# Governing Agentic AI at Enterprise Speed

A business solutions white paper for CTOs and CIOs

## Executive summary

Enterprise AI has moved beyond the era of assistive tools and into the era of autonomous and semiautonomous agents. This shift changes the governance challenge in a material way. Traditional governance models were designed for people, deterministic applications, and relatively stable business processes. Agentic systems behave differently. They operate dynamically, interact with enterprise data at machine speed, make tool calls, retrieve information from multiple sources, and may initiate actions before a human reviewer has time to intervene.

For CTOs and CIOs, this creates a new business mandate. The organization must be able to accelerate AI adoption while preserving trust, control, auditability, regulatory defensibility, and operational resilience. The question is no longer whether enterprises will deploy agents. They will. The more important question is whether those agents can be governed in a way that allows the organization to innovate safely, confidently, and at scale.

The answer is not simply more policy. Policies, principles, review boards, and employee training remain essential, but they are not sufficient on their own. In the agentic era, governance must become executable. It must be enforced at the point where agents access, retrieve, transform, and act on enterprise data. That means moving governance closer to the execution boundary, particularly the data layer.

This white paper sets out a business and technical framework for governing agentic AI.

It explains why traditional governance is no longer enough, how unmanaged agents create both compliance risk and innovation drag, why the data layer is the logical enforcement point, and how CTOs and CIOs can implement a phased roadmap that enables governed acceleration rather than defensive paralysis.

## The new governance problem: Agents need controls where they work

The enterprise AI landscape is changing from systems that advise to systems that act. In earlier phases of AI adoption, a user typically prompted a model, reviewed the output, and decided whether to use it. Governance could therefore rely heavily on acceptable-use policies, training, human review, and application-level controls. Those mechanisms are still useful, but they were designed for an operating model in which humans remained the primary actors.

Agentic AI changes that operating model. An agent may retrieve data, write code, update records, route tickets, generate documents, initiate workflows, summarize sensitive information, call APIs, or interact with systems of record. In many cases, these actions occur with limited or no immediate human oversight. The agent is no longer merely producing an answer for a person to evaluate. It is participating directly in the execution of work.

This creates two categories of enterprise failure:

- **Visible, familiar, and manageable:** Unauthorized access, data leakage, failed audits, privacy exposure, incorrect decisions, regulatory violations, or production incidents caused by excessive permissions or poor control design. These failures are serious, and they are the ones most organizations already recognize.
- **Insidious, paralyzing, and slow to heal:** The second category is less visible but strategically more damaging. When an unmanaged agent causes a material failure, leadership confidence deteriorates. AI initiatives slow down. Every new deployment attracts additional review cycles. Engineering teams hesitate. Security, legal, compliance, and risk functions become more cautious.

> *"Business leaders begin to question whether the organization can safely operationalize AI at all. The data may be restored, and the system may be repaired, but trust in AI-enabled innovation is harder to recover.*

**Rob Feldman**
Chief Legal Officer, EnterpriseDB (EDB)

This is the central problem for technology executives: Unmanaged AI does not merely increase risk—it reduces the organization's capacity to move quickly.

The goal of governance, therefore, should not be to slow AI adoption. The goal should be to create an architecture and operating model that allow the enterprise to move faster because controls are already built into the way agents operate.

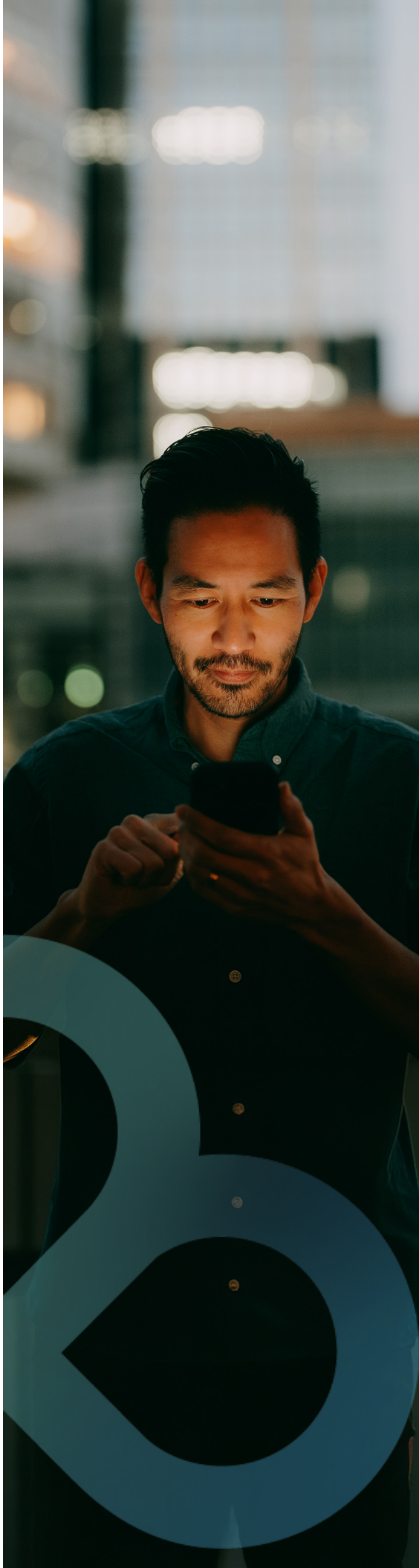
## Traditional governance models fall short because the assumptions no longer work well

