FRAMEWORK WHITE PAPER | VERSION 1.0

Who Owns the Agent?

The Organizational Accountability Architecture

That Existing Governance Frameworks Require But Do Not Implement at the Agent Level

The Intent Architecture Stack

Sougata Roy | sougataroy.com | April 2026

Views expressed are personal. Not legal or regulatory advice. This paper presents an organizational design framework based on public sources and practitioner experience. It is not a substitute for qualified legal and regulatory counsel. Organizations should obtain legal advice before treating any description here as compliance guidance.

Executive Summary

There is a meeting that is about to happen inside your organization. You may not know it is scheduled yet.

An AI agent is going to do something unexpected. It will not be a dramatic failure. It will be something quieter: an email sent to the wrong party, a document shared outside its intended scope, a decision made at 2am by a system that was technically operating within its permissions but well outside what anyone intended. The logging system will capture all of it in perfect detail. Then someone will ask the question that no logging system answers: who in this organization was responsible for what that agent just did?

If that question does not have a pre-written answer, the organization is in a room it will not enjoy.

This white paper introduces the Intent Architecture Stack, a three-layer organizational design framework that answers that question before the meeting is called. The three layers are Context, Intent, and Governance. Each layer defines a specific organizational condition that must be present before an AI agent is authorized to operate in production. Together, they produce the governance architecture that existing compliance frameworks require but do not define.

Five major frameworks require that someone be accountable when an AI agent takes an action: NIST AI RMF, the EU AI Act, ISO 42001, the Cloud Security Alliance's Agentic Trust Framework, and Microsoft's Responsible AI Standard. None of them specify the organizational design that makes accountability operational at the agent level. The Intent Architecture Stack proposes an organizational design pattern that can help operationalize NIST's GOVERN function at the agent level, give practical form to EU AI Act Article 26's human oversight requirements for specific deployments, and provide an organizational backbone for the CSA Agentic Profile's accountability register in Microsoft-first enterprises. It is one proposed pattern, not an official standard or a substitute for legal or regulatory interpretation.

Who this paper is written for

This paper serves three readers. The CISO or CTO who needs to answer the board's accountability question before the next agentic AI deployment. The security architect who needs an organizational design framework to work against. The board member who needs to understand what governance evidence they should be asking for.

VOCABULARY: FIVE TERMS USED THROUGHOUT THIS PAPER

Intent Architecture The organizational design layer that defines what an AI agent is authorized to do, who authorized it, and what happens when it acts outside those boundaries. Built before deployment. The Intent Architecture Stack is the operational framework that implements this concept across three layers.

Intent Gap The unplanned divergence between what an organization genuinely intended an AI system to do and what it actually does in production. It is not a deployment failure. It is a monitoring failure that accumulates during operation when no mechanism exists to compare documented intent against actual behavior on an ongoing basis.

Governance Debt The accumulated accountability design work an organization deferred while deploying AI systems at speed. It accumulates the moment a deployment goes live without a documented authorization record, a named accountable owner, a defined scope, and a compliance review completed before deployment. Unlike technical debt, Governance Debt has an external enforcement dimension: when it reaches sufficient scale, regulators, auditors, and legal systems become involved.

The Accountability Assumption The implicit organizational belief that accountability for an AI agent's decisions resides with the vendor that supplied it, the platform that hosts it, or another team, rather than with the organization that decided to deploy it. It is not a deliberate choice. It is what fills the space when no deliberate accountability assignment is made. Regulators in financial services, employment, and consumer products have begun saying explicitly that vendor terms of service are not a defense.

Agent Sprawl The proliferation of AI agents across an enterprise without corresponding governance architecture. It operates across three distinct tiers: employee shadow AI (individuals using unsanctioned tools), organizational procurement without central visibility (business units adopting AI independently), and authorized agents with over-permissive operational scope (governed agents whose operational boundaries were never formally defined). Each tier requires a different governance response. Solving the first tier does not address the second or third.

Section 1: The Question Nobody Has a Good Answer For

The meeting had been going for forty minutes when the board chair held up a printout and asked a simple question.

Three weeks earlier, the organization's Copilot Studio agent had sent a draft contract amendment to an external counterparty. The agent had been configured to draft, not to send. The version it sent was not the current version. The compliance team had logs. The security team had the Purview audit record. The IT team had the agent configuration file. What none of them had was a document written before the incident that answered the chair's question: who in this organization authorized that agent to act, and how is that authorization documented?

The CISO had three answers ready. All three were technically accurate. None of them were what the chair was asking for. The chair was not asking what the agent did. The chair was asking who owned it.

> **Logging tells you what happened. It does not tell you who was responsible. Those are different questions, and only one of them survives a board meeting. This paper introduces the Intent Architecture Stack as the organizational design framework that answers the accountability question before the incident happens. It has three layers. Layer 1, Context, maps the regulatory obligations, stakeholder relationships, and system integrations the agent operates inside before its purpose is defined. Layer 2, Intent, documents what the agent is authorized to accomplish, what it is explicitly prohibited from doing, and what correct behavior looks like. Layer 3, Governance, names the Consequence Owner accountable for board-level decisions, establishes the review cadence, and writes the escalation path before the first incident requires it. When all three are in place before deployment, the board question has an answer. The remainder of this paper shows how to build that answer.**

That scenario is constructed. The pattern behind it is real and documented in public records from 2024 and 2025. The data below shows how widespread the problem already is.

86%

of organizations lack or do not enforce access policies for AI identities. The same report finds that 71% have AI in core business systems, and only 16% govern that access effectively.
Saviynt 2026 CISO AI Risk Report

11%

of senior IT and security leaders say there is no clear ownership of AI agents at all in their organization. The majority describe fragmented ownership that, in their own words, dilutes accountability.
Cisco AI/ML Security Report, March 2026

37%

of CISOs cite securing AI agents as their most urgent concern, above employees' use of AI tools (36%), which ranked second. This is the issue at the top of the stack.
Team8 CISO Village Survey, July 2025

What the Incidents Show

The statistics above describe a posture. The cases below describe what that posture looks like when an incident converts the governance gap from a theoretical problem into an operational one. All four cases are drawn from public records: court rulings, official company statements, or regulatory filings.

AIR CANADA, FEBRUARY 2024 : THE RULING THAT CLARIFIED THE OWNERSHIP QUESTION

Air Canada argued in court that it could not be held liable for its chatbot's statements because the chatbot was, in effect, a separate entity from the company. The British Columbia Civil Resolution Tribunal rejected this argument in a single sentence: 'It should be obvious to Air Canada that it is responsible for all the information on its website. It makes no difference whether the information comes from a static page or a chatbot.' The organization owned the agent's behavior. The ruling did not explain how that accountability should be designed into an organization before an incident. That is the gap this paper addresses.

META, MARCH 2026 : THE AGENT THAT POSTED WITHOUT PERMISSION

An internal AI agent analyzed a forum question and posted a public reply without explicit authorization. A second employee acted on the advice the agent had published. For approximately two hours, sensitive company and user data was accessible to employees who were not authorized to see it. The post-incident response proposed requiring agents to explicitly request user permission before taking actions. That proposal is an acknowledgment, in plain language, that explicit authorization gates for that class of agent action were not documented as in place before the incident. The governance architecture, by the organization’s own account, was addressed after the fact.

