From AI Pilots to Autonomous Operations

Why Enterprises Need a Context Engine

AI adoption is moving faster than the operating models and architectures required to support it. As enterprises shift rapidly from AI experimentation to implementation, too many organizations still rely on systems and processes built for a pre-AI era.

Leaders see this tension directly. Scaling AI exposes persistent issues: siloed data, unclear ownership, incomplete governance, and growing concerns about trust, risk, and cost. So, at the same time many companies are beginning to report productivity gains created by AI, they are also seeing opaque cost behavior and data practices that still impede the successful deployment of real-time, autonomous AI at scale.

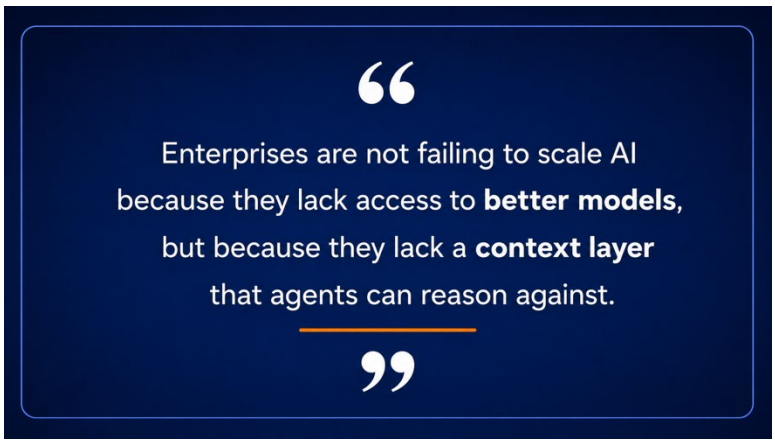
By this point, the central challenge in enterprise AI is not access to better models. It is building governed, reusable context and the operating discipline to manage it. That requires a unified architectural foundation — a context engine that brings fragmented data, policies, and expertise into a reliable knowledge layer for humans and AI agents alike across cloud, on-premises, and sovereign environments.

Enterprise AI Needs a New Operating Model to Support It

AI adoption at the workforce and project levels is accelerating. Recent enterprise surveys report that access to approved AI tools has increased from under 40% to around 60% among workers over the past year, and roughly two-thirds of organizations cite productivity and efficiency gains from AI. Yet enterprises are scaling AI usage faster than they are scaling the knowledge layers needed to make AI dependable.

The adoption-readiness gap is evident in the number of AI initiatives that stall. Most enterprises can stand up an internal copilot, a domain assistant, or an early task-oriented agent. But challenges emerge when those pilots must ramp up to meet the requirements for reliable enterprise-scale deployment. In Deloitte's State of AI in the Enterprise 2026 study, a large majority of organizations said they expect to customize autonomous agents for their business, but only a small minority reported having mature governance models in place, and many expect it will take a year or more to fully implement AI governance.

The failure modes are consistent. Governance teams need to understand which data, prompts, and policies shaped an agent's output. Operations teams need predictable performance, rollback paths, and monitoring. Finance teams need AI costs to remain predictable and transparent as workloads scale.



These practical shortcomings tie in with a critical insight that most organizations learn the hard way: Enterprises are not failing to scale AI because they lack access to better models, but because they lack a context layer that agents can reason against — one that is built with governance and operational consistency in mind. Buying a more capable model does not close the governance gap. It does not resolve inconsistent policies, fragmented semantics, or the absence of decision traces. It simply exposes the architectural shortfall at higher speed and greater cost.

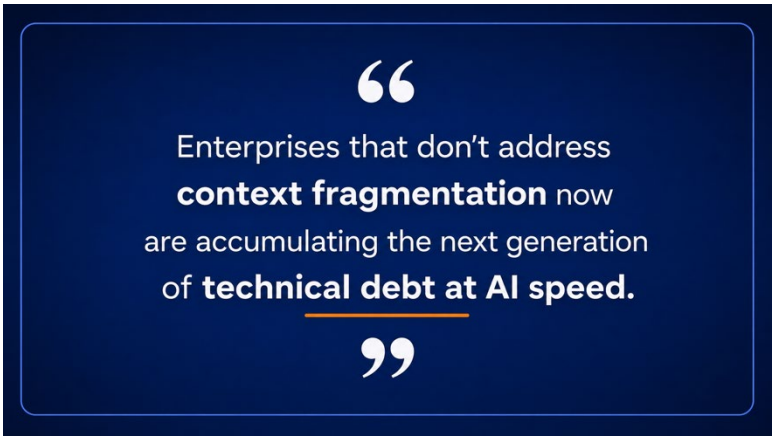
Organizations that fail to address these problems are likely to keep cycling through pilots without reaching production. By contrast, the enterprises that successfully move from pilots to autonomous operations will be those that build shared context, shared control, and shared accountability into the platform from the beginning, then exhibit good operating discipