

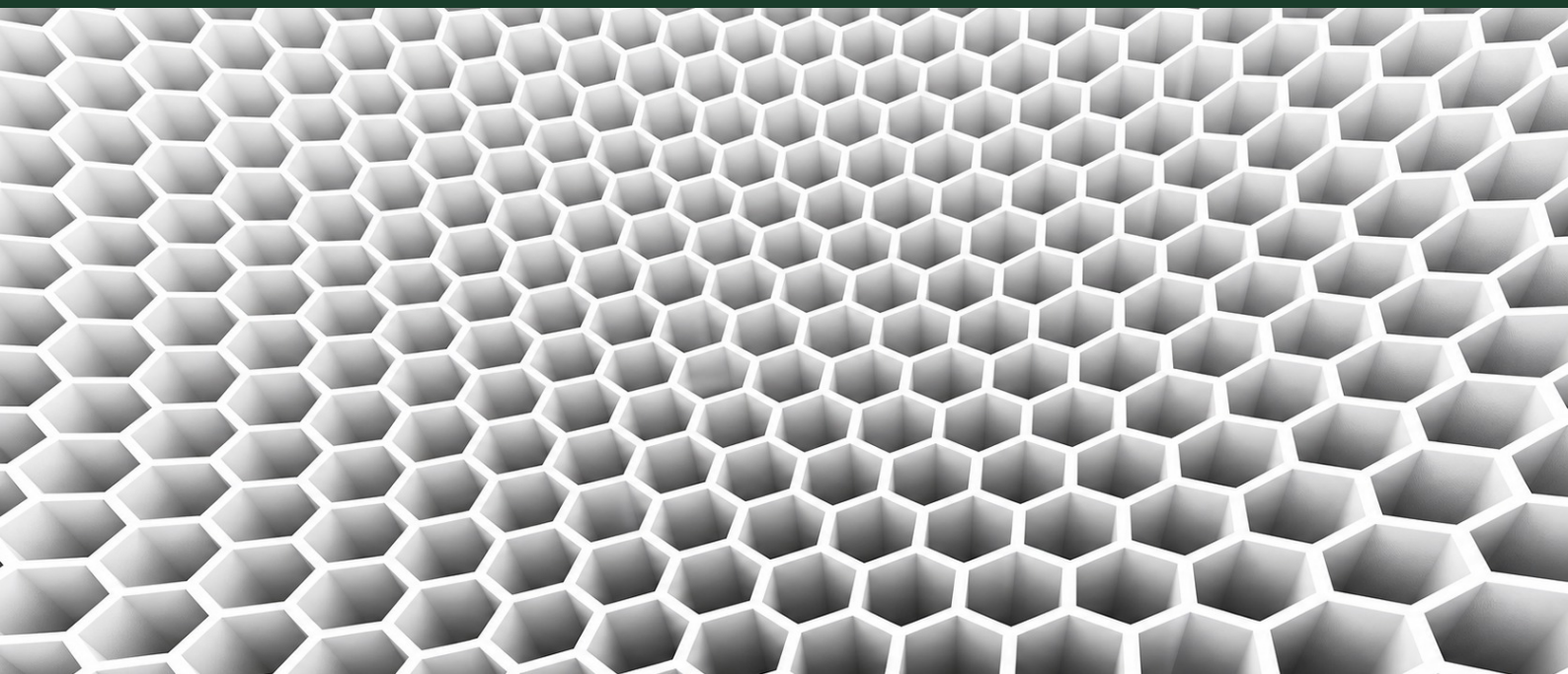
BEYOND SIGN-OFF

Who Signs Off When the Agent Decides? — Accountability Design for Autonomous Delivery

On Reversibility, Blast Radius, and the Accountability Architecture That Delivery Governance Now Requires

Giovanni Leonardi

January 2026



Contents

1. Executive Summary	3
2. The Moment the Question Changed	3
3. What We Built Governance to Do	4
4. The Two Positions That Both Fail	4
5. Two Axes, Four Envelopes	5
6. Owning the Boundary, Not the Action	7
7. Assurance by Sampling	7
8. The Governance We Have Not Yet Imagined	8

Who Signs Off When the Agent Decides? — Accountability Design for Autonomous Delivery

Programme · No. 276
January 2026

“The senior manager who signs off on forty changes in a batch without reading any of them — a practice common enough that most delivery professionals will recognise it without further description — is exercising a fiction of accountability.”

