evals, fallback paths
T3 — Low	Customer-visible or costly	Customer communications, ops decisions, published content	Governed pipeline; verification stage; human review on sampled or flagged output; full tracing
T4 — Near-zero	Regulatory, financial, legal, safety	Filings, financial analysis, compliance, contractual output	Deterministic control flow; independent generator/verifier separation; citation-level traceability; mandatory degradation disclosure; human sign-off gates; immutable audit log

6.3 Doctrine into mechanism

Class T3 and T4 architecture imports five mechanisms that mature reliability engineering already trusts:

- **Generator/verifier separation.** The component that produces an output must be structurally different from the component that checks it — different model, different prompt frame, ideally different failure modes. Self-grading is not verification.
- **Immutable facts, cited claims.** Once a fact is computed and verified, it is frozen; every downstream claim must cite a frozen fact by identifier, and citations are validated deterministically in code. An output that cannot cite is stripped or blocked, never shipped on trust.
- **Error budgets per workflow.** Borrowed from SRE (Section 2.2): each workflow carries an explicit reliability target agreed with the business owner, measured continuously. Inside budget, optimize for cost and speed; budget exhausted, feature work stops and reliability work starts. This converts “is it good enough?” from an argument into a dashboard.
- **Mandatory degradation disclosure.** When any component falls back — a model unavailable, a verification skipped, retrieval thin — the output must say so, visibly and in the audit log. Silent degradation is how probabilistic systems destroy trust; disclosed degradation is how they earn it.
- **Human review as a designed control, not a vibe.** The human-review rate is a governed variable per workflow class: set deliberately, measured continuously, and reduced only when eval evidence justifies it. Review-rate reduction is the *earned dividend* of demonstrated reliability — and one of the cleanest ROI levers the function owns, since review labor is usually the dominant cost term in T3/T4 unit economics.

The doctrine's final clause is cultural. Holding it requires respect for the trade-off, not contempt for either camp: agents are the correct answer where their preconditions hold, and the discipline's job is to *know the preconditions*, classify honestly, and refuse to let fashion — in either direction — override the error-tolerance math.

Part III — The Operating Model

7. The Portfolio: One Inventory, One Lifecycle

The first artifact of an AI Operations function — usually deliverable within thirty days — is the thing almost no enterprise possesses: a **single, complete inventory** of every AI initiative, tool, embedded SaaS AI feature, and shadow-AI pattern in the company, each tagged with owner, workflow, tolerance class (Section 6), spend, and lifecycle stage. The inventory is not bureaucracy; it is the precondition for every other capability in this document. You cannot govern, route, price, or optimize what you cannot enumerate.

Every item in the inventory then lives on one lifecycle. The lifecycle is the operating model's spine: a standard path from idea to retirement, with stage gates whose criteria are public, so that any team in the company knows exactly what it takes to get an AI capability into production and keep it there.

7.1 The six stages

- **Intake.** Any function may propose. The proposal names the workflow, the business metric it should move, the error-tolerance class, and the current baseline (cost, cycle time, quality). No baseline, no entry — a rule that alone kills most theater.
- **Qualification.** AI Operations scores the proposal on value density (metric impact × volume), feasibility, tolerance class, and strategic fit; the portfolio is prioritized like an S&OP plan — demand for AI against the enterprise's finite capacity to deploy well.
- **Pilot.** Time-boxed, instrumented from day one, run against the golden evaluation set built *before* the pilot starts, with explicit success criteria signed by the business owner. A pilot without pre-committed kill criteria is a demo with a budget.
- **Production hardening.** The gate everything currently dies at. Verification stages, fallbacks, degradation disclosure, audit logging, cost instrumentation, human-review policy, runbooks, and the adoption plan (Section 15) — built to the tolerance class's minimum architecture.

- **Operate &** optimize. The longest stage and the least glamorous: SLO monitoring, error-budget accounting, routing re-bids as the model market moves, review-rate reductions earned by eval evidence, quarterly unit-cost reviews. This is where the compounding happens.
- **Retire / re-platform.** Models deprecate, vendors fail, better patterns emerge. Every workflow carries a documented exit: what replaces it, how the eval set transfers, what the switching cost is. Planned retirement is a capability; unplanned retirement is an outage.

7.2 Gate criteria are the culture

Stage gates do the cultural work. When the pilot-to-production gate publicly requires a measured error rate against a golden set, a unit-cost estimate, and a signed business-owner commitment to the adoption plan, three things happen: weak proposals self-select out at intake; engineering teams design for the gate from day one because the gate is known; and the phrase “the pilot went well” loses currency, replaced by numbers. The gates are where the reliability doctrine (Section 6) and the economics (Section 10) stop being documents and start being enforced.

7.3 Normative requirements (AIOPS-PORT)

- **AIOPS-PORT-01 (MUST).** The organization MUST maintain a single inventory of all AI initiatives, tools, embedded SaaS AI features, and vendors, each tagged with owner, workflow, tolerance class, spend, and lifecycle stage.
- **AIOPS-PORT-02 (MUST).** Every inventory item MUST have exactly one accountable owner.
- **AIOPS-PORT-03 (MUST).** Every AI-assisted workflow MUST be classified by error-tolerance class (T1–T4) before entering pilot.
- **AIOPS-PORT-04 (MUST).** Every workflow MUST pass through the standard lifecycle (intake → qualification → pilot → production hardening → operate & optimize → retire); stage-gate criteria MUST be published and versioned.
- **AIOPS-PORT-05 (MUST).** Intake proposals MUST name the target business metric and freeze a pre-deployment baseline. Proposals without a baseline MUST NOT enter qualification.
- **AIOPS-PORT-06 (MUST).** Pilots MUST be time-boxed and MUST define kill criteria before starting.
- **AIOPS-PORT-07 (MUST).** Pilots MUST be evaluated against a golden test set built before the pilot begins.

- **AIOPS-PORT-08 (MUST).** Production release MUST require: a measured error rate against the golden set, a unit-cost estimate, and business-owner sign-off on the adoption plan.
- **AIOPS-PORT-09 (MUST).** Every production workflow MUST carry a documented retirement plan: replacement path, evaluation-set transfer, and switching cost.
- **AIOPS-PORT-10 (SHOULD).** The organization SHOULD run portfolio prioritization on a fixed cadence, balancing demand for AI capability against deployment capacity.

8. Workflow Redesign: Where the Value Actually Is

The highest-leverage skill in AI Operations is not prompt engineering. It is the operator's oldest skill: **process decomposition** — taking a workflow apart into its steps, finding where time, cost, and error concentrate, and re-architecting the flow. AI merely changes the economics of certain step types; the method is classical industrial engineering.

8.1 The decomposition method

- **Map the workflow as it actually runs** — not the process doc. Time-and-motion it: steps, handoffs, queues, rework loops, and the error rate at each step. (Shadow AI usage along the flow is marked as demand signal.)
- **Classify each step** by cognitive type: retrieve, extract, classify, transform, draft, verify, decide, communicate. AI's leverage is wildly uneven across these types — enormous on retrieve/extract/draft/first-pass-verify, modest on judgment-heavy decide steps, and often negative when forced onto steps requiring accountable human commitment.
- **Find the constraint.** Theory-of-constraints logic applies unchanged: accelerating a non-bottleneck step produces nothing but local vanity metrics. The AI goes where the queue is.
- **Re-architect the flow, not the step.** The common failure is "bolt-on AI": the step gets faster, the workflow doesn't, because the flow was designed around human batch rhythms. Redesign sequencing, batching, and handoffs around the new step economics — this is where 10x outcomes hide, and it is pure operations work.
- **Redesign the human role deliberately.** In T3/T4 workflows the human moves from producer to reviewer-and-exception-handler. That role must be designed: what they see, what they can override, what evidence accompanies each output, and how their corrections feed back into evals. An undesigned reviewer role decays into rubber-stamping, which is worse than no review because it launders machine error with human authority.
- **Instrument before and after.** The baseline from intake (Section 7) is the denominator of every ROI claim the function will ever make. Guard it.