Most enterprise governance systems rest on assumptions that made sense in a human-centered technology environment. They assume that the primary actor is a person or a deterministic application. They assume that review can occur before execution. They assume that data access flows through known systems, follows predictable patterns, and can be monitored through established application and infrastructure controls.

Agentic AI challenges each of those assumptions.

- **The first change is the nature of the actor.** An agent is a digital actor that may act on behalf of a person, a business process, an application, or another agent. Its behavior can vary depending on the prompt, context, retrieved information, model behavior, available tools, memory, and environmental conditions. Unlike traditional software, an agent may not follow a single, fixed execution path. Unlike a human employee, it may operate continuously, at high speed, and across multiple systems simultaneously. As a result, governance cannot depend only on static preapproval. The relevant question used to be simply whether a user or application was allowed to access a resource. The better question now is whether this agent, acting on behalf of this user, for this purpose, in this session, is allowed to perform this action on this data right now. That question is contextual, and it must be answered at runtime.
- **The second change is speed.** Human governance processes operate at human speed. An analyst may run a limited number of queries in a day. An agent may run a comparable number in seconds. A misconfigured permission, prompt injection, flawed tool call, or unauthorized action can propagate before a human reviewer even sees an alert. This creates what can be called the millisecond problem: Governance that depends on delayed review or after-the-fact analysis cannot reliably prevent agent-speed failures.
- **The third change is data flow.** Enterprise data no longer moves only through structured applications and reporting systems. Agentic AI relies on embeddings, vector stores, retrieval-augmented generation, context windows, document repositories, logs, APIs, memory, and databases. A document uploaded for one purpose may later influence multiple agent responses. Sensitive information may be retrieved, summarized, embedded, or passed into a context window in ways that traditional application controls do not fully capture.

These changes do not make existing governance obsolete. They do, however, expose its limits. Principles, policies, and oversight bodies define intent. Agentic AI requires an additional layer: deterministic enforcement at the point of action.



## The business consequence of ungoverned agents: Pace and scale of exposure are exponentially higher

The risks associated with agentic AI are often described in technical or legal terms, but CTOs and CIOs should understand them as business risks. Unauthorized data access, privacy exposure, copyright issues, discriminatory outputs, poor decision traceability, excessive privileges, insecure tool use, and unapproved changes to systems of record all create potential legal, operational, financial, and reputational consequences. Existing legal frameworks around negligence, privacy, intellectual property, discrimination, and duty of care will continue to apply even as the technology changes.

What is different is the pace and scale at which harm can occur. AI does not remove the need for traditional enterprise protections. It increases the importance of those protections because the system can act faster, affect more data, and create downstream consequences more quickly.

Organizations that fail to govern agents are likely to fall into one of two patterns.

Some will avoid deploying meaningful AI capabilities because they fear production incidents, unclear regulatory expectations, customer harm, or reputational exposure. This path leads to AI paralysis. The organization remains formally safe but strategically slow.

Others will move aggressively, without sufficient control. They will create an unmanaged agent workforce, often through a combination of business pressure, fragmented tooling, shadow AI, and unclear ownership. This path may produce short-term productivity gains, but it creates accumulating risk. Eventually, a failure will force a more restrictive posture, and the organization will lose momentum.