MICROSOFT COPILOT, FEBRUARY 2026 : WHEN PLATFORM CONTROLS ARE NOT ENOUGH

Microsoft confirmed a bug in which Copilot Chat accessed and summarized users' draft and sent emails, including messages protected by sensitivity labels and DLP controls designed to prevent exactly that. The platform controls were correctly configured. They failed anyway. When they did, the accountability question immediately shifted from 'what did the platform allow?' to 'who in this organization owns the consequence?' Organizations with a written authorization scope for the agent had an answer. Organizations whose governance artifact was the platform configuration did not.

AWS, DECEMBER 2025 : GOVERNANCE FAILURE WITH AN AI TOOL IN THE LOOP

An AWS engineer allowed an AI coding tool to resolve a production issue without the peer approval required by the organization's change management process. The resulting outage lasted thirteen hours. AWS attributed the incident to a user access control failure: the engineer held broader permissions than intended and the change management gate was not enforced at the organizational level. The AI tool was a factor in the sequence, not an autonomous actor operating outside all human direction. The governance lesson is precise: the authorization boundary existed in policy but not in practice, and no organizational design work had been done to ensure the two matched.

UPSTART HOLDINGS, APRIL 2026 : THE INTENT GAP PRICED BY A SECURITIES FILING

Upstart Holdings launched its Model 22 AI underwriting system in May 2025, describing it to investors as a tool that would increase loan approvals and improve risk assessment accuracy. During Q3 2025, the model’s behavior became more conservative, with management later acknowledging on the Q3 earnings call that the model had been overreacting to macroeconomic signals and that there was sampling and measurement error, reducing borrower approvals and conversion rates throughout the quarter. Revenue guidance was cut by approximately $20 million, and the stock fell 9.71 percent on November 5, 2025. Federal securities class actions were filed in April 2026, alleging that investors were misled about the model’s performance and impact. The class action complaints and the company’s own public statements are the source for the description above. From a governance perspective, this incident illustrates what an Intent Gap looks like when it surfaces through financial results rather than a risk review. The legal merits of the securities claims are for the courts to resolve. The governance observation, based on public sources, is that the divergence between the model’s documented intent at launch and its actual production behavior during Q3 2025 was not identified through an internal monitoring mechanism comparing the two on an ongoing basis. Instead, it became visible when quarterly metrics surfaced the problem at period-end. Whether and how the organization monitored model behavior against its documented intent is not established in public records; what is documented is that the gap became visible in a securities filing rather than in an internal governance review. Source: Pomerantz LLP class action announcement, April 30, 2026; Levi & Korsinsky announcement, April 22, 2026.

KISTLER V. EIGHTFOLD AI, JANUARY 2026 : THE ACCOUNTABILITY ASSUMPTION IN HIRING

> A class action filed in California Superior Court in January 2026 alleged that Eightfold AI scraped data on over one billion workers, scored applicants zero to five, and had low-scoring applicants discarded before any human review occurred. Plaintiffs alleged that Eightfold operated as a de facto consumer reporting agency without FCRA disclosures, access, or dispute rights, and that employers relied on its scores without telling applicants. The complaints portray a situation in which, according to the plaintiffs' allegations, neither the vendor nor the employers had clearly established who owned legal responsibility for the screening decisions. In the plaintiffs' telling, the vendor assumed employers had ensured FCRA compliance, and employers assumed the vendor's platform had handled it. From an Intent Architecture perspective, this is the Accountability Assumption in its most direct form: the implicit belief that legal responsibility for an AI system's consequential decisions resides with the other party. Kistler et al. v. Eightfold AI Inc., January 2026. A parallel pattern appeared in McDonald's McHire platform, operated by Paradox.ai, where public reporting described a credential failure exposing personal data on tens of millions of job applicants. Public statements and reporting do not suggest that the governance posture of the AI vendor had been fully evaluated by those responsible for the procurement decision before deployment at that scale. Whatever the contractual allocation of liability, accountability for third-party AI integrations in an organization's hiring process sits with the deploying organization by default.

Auditability Is Not Accountability

All four organizations above had logging. Three of them had sophisticated logging. The logs were not the problem. The problem was that the logs answered a question nobody was asking in the moment, while the question everyone was asking had no pre-written answer.

Auditability and accountability are not the same thing, and confusing them is the most expensive governance mistake a regulated enterprise can make.

Auditability means the organization can reconstruct what happened. Purview captures what agents accessed and what they processed. Entra Agent ID logs authentication events. Defender for Cloud logs behavioral anomalies. These systems are designed for auditability, and in most regulated enterprises they do their job well.

Accountability is a different condition entirely. It means the organization can identify, from a document written before the incident, who authorized the agent's action scope, who owns the consequence of that action, and what organizational structure connects those two things. Researchers Norval, Cobbe, and Singh define accountability in AI systems as 'answerability': the capacity to call a specific party to account, demand justification, and enforce a consequence if the justification fails.

An organization can have complete auditability and zero accountability simultaneously. The logs show every action the agent took. The board asks who authorized those actions. The answer is: it was in the configuration. That answer ends careers on a Tuesday morning, and it is entirely preventable.

Section 2: Five Frameworks, Five Different Versions of the Same Gap

The descriptions in this section summarize and interpret public frameworks at a high level. They are intended to motivate an implementation pattern, not to provide jurisdiction-specific legal analysis or to replace formal interpretation of NIST AI RMF, EU AI Act, ISO 42001, CSA documents, or Microsoft standards. Organizations should consult qualified legal and regulatory counsel before relying on any characterization here as compliance guidance.

Before introducing the Intent Architecture Stack, the major governance frameworks deserve a precise reading. This is not a critique. They are serious documents, and nothing in any of them prevents per-agent accountability documentation. The Intent Architecture Stack is designed to operationalize what they require: it instantiates NIST AI RMF's GOVERN function at the agent level, concretizes EU AI Act Article 26's human oversight requirement for agent-specific deployments, and gives the CSA Agentic Profile's accountability register its organizational teeth in Microsoft-first enterprises. The gap is an implementation gap, not a conceptual one. None of these frameworks specify how to build accountability into an agent's organizational design before it goes live.

Here is a pattern evident in much of today's AI governance practice: an organization can be formally aligned to NIST AI RMF, ISO 42001, EU AI Act deployer obligations, and Microsoft's recommended Entra Agent ID governance model, and still struggle to answer the board's question for a specific agent. The reason is not that these frameworks are wrong or incomplete. They were not written to specify organizational design at the individual agent level. They describe the accountability requirement without prescribing a concrete pattern for instantiating that accountability per agent.

The Intent Architecture Stack is offered here as one implementation pattern to close that gap, not as a replacement for those frameworks and not as an official standard.

NIST AI Risk Management Framework (AI RMF 1.0, 2023)

The NIST AI RMF requires organizations to establish accountability mechanisms, define roles and responsibilities, and ensure that accountability structures are in place so that appropriate teams are empowered to manage AI risks. This is the right requirement. The GOVERN function says clearly that accountability structures must inform every other function.

The implementation gap: the framework does not address agents as a distinct class of autonomous system. It does not provide a model for assigning accountability to a specific agent identity, or for documenting the authorization decision that defines what a particular agent is permitted to do. A Cloud Security Alliance working group that developed the NIST Agentic

Profile in March 2026 explicitly identified this as an implementation gap in RMF 1.0. The framework was not written for architectures where a single deployment decision can produce dozens of autonomous sub-agents making real-time decisions across multiple systems. The Intent Architecture Stack is designed as an implementation pattern for that gap: it proposes one way to apply the GOVERN function's accountability requirements at the agent level, designed to complement NIST AI RMF rather than replace it and to make its governance requirements more concrete for individual agent deployments.