8.2 Selection heuristics

Portfolio experience compresses into a few reliable heuristics for where AI-assisted redesign pays first: high-volume, document-heavy, rule-rich workflows with measurable outputs and existing quality standards (claims, KYC, invoice processing, contract review triage, customer-operations correspondence); workflows already carrying a review step, because the verification culture exists and the review rate becomes the ROI lever; and workflows whose constraint is *reading and routing* rather than judging. Conversely, the reliably bad first targets are workflows with contested ground truth, workflows whose real function is political, and workflows too low-volume to amortize the hardening cost of their tolerance class.

8.3 Normative requirements (AIOPS-WFR)

- **AIOPS-WFR-01 (MUST).** Workflows MUST be mapped as they actually run — steps, handoffs, queues, rework loops, per-step error rates — before AI is introduced.
- **AIOPS-WFR-02 (MUST).** Baselines MUST be measured end-to-end at the workflow level, not the step level.
- **AIOPS-WFR-03 (MUST).** The human role in every AI-assisted workflow MUST be explicitly designed: review responsibilities, override authority, evidence presented to the reviewer, and a feedback path into evaluation sets.
- **AIOPS-WFR-04 (SHOULD/MUST).** Redesigns SHOULD target the workflow constraint; reported gains MUST be measured on end-to-end flow.
- **AIOPS-WFR-05 (SHOULD).** Human corrections SHOULD feed back into evaluation sets as test cases.

9. Governance and Reliability: The Evaluation Regime

Governance in most enterprises is a document. In AI Operations it is a *control system* — implemented in the workflows themselves, evidenced continuously, and audited independently. This section specifies the layer that makes the phrase “governed AI” mean something a risk officer can inspect.

9.1 Evaluation is the instrument panel

Every reliability claim in this document reduces to a measurement question: *how do you know the error rate, and how would you prove it?* The evaluation regime answers with four instruments, run per workflow:

- **Golden test sets** — labeled examples with known correct answers, built with the business owner before the pilot, versioned like code, refreshed as the input distribution drifts. Set size follows from the claim: asserting a sub-1% error rate with statistical honesty requires test sets in the high hundreds to thousands, not a demo's dozen. The claim's precision is bounded by the set's size, and the function says so out loud.
- **Faithfulness / groundedness measurement** — for retrieval-based workflows, the fraction of output claims supported by retrieved evidence. Enforced structurally (claims must cite; uncited claims are stripped) and measured statistically (sampled audits score faithfulness over time). Standard tooling (e.g., RAGAS-style metrics: faithfulness, answer relevance, context precision/recall) makes this reportable without inventing anything.
- **LLM-as-judge with guardrails** — model-graded evaluation scales sampling far beyond human capacity, but is itself a measurement instrument with known failure modes (position bias, verbosity bias, self-preference). Doctrine: judges are calibrated against human-labeled subsets, never grade their own generator family where avoidable, and their disagreement rate with humans is itself tracked.
- **Per-stage error accounting** — in multi-stage pipelines, error is measured where it enters, not where it surfaces. A per-stage error ledger tells you whether to fix retrieval, extraction, generation, or verification — and turns "the AI was wrong" from an anecdote into a routable defect.

9.2 The governance gate set

Tolerance class (Section 6.2) determines which gates a workflow must implement. The full T4 set, from which lower classes subtract:

- **Input gates** — schema validation, data-classification checks (does this workflow have the right to see this data?), and prompt-injection screening on untrusted content.
- **Process gates** — generator/verifier separation; immutable fact lists with deterministic citation validation; bounded retries with logged fallbacks.
- **Output gates** — no unverified claim released; mandatory degradation disclosure; policy filters (prohibited content, commitments the enterprise hasn't authorized the machine to make).
- **Release gates** — human sign-off where the tolerance class demands it, with the reviewer shown the evidence trail, not just the answer.
- Audit gates (System 3*) — immutable logging of inputs, retrieved evidence, model versions, prompts, outputs, and reviewer actions, retained to the workflow's regulatory horizon; plus the independent sampled-review program that inspects production outputs without asking the owning team first.

9.3 Model risk management, inherited not invented

Regulated finance solved a version of this problem a decade ago. Supervisory guidance on model risk management (in the U.S., the Federal Reserve and OCC's SR 11-7 lineage) established the pattern: an inventory of models, documented purpose and limitations, independent validation, ongoing monitoring, and clear ownership. AI Operations generalizes that pattern to every industry, because the underlying logic — *any model used for decisions carries risk proportional to its use, and that risk is managed through inventory, validation, and monitoring* — is not a banking rule. It is simply what managing probabilistic systems looks like. Enterprises in regulated sectors will recognize the vocabulary; enterprises outside them inherit a decade of tested practice for free. Emerging AI-specific regimes (the EU AI Act's risk-tiering, NIST's AI RMF) follow the same architecture, which means a workflow portfolio classified by tolerance class and evidenced by the gate set above is, to a first approximation, *already shaped like compliance* with regulation that has not finished arriving.

9.4 Incidents

AI workflows fail in production — wrong outputs shipped, degradation undisclosed, drift undetected. The response is standard incident discipline transposed: severity classification by tolerance class and blast radius, containment (gate the workflow, raise the review rate to 100%, or roll back the route), disclosure obligations mapped in advance with legal, and blameless postmortems whose corrective actions land in the eval set — every incident becomes a permanent test case. A function that cannot answer "what happens when the AI is wrong?" with a rehearsed procedure does not yet deserve T3/T4 workflows.

9.5 Normative requirements (AIOPS-GOV)

- **AIOPS-GOV-01 (MUST).** The organization MUST maintain an AI governance constitution — risk appetite, tolerance classes, prohibited uses, escalation rights — with executive and legal sign-off, reviewed at least annually.
- **AIOPS-GOV-02 (MUST).** Every T3+ workflow MUST maintain a versioned golden test set; refresh age MUST be tracked.
- **AIOPS-GOV-03 (MUST).** Error-rate claims MUST carry confidence bounds, and claimed precision MUST NOT exceed what the test-set size statistically supports.
- **AIOPS-GOV-04 (MUST).** T3+ retrieval workflows MUST enforce structural groundedness: uncited claims MUST be stripped or blocked before release.
- **AIOPS-GOV-05 (MUST).** T4 workflows MUST implement generator/verifier separation; the verifier MUST be structurally different from the generator.