> *"The more durable path is governed acceleration. In this model, agents are not blocked from acting. They are identified, scoped, monitored, constrained, audited, and improved through a control architecture designed for machine-speed execution. The enterprise moves faster because leadership, engineering, legal, security, and business stakeholders trust the operating model.*
>
> **Rob Feldman**
> Chief Legal Officer,
> EnterpriseDB (EDB)

## Governance must move to the data layer

The critical architectural shift is that governance must move closer to where agents act. In many enterprise environments, that point is the data layer.

Agents create value by interacting with data. They query it, retrieve it, summarize it, classify it, transform it, and—in more advanced use cases—update it. If the enterprise cannot enforce policy at the point where an agent touches data, then governance remains largely advisory. A policy that says an agent should not access a certain class of data is meaningful only if the system can deny that access when the agent requests it. A principle that says AI must be transparent is meaningful only if the organization can reconstruct what the agent did, what data it touched, which user it acted for, which intent it pursued, and which outcome resulted.

This requires a shift from static access control to contextual authorization. Traditional access control asks whether a user or application is allowed to access a resource. Agentic governance must evaluate identity, delegation, purpose, action, data sensitivity, business intent, session context, and risk. The control decision must account for both who is acting and why the action is being attempted.

This is also why deterministic controls are essential. Agent behavior may be probabilistic, but governance cannot be. The organization should not rely on the model to choose to follow policy. The policy must be enforced by the system. Role-based access control, attribute-based access control, row-level security, column-level security, data classification, masking, filtering, tool permissions, policy-as-code, audit trails, intent validation, and rollback mechanisms are all part of the governance fabric.

> *"The objective should not be to eliminate uncertainty from AI behavior. That is unrealistic. The objective is to ensure that uncertain behavior occurs within controlled, observable, and enforceable boundaries.*

**Priyanka Jain**
VP of Product, EnterpriseDB (EDB)

## The digital leash as an operating concept: The five basic architectural requirements

A practical way to think about agentic governance is the idea of a digital leash.

The purpose of the leash is not to prevent the agent from doing useful work. It is to define how far the agent can go, what it can touch, what it can change, what requires escalation, and how the organization can reconstruct events if something goes wrong.

- **The first requirement is verifiable agent identity.** Agents should not operate through generic service accounts with broad permissions. Each agent should have a unique identity and an assigned owner. Its approved purpose, permitted tools, authorized data scope, risk classification, and lifecycle status should be known. Without identity, there is no reliable accountability.
- **The second requirement is on-behalf-of context.** Agents frequently act for users. If an agent performs an action for a sales representative, support engineer, developer, analyst, or executive, the system should preserve that relationship. The agent's authority should be derived from both its own scope and the authority of the user or process it represents. This context is essential for access control, audit, escalation, and incident response.
- **The third requirement is runtime policy enforcement.** The system must be able to evaluate policy at the moment of action. It should block restricted data access, deny actions that exceed the agent's scope, require approval for high-impact operations, limit retrieval from sensitive repositories, and prevent write actions where intent and authority do not align.
- **The fourth requirement is observability.** Technology leaders need a reliable record of what the agent attempted, what it accessed, what tools it called, what prompts or retrieved information influenced the action, what policies were evaluated, what was allowed or denied, and what changed as a result. This is not merely a compliance requirement. It is the basis for operational confidence.
- **The fifth requirement is reconstruction and feedback.** When an agentic system fails, the organization needs the ability to investigate the incident, understand causality, improve controls, refine policies, and prevent recurrence. This creates a learning loop. Governance becomes an improvement mechanism as well as a control function.

## Governance requires cross-functional ownership

Agentic AI governance cannot belong to a single function. Legal may be a natural home for policy stewardship because organizations already rely on legal teams to interpret obligations, manage risk, and translate principles into practice. But legal cannot govern agentic AI alone. The operating model must bring together technology, data, security, privacy, product, engineering, compliance, risk, operations, and business leadership.

### 01 Define AI principles

These principles serve as the organization's internal constitution for AI use. They should articulate how the enterprise will use AI responsibly, safely, transparently, and in alignment with customer trust. They should also evolve. As AI capabilities, customer expectations, regulatory requirements, and enterprise use cases change, the principles should be reviewed and updated.

### 02 Establish an AI governance team

This team should not exist only to approve or reject projects. Its more important role is to create scalable patterns that allow the business to adopt AI responsibly. It should define risk tiers, acceptable-use standards, control requirements, escalation paths, incident review processes, and production-readiness criteria. It should also help ensure that AI governance is not separated from the way systems are actually designed and operated.