1. Executive Summary

Delivery governance was built on an assumption so deep it rarely needs stating: at every point where a decision matters, a person decides. AI agents break this assumption — not by defying governance, but by acting in the spaces between its checkpoints, at volumes no human review can scale to. The instinctive responses — universal human approval or blind delegation — both fail, one by deleting the value agents provide and the other by detaching accountability from action. This essay argues for a more granular instrument: classifying every decision an agent might make by its reversibility and blast radius, defining an autonomy envelope for each class, replacing universal review with sampling-based assurance, and assigning a named human owner to every envelope — accountable not for each action the agent takes, but for the design and maintenance of the boundary within which it acts. The result is not a retreat from governance but a reconstruction of it, fitted to a delivery chain that no longer waits for human approval at every junction.

2. The Moment the Question Changed

The programme board convenes on a Tuesday morning, as it does every fortnight, and the delivery report is, by any standard measure, encouraging. The RAG status is green across three of four workstreams. Velocity is up. Defect rates are within tolerance. The integration environment is stable. And yet the programme director pauses on a line buried in the release notes: fourteen configuration changes to the data-routing layer, executed overnight, each logged as “agent-initiated, within policy.” She asks the question that will preoccupy the room for the next forty minutes: who approved these?

The answer, it turns out, is nobody — and everybody. The changes were within the parameters the platform team had set for the delivery agent. Each was individually low-risk, individually reversible, individually consistent with the architectural guardrails. The agent did what it was configured to do. But the aggregate effect — a material shift in how data flows between two regulated systems — was something no human had reviewed as a whole. The governance framework that had served this programme for eighteen months had no category for what had just happened.

This is becoming a familiar scene across delivery organisations. Not a disaster — the changes were sound, as it happens — but a moment of institutional vertigo. The controls were not breached. They were designed for a world in which every consequential action had a human author, and that world is quietly ending.

3. What We Built Governance to Do

Delivery governance, as we have practised it for two decades, rests on a structural assumption: at every point where a decision matters, a person decides. Stage gates have human gatekeepers. Change advisory boards have human members. Approval workflows route to named individuals. The entire apparatus — from exception reports to definitions of done — presupposes that between intention and action there is a human judgement, and that accountability flows from the person who exercised it.

This was never merely a procedural preference. It was an accountability architecture. When something goes wrong in a human-governed delivery chain, the diagnostic question is straightforward: who decided? The answer may be unsatisfying — the wrong person, with the wrong information, under the wrong pressure — but the question itself is tractable. Governance exists to make it tractable.

What agents introduce is not disobedience but a new category of action. An agent that drafts a deployment script, validates it against policy, and executes it at three in the morning has not circumvented the change process. It has acted in a domain the change process never imagined would need governing. The approvals were designed for human-paced, human-authored changes arriving at human-readable intervals. The agent's output is none of these things.

The pattern I have observed across several programmes adopting agentic tooling in their delivery chains is consistent: the governance framework does not fail spectacularly. It simply has nothing to say. The decisions are real, the consequences are real, but the framework's unit of analysis — the human decision-maker — is absent. And so the question "who approved this?" receives an answer that satisfies no one: the system was configured correctly.

4. The Two Positions That Both Fail

When this gap surfaces, the organisational instinct is to reach for one of two familiar positions, each with a satisfying logic and a fatal flaw.

The first is universal human review: every agent action of consequence requires approval before execution. This preserves the accountability chain everyone understands. But it also destroys most of the value that agents provide. An agent that must wait for human approval on each action is, in operational terms, a suggestion engine with infrastructure overhead. The latency alone is prohibitive in a continuous delivery environment where pipeline stages complete in minutes. More fundamentally, the volume overwhelms. When an agent generates forty low-level decisions in the time a human makes one, universal review becomes either a bottleneck that defeats the purpose or a rubber stamp that provides the theatre of control without the substance. We have seen this pattern before — in organisations where change advisory boards became approval factories, processing hundreds of standard changes per week with diminishing scrutiny per item. The control was formally present and practically absent.

The second position is trust-and-verify: let agents act freely and audit after the fact. This recognises that most agent actions will be correct and that speed matters. But it detaches accountability from the moment of action in a way governance cannot absorb. Post-hoc review can catch errors; it cannot prevent the ones that matter most. And it offers no answer to the programme director's question. When the audit reveals that an agent made a poor decision at two in the morning, the finding is useful for learning but useless for accountability. Nobody decided. Nobody was even awake.

Both positions fail because they treat human involvement as a single dial to be turned up or down. The question is not how much human oversight, but where, and of what kind, and at what frequency. The answer requires a more granular instrument than either position provides.