- **AIOPS-GOV-06 (MUST/SHOULD).** LLM-as-judge instruments MUST be calibrated against human-labeled subsets, and judge–human disagreement rates MUST be tracked. Judges SHOULD NOT grade outputs from their own model family.
- **AIOPS-GOV-07 (MUST).** Multi-stage pipelines MUST account for error at each stage, not only at the output.
- **AIOPS-GOV-08 (MUST).** Any fallback, skipped verification, or component degradation MUST be disclosed on the output and in the audit log. Silent degradation MUST NOT occur.
- **AIOPS-GOV-09 (MUST).** T4 workflows MUST maintain immutable audit logs — inputs, retrieved evidence, model versions, prompts, outputs, reviewer actions — retained to the workflow's regulatory horizon.
- **AIOPS-GOV-10 (MUST).** The AI Operations function MUST retain an independent audit channel (System 3*): the standing right and technical means to sample production outputs from any workflow without the owning team's pre-approval.
- **AIOPS-GOV-11 (MUST).** Every T3+ workflow MUST have an incident-response procedure — severity classification, containment, disclosure mapping, blameless postmortem — and corrective actions MUST become permanent test cases.
- **AIOPS-GOV-12 (MUST/MAY).** Human review rates MUST be set per tolerance class and MAY be reduced only on documented evaluation evidence, recorded in an earned-reduction log.
- **AIOPS-GOV-13 (MUST).** Every T3+ workflow MUST carry an explicit error budget agreed with the business owner, measured continuously.
- **AIOPS-GOV-14 (MUST).** When a workflow's error budget is exhausted, feature work MUST stop and reliability work MUST take priority until the workflow is back inside budget.

Part IV — The Economics

10. AI FinOps: Making AI Economically Legible

AI is the first enterprise capability whose *quality* and *cost* trade against each other call by call. Cloud FinOps managed a cost axis; AI economics manages a cost-quality frontier. This section defines the economic machinery: the unit-cost model, the attribution structure, the routing economics, the vendor portfolio, and the ROI discipline.

10.1 The unit: cost per workflow execution

The sovereign economic metric is the **fully loaded cost per workflow execution** — the AI equivalent of landed cost per unit:

- **Direct model cost** — tokens across every call in the pipeline, including retries, verification passes, and judge calls (verification is a real cost and must be priced, not hidden).
- **Infrastructure cost** — retrieval, vector stores, orchestration compute, logging, and storage of the audit trail, allocated per execution.
- **Asset amortization** — the build and maintenance of prompts, golden sets, and integration code, amortized over execution volume; this term is why low-volume T4 workflows are often uneconomic and should fail qualification.
- **Review labor** — the human review rate × time per review × loaded labor cost. In T3/T4 workflows this is routinely the *largest* term, which is precisely why earned review-rate reduction (Section 6.3) is the function's cleanest ROI lever.
- **Error cost** — the expected downstream cost of the residual error rate. Hard to estimate, essential to include even as a band: it is the term that keeps quality in the economic model and prevents the false economy of routing everything to the cheapest model.

Cost per execution, tracked per workflow per month, alongside the workflow's quality SLO, gives the enterprise the two-axis view every optimization decision needs. The target state is a portfolio dashboard where every workflow is a point on the cost-quality plane, moving down and to the right over time.

10.2 Attribution: showback first, chargeback when mature

Fragmented AI spend (Section 1.3) is reassembled through standard FinOps mechanics: tagging discipline on every API key and workload, per-workflow and per-business-unit cost views, monthly showback to make consumption visible, and chargeback only once measurement is trusted — premature chargeback teaches teams to hide usage, recreating shadow AI inside the official system. The partnership with Finance is structural: Finance owns the ledger; AI Operations owns the denominator (executions, outcomes) that turns spend into unit economics.

10.3 Routing economics and the supplier portfolio

Section 4 established routing as carrier selection; the economic practice is a quarterly **route review**: for each task lane, re-run the lane's eval set against the current model market and move volume when a cheaper option clears the quality bar. The vendor portfolio is managed like a supplier base — concentration limits (no critical lane sole-sourced without a tested fallback), deprecation calendars tracked like contract expiries, price-move

alerting, and negotiation posture built on demonstrated switching ability. The compounding effect is large: model prices per unit of capability have fallen consistently and steeply; an enterprise with routing discipline captures that curve automatically, while an enterprise standardized on one provider captures only what that provider chooses to pass through.

10.4 ROI discipline

Every workflow's return is computed against its intake baseline (Section 7) using the business owner's metric — cycle time, cost per unit, throughput, quality, revenue — never the AI team's proxy metrics. Three rules keep the number honest: returns are claimed only on *measured* deltas against the frozen baseline; benefits and costs are both fully loaded (the review labor and hardening cost count); and "time saved" converts to value only through one of three doors — headcount avoided, capacity redeployed to named work, or volume growth absorbed without hiring. Time saved that walks out the door as slack is not ROI, and reporting it as ROI is how AI programs lose the CFO. The portfolio-level number — total measured return over total fully loaded spend — is the single figure the board sees, and the function lives or dies by its trajectory.

10.5 Normative requirements (AIOPS-ECON)

- **AIOPS-ECON-01 (MUST).** Fully loaded cost per workflow execution MUST be computed monthly for every production workflow, including direct model cost, infrastructure, asset amortization, review labor, and expected error cost.
- **AIOPS-ECON-02 (MUST).** Verification passes, retries, and judge calls MUST be included in direct model cost — priced, never hidden.
- **AIOPS-ECON-03 (MUST).** All AI spend MUST be attributable to a workflow and business unit through tagging discipline on API keys and workloads.
- **AIOPS-ECON-04 (MUST/MAY).** Showback MUST precede chargeback; chargeback MAY be introduced only once measurement is trusted by the teams being charged.
- **AIOPS-ECON-05 (MUST).** Route reviews MUST run at least quarterly per task lane: re-run the lane's evaluation set against the current model market and move volume when a cheaper option clears the quality bar.
- **AIOPS-ECON-06 (MUST NOT).** No critical task lane MAY be sole-sourced without a tested fallback; concentration limits MUST be defined for the provider portfolio.
- **AIOPS-ECON-07 (MUST).** Model deprecation calendars MUST be tracked for every provider and model in a production lane, treated like contract expiries.
- **AIOPS-ECON-08 (MUST).** ROI MUST be computed only against frozen intake baselines using the business owner's metric; Finance MUST co-sign the numerator.

- **AIOPS-ECON-09 (MUST).** "Time saved" MUST convert to claimed return only through one of three doors: headcount avoided, capacity redeployed to named work, or volume growth absorbed without hiring.
- **AIOPS-ECON-10 (MUST).** Benefits and costs MUST both be fully loaded; review labor and hardening cost count against the return.