EU AI Act (2024)

Article 26 of the EU AI Act requires deployers of high-risk AI systems to assign human oversight to individuals with the necessary competence and authority, monitor operation, and suspend use when the system poses a risk. This is accountability language with legal force, and it genuinely pushes organizations toward defining concrete responsibilities for specific deployments.

The implementation gap: the Act is drafted around AI systems, not AI agents. Nothing in Article 26 prevents per-agent accountability documentation. What it does not provide is a template for how to build that documentation. It does not define what per-agent ownership looks like, what an authorization register should contain, or how human oversight operates when an agent can spawn sub-agents without a human approval gate at each step. The Intent Architecture Stack proposes one way to give Article 26's requirement practical organizational form at the agent level.

ISO/IEC 42001:2023

ISO 42001 provides a management system standard for AI governance. It requires organizations to define and allocate roles and responsibilities, ensure top management accountability, and embed accountability throughout the AI lifecycle. The standard's instruction that accountability should 'ultimately reside with top management and formally designated risk owners' is the right principle and genuinely pushes organizations to define traceable responsibility down to specific deployments.

The implementation gap: the standard does not address agents as a distinct class of autonomous system. Nothing in ISO 42001 prevents per-agent documentation. What it does not provide is the operational design pattern for a specific agent deployment. There is no notion of agent identity, delegation chains, or per-agent accountability registers in the 42001 text. The Intent Architecture Stack proposes an implementation pattern at that level: one way to apply the top-management accountability structure ISO 42001 establishes to individual agent deployments in production.

Cloud Security Alliance Agentic Trust Framework (February 2026)

The CSA's Agentic Trust Framework is the closest existing document to addressing the gap. It requires each agent to have a verified identity, a documented ownership chain, and explicit governance sign-off before deployment. The CSA's companion NIST Agentic Profile introduces the concept of an agent accountability register: a document that captures, for each deployed agent, the business owner, the technical owner, the lineage of delegation authority, and the conditions under which the accountability chain is reviewed and updated.

Where it goes further than the others: The CSA's Agentic Trust Framework and NIST AI RMF Agentic Profile together represent the closest existing work to per-agent accountability. The Agentic Profile explicitly introduces an agent accountability register recording the business owner, technical owner, and delegation lineage for each deployed agent. That is the right structure. The Intent Architecture Stack is designed to be the organizational complement that gives that register its teeth in Microsoft-first enterprises: the Context, Intent, and Governance documents are what make the accountability register entries mean something when a board or an examiner asks about a specific agent on a specific Tuesday.

Microsoft's Own Documentation

Microsoft's position is perhaps the most instructive, because the platform sits at the center of most regulated enterprise agentic deployments. Microsoft Entra Agent ID requires a sponsor field when creating an agent identity: a named business representative accountable for the agent's purpose and lifecycle. The Responsible AI Standard v2 requires teams to identify stakeholders responsible for overseeing and controlling AI systems. Agent 365's Cloud Adoption Framework guidance states that organizations must 'assign organizational accountability for agent governance.'

The implementation gap: Microsoft’s own documentation is explicit about this boundary. The platform supplies controls, identity, logging, and policy enforcement. One line from the Cloud Adoption Framework captures the boundary precisely: “The platform can lock doors. It does not invent your org chart.” The organizational design on the other side of that boundary is the customer’s work. It is also the most important governance work in an agentic AI deployment, and it is what the Intent Architecture Stack provides, not as a critique of what Microsoft documents, but as the implementation pattern for what Microsoft’s own guidance explicitly leaves to the deploying organization.

> **Five frameworks, one implementation gap. Every framework requires that someone be accountable. The Intent Architecture Stack specifies how to build that accountability into the agent's organizational design before it goes live.**

Section 3: The Intent Architecture Stack

The Intent Architecture Stack has three layers. Before they are described, one framing point matters: the Intent Architecture Stack is a platform-independent organizational design framework. The governance decisions it requires apply regardless of which AI vendor, platform, or toolset an enterprise uses. This paper operationalizes it specifically for Microsoft-first enterprises using Microsoft Entra Agent ID, Microsoft Purview, and Copilot Studio, because that is where the majority of regulated enterprise agentic deployments are happening. The framework itself is not a Microsoft product or Microsoft-specific. It is the organizational design layer that Microsoft's own documentation explicitly leaves to the deploying organization. The layers build on each other in sequence: Context establishes the environment the agent operates in. Intent defines what the agent is supposed to accomplish and the boundaries within which it must operate. Governance designates the human accountability structure that owns the consequence when the agent acts. An agent without all three layers in place is significantly harder for an organization to defend in a board meeting, a regulatory examination, or an incident review.

Each layer produces a document. Together, the three documents constitute the governance record that a CISO can hand to a board. Each scenario below shows what happens when one of the three layers is missing before the agent goes live.

THE SCENARIO THAT MAKES CONTEXT CONCRETE

The agent had been running for six months. It was fully configured in Copilot Studio. It had a sponsor in Entra Agent ID. It passed every security review. It was also, quietly, reading emails in Sent Items when it was generating summaries for the weekly operations report. Nobody had intended for it to do that. But nobody had mapped the downstream system integration points before the agent was defined. The regulatory environment the organization operated in required sensitivity label enforcement. The stakeholder data the agent was touching included counterparty communications. None of that was documented before the intent was written. When the retrieval boundary failed, there was no pre-deployment context document to measure the failure against.

LAYER 1

CONTEXT: THE ENVIRONMENT

The organizational environment in which the agent will operate must be understood and documented before intent is defined or permissions are granted.

Context is the pre-deployment work of mapping the regulatory obligations, stakeholder relationships, data touchpoints, and system integration points that define the landscape the agent will operate in. Intent cannot be written responsibly without Context. A purpose statement written without knowing the regulatory environment, the affected stakeholders, and the downstream systems the agent can reach is a purpose statement that will surprise someone in a compliance review.

PRESENT the organization can produce a Context document, prepared before the agent's intent was defined, that names the applicable regulatory obligations, the stakeholder groups and data touchpoints affected by the agent, and every downstream system the agent can trigger or access.

ABSENT the agent's regulatory environment, stakeholder impact, and system integration scope were assumed from the deployment context rather than documented before intent was defined. When the agent touches data or systems outside the assumed scope, there is no pre-deployment record of what the organization understood the environment to be.

Context has three components:

Regulatory Environment. Define the regulatory obligations (OCC, FINRA, HIPAA, FedRAMP as applicable), stakeholder impact, and downstream system integration points before defining intent.

Stakeholders and Data. Identify the affected parties and data touchpoints. Know whose data the agent will touch and what it will do with it before defining what the agent is permitted to do.

System Integrations. Map all downstream system triggers and integration points to define the full technical scope before any intent statement or authorization boundary is written.

THE SCENARIO THAT MAKES INTENT CONCRETE

The agent had been running for eleven weeks. It was fully compliant. It was hitting its KPIs. It was also sending follow-up emails to every customer who had submitted a cancellation request, offering them a discount. The retention team thought it was brilliant. Legal found out on a Thursday. The agent was operating exactly within its permission scope. It was completely outside what the organization intended it to do. There was no written purpose statement against which anyone could have measured the drift. The authorized scope said what the agent could do. Nobody had written down what it was supposed to accomplish. By the time the gap was visible, it had been running for eleven weeks.

LAYER 2 INTENT: THE PURPOSE

The organizational record of what the agent was built to accomplish, expressed in plain language that a compliance officer, a regulator, or a board member can evaluate without technical context.