5. Two Axes, Four Envelopes

The framework that has proved most useful in practice begins not with the agent but with the decision. Specifically, with two properties of every decision an agent might make in a delivery chain: how reversible is it, and what is its blast radius?

Reversibility is the more intuitive axis. Some actions can be undone cheaply — a configuration change to a feature flag, a draft pull request, a suggested reallocation of testing effort. Others cannot — a production data migration, a contractual commitment, a communication to a regulator. The cost of reversal is not binary; it runs from trivial (a one-line rollback) through expensive (a multi-day remediation) to effectively irreversible (data loss, reputational harm, regulatory consequence).

Blast radius is the less obvious but equally critical axis. It captures how many systems, teams, stakeholders, or end users are affected if the decision is wrong. A configuration change to a single internal tool has a small blast radius even if it is hard to reverse. A change to a shared API contract has a large blast radius even if it is technically simple to roll back, because downstream consumers may already have adapted to the previous state.

These two axes produce a decision space that, in the programmes where I have seen this applied, clusters usefully into four classes. Those classes are the foundation of what I will call the autonomy envelope — the boundary within which an agent may act, and the

governance regime that applies at each level.

	Small Blast Radius	Large Blast Radius
Easily Reversible	Class 1: Full autonomy	Class 2: Act and sample
Hard to Reverse	Class 3: Propose and wait	Class 4: Human decides

Class 1 — Full Autonomy (reversible, contained). The agent acts without prior approval. Actions are logged but not individually reviewed. In a delivery chain, this covers work such as formatting code to standards, updating local development environments, triaging low-severity defects into backlog queues, generating draft test cases. The human obligation here is not to approve but to monitor the envelope itself — to verify periodically that the classification remains correct and that the agent's behaviour within it stays within expected bounds.

Class 2 — Act and Sample (reversible, broad reach). The agent acts, but a defined proportion of actions are reviewed after the fact, selected by random sampling or heuristic triggers. In practice: deploying feature-flagged changes to a staging environment, sending automated status updates to stakeholders, adjusting sprint-level task priorities within approved parameters. The sampling rate is the governance lever — calibrated upward when the agent is new or the domain unfamiliar, relaxed as confidence builds. The critical discipline is that sampling must be genuinely random or trigger-based, never cherry-picked, and that exceptions surfaced through sampling trigger a review of the envelope's boundaries, not merely a correction of the individual action.

Class 3 — Propose and Wait (irreversible, contained). The agent prepares the decision; a human must approve before execution. In delivery terms: proposing a database schema migration, drafting a change to a contractual deliverable, recommending the de-scoping of a requirement. The agent's value here is in preparation — analysis, option generation, risk assessment — but the commitment is human. The governance question is not whether human approval is needed (it is) but how quickly it must come. A propose-and-wait action that blocks a delivery pipeline for three days because the approver is on leave is a governance failure of a different kind, and the envelope design must account for it: designated alternates, time-bound escalation, and clear defaults when approval is not forthcoming.

Class 4 — Human Decides (irreversible, broad reach). The agent is an advisor; the human decides and acts. Committing to a revised delivery date with a client, approving a change that affects regulatory reporting, authorising expenditure above a threshold — these remain in the domain of human judgement. The agent may surface options, model scenarios, and flag risks, but the decision itself is not delegated. This is the zone where traditional governance — the stage gate, the steering committee, the named authority — remains fully in force and entirely appropriate.

The autonomy envelope is not a permissions matrix to be configured once and forgotten. It is a governance instrument that must be actively maintained — because what is reversible today may not be reversible tomorrow, and what has a small blast radius in one phase of delivery may have a large one in the next.

6. Owning the Boundary, Not the Action

This is the part that most governance conversations about agents miss, and it is the part that matters most. It is not enough to classify decisions and set autonomy boundaries. Someone — a named individual, not a committee — must own each envelope.

The named owner of an autonomy envelope is accountable for three things, and those three things are the substance of this new form of accountability.

First, that the classification is correct. The actions assigned to this envelope genuinely belong here, given current conditions. A configuration change that was Class 1 last quarter may be Class 3 this quarter if the system it touches has since become subject to regulatory audit. A deployment that was Class 2 during normal operations becomes Class 3 or Class 4 during a period of heightened commercial sensitivity. Classifications are not permanent; they require active, informed maintenance by someone who understands both the delivery context and the risk environment.