Part V — The Measurement System

11. The KPI Catalog

A discipline is its metrics. This section expands the KPI set into an operational catalog: definition, formula, and the gaming risk each metric carries — because every metric that matters will be gamed, and the catalog is not complete until the gaming vector is named and countered. Metrics are grouped into four families that answer the four executive questions: *Is it working? Is it used? Is it safe? Is it worth it?*

11.1 Value & economics — "Is it worth it?"

KPI	Definition & formula	Gaming risk / counter
AI ROI (portfolio)	Σ measured business return ÷ Σ fully loaded AI spend, trailing 12 months. Returns only from frozen intake baselines.	Soft "time saved" inflation → the three-door rule (10.4); Finance co-signs the numerator
Cost per AI workflow execution	Fully loaded unit cost per Section 10.1, per workflow, monthly trend.	Hiding review labor or retries → cost model is standardized and audited, not self-reported
Time-to-ROI	Days from intake approval to cumulative return exceeding cumulative cost, per workflow.	Cherry-picking easy workflows → reported alongside portfolio mix by tolerance class
AI spend vs. budget	Actual fully loaded spend vs. plan, with variance decomposition (volume vs. rate vs. mix).	Burying spend in SaaS renewals → inventory completeness is itself a KPI (11.4)
Model utilization	Share of paid capacity (committed contracts, provisioned throughput) actually consumed.	Over-committing for discounts → commitments require route-review evidence

11.2 Operational performance — “Is it working?”

KPI	Definition & formula	Gaming risk / counter
Production deployment rate	Workflows passing the production gate ÷ qualified intakes, and median days pilot→production.	Lowering the gate → gate criteria are versioned; System 3* audits gate evidence
Cycle time reduction	Baseline vs. current end-to-end cycle time per redesigned workflow.	Measuring the step, not the flow → baselines are end-to-end by rule (8.1)
Throughput improvement	Units processed per period per FTE-equivalent vs. baseline.	Quality dumping → always reported paired with the workflow’s quality SLO
Workflow automation %	Share of workflow steps executed by AI within governed pipelines, weighted by step time.	Automating trivial steps → time-weighted, not step-counted
SLO attainment	% of workflows inside their error budget and latency targets this period.	Sandbagging targets → targets set with business owners; changes are logged

11.3 Adoption — “Is it used?”

KPI	Definition & formula	Gaming risk / counter
AI adoption rate	Weekly active users of sanctioned AI workflows ÷ eligible users, per function.	Counting logins, not use → “active” = completed executions, not sessions
Depth of use	Executions per active user per week; distribution, not just mean.	Power-user skew hiding non-adoption → report median and quartiles
Shadow AI reduction	Estimated unsanctioned usage (from network telemetry + surveys) trend, and conversion rate into sanctioned equivalents.	Driving it further underground → amnesty-based conversion program (Section 15)
Customer satisfaction impact	CSAT/NPS delta on AI-touched interactions vs. control.	Selective routing of easy cases → randomized holdouts where feasible

11.4 Governance & risk — “Is it safe?”

KPI	Definition & formula	Gaming risk / counter
Measured error rate	Per T3/T4 workflow: errors per execution against golden-set and sampled-audit evidence, with confidence bounds.	Stale golden sets → set refresh age is tracked; 3* audits sample live traffic

KPI	Definition & formula	Gaming risk / counter
Faithfulness score	% of output claims supported by cited evidence (retrieval workflows).	Citing weak evidence → context precision audited alongside
Human review rate	% of outputs receiving human review, per workflow, vs. its class policy — with the earned-reduction log.	Rubber-stamping → reviewer correction rate is tracked; zero-correction reviewers trigger audit
Governance compliance	% of production workflows fully implementing their class's gate set, evidenced.	Paper compliance → evidence is machine-generated logs, not attestations
Inventory completeness	Estimated share of enterprise AI usage captured in the portfolio inventory.	Definitional shrinkage → survey + telemetry triangulation quarterly
Incident rate & MTTR	T3/T4 incidents per 10k executions; mean time to contain and to resolve.	Reclassifying incidents → severity rubric is fixed; postmortems are 3*-sampled

12. Dashboard Architecture: Three Altitudes

The measurement system reports at three altitudes, each with a different question and cadence, all drawing from the same instrumented source of truth — never from slideware.

- **Board / CEO (quarterly):** one page. Portfolio ROI trend, total fully loaded spend, production workflow count by tolerance class, adoption trend, governance compliance %, and the two or three incidents worth knowing about. The purpose is singular: demonstrate that AI is a *managed capability* with a trajectory, not a collection of experiments.
- **Executive / functional (monthly):** per business unit — workflows live and in pipeline, unit costs and trends, cycle-time and throughput deltas against baseline, adoption depth, review rates, and the S&OP-style view: capability demand vs. deployment capacity, with the prioritization queue visible so that "why isn't my project moving" has a transparent answer.
- **Operational (continuous):** per workflow — SLO and error-budget status, live cost per execution, routing mix, drift indicators, degradation events, incident status. This is the working surface of the AI Operations team itself and the evidence layer every altitude above summarizes.

Design rule: every number on the board page must be traceable, in two clicks, to the operational telemetry that produced it. The moment dashboard numbers are assembled by hand, the function has lost its System 3* integrity — it is now grading its own homework.

12.1 Normative requirements (AIOPS-MEAS)

The requirements below govern the measurement system defined in Sections 11 and 12.

- **AIOPS-MEAS-01 (MUST).** The organization MUST report KPIs across all four families — value & economics, operational performance, adoption, governance & risk — at the cadence defined by the three-altitude system.
- **AIOPS-MEAS-02 (MUST).** Every reported KPI MUST have its gaming vector named and a counter defined before the KPI enters a dashboard.
- **AIOPS-MEAS-03 (MUST).** Adoption MUST be counted as completed executions, never logins or sessions; depth of use MUST be reported as a distribution (median and quartiles), not a mean.
- **AIOPS-MEAS-04 (MUST/MUST NOT).** Every board-level number MUST be traceable to production telemetry; dashboard numbers MUST NOT be hand-assembled.
- **AIOPS-MEAS-05 (MUST).** SLO targets MUST be set jointly with business owners, and target changes MUST be logged.
- **AIOPS-MEAS-06 (MUST/SHOULD).** Reviewer correction rates MUST be tracked per workflow; reviewers with sustained zero-correction rates SHOULD trigger an audit of the review design.

Part VI — Organization, Maturity, Adoption

13. Organizational Design

13.1 Reporting line

The function should report to the **Chief Operating Officer**. The reasoning is structural, not political: AI Operations' accountability is denominated in operational and financial outcomes across functions, and the COO is the executive who already owns cross-functional outcome accountability. Alternate homes work with caveats — a Chief AI Officer (where one exists with real authority), a Chief Digital or Transformation Officer. Two homes reliably fail: reporting into engineering converts the function back into MLOps (artifact accountability displaces outcome accountability), and reporting into IT frames AI as a support service to be ticketed rather than an operating capability to be run. The reporting line is a statement about what kind of thing the enterprise believes AI is.

13.2 Shape: thin hub, embedded spokes

Per the VSM (Section 5), the shape is hub-and-spoke by construction. The hub is deliberately thin — for a mid-size enterprise, often five to ten people: portfolio lead, evaluation/reliability lead, AI FinOps analyst, adoption/change lead, and workflow architects. The spokes are embedded: each major business unit carries workflow owners and champions who live in the business, use its language, and are accountable inside its P&L. The hub owns Systems 2–5 (standards, portfolio control, audit, intelligence, policy); the spokes own System 1 (the workflows). Headcount grows with the *portfolio's tolerance-class mix*, not its raw count — T4 workflows are expensive to govern and should be; T1/T2 workflows should be nearly self-service on the hub's paved road.

13.3 RACI

Decision / activity	AI Ops	Engineering	Business unit	Finance	Legal/Sec
Portfolio prioritization	A/R	C	R	C	
Workflow redesign	A/R	C	R	I	C
System build & integration	C (requirements, gates)	A/R	C	I	C
Production release (gate)	A	R	R (sign-off)	I	C (T4: R)
Error tolerance & SLOs	A/R	C	R (co-sign)	I	C
Model/vendor selection & routing	A/R	C	I	C	
AI budget & unit economics	R	I	C	A/R	I
Governance policy (System 5)	R (drafts, enforces)	C	A (approves)		
Incident response	A/R	R	C	I	C (disclosure: A)
Adoption & enablement	A/R	I	R	I	

13.4 Normative requirements (AIOPS-ORG)

- **AIOPS-ORG-01 (SHOULD/MUST NOT).** The AI Operations function SHOULD report to the Chief Operating Officer; it MUST NOT report into Engineering or IT.