### 03 Enable the workforce

Employees need training, approved tools, practical examples, and clear guidance. Many employees do not fear AI itself; they fear being unable to use it effectively or safely. If organizations do not provide sanctioned pathways, employees will often find their own. A strong AI literacy program reduces shadow AI, improves adoption quality, and turns governance into a shared enterprise capability.

## A technical control architecture for agentic AI

A mature agentic AI governance architecture should begin with an agent registry. The registry establishes a single inventory of agents, their owners, purposes, permissions, data access scopes, approved tools, risk levels, and lifecycle status. This becomes the foundation for accountability and operational management.

The registry should connect to a policy engine that translates business, legal, security, and data governance requirements into executable controls. These controls should be evaluated dynamically based on agent identity, user context, intent, action, and data sensitivity. The more policy remains disconnected from execution, the less effective it will be.

The data layer should then enforce those controls. This may include role-based access, attribute-based access, row-level security, column-level security, data masking, data classification, query restrictions, policy evaluation, and audit logging. For agentic systems, these controls should be applied to users and applications—and also to agents acting on behalf of users.

Prompt injection defense and tool-misuse protection should be part of the architecture, particularly when agents interact with untrusted documents, external content, code repositories, or APIs. These controls are necessary, but they should not be treated as sufficient. Model-layer guardrails reduce risk, but execution-layer enforcement determines what the agent can actually do.

Memory governance is also essential. Agent memory can become a hidden channel for data propagation. The organization needs controls over what agents can remember, retrieve, persist, summarize, share, or delete. Without memory governance, sensitive data may outlive its intended context and influence future actions in ways that are difficult to see.

As agents gain write authority, rollback becomes critical. Write actions should be scoped, monitored, reversible, and subject to escalation when risk thresholds are met. The enterprise should be able to undo incorrect or unauthorized actions, particularly when agents update records, alter configurations, or interact with systems of record.

Finally, audit and reconstruction capabilities must be designed into the platform. Logs should support a complete view of agent sessions, including prompts, retrieved context, tool calls, policy decisions, data accessed, outputs generated, and actions performed. Drift and anomaly detection should identify unusual access patterns, unexpected behaviors, excessive query rates, privilege misuse, or deviations from approved purpose.

> *"Intent validation becomes increasingly important as agents move beyond read-only use cases. The system should evaluate whether the requested action aligns with the agent's approved purpose and the authority of the user or workflow it represents. An agent designed to summarize support tickets should not suddenly make production configuration changes simply because a prompt, retrieved document, or tool pathway suggests doing so.*
>
> **Priyanka Jain**
> VP of Product,
> EnterpriseDB (EDB)

## A phased roadmap for implementation

Agentic AI governance does not need to be perfected before an organization begins. In fact, trying to solve everything at once can slow adoption and create unnecessary complexity. A phased approach allows enterprises to build confidence, prove controls, and expand responsibly.

**The first phase should focus on foundation-building through low-risk and read-only use cases.** These may include internal productivity agents, read-only analytics assistants, knowledge discovery over approved data products, internal documentation agents, and insight agents that do not have write authority. The purpose of this phase is to establish the basic governance architecture. Each agent should receive a verifiable identity, a defined owner, a clear purpose, and a bounded scope. On-behalf-of context should be preserved. Existing access policies should be enforced at the data layer, and audit trails should be established so that the organization can see what the agent did. This phase is not about perfection. It is about creating the governed foundation on which later use cases can build.

**The second phase should expand into controlled workflow automation.** At this stage, agents may generate documents, route tickets, update records in limited domains, prepare orders, orchestrate internal workflows, or recommend actions. The control environment must now evaluate what agents are trying to do as well as what they can see. Dynamic intent validation becomes more important. Prompt injection defenses and memory governance should be strengthened. Limited write actions may be introduced, but they should be scoped, monitored, and reversible. Escalation thresholds should be clear, and rollback mechanisms should be tested. The business objective of this phase is to prove that governance holds when agents begin to participate in execution.

**The third phase involves high-stakes deployment.** These use cases may include customer-facing agents, financial workflows, regulated processes, production system changes, compliance-sensitive decision support, and system-of-record interactions. At this level, on-behalf-of context must be enforced end to end. Runtime policy enforcement must operate at execution boundaries. Write actions must be strictly governed. Real-time anomaly detection should flag drift or suspicious behavior. Full session reconstruction should be available for investigation, audit, and continuous improvement. The organization should also be prepared to engage customers, regulators, and auditors with evidence of how its AI governance model works.