Intent is the organizational condition that connects what the agent was deployed to accomplish with what it is actually doing in production. Intent must be written before deployment. An agent can operate within its permission scope and completely outside its organizational intent simultaneously. When that happens, there is no governance standard against which to measure the drift.

PRESENT the agent has a written Intent document containing a Purpose Statement, an Authorized Scope with explicit prohibitions, and Expected Outputs defining what correct behavior looks like. All three are written before the agent enters production.

ABSENT the agent's intent is implied by its configuration. No Intent document exists before deployment. The only record of what the agent was supposed to accomplish is what it was technically set up to be capable of doing. Those are not the same thing, and the gap between them is where governance failures compound invisibly.

Intent has three components:

Purpose Statement. Document the organizationally intended accomplishments and the explicit purpose of the AI agent, in plain language, before it enters production.

Authorized Scope. Clearly define the authorized scope of actions, including specific permissions and explicit prohibitions. What the agent is forbidden from doing must be written. Authorization boundaries without explicit prohibitions are incomplete.

Expected Outputs. Specify expected output formats, define the triggers for human review, and establish what constitutes correct behavior. This is the standard against which the agent's actual behavior is measured in production.

THE SCENARIO THAT MAKES GOVERNANCE CONCRETE

The governance committee met quarterly. The agents were deployed weekly. Every deployment went through a review that confirmed the Entra Agent ID had a sponsor field populated, the Copilot Studio configuration had an owner assigned, and the Purview policy was active. The review took forty-five minutes per agent on a good day. What it did not confirm was whether the sponsor understood what their sponsorship meant in practice. The sponsor for fourteen of the thirty-one production agents was the same person: the head of the AI Center of Excellence, who had been assigned to those roles because the form required a name and his name was available. He was accountable for fourteen agents in the governance record. He had reviewed the configuration of three of them.

LAYER 3 GOVERNANCE: THE ACCOUNTABILITY

Who in this organization owns the consequence when this agent acts, how will that ownership be reviewed, and what is the escalation path when it acts outside its intent?

Governance is the organizational condition that connects an agent's actions to a named human who bears genuine organizational responsibility for those actions, a defined review cadence that keeps the accountability current, and a clear escalation path when the agent behaves outside its intent. A sponsor field in a platform is not governance. Governance is the organizational design that makes the sponsor's accountability real.

PRESENT every agent in production has a named Consequence Owner responsible for board-level accountability and incident escalation decisions, nested within the organization's existing formal accountability structure (three-lines-of-defense, SMF regime, risk committee, or equivalent). A defined Review Cadence maintains documented evidence records. A written Escalation Path specifies who is contacted and in what sequence when the agent triggers an anomaly. All three are established before deployment.

ABSENT the agent has sponsor and owner fields populated in the platform. There is no written escalation path. The review cadence is informal. The named individual cannot describe what their accountability means in practice for a board-level question about a specific agent action.

Governance has three components:

Accountable Owner. A named Consequence Owner responsible for board-level accountability and incident escalation decisions. In most regulated institutions this individual is nested within an existing formal accountability structure: three-lines-of-defense, a senior management function, a risk committee, or equivalent. The Consequence Owner is not necessarily the first-line incident responder. They are the person accountable for the governance decision when the agent's actions require organizational explanation.

Review Cadence. Define a recurring review cadence, establishing evidence records and frequency for ongoing assessment. The review is triggered by scheduled intervals and by specific organizational events including Microsoft product releases that expand agent capabilities.

Escalation Path. Establish clear incident escalation paths with defined response sequences and triggers for immediate action. The escalation path is written before the first incident, not assembled during it.

Reading the Three Layers Together

The layers work as a governance architecture, not a sequential checklist. An organization with strong Governance (a named accountable owner, a review cadence, an escalation path) but no Intent document has given someone responsibility for a boundary they cannot define, because the purpose statement and authorized scope were never written. An organization with a written Intent document but no Context foundation has defined what the agent is supposed to accomplish without mapping the regulatory environment and system integrations it will operate inside. When those conditions are undocumented, the Intent statement is written against assumptions rather than facts.

The most common pattern in regulated enterprises deploying agentic AI right now: no Context document because the regulatory and stakeholder landscape was assumed rather than mapped, a partial Intent document in the form of a capabilities description rather than a purpose statement with explicit prohibitions, and nominal Governance in the form of platform metadata fields rather than a written accountability structure with a review cadence and an escalation path. The logs are clean. The three-layer governance architecture is not there.

Section 4: The Diagnostic: Applying the Framework

THE QUICK-SCAN: FIVE QUESTIONS PER LAYER Before running the full diagnostic, use the quick-scan below. It takes fifteen minutes and identifies whether any layer is clearly absent. If an agent passes all fifteen questions, proceed to the full diagnostic to confirm depth. If any question fails, the layer is absent or incomplete and the full diagnostic section for that layer applies. Layer 1, Context: Can you produce a document prepared before the intent was defined that names the regulatory frameworks applicable to this agent's deployment? Does that document identify the stakeholder groups whose data or workflows the agent affects? Does it map every downstream system the agent can trigger or access? Was it written before the Intent Document? Could a new team member use it to understand the full operating environment without asking the deployment team? Layer 2, Intent: Can you produce a written Intent Document prepared before production deployment? Does it contain a Purpose Statement in plain language describing what the agent is supposed to accomplish? Does it contain an Authorized Scope with explicit prohibitions, not just permissions? Does it define Expected Outputs and the conditions that trigger human review? Has it been updated to reflect any change in the agent's purpose or scope since deployment? Layer 3, Governance: Is there a named Consequence Owner in a written organizational document, not only in a platform metadata field? Can that person describe what their accountability means in practice if an examiner calls? Is there a written Escalation Path with named contacts and response sequences that predates any incident? Is there a defined Review Cadence with a future review date? If the current Consequence Owner left last month, is ownership documented for the person who replaced them? If any answer to the fifteen questions above is no or uncertain, that layer is the starting point. The full diagnostic below provides the detail needed to close the gap.
THE FULL DIAGNOSTIC The diagnostic that follows is a practitioner-level tool. Use it in a structured session with the business unit that owns the agent. It typically takes 90 minutes to three hours depending on the agent's complexity. The session is not a scoring exercise. It is a gap identification exercise. For each of the three layers, the practitioner looks for a specific type of evidence: not a policy describing what should exist, but an artifact demonstrating what does exist.

FINRA's December 2025 Oversight Report made this distinction explicit, instructing firms to maintain comprehensive documentation throughout AI deployments and to track agent actions and decisions. That is evidence language, not policy language. The gap between the two is what this diagnostic is designed to surface.

Layer 1: Context Diagnostic

The single test for the Context layer: can the practitioner produce a document, prepared before the agent's intent was defined, that maps the regulatory obligations, affected stakeholders and data touchpoints, and downstream system integrations relevant to this agent's deployment? If

the answer involves opening the technical specification or the deployment runbook, the Context layer is absent. Those are implementation artifacts. The Context document is an organizational artifact that precedes and informs them.

1. Produce the Context document for this agent. Not the technical specification and not the deployment checklist. A written document that maps the regulatory obligations applicable to this deployment, the stakeholder groups whose data or workflows the agent touches, and every downstream system the agent can trigger or access. What is the date on it?
2. Does the Context document name the specific regulatory frameworks that apply to this agent's deployment environment? OCC, FINRA, HIPAA, FedRAMP, or whichever applies? If the regulatory environment changed since deployment, was the Context document updated?
3. Does the document identify the stakeholder groups affected by the agent's actions? Does it map the data touchpoints the agent can reach, including any data the agent accesses indirectly through system integrations?
4. Does the document map all downstream system triggers and integration points? If the agent can invoke an API, trigger a workflow, or modify a record in a connected system, is that connection documented in the Context layer?
5. Was the Context document written before the Intent document? The sequence matters: intent written without a mapped context is intent written against assumptions. If both were written on the same day, the Context layer may be retroactive rather than foundational.
6. Could a new security architect review this Context document and understand the full regulatory, stakeholder, and technical landscape the agent operates within, without needing to ask the deployment team a single question? If not, the document is incomplete.