- **AIOPS-ORG-02 (MUST).** The function MUST be structured hub-and-spoke: a central hub owning standards, portfolio control, audit, intelligence, and policy (Systems 2–5), with workflow ownership embedded in the business units (System 1).
- **AIOPS-ORG-03 (MUST).** Decision rights across AI Operations, Engineering, business units, Finance, and Legal MUST be explicitly documented, with exactly one accountable party per decision category.
- **AIOPS-ORG-04 (SHOULD).** Hub headcount SHOULD scale with the portfolio's tolerance-class mix, not its raw workflow count.

14. The Maturity Model

Enterprises need to locate themselves before they can move. Five levels, each defined by what the organization can *truthfully answer*:

Level	Name	Defining condition	Can truthfully answer
L0	Ad hoc	Shadow AI dominant; no inventory; spend invisible; pilots are demos	Nothing — "who is using AI for what?" is unanswerable
L1	Inventoried	Single portfolio inventory exists; spend roughly assembled; tolerance classes assigned	"What AI do we have, who owns it, what does it cost?"
L2	Governed	Lifecycle and gates enforced; golden sets on T3/T4; audit logging; incident process	"What is the error rate, and who approved it?"
L3	Measured	Unit economics live per workflow; ROI computed against frozen baselines; routing reviews running; dashboards at three altitudes	"What does each workflow cost and return, and which lever moves it?"
L4	Optimizing	Error budgets drive work allocation; review rates reduced on evidence; portfolio actively rebalanced; AI capability enters strategic planning as a costed input	"Where should the next dollar of AI investment go, and why?"

Two properties of the model matter. First, levels are *sequential in evidence*: an enterprise cannot honestly claim L3 economics over an L0 inventory, and attempts to skip levels produce dashboards no one trusts. Second, the model doubles as the function's own roadmap — the first-365-days playbook in Section 16 is, structurally, the path from L0 to early L3.

14.1 Normative requirements (AIOPS-MAT)

- **AIOPS-MAT-01 (MUST).** Maturity levels MUST be claimed sequentially in evidence; an organization MUST NOT claim a level without satisfying the evidence requirements of every level below it.

- **AIOPS-MAT-02 (SHOULD).** The organization SHOULD self-assess against the maturity model at least annually, reporting the profile per dimension rather than an average.

15. Adoption and the Shadow AI Conversion

Deployment without adoption is inventory. The adoption discipline has three components, and one hard-won principle governs all of them: *adoption is earned by removing friction, never commanded by policy.*

- **The paved road.** The sanctioned path must be genuinely better than the shadow path — faster access, better models, pre-approved data handling, and self-service onboarding for T1/T2 use. People took the shadow path because it was the best available; the conversion strategy is to change which path that is. An amnesty-based conversion program — inventory what people actually use, sanction and instrument the valuable patterns, retire only the genuinely dangerous ones — turns the shadow map into the demand roadmap.
- **Role-level enablement, not tool training.** Training on "how to use the AI tool" decays in weeks. Enablement that sticks is workflow-specific: for this role, in this process, here is what the machine now does, here is your new role (review, exception handling, judgment), here is what you can override and how your corrections improve the system. The reviewer role in particular (Section 8.1) is designed, staffed, and respected — it is quality engineering, not leftover work.
- **Executive enablement.** Leaders who do not use the capability cannot govern it. A standing executive fluency program — hands-on, on real company workflows, quarterly — is disproportionately valuable: it converts governance conversations from theology to engineering, because the executives have touched the thing.

Adoption metrics (Section 11.3) close the loop: weekly active usage by function, depth distribution, and shadow-AI trend are reviewed monthly with each business unit, and persistent non-adoption is treated as a *design signal* — the workflow, the enablement, or the incentive structure is wrong — before it is treated as a compliance problem.

15.1 Normative requirements (AIOPS-ADOPT)

- **AIOPS-ADOPT-01 (MUST).** The organization MUST maintain a shadow-AI conversion program; discovery MUST be amnesty-based, and shadow usage MUST be treated as demand signal before it is treated as violation.
- **AIOPS-ADOPT-02 (MUST).** Enablement MUST be role- and workflow-specific, including explicit training on the redesigned human role (review, override, exception handling).
- **AIOPS-ADOPT-03 (SHOULD).** A standing executive fluency program SHOULD run at least quarterly, hands-on, on real company workflows.

- **AIOPS-ADOPT-04 (MUST).** Persistent non-adoption MUST be investigated as a design signal before it is escalated as a compliance problem.

Part VII — Execution

16. The First 365 Days

This section is non-normative implementation guidance.

The playbook below is the maturity model in motion: L0 to early L3 in one year, sequenced so that every phase funds the credibility of the next. The single organizing principle: **show a measured win early, and never publish a number the function cannot defend under audit.**

Days 1–30: See the whole board

- Build the portfolio inventory — every initiative, tool, embedded SaaS AI feature, API key, and known shadow pattern; owner, workflow, spend, tolerance class. Triangulate telemetry, expense data, and an amnesty-framed survey.
- Assemble the first fully loaded spend picture with Finance — imperfect is fine; visible is the point.
- Classify the top 20 workflows by tolerance class and value density; identify the two or three candidates for the first governed wins.
- Stand up the governance skeleton: draft the System 5 constitution (risk appetite, tolerance classes, prohibited uses) and get executive sign-off on the *framework*, not yet the details.
- Deliverable: the inventory readout to the executive team — almost always the first time anyone has seen the whole board. This artifact alone typically justifies the role.

Days 31–90: First governed wins

- Select 2–3 workflows — at least one T3 to prove the governance machinery, none T4 yet — with committed business owners and clean baselines.
- Freeze baselines; build golden sets with the owners; define SLOs and review policy; instrument cost per execution from day one.
- Run the pilots through the full lifecycle gates — deliberately, to debug the gates themselves, not just the workflows.

- Launch the intake process publicly: any function can propose; criteria are published; the queue is visible.
- Stand up the operational dashboard (altitude three) for the pilot workflows.

Days 91–180: Production and proof

- Take the first workflows through the production gate: hardened, gated, logged, reviewed, adopted — and publish the first measured deltas against frozen baselines.
- First quarterly route review: eval-driven model selection per lane; capture the first routing savings (usually the fastest hard-dollar win in the whole playbook).
- Launch the shadow-AI conversion program on the paved-road model (Section 15).
- Begin monthly executive dashboards (altitude two); agree the board page format with the CEO/CFO.
- First independent 3* audit of a production workflow — establish the norm early that audit is routine, not accusation.

Days 181–365: Scale the system, not the headcount

- Expand the portfolio through the now-proven lifecycle; admit the first T4 workflow only when the T3 evidence base has earned it.
- Implement showback with Finance; publish unit-cost trends per workflow; begin earned review-rate reductions where eval evidence supports them.
- Codify the paved road so T1/T2 use is self-service against hub standards — the hub's leverage is standards, not throughput.
- First annual System 5 review: tolerance classes, risk appetite, and prohibited-use list revisited with Legal and the executive team against a year of evidence.
- Year-one board readout: portfolio ROI with Finance co-sign, spend trajectory, governance compliance, adoption trend, maturity self-assessment against Section 14 — and the year-two plan priced as an investment case, not a budget ask.