This phased approach allows the enterprise to build from confidence rather than fear. It also prevents governance from becoming an abstract exercise. Each phase produces operational learning that improves the next.

## Measuring success

The success of agentic AI governance should not be measured only by the absence of incidents. A governance model that prevents all AI adoption is not successful. The better measure is whether the organization can increase AI use while maintaining control and trust.

Useful indicators include the percentage of agents with verified identities and assigned owners, the percentage of agent actions that preserve on-behalf-of context, the share of agent data access governed at the data layer, policy decision latency, the number of high-risk actions blocked or escalated, the time required to reconstruct an incident, and the mean time to detect anomalous behavior. Organizations should also track the number of governed use cases promoted from pilot to production, employee AI literacy participation, reduction in shadow AI, rollback effectiveness, and audit readiness for agentic workflows.

These measures matter because they connect governance to business enablement. The objective is to give leadership confidence that AI can be deployed responsibly, give engineering teams repeatable patterns for delivery, and give risk functions evidence that controls are working.

## The strategic implications for CTOs and CIOs

The next phase of enterprise AI competition will not be won simply by the organizations with the most models, experiments, or agents. It will be won by organizations that can operationalize AI safely at scale.

For CTOs, the mandate is architectural. They must design systems in which agents can act within deterministic, observable, and enforceable boundaries. This requires identity, policy enforcement, data-layer controls, auditability, rollback, and monitoring to be built into the execution fabric.

For CIOs, the mandate is operational. They must ensure that AI adoption improves productivity without creating unmanaged risk, fragmented tooling, uncontrolled data flows, or compliance exposure. This requires governance processes that are practical enough for the business to use and strong enough for leadership to trust.

For both roles, the central insight is that governance must become part of how AI systems run, not merely part of how AI projects are reviewed. Organizations that achieve this will be able to move faster because their control environment supports innovation. Organizations that do not will either slow themselves through fear or expose themselves through unmanaged adoption.

## Recommended path forward

Technology leaders should begin by creating a clear inventory of current and planned agents. The organization needs to know which agents exist, who owns them, what they are intended to do, what systems and data they touch, which tools they can call, and what level of risk they introduce. Without this inventory, governance will remain reactive.

The next step is to define agent identity standards. Generic service accounts and excessive privileges are not sustainable in an agentic environment. Every agent should be identifiable, scoped, and accountable. The enterprise should also establish on-behalf-of context so that agent actions can be connected to the user, process, or business purpose they represent.

Leaders should then map existing data-layer controls and identify where they can be extended to agents. Many organizations already have role-based access, row-level security, data classification, masking, and audit capabilities. The challenge is to make these controls work in an agentic context and to convert policies into executable enforcement wherever possible.

At the same time, the organization should formalize cross-functional governance. AI principles, governance teams, acceptable-use standards, workforce training, and incident review processes provide the management structure around the technical architecture. Governance should be positioned as an enabler of responsible adoption, not as a blocker.

Finally, enterprises should start with low-risk governed deployments and expand deliberately. Read-only agents and internal productivity use cases provide a practical foundation. Controlled workflow automation can follow once identity, policy enforcement, auditability, and rollback capabilities are proven. High-stakes customer-facing and system-of-record use cases should come only after the operating model has demonstrated that it can govern both data access and agent action.

## Conclusion

Agentic AI forces enterprises to rethink governance. While every organization needs principles, policies, and human oversight, the question now is whether those principles and policies can be enforced at the speed and execution boundary at which agents operate.

The right response is not to slow AI adoption indefinitely, nor is it to allow unmanaged agents to proliferate. The right response is governed acceleration. In this model, agents are identifiable, scoped, monitored, policy bound, auditable, and capable of being improved through feedback and reconstruction.

For CTOs and CIOs, this is the business solution: Build an AI governance architecture that protects the enterprise while increasing its ability to innovate. The organizations that master this discipline will do more than merely reduce risk. They will create the trust, control, and operational confidence required to scale AI as a core enterprise capability.