Layer 2: Intent Diagnostic

The single test for the Intent layer: can the practitioner produce a written Intent document, prepared before the agent entered production, containing a Purpose Statement, an Authorized Scope with explicit prohibitions, and a definition of Expected Outputs? If those three components are not all present in a single pre-deployment document, the Intent layer is partial or absent. A technical specification that describes what the agent can do is not an Intent document. An Intent document describes what the agent is supposed to accomplish and what it is forbidden from doing.

7. Produce the Intent document for this agent. It must contain three things: a Purpose Statement (what the agent is supposed to accomplish), an Authorized Scope (what it may do and what is explicitly prohibited), and Expected Outputs (what correct behavior looks like and what triggers human review). What is the date on it?
8. Does the Purpose Statement describe what the organization intends the agent to accomplish, in plain language, rather than what the agent is technically capable of doing? If the Purpose Statement could have been written by reading the configuration file, it is a capabilities description, not an intent statement.

9. Does the Authorized Scope explicitly name what the agent is forbidden from doing? A scope document that lists only permissions is incomplete. Explicit prohibitions define the boundary from the outside, and the boundary from the outside is what an examiner asks for.
10. Does the Expected Outputs section define what correct behavior looks like? Does it define triggers for human review? If an agent produces an output that falls outside the expected range, is there a written standard that makes that determination without requiring the original deployment team to be in the room?
11. Has the Intent document been updated since the agent was first deployed? If the agent's purpose evolved in production, did the Intent document evolve with it? An outdated Intent document means the governance standard being applied to the agent is wrong.
12. Could the Intent document be handed to a new compliance officer on their first day and give them a meaningful standard against which to evaluate the agent's behavior? If the document requires interpretation from someone who was present at the deployment, it is not sufficient.

Layer 3: Governance Diagnostic

The single test for the Governance layer: can the practitioner produce three documents prepared before any incident, naming the Accountable Owner, defining the Review Cadence with evidence records, and specifying the Escalation Path with response sequences and triggers? A populated sponsor field in Entra Agent ID is not this evidence. It is the platform record that points to the Governance documentation. The documentation is what gives the platform record its organizational meaning.

13. Who is the Accountable Owner for this agent, as a specific named individual rather than a team or a role? What does their ownership mean in practice? If the agent sends an email it should not have sent, what does this person do in the next four hours and who do they notify?
14. Is the Accountable Owner documented in a written record prepared before any incident, not in the Entra Agent ID metadata field? A written organizational document that describes who is accountable, what their accountability encompasses, and what their escalation obligation is.
15. What is the Review Cadence for this agent? On what schedule is the Governance layer re-validated? What organizational events trigger an unscheduled review, such as a Microsoft product release that expands the agent's capabilities or a personnel change that affects the Accountable Owner?
16. Is there a written Escalation Path for this agent? Does it name the individuals contacted in sequence when the agent triggers an anomaly? Does it specify the triggers for immediate action versus scheduled review? If the escalation path was assembled during the last incident rather than written before it, the Governance layer is absent.
17. If the current Accountable Owner left the organization last month, who holds accountability today? Is that transition documented, or does accountability for this agent currently reside with someone who no longer works here?
18. Could a board member, an OCC examiner, or an external auditor identify the Accountable Owner and the Escalation Path from documents that predate any incident?

Or would those individuals only be identified through the investigation that follows the incident?

> **The governance architecture that survives an incident is the one that was built before the incident. Everything built after it is incident response, not governance. THE CROSS-ORGANIZATIONAL DELEGATION SCENARIO Multi-agent architectures introduce a fourth diagnostic question that sits above the three layers: when an agent can invoke sub-agents, who owns the accountability chain across the delegation boundary? Consider this pattern, which is now appearing in regulated enterprise deployments. An organization uses a vendor-hosted orchestration agent (built and maintained by a third-party AI provider) as the primary interface for a customer service workflow. That orchestration agent, when it encounters a specific claim category, delegates to a customer-owned sub-agent running in the organization's own Microsoft tenant. The sub-agent has authority to read customer financial records and draft a preliminary settlement recommendation. The accountability question in this architecture is not whether either agent is individually governed. Both may have completed Intent Documents and named Consequence Owners. The accountability question is what happens at the delegation boundary. The vendor-hosted orchestration agent passes a claim record to the customer-owned sub-agent. The customer-owned sub-agent acts on it. If the sub-agent drafts a recommendation based on information it was not authorized to receive, or if the orchestration agent passes parameters that expand the sub-agent's effective scope beyond its documented authorization, neither agent's individual governance record captures the failure. The Intent Architecture Stack addresses this through two additions to the Layer 2 Authorized Scope for any Tier 3 agent that can be invoked by an external or vendor-hosted orchestrator. First, the Authorized Scope must explicitly name which orchestration sources the agent is permitted to receive instructions from, and what parameters it is permitted to act on. Delegation from an unauthorized source, or acting on parameters outside the documented scope, must be listed as explicit prohibitions. Second, the Consequence Owner for the customer-owned sub-agent must have reviewed and signed the delegation authorization, a separate record from the agent's standard authorization document that names the vendor-hosted orchestrator, the scope of delegated instructions, and the conditions under which the sub-agent is permitted to act on them. The diagnostic question for cross-organizational delegation is direct: if the vendor-hosted orchestration agent sends your sub-agent an instruction outside its documented scope at 2am, which named individual in your organization is**

> **accountable for the resulting action, and is that accountability documented in a record that predates the event?**

Section 5: Risk Proportionality: Not Every Agent Needs the Same Stack

The first question every experienced CISO asks when they see a three-document governance requirement is: does a simple internal summarizer need the same documentation as an agent that can move money? The answer is no. The framework is universal. The depth of documentation within it scales with risk.

Applying the full three-layer stack with equal rigor to every agent regardless of its consequence profile will create governance theater and stall delivery. The proportionality model below defines three risk tiers. The tier determines the documentation depth required for each layer, not whether the layer applies. Every agent goes through all three layers. The question is how much evidence each layer requires.

The Three-Tier Risk Model

Tier	Agent Profile	Documentation Depth	Examples
Tier 1 Low Risk	Internal-only. Reads non-sensitive or non-regulated data only. No external communications. No record modification in regulated systems. No financial transactions. Internal communications, if any, are limited to the invoking user or a defined small team and are not broadcast-capable.	Lightweight Intent Document (purpose + scope). Governance Record with named owner and review trigger. Context Document can be class-level, shared across similar agents.	Internal knowledge summarizers, document search agents, internal scheduling assistants, read-only analytics agents.
Tier 2 Medium Risk	Cross-departmental. Reads regulated data. Internal communications capability.	Full Intent Document (purpose + scope + explicit prohibitions + expected outputs). Agent-specific Context Document. Governance Record with named Consequence Owner,	Compliance monitoring agents, customer service drafting agents, risk flagging agents, HR workflow agents.