17. The Role: Head of AI Operations

Everything preceding compresses into a role specification. This section is written to be lifted directly into a req.

17.1 Position summary

The Head of AI Operations is accountable for transforming AI from isolated tools and pilots into a scalable, governed, economically legible enterprise operating capability. The role owns the AI portfolio lifecycle, the governance and evaluation regime, the economics,

adoption, and the measurement system — and is judged on business outcomes: cost, cycle time, throughput, quality, revenue, and audited risk posture. Engineering builds AI; this role ensures AI improves the operations and economics of the enterprise.

17.2 Mandate (first year)

- Build the complete enterprise AI inventory and the fully loaded spend picture (Days 1–30).
- Establish the operating model: lifecycle, gates, intake, and the governance constitution with executive sign-off.
- Deliver 2–3 governed production workflows with measured deltas against frozen baselines.
- Stand up the evaluation regime (golden sets, faithfulness measurement, per-stage error accounting) on all T3+ workflows.
- Implement unit economics and showback with Finance; run the first routing reviews and capture the savings.
- Deliver the three-altitude dashboard system, through to the board page.
- Reach maturity level L2 fully and early L3 (Section 14) by day 365, evidenced.

17.3 Competency profile

Competency	What it looks like in practice
Operations leadership (10–15+ yrs)	Has owned cross-functional outcome accountability at scale — supply chain, customer operations, risk operations, transformation. Has held a number that mattered.
AI systems literacy	Can read an architecture, interrogate an eval, and hold the workflow-vs-agent trade-off with engineers — LLMs, RAG, orchestration, structured outputs, evaluation methods. Builder-level fluency is a strong plus; researcher depth is not required.
Reliability & governance instinct	Thinks in error rates, control limits, audit trails, and disclosure — treats “how do you know it’s right?” as the first question, not a compliance afterthought.
Financial fluency	Builds unit-cost models, defends an ROI methodology to a CFO, manages a vendor portfolio and its contracts.
Translation	Speaks risk officer, CFO, engineer, and frontline operator — natively, in the same meeting. This is the load-bearing skill of the role.
Change leadership	Has moved real organizations through operating-model change; understands adoption as design, not decree.

17.4 Ideal background — and why it is an operator

The historical pattern (Section 2.5) predicts, and the competency profile confirms: the strongest candidates are **operations leaders who have built AI systems**, not AI engineers who have observed operations. The hard problems of the role — accountability design, cross-functional coordination, unit economics, adoption, governance that survives audit — are twenty-year operations problems wearing a new technical surface. The technical fluency is trainable and verifiable; the operator judgment is the scarce input. The profile in one line: *fifteen years of owning operational outcomes, plus demonstrated hands-on fluency with modern AI systems, plus a coherent reliability philosophy.* Candidates who have personally built governed, evaluated, multi-model AI pipelines — even at small scale — should be weighted heavily: they have run the entire discipline in miniature and know where it bites.

17.5 Reporting, team, path

Reports to the COO (Section 13.1). Starts with a thin hub (Section 13.2) and embedded champions. Career path: Head of AI Operations → VP of AI Operations → Chief AI Officer or COO — the role is, structurally, COO training for the AI-era enterprise, because it rehearses the same portfolio: cross-functional outcomes, economics, risk, and change.

17.6 Normative requirements (AIOPS-ROLE)

- **AIOPS-ROLE-01 (MUST).** The enterprise MUST designate a single accountable executive for AI Operations with a cross-functional mandate spanning engineering, finance, risk, and the business lines.
- **AIOPS-ROLE-02 (MUST).** The role MUST be judged on business outcomes — cycle time, cost per unit, throughput, quality, revenue, audited risk posture — not on technology outputs.

18. Objections and Rebuttals

A new discipline earns its place by answering its skeptics directly. The six standard objections:

18.1 “Isn’t this the Chief AI Officer’s job?”

Where a CAIO exists, the roles are complements, not rivals: the CAIO owns *what AI should do for the enterprise* — strategy, ambition, external posture; AI Operations owns *whether it actually does it, at what cost, at what risk*. Strategy without an operating discipline is the pilot graveyard with better slides. In practice most CAIO offices are small strategy functions with no lifecycle machinery, no unit economics, and no audit channel — exactly the gap this specification fills. Where no CAIO exists, AI Operations reporting to the COO covers the operating mandate, and the strategy question resolves upward normally.

18.2 “Isn’t this just MLOps / platform engineering?”

Section 2.4 in one line: MLOps’ customer is the model; AI Operations’ customer is the workflow and the P&L. MLOps cannot answer “what did this workflow return against its baseline, who approved its error rate, and why is Finance confident in the number?” — not because MLOps is deficient, but because those are not engineering questions. The functions are adjacent layers, and both are necessary.

18.3 “Aren’t we too early? The technology is still changing.”

Technology volatility is the argument *for* the function, not against it. Precisely because models, prices, and capabilities shift quarterly, the enterprise needs the layer that absorbs volatility — routing abstraction, portable eval sets, supplier management, lifecycle discipline — so the business does not have to re-platform every time the market moves. FinOps was not premature in 2014 because cloud pricing kept changing; the changing was the point. Waiting for AI to “settle” means running the chaos phase (Section 2) longer than competitors who did not wait — and the chaos phase compounds: shadow AI entrenches, spend fragments further, and un-instrumented workflows accumulate baselines that can never be reconstructed.

18.4 “Engineering / IT already owns this.”

Then the org chart can answer four questions: Who owns the single AI inventory? Who signs the error tolerance for each customer-facing workflow? Who computes fully loaded cost per execution? Who is accountable when adoption fails? In enterprises where “engineering owns AI,” the honest answers are: no one, no one, no one, and everyone — which is the operations gap restated. Engineering owning AI *systems* is correct and unchanged by this specification. Owning systems is not owning outcomes.

18.5 “Can’t we just buy this from consultants?”

Consultants can accelerate the standing-up — the inventory sprint, the operating-model draft, the first gates. They cannot *be* the function, for the same reason no enterprise outsources its supply chain organization while keeping its margins: the discipline’s value compounds through continuous ownership — baselines, eval sets, supplier leverage, and institutional trust in the numbers are assets that accrue to whoever holds them daily. Rent the setup; own the capability.

18.6 “Won’t AI eventually manage itself?”

Some operational subtasks will progressively automate — routing optimization, drift detection, even portions of evaluation. But the core of the function is the part that cannot delegate to the managed system itself: setting error tolerances the enterprise will stand behind, owning the audit channel that inspects the machine independently, negotiating

with suppliers, allocating capital, and being the human a regulator, a board, and a customer hold accountable. A control system graded by itself is not a control system (Section 5.2). The function's tools will change; its accountability cannot.

19. Conclusion: The Eighteen-Month Window

This specification has argued a single thesis from four directions. Historically: every enterprise technology wave has been made valuable by an operations discipline arriving three to five years behind the technology, and AI is at that point on the arc now. Structurally: an AI portfolio is a supply chain — suppliers, routing, QC, unit cost, compliance — and yields to the same management science. Doctrinally: reliability is a designed, measured, class-by-class property, purchased with governance and evidenced by evaluation, not an emergent gift of better models. Economically: the function pays for itself through routing discipline, review-rate reduction, shelfware elimination, and the conversion of shadow demand into governed capability — before counting a single workflow's cycle-time gains.