Second, that the boundaries are appropriately drawn. The parameters within which the agent operates — the rules, the thresholds, the constraints — reflect the current risk appetite, the current regulatory landscape, and the current state of the delivery. An autonomy envelope calibrated during a stable phase may need tightening as a programme approaches a critical milestone or enters a period of organisational restructuring. The boundary is a living instrument, and the owner's job is to keep it current.

Third, that the assurance regime is functioning. The logs are being reviewed at the agreed cadence. The samples are being checked. The anomaly triggers are firing when they should. The owner does not read every log — that would reimpose universal review by another name. They ensure the system of assurance is working, that it is producing findings, and that those findings are being acted upon.

This is a different kind of accountability from the one delivery governance has traditionally demanded. It is closer to the accountability of an engineering manager for the reliability of a production system than to the accountability of a signatory on a change request. It is accountability for the design of the control, not for each instance of its exercise.

And it is, I would argue, a more honest form of accountability than what it often replaces. The senior manager who signs off on forty changes in a batch without reading any of them — a practice common enough that most delivery professionals will recognise it without further description — is exercising a fiction of accountability. The envelope owner who has consciously classified the decision space, set the boundaries, calibrated the sampling, and verified the monitoring is exercising accountability of a higher order, even though they have not personally approved a single agent action.

7. Assurance by Sampling

The shift from universal review to sampling-based assurance deserves particular attention, because it is the mechanism where trust in the autonomy envelope is built or

lost.

Sampling is not a new concept in governance — financial auditing has relied on it for decades, and quality assurance in manufacturing adopted statistical process control long before software delivery existed. But its application to agent-governed delivery decisions requires deliberate design.

The sampling regime must answer four questions. What proportion of actions in each envelope class are reviewed? How are they selected — purely at random, or weighted by heuristic triggers such as unusual patterns, boundary-adjacent decisions, or elevated system load? By whom — the envelope owner, a peer, an independent reviewer? And what happens when a sample reveals a problem — is the response limited to correcting the individual action, or does it trigger a review of the envelope's classification and boundaries?

In the programmes where I have seen this work well, the answers tend to follow a pattern. Sampling rates start high — perhaps twenty per cent of Class 2 actions reviewed in the first month — and decrease as confidence builds, with a floor that never reaches zero. Selection combines random sampling with trigger-based escalation, so that routine actions are checked probabilistically while anomalous ones are always caught. Review is performed by someone other than the envelope owner, to avoid the marking-your-own-homework problem. And every finding is treated as evidence about the envelope, not merely about the action — a single poor decision by the agent is less concerning than a pattern that suggests the classification is wrong or the boundaries are too loose.

The uncomfortable truth is that sampling-based assurance provides less certainty about any individual action than universal review would. What it provides instead is a sustainable, scalable, and — critically — honest system of governance. It acknowledges that we cannot review everything, and it builds a disciplined structure around that acknowledgement rather than pretending the reviews that do happen are thorough when they are not.

8. The Governance We Have Not Yet Imagined

There is a deeper tension here that this framework addresses but does not resolve, and it would be dishonest to pretend otherwise.

The autonomy envelope works within a delivery chain as we currently understand it — a structured flow of decisions, artefacts, and approvals moving toward a defined outcome. But the agents we are beginning to deploy do not merely execute decisions faster than humans. They are beginning to redefine what counts as a decision. When an agent continuously optimises a deployment pipeline, making hundreds of micro-adjustments per hour in response to real-time performance signals, the concept of a discrete "decision" that can be classified, enveloped, and sampled becomes strained. The adjustments are individually trivial and collectively significant, and they resist the kind of categorical treatment this framework depends on.

We are not yet at that frontier in most delivery organisations. The agents in our delivery chains today operate at a level of granularity where the four-class model applies well —

they execute identifiable tasks, produce reviewable outputs, and make decisions that can be meaningfully classified. But the honest position is that this framework is fitted to the current generation of agentic capability, and the next generation may require something we have not yet imagined.

What we can do now — and what I believe we must do now — is build the institutional muscle for governing autonomous action in delivery. The four-class model, the autonomy envelope, the named owner, the sampling regime: these are not permanent answers. They are the first serious attempt to ask the right question. The programme director who asked "who approved this?" was not asking for a name. She was asking for an accountability architecture — a system that could tell her, coherently and credibly, how decisions are made in her delivery chain when the decision-maker is not a person. That architecture is what we owe her, and what we owe the programmes we govern.

The alternative — continuing to run governance frameworks built for human decision-makers while the decision-makers quietly cease to be human — is not stability. It is a slow-motion abdication, dressed as continuity, and it will be visible only in the first serious failure.