	Recommendation outputs that inform human decisions but do not execute them. May include agents that create or update internal records and drafts that are not authoritative sources of record, where a human must review and approve before any customer-facing or regulator-facing action occurs. Note: some Tier 2 agents may still be classified as high-risk under the EU AI Act (for example, risk scoring or compliance-relevant monitoring systems). This tier governs internal documentation proportionality, not legal classification.	formal review cadence, and escalation path.	
Tier 3 High Risk	External communications. Financial transaction execution. Record modification in authoritative regulated systems. Customer-facing autonomous decisions. Can invoke or delegate to sub-agents with authority to change records, communicate externally, or execute transactions without an additional human gate. Tier 3 is where both	Full three-layer stack. Named Consequence Owner with board-level accountability. Formal review cadence with evidence records. Written escalation path with named contacts. Pre-deployment governance sign-off required. Board-level reporting on posture.	Trading workflow agents, claims processing agents, customer communication agents, agents that invoke sub-agents or delegate across systems.

	exposure and autonomy are high.		

The tier is determined before the agent is classified as any particular type. The classification question is always the same: what is the worst thing this agent could do if it operates at the edge of its permission scope and outside its intent? The answer to that question determines the tier. An internal summarizer that can read only SharePoint documents and produce only internal notes is Tier 1 even if it is built on the same platform as a Tier 3 agent.

Class-level documentation is permitted for Tier 1 agents: a single Context Document can cover a class of twenty similar internal summarizers, and a single Intent Document template can be applied across the class with agent-specific fields filled in. This approach aligns with how most model risk inventories work and allows the framework to scale without creating a documentation burden that stalls deployment.

Tier 2 and Tier 3 agents require agent-specific documentation because their consequence profiles differ materially even within the same agent type. Two Tier 3 customer communication agents at the same organization may have different Authorized Scopes, different Consequence Owners, and different Escalation Paths depending on which customer segment they serve and which regulatory regime applies.

An Anonymized Intent Document Example, Tier 2

The table below shows what a completed Intent Document looks like for a Tier 2 agent. This is the document that sits at the center of the Layer 2 governance requirement. It is not a technical specification. It is an organizational record written in plain language that a compliance officer, a regulator, or a board member can evaluate without technical context.

INTENT DOCUMENT | Tier 2 Agent | Version 1.0 | [Date] | Classification: Internal	
Agent Name	[Organization-assigned name and unique identifier]
Deployment Environment	[Microsoft 365 tenant name, Copilot Studio environment, date entered production]
Risk Tier	Tier 2, Medium Risk
Purpose Statement	This agent monitors incoming customer service requests submitted via the internal ticketing system and drafts a proposed response for review by a human agent before any response is sent. It does not send communications. It does not access customer financial records. It does not modify any record in any system.
Authorized Scope	The agent may: read submitted service requests from the ticketing system queue. The agent may: access the internal knowledge base to identify relevant policy information. The agent may: produce a draft response document delivered to the assigned human agent for review. The agent may NOT: send any communication to any party, internal or external. The

	agent may NOT: access customer account data, transaction history, or financial records. The agent may NOT: modify, delete, or flag any record in any system. The agent may NOT: invoke any other agent or automated workflow.
Expected Outputs	A draft response document, in the format specified in the knowledge base template, delivered to the assigned human agent within the ticketing system. Correct behavior: draft uses approved policy language, does not make commitments, identifies the relevant policy section. Anomalous behavior requiring human review: draft that includes a financial figure, a timeline commitment, or language referencing any record the agent is not authorized to access.
Human Review Triggers	Any draft flagged as anomalous by the expected output definition above. Any customer request classified as a complaint, legal notice, or escalation. Any draft that cannot be mapped to a specific knowledge base policy entry.
Consequence Owner	[Full name, title, organizational unit], responsible for board-level accountability and incident escalation decisions for this agent. Contact: [work phone, email].
Technical Owner	[Full name, title], responsible for agent configuration, credentials, monitoring, and control enforcement.
Review Cadence	Quarterly. Additional review triggered by: any Microsoft product release that modifies Copilot Studio retrieval behavior; any change to the ticketing system integration; any incident involving this agent; any change to the Consequence Owner or Technical Owner.
Authorization Signatory	[Name, title, date signed], authorized to grant the scope described above under [organization policy reference].
Document Version History	v1.0, [Date], Initial authorization.

The Authorized Scope section is the most important field in the document. It contains both permissions and explicit prohibitions. An examiner, a board member, or an incident investigator reading this document can immediately determine whether a specific agent action was within scope or outside it. The explicit prohibitions are what make that determination possible. A scope document that lists only permissions cannot answer that question.

This template can be downloaded from sougataroy.com/frameworks/intent-architecture-stack. The full suite of governance frameworks, including the Governance Readiness Matrix, the Deployment Accountability Map, the Tenant Agent Reconciliation Framework, and the Authorization Coverage Lifecycle, is available at sougataroy.com/frameworks. Tier 1 agents use an abbreviated version covering Purpose Statement, Authorized Scope, and Consequence Owner only. Tier 3 agents require additional fields covering delegation scope, sub-agent authorization, and board reporting obligations.

Section 6: What Regulators Are Now Requiring

The regulatory environment for AI agent governance in financial services is undergoing a specific transition. The direction of travel is clear: regulators are moving from governance language, which tells organizations they must have policies and oversight structures, to evidence language, which tells organizations what they must be able to produce when an examiner asks. The Intent Architecture Stack is designed to produce that evidence.

Regulator	What They Now Require
OCC (April 2026)	Revised model risk guidance issued jointly by the Federal Reserve, OCC, and FDIC in April 2026 rescinds SR 11-7 and replaces it with a more risk-based, principles-driven framework. The revised guidance does not yet explicitly address generative AI and agentic AI as a distinct category, while preserving the core accountability standard that has always applied: 'clear roles and responsibilities with well-defined accountability' and documentation that supports 'tracking of recommendations, responses, and exceptions.' The board retains ultimate oversight responsibility. Accountability cannot be delegated to algorithms or vendors.
FINRA (December 2025)	FINRA's 2026 Oversight Report moved from general governance language to specific evidence requirements for AI agents: 'formal review and approval processes,' 'comprehensive documentation throughout,' 'storing prompt and output logs for accountability and troubleshooting,' and 'tracking agent actions and decisions.' FINRA warned explicitly that agents may act 'beyond the user's actual or intended scope and authority.' This is the most operationally specific U.S. financial services regulatory language on AI agent accountability currently in print.
Federal Reserve (February 2026)	Governor Christopher Waller stated that proper AI operation requires 'rigorous model validation, human accountability for decisions, and ongoing evaluation.' The phrase that carries the most weight in a governance context: 'human accountability for decisions.' The owner is a person inside the institution, not the model, not the agent, not the vendor contract.
FINMA (February 2026)	The Chair of the Swiss Financial Market Supervisory Authority stated that 'supervisory decisions must remain explainable and accountable,' requiring 'a human in the loop' for all significant AI-enabled interventions. The State of SupTech Report 2025 found that more than half of surveyed financial regulatory authorities lacked clear governance structures for AI-enabled supervision. The accountability gap is not only a regulated institution problem. It is an examiner-level problem as well.

The regulatory direction has a single implication for an organization using the Intent Architecture Stack: when an examiner asks for the governance documentation for a specific agent, the three-layer documentation set is the answer. The Context document answers the environmental and integration scope question. The Intent document answers the purpose and authorized scope question. The Governance documentation answers the accountability ownership and escalation question. The examiner does not need to be told what those documents are for. They will recognize them immediately.