The window claim is the practical conclusion. Disciplines confer most of their advantage on early institutionalizers: the enterprises that ran FinOps in 2016 spent the next five years compounding cost advantages their competitors could not see, let alone match. AI Operations is at the same moment. The enterprises that stand up the function in the next eighteen months will spend that period building inventories, baselines, eval sets, supplier leverage, and audit credibility — assets that cannot be bought retroactively, because they are made of *time and measurement*. Their competitors will spend the same period discovering, pilot by dead pilot, that the missing ingredient was never the model.

Engineering builds AI. AI Operations makes it pay, keeps it honest, and proves both. Every company will eventually have this function. The only open question is whether it is built deliberately, now — or assembled in retrospect, after the audit, the write-off, or the competitor's margin makes the case unanswerable.

Appendices

Appendix A — Glossary

Term	Definition
AI Operations	The discipline of deploying, governing, measuring, and optimizing AI as an enterprise operating capability to produce measurable business outcomes.

Term	Definition
AI-assisted workflow	A business process in which one or more steps are executed by AI under defined governance; the unit of management of AI Operations.
Tolerance class (T1–T4)	Classification of a workflow by the cost of an undetected error; dictates minimum control architecture (Section 6).
Governed workflow / compound AI system	AI system whose control flow lives in deterministic code; models perform bounded tasks and code decides sequencing, verification, and release.
Agent	AI system in which the model holds the control loop — choosing tools, actions, and termination at runtime.
Golden set	Versioned, labeled test set with known correct answers, used to measure a workflow's error rate.
Faithfulness / groundedness	The degree to which output claims are supported by retrieved evidence; enforced structurally and measured statistically.
LLM-as-judge	Using a model to grade outputs against a rubric; a scalable but bias-prone instrument, calibrated against human labels.
Error budget	The explicitly chosen allowable failure rate for a workflow; consumption governs the balance between feature and reliability work.
Degradation disclosure	The requirement that any fallback or skipped verification be visibly flagged on the output and in the audit log.
Model routing	Policy-driven assignment of task lanes to models based on eval-verified quality at lowest cost and acceptable latency.
Cost per workflow execution	Fully loaded unit cost: model tokens, infrastructure, asset amortization, review labor, and expected error cost.
Shadow AI	Unsanctioned, uninstrumented AI usage; treated as demand signal and converted via the paved road.
Paved road	The sanctioned path made genuinely superior to shadow alternatives — the core adoption mechanism.
System 3* (audit channel)	Independent, sporadic direct inspection of production outputs, bypassing self-reporting (from Beer's VSM).
Showback / chargeback	Cost visibility mechanisms: reporting consumption to owners (showback) vs. billing them (chargeback).

Appendix B — The Board Page (template)

The quarterly one-page format, per Section 12:

- **Headline:** portfolio ROI (trailing 12 months, Finance co-signed) and trend arrow.

- **Economics:** total fully loaded AI spend vs. plan; cost-per-execution trend for the top five workflows.
- **Portfolio:** production workflows by tolerance class; pipeline count; median pilot-to-production days.
- **Adoption:** weekly active usage by function (trend); shadow-AI trend.
- **Governance:** compliance % against gate sets; measured error rates on T4 workflows with confidence bounds; incidents this quarter with status.
- **Decision requests:** the one to three items requiring board or CEO action, priced.

Appendix C — Maturity Self-Assessment

Score each dimension 0–4 against the level definitions in Section 14; the profile, not the average, is the diagnosis — a spiky profile (L3 economics, L0 governance) is itself a risk finding.

- **Inventory:** Is there one complete list of all AI usage, owned and current?
- **Lifecycle:** Do public stage gates exist, and does anything actually fail them?
- **Evaluation:** Do golden sets exist for every T3/T4 workflow, and does an independent channel sample live traffic?
- **Economics:** Is fully loaded cost per execution computed per workflow, and does Finance co-sign the ROI numerator?
- **Adoption:** Is usage measured by completed executions, and is shadow AI trending down via conversion rather than prohibition?
- **Governance:** Can the enterprise answer, for any T4 output, "what evidence supports it and who approved the process" — in minutes, from logs?

Appendix D — Conformance Checklist

This checklist maps all 57 normative requirements to the maturity level (Section 14) at which each must first be satisfied. Levels are cumulative: claiming a level requires satisfying every requirement at that level and all levels below it (AIOPS-MAT-01). The Evidence column is completed by the organization with a pointer to the artifact that demonstrates satisfaction — an inventory export, a gate record, a dashboard, a log.

D.1 Level L1 — Inventoried (8 requirements)

ID	Requirement (summary)	Evidence
AIOPS-PORT-01	Single complete AI inventory with owner, class, spend, stage	

ID	Requirement (summary)	Evidence
AIOPS-PORT-02	Exactly one accountable owner per inventory item	
AIOPS-PORT-03	Tolerance class assigned before pilot	
AIOPS-ORG-01	Reports to COO; never Engineering or IT	
AIOPS-ROLE-01	Single accountable executive, cross-functional mandate	
AIOPS-ROLE-02	Role judged on business outcomes, not technology outputs	
AIOPS-MAT-01	Maturity claimed sequentially in evidence	
AIOPS-MAT-02	Annual self-assessment, profile per dimension	

D.2 Level L2 — Governed (26 requirements)

ID	Requirement (summary)	Evidence
AIOPS-PORT-04	Standard lifecycle with published, versioned gate criteria	
AIOPS-PORT-05	Intake names metric and freezes baseline; no baseline, no entry	
AIOPS-PORT-06	Pilots time-boxed with pre-committed kill criteria	
AIOPS-PORT-07	Pilots evaluated against golden set built before pilot	
AIOPS-PORT-08	Production gate: measured error rate, unit cost, owner sign-off	
AIOPS-PORT-09	Documented retirement plan per production workflow	
AIOPS-WFR-01	Workflows mapped as actually run before AI introduced	
AIOPS-WFR-02	Baselines measured end-to-end, not per step	
AIOPS-WFR-03	Human role explicitly designed: review, override, evidence, feedback	
AIOPS-GOV-01	Governance constitution with executive and legal sign-off	
AIOPS-GOV-02	Versioned golden set per T3+ workflow; refresh age tracked	
AIOPS-GOV-03	Error claims carry confidence bounds bounded by set size	
AIOPS-GOV-04	Structural groundedness on T3+ retrieval: uncited claims stripped	
AIOPS-GOV-05	Generator/verifier separation on T4 workflows	
AIOPS-GOV-06	LLM judges calibrated against human labels; disagreement tracked	
AIOPS-GOV-07	Per-stage error accounting in multi-stage pipelines	
AIOPS-GOV-08	Mandatory degradation disclosure; no silent degradation	
AIOPS-GOV-09	Immutable audit logs on T4, retained to regulatory horizon	

ID	Requirement (summary)	Evidence
AIOPS-GOV-10	Independent audit channel (System 3*) with standing access	
AIOPS-GOV-11	Incident procedure per T3+ workflow; corrective actions become tests	
AIOPS-GOV-12	Review rates set per class; reductions earned and logged	
AIOPS-GOV-13	Explicit error budget per T3+ workflow, measured continuously	
AIOPS-MEAS-05	SLO targets set with business owners; changes logged	
AIOPS-ORG-02	Hub-and-spoke structure: central Systems 2–5, embedded System 1	
AIOPS-ORG-03	Decision rights documented; one accountable party per category	
AIOPS-ADOPT-02	Role- and workflow-specific enablement including reviewer role	