## Requirements that matter

The following requirements should be treated as practical evaluation criteria for any enterprise platform, architecture, or operating model intended to support governed agentic AI. They are the requirements that determine whether governance is merely documented or actually enforceable in production.

| Requirement | Why it matters | What good looks like |
|---|---|---|
| **Verifiable agent identity** | Agents must be accountable digital actors, not anonymous processes or over-permissioned service accounts. | Every agent has a unique identity, owner, approved purpose, risk classification, and lifecycle status. |
| **On-behalf-of user context** | Agent actions need to reflect the authority and accountability of the person, role, or workflow they represent. | Every action can be tied to the agent, the user or process it served, the session, and the business purpose. |
| **Runtime policy enforcement** | Policies must be enforced when the agent acts, not merely reviewed before deployment. | The system evaluates access, intent, action, data sensitivity, and risk at execution time. |
| **Data-layer controls** | Agents create value by touching data, so governance must operate where data access and action occur. | Row-level security, column-level security, masking, classification, access control, and audit policies apply to agent activity. |
| **Dynamic intent validation** | Agent permissions should depend not only on identity but also on what the agent is trying to do in the moment. | The platform can determine whether a requested action aligns with the agent's approved purpose and user authority. |
| **Scoped write authority** | Write actions create higher enterprise risk than read-only access. | Agents can update, create, or delete only within approved boundaries, with escalation for high-impact actions. |
| **Rollback and reversibility** | Agentic errors can occur quickly and at scale, so recovery must be designed in. | Material changes can be reversed, reconstructed, or remediated without relying on manual guesswork. |
| **Full session traceability** | Auditability depends on knowing what happened, not merely that something happened. | Logs capture prompts, retrieved context, tools used, data accessed, policy decisions, outputs, and downstream actions. |
| **Prompt injection and tool-misuse defense** | Agents are vulnerable to malicious or unintended instructions, especially when interacting with external or untrusted content. | Guardrails detect, isolate, and block attempts to override policy or misuse tools. |
| **Memory governance** | Agent memory can become a hidden channel for sensitive data persistence and reuse. | The organization controls what agents can store, retrieve, summarize, share, retain, and delete. |
| **Anomaly and drift detection** | Agent behavior can shift over time as context, tools, data, and model behavior change. | Monitoring detects unusual access patterns, excessive activity, privilege misuse, and deviation from approved purpose. |
| **Cross-functional governance ownership** | Technical controls need legal, risk, security, product, data, and business alignment. | A governance team defines standards, risk tiers, escalation paths, and production-readiness criteria. |
| **Workforce AI literacy** | Employees need safe, approved ways to use AI effectively. | Training, approved tools, and clear guidance reduce shadow AI and improve adoption quality. |
| **Incident reconstruction** | The enterprise must be able to learn from agent failures and improve controls. | Post-incident review can reconstruct causality and feed improvements back into policies, controls, and training. |
| **Measurable governed acceleration** | Governance should increase confidence and deployment velocity, not become a static compliance artifact. | Leaders can track adoption, control effectiveness, incident response, audit readiness, and business impact. |

# About the authors

## Rob Feldman, Chief Legal Officer, EDB

Rob Feldman is responsible for the worldwide legal and compliance functions at EDB, including EDB's Responsible AI initiatives. An experienced executive and lawyer, he builds high-performing legal teams to support growing technology companies in dynamic business and regulatory environments. He spent more than a decade in private practice as a technology company litigator, focused in securities fraud defense, intellectual property disputes, and government and internal investigations. He also serves on the U.N. Global Compact Legal Council, providing strategic guidance on global regulatory environments to help businesses drive transformative, long-term impact.

## Priyanka Jain, VP of Product, AI & Governance, EDB

Priyanka Jain is responsible for the product, data, and AI governance functions at EDB, where she is defining how enterprises govern agentic AI at the data layer. A seasoned product executive, she has spent her career at the intersection of data, analytics, and regulation, building the platforms that let organizations adopt advanced technology without surrendering control or trust. She spent almost a decade at IBM, leading data management and advanced analytics, and went on to build data and AI platforms for some of the most heavily regulated industries in the world, including healthcare and pharmaceuticals.

EDB POSTGRES AI

EDB Postgres AI is the first open, enterprise-grade sovereign data and AI platform, with a secure, compliant, and fully scalable environment, on premises and across clouds. Supported by a global partner network, EDB Postgres AI unifies transactional, analytical, and AI workloads, enabling organizations to operationalize their data and LLMs where, when, and how they need them. For more information, visit enterprisedb.com