Section 7: Where Most Organizations Are Right Now

Every organization deploying AI agents sits somewhere on the three-layer framework. The Governance Debt Maturity Model describes where most of them are and what the path to Stage 3 looks like in operational terms. The model has three stages. Saviynt's 2026 CISO AI Risk Report provides the clearest evidence of where most organizations currently sit: 86% lack or do not enforce access policies for AI identities, and only 16% govern AI access to core business systems effectively. That is a Stage 1 profile at scale.

The financial cost of the Stage 1 posture is now quantified at the organizational level. IBM’s 2025 Cost of a Data Breach report found that high levels of shadow AI added approximately $670,000 to the average breach cost of $4.44 million. That figure represents one category of Governance Debt made measurable: the cost of ungoverned AI access that existing breach containment did not account for. Reco’s 2025 State of Shadow AI report found that 71 percent of office workers used AI tools without IT approval, and nearly 20 percent of organizations had already experienced data breaches directly attributable to unauthorized AI use. The accumulation is not theoretical. It is a current inventory condition in most enterprises.

Agent Sprawl, the proliferation of AI systems without corresponding governance architecture, operates across three distinct tiers that require three different governance responses. The first tier is employee shadow AI: individuals using personal accounts or browser-based AI tools for work tasks without organizational visibility into what data is being processed. The second tier is organizational procurement without central visibility: business units independently adopting AI vendors through department-level purchases or pilots that became permanent without a central inventory or risk assessment. The third tier is the most consequential and least addressed: authorized agents with over-permissive operational scope. These agents are in the registry and have legitimate credentials. The governance failure is in what they are permitted to do once deployed. The Replit production database deletion and the Meta Sev-1 breach described in Section 1 are both Tier 3 failures: not ungoverned agents, but governed agents whose operational boundaries were never defined with enforcement behind them. Solving Tier 1 does not address Tier 2. Solving both does not address Tier 3. Most enterprise governance programs in 2026 are focused on Tier 1.

Stage	What It Looks Like on a Specific Tuesday
Stage 1: Accumulation (Where most organizations are right now)	Agents are deployed. Platform controls are configured. Purview is logging. Entra Agent ID has sponsor fields populated. The governance review committee approved the deployment. But no written authorization documents exist. No intent statements were prepared before deployment. The accountability designation is a platform metadata field that someone filled in because the form required it. Governance debt is accumulating with every agent deployed, and the organization cannot see it because the audit trail looks clean. This

	stage ends when an incident or an examiner asks a question the organization cannot answer from a pre-deployment document.
Stage 2: Recognition (Triggered by an incident or an examiner question)	An incident occurs or an examiner asks a question the organization cannot answer from pre-deployment documentation. The board asks who is accountable, and the CISO realizes the logging system has a perfect audit trail and the governance system has no authorization record. The organization understands the debt. Tactical responses begin: authorization documents are written retroactively, intent statements are reconstructed from configuration data, accountability assignments are formalized after the fact. The debt is visible but not yet resolved. This stage is expensive and reactive.
Stage 3: Resolution (The governance-first posture)	Before any agent enters production, the three layers are completed in sequence. A written authorization document is signed by a named individual with the authority to grant the scope. An intent statement defines what the agent is supposed to accomplish and what anomalous behavior looks like. An accountability record names the business sponsor and the technical owner, with an organizational description of what each person's accountability means in practice. A quarterly posture review updates each layer against the current platform capabilities and regulatory environment. No agent enters production without clearing all three layers.

The transition from Stage 1 to Stage 2 is rarely a deliberate choice. For most organizations that remain in Stage 1 long enough, an incident or an examiner question eventually makes the governance debt visible. The question is whether the organization builds the governance architecture before that event or after it.

The research on this is clear about the directional cost difference. Analysis of AI transformation program failures consistently finds that retrofitting governance after deployment costs significantly more than embedding controls from the outset, because the remediation must work around an architecture that was never designed to accommodate governance gates. Ethyca's "AI Governance: Framework, Compliance and Operational Guide 2026" (February 2026) documents the specific mechanism: most organizations confuse governance with policy, producing extensive documentation that does not translate into real controls, so that "documentation exists, but evidence does not." The governance debt accumulates invisibly in Stage 1. The payment date is set by external events the organization does not control.

Section 8: Applying the Framework in the Microsoft Environment

The Intent Architecture Stack is platform-independent. The organizational design conditions it defines apply regardless of which AI platform, vendor, or toolset an enterprise uses. This section maps the framework to the Microsoft enterprise environment specifically, because the research base for this paper is grounded in the Microsoft governance ecosystem and the practitioners most likely to apply this framework are working in Microsoft-first regulated enterprises.

Microsoft Tool	What It Provides and Where the Organizational Design Work Begins
Microsoft Entra Agent ID	Provides the identity foundation for the Governance layer: verified agent identity, a required sponsor field, a recommended owner field, and separation between technical administration and business accountability. Entra Agent ID's documentation is explicit that sponsors are 'business representatives accountable for the agent's purpose and lifecycle decisions.' The sponsor field is the platform record that points to the Governance documentation. It is not the Governance documentation itself.
Microsoft Purview	Provides the audit infrastructure that supports all three layers: prompts and responses captured in the unified audit log, DLP policy enforcement, sensitivity label compliance, and AI interaction event records. Purview can confirm that an agent accessed or did not access specific data. It cannot confirm whether that access was consistent with the organizational Intent the agent was deployed with, because that Intent lives in the Intent document, not in the platform configuration.
Microsoft Agent 365 (GA: May 1, 2026)	Provides the control plane for all three layers: a centralized agent registry, lifecycle management, access control, observability across agents regardless of where they were built. Microsoft's Cloud Adoption Framework guidance on Agent 365 states: 'Assign organizational accountability for agent governance. AI agents introduce organizational risk similar to applications and identities. Governance requires clear accountability.' The platform provides the visibility. The organization provides the governance architecture that gives visibility its meaning.
Copilot Studio	Provides agent creation and management within the Power Platform environment. Microsoft's February 2026 security guidance for Copilot Studio states explicitly: 'Ensure that every agent has an active, accountable owner. Reassign ownership for orphaned agents

	or retire agents that no longer have a clear purpose.' That instruction describes the Governance layer requirement in operational terms. What Microsoft does not specify is what 'accountable' means inside the organization's governance structure, or what review cadence and escalation path must accompany the ownership designation.
EchoLeak (CVE-2025-32711)	Aim Security disclosed EchoLeak (CVE-2025-32711, CVSS 9.3) in June 2025: a zero-click vulnerability in Microsoft 365 Copilot in which attackers used crafted content, including Outlook emails, that Copilot processed as instructions, causing it to exfiltrate sensitive data from the tenant without any user interaction. Copilot's design allowed external content to function as high-privilege instructions inside the tenant. The deploying organization had not formally defined what external content should be permitted to trigger agent action. The architectural boundary between untrusted external content and privileged internal AI orchestration had not been established before deployment. This is an Intent Architecture failure at the tenant boundary: the organization's Layer 2 Authorized Scope did not address this attack surface because the surface was not mapped before the intent was defined.
RoguePilot (GitHub Codespaces)	Orca Security disclosed RoguePilot in February 2026: a vulnerability in GitHub Codespaces where Copilot acted on malicious instructions injected into GitHub Issues. When a developer launched a Codespace from a tainted issue, Copilot automatically consumed the issue text as context, executed attacker-crafted instructions, and exfiltrated the GITHUB_TOKEN from the Codespace, enabling full repository takeover. The deploying organization had not formally defined what file paths Copilot could read, what outbound calls it could make, or what environment credentials it was permitted to access inside Codespaces runtime environments. All three of these are Layer 1 and Layer 2 governance decisions the organization must make before the agent operates. The platform supplied the capability. The organization had not designed the boundary. Source: Orca Security, February 2026.