D.3 Level L3 — Measured (20 requirements)

ID	Requirement (summary)	Evidence
AIOPS-PORT-10	Portfolio prioritization on a fixed cadence	
AIOPS-WFR-04	Redesign targets the constraint; gains measured end-to-end	
AIOPS-ECON-01	Fully loaded cost per execution, monthly, per workflow	
AIOPS-ECON-02	Verification, retries, judge calls priced into direct cost	
AIOPS-ECON-03	All spend attributable to workflow and business unit	
AIOPS-ECON-04	Showback before chargeback; chargeback only when trusted	
AIOPS-ECON-05	Quarterly route reviews per task lane, eval-driven	
AIOPS-ECON-06	No sole-sourced critical lane without tested fallback	
AIOPS-ECON-07	Deprecation calendars tracked like contract expiries	
AIOPS-ECON-08	ROI only against frozen baselines; Finance co-signs numerator	
AIOPS-ECON-09	Time saved converts only through the three doors	
AIOPS-ECON-10	Benefits and costs both fully loaded	
AIOPS-MEAS-01	KPIs reported across all four families at defined cadence	
AIOPS-MEAS-02	Every KPI has its gaming vector named and countered	
AIOPS-MEAS-03	Adoption = completed executions; depth as distribution	
AIOPS-MEAS-04	Board numbers traceable to telemetry; never hand-assembled	
AIOPS-MEAS-06	Reviewer correction rates tracked; zero-correction triggers audit	

ID	Requirement (summary)	Evidence
AIOPS-ADOPT-01	Amnesty-based shadow-AI conversion program running	
AIOPS-ADOPT-03	Standing quarterly executive fluency program	
AIOPS-ADOPT-04	Non-adoption investigated as design signal first	

D.4 Level L4 — Optimizing (3 requirements)

ID	Requirement (summary)	Evidence
AIOPS-GOV-14	Exhausted error budget stops feature work until back in budget	
AIOPS-WFR-05	Human corrections feed evaluation sets as test cases	
AIOPS-ORG-04	Hub headcount scales with tolerance-class mix	

Appendix E — Workflow Intake Form (template)

Per Section 7.1 and AIOPS-PORT-05: no baseline, no entry.

- **Workflow name and description:**
- **Proposing function / business unit:**
- **Accountable owner** (one name — AIOPS-PORT-02):
- **Business metric this workflow should move** (cycle time, cost per unit, throughput, quality, revenue):
- **Frozen baseline:** current value of that metric, how it was measured, measurement date, and who signed it:
- **Estimated volume** (executions per week/month):
- **Proposed tolerance class (T1–T4)** and justification — state the cost of an undetected error (use Appendix F):
- **Data touched** and its classification (does this workflow have the right to see this data?):
- **Known shadow-AI usage** in this workflow today (amnesty basis — demand signal, not violation):
- **Requested timeline and any hard deadlines:**

Appendix F — Tolerance-Class Rubric (template)

Classify by answering the questions in order; the first “yes” from the top sets the class. When in doubt between two classes, classify up — declassification is earned later with evaluation evidence.

- **Is the output regulatory, financial, legal, contractual, or safety-relevant — or does it leave the company under the company's name without human sign-off? → T4.** Minimum architecture: deterministic control flow, generator/verifier separation, citation-level traceability, mandatory degradation disclosure, human sign-off gates, immutable audit log.
- **Is the output customer-visible, does it drive an operational decision, or is an undetected error expensive to unwind? → T3.** Minimum architecture: governed pipeline, verification stage, human review on sampled or flagged output, full tracing, error budget.
- **Would an undetected error cause rework or minor confusion, but a human would likely catch it downstream? → T2.** Minimum architecture: structured outputs, spot-check evals, fallback paths.
- **Is the output a draft, an idea, or an internal search result that a human will inevitably rework? → T1.** Agentic patterns acceptable; light logging.

Appendix G — Production Gate Checklist (template)

The gate record for AIOPS-PORT-08. Every item is checked with a pointer to evidence before a T3+ workflow enters production; the completed record is retained in the audit log.

- **Evaluation:** measured error rate against the golden set, with confidence bounds and set size stated (AIOPS-GOV-02, GOV-03)
- **Input gates:** schema validation, data-classification check, prompt-injection screening on untrusted content
- **Process gates:** generator/verifier separation implemented (T4), bounded retries with logged fallbacks
- **Output gates:** uncited claims stripped or blocked (retrieval workflows), degradation disclosure wired and tested (AIOPS-GOV-04, GOV-08)
- **Audit:** immutable logging of inputs, evidence, model versions, prompts, outputs, reviewer actions; retention period set to regulatory horizon (AIOPS-GOV-09)
- **Error budget:** target agreed and signed by the business owner (AIOPS-GOV-13)
- **Review policy:** human review rate set per tolerance class; reviewer sees the evidence trail, not just the answer (AIOPS-GOV-12)
- **Economics:** cost per execution instrumented from day one (AIOPS-ECON-01)
- **Incident response:** severity rubric, containment plan, and disclosure mapping in place (AIOPS-GOV-11)

- **Retirement plan:** replacement path, eval-set transfer, switching cost documented (AIOPS-PORT-09)
- **Adoption:** business-owner sign-off on the adoption plan; enablement scheduled (AIOPS-PORT-08, ADOPT-02)

Appendix H — References

Externally sourced claims in this specification trace to the references below, keyed to the section where each is used. Frameworks original to this specification — the supply-chain isomorphism, the VSM instantiation for enterprise AI, the T1–T4 tolerance-class doctrine, and the AIOPS requirement set — carry no external citation because they originate here.

- Gartner, *Gartner Predicts 30% of Generative AI Projects Will Be Abandoned After Proof of Concept by End of 2025*, press release, July 2024. (Section 1.1)
- MIT NANDA initiative, *The GenAI Divide: State of AI in Business 2025*, 2025. Note: the study's methodology has been publicly debated; it is cited here for the shape of the finding, consistent with Section 1.1's framing. (Sections 1.1, 1.2)
- Central Computer and Telecommunications Agency (UK), *IT Infrastructure Library (ITIL)*, first published 1989. (Section 2.1)
- Beyer, B., Jones, C., Petoff, J., Murphy, N. R. (eds.), *Site Reliability Engineering: How Google Runs Production Systems*, O'Reilly Media, 2016. (Sections 2.2, 6)
- FinOps Foundation, established 2019 under the Linux Foundation. (Sections 2.3, 10)
- Board of Governors of the Federal Reserve System and Office of the Comptroller of the Currency, *SR 11-7: Guidance on Model Risk Management*, April 2011. (Section 9.3)
- National Institute of Standards and Technology, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, January 2023. (Section 9.3)
- European Union, *Regulation (EU) 2024/1689 (Artificial Intelligence Act)*, entered into force August 2024. (Section 9.3)
- Beer, Stafford, *Brain of the Firm*, 1972; *The Heart of Enterprise*, 1979. (Section 5)
- Internet Engineering Task Force, *RFC 2119: Key words for use in RFCs to Indicate Requirement Levels*, 1997. (Section 3.5)

— End of specification —

AI Operations Specification v1.0 ·