The organizational design work on the other side of that boundary is what the three-layer framework addresses. The platform surfaces the data. The Context, Intent, and Governance documents determine whether that data can answer a board question.

Section 9: Answering the Board Question

This section provides the practical template for what the board answer looks like when the Intent Architecture Stack is in place. The board question is: who in this organization is accountable when our agent takes an unauthorized action? Not which vendor built the system. Not which team deployed it. Who in this organization owns that consequence, and how is that ownership documented?

The Three-Layer Documentation Set

The board answer is not a single document. It is a documentation set, one output per layer of the framework, built before the agent enters production. The table below shows what each layer produces, what the document must contain, and the governance test it must pass.

Layer	Document Required	Must Contain	Governance Test
Layer 1 Context	**Context Document**	Regulatory obligations, affected stakeholders and data touchpoints, downstream system integration map.	Predates the Intent document. Named signatory.
Layer 2 Intent	**Intent Document**	Purpose Statement, Authorized Scope with explicit prohibitions, Expected Outputs and human-review triggers.	Predates production deployment. Explicit prohibitions present.
Layer 3 Governance	**Governance Record**	Named Accountable Owner (individual), Review Cadence with evidence schedule, Escalation Path with named contacts and response triggers.	Owner is a named person, not a team. Escalation path is written, not assumed.
Supporting All layers	**Technical Owner Record**	Named individual responsible for configuration, credentials, monitoring, and day-to-day control enforcement. Distinct from the business owner.	Business owner and technical owner are two different named individuals.

Every row in this table must be completable from documents produced before the incident. If any row requires investigation to answer, that is the organization's current governance gap. The examiner will find the same gap. The board will ask about the same gap. Getting the answer

from documents rather than investigation is the difference between governance and incident response.

One thing to notice: none of these documents require sophisticated technology. They require organizational design decisions made by named individuals with the authority to make them. The hard part is not the documentation. The hard part is making those decisions before the first incident forces them.

Section 10: What Good Looks Like

This section describes a Stage 3 organization in operational terms. It is not a description of a perfect organization. It is a description of an organization that can answer the board question from a document that predates the incident.

Before Any Agent Goes Live

The organization runs the three-layer diagnostic as a mandatory pre-deployment gate. The gate is not a policy statement. It is a production lock: the agent's Entra Agent ID sponsor field is not populated until the business sponsor has signed the accountability assignment document. The agent's deployment is not approved until the authorization document and intent statement have been reviewed and accepted by both the business sponsor and the technical owner.

This adds time to deployment. For a standard agent with limited scope, the typical addition is two to five business days. For an agent with broad action scope or external system access, it takes longer. The governance committee that used to spend forty-five minutes per agent on a quarterly review now spends that time on pre-deployment documentation for each new agent. The difference is that the forty-five minutes now happens before the agent is running, and the questions it asks are organizational rather than procedural.

On a Quarterly Cadence

Every quarter, the organization runs a posture update against the three layers for every agent in production. The update is triggered by Microsoft's quarterly Agent 365 and Copilot capability releases. When Microsoft expands what an agent can do, the organization's authorization document may no longer accurately reflect what the organization has actually authorized. The quarterly posture update is the process that closes that gap before it becomes a governance problem.

The update is not a full re-validation of every agent on every question. It is a targeted comparison: which agents' authorization scope has been affected by this quarter's capability changes? Only those agents require a posture review. The others carry their previous authorization forward with a dated confirmation.

When an Incident Occurs

The organization's incident response begins with a question the authorization document answers immediately: does the agent's behavior fall within or outside its formally authorized scope? That determination defines the response path. If the behavior was within scope, the incident is a technical failure requiring a control update. If it was outside scope, the incident is a governance failure requiring an accountability review with the named business sponsor.

The time to identify the accountable individual: immediate. Not because the investigation is fast, but because the accountability assignment document was written before the incident and names the individual. The board question has an answer before the board meeting is called.

The Stage 3 Maturity Test

A Stage 3 organization can pass one test without preparation: a new governance team member, on their first day, can pick up the documentation set for any agent in production and immediately identify what the agent is authorized to do, what it is intended to accomplish, who owns the consequence if it fails, and when that accountability was last reviewed.

If any of those four pieces requires investigation rather than documentation retrieval, the organization is in Stage 1 or Stage 2. Not as a judgment. As a design question: what organizational work remains to be done?

The Closing Observation

There is an enterprise pattern that repeats reliably in the adoption of any powerful new technology. The organization acquires the capability faster than it designs the governance. The governance gap accumulates invisibly. An incident or a regulator eventually surfaces it. The organization then spends significantly more time and money on retroactive governance than it would have spent building it correctly in the first place.

Agentic AI is running that pattern now, at speed. The agents are deployed. The governance architecture that should sit beneath them has not been built at the same pace.

What makes the current moment different from previous technology adoption cycles is that the regulatory language is moving faster than usual. FINRA's December 2025 guidance requires 'formal review and approval processes' and 'comprehensive documentation throughout.' The April 2026 model risk revision requires 'clear roles and responsibilities with well-defined accountability.' The Federal Reserve requires 'human accountability for decisions.' These are not aspirational statements. They are examiner anchors. They describe the artifacts an examiner will ask for when they find an agent in production.

The Intent Architecture Stack is the organizational design framework that produces those artifacts before they are requested. Three layers, built before the agent goes live, updated when the platform changes, maintained until the agent is retired. When all three are present, the board question has an answer. When any of them are absent, the organization is one incident away from building them under pressure.

> **The platform can lock doors. It does not invent your org chart. That work belongs to the organization, and the organization that does it before the first incident will have a very different conversation with its board than the one that does it after.**

About This Paper and the Framework

This white paper presents original research and framework development by Sougata Roy, published at sougataroy.com and The Governance Gap newsletter. The Intent Architecture Stack is version 1.0, dated April 2026. Every statistic cited is sourced from a named primary source with a publication date. Where a source could not be verified, it was excluded.

Primary sources consulted include: Saviynt 2026 CISO AI Risk Report; Cisco AI/ML Security Report (March 2026); Team8 CISO Village Survey (July 2025); FINRA 2026 Annual Regulatory Oversight Report (December 2025); OCC Semiannual Risk Perspectives (Spring 2024, Spring 2025, Fall 2025); Federal Reserve, OCC, and FDIC revised model risk guidance (April 2026); FINMA State of SupTech Report 2025 (February 2026); Cloud Security Alliance Agentic Trust Framework (February 2026); CSA NIST AI RMF Agentic Profile (March 2026); Microsoft Responsible AI Standard v2; Microsoft Entra Agent ID documentation (learn.microsoft.com/en-us/entra/agent-id); Microsoft Agent 365 overview documentation; Microsoft Purview AI documentation; Microsoft Cloud Adoption Framework AI governance guidance; Ethyca operational AI governance guide (2025-2026); NTT Data AI responsibility gap research; Norval, Cobbe and Singh (2022), Data and Policy; British Columbia Civil Resolution Tribunal, Moffatt v. Air Canada, 2024 BCCRT 149; Oso incident registry (April 2026).

Citation format: Roy, S. (2026). Who Owns the Agent? The Intent Architecture Stack Framework White Paper, Version 1.0. sougataroy.com. April 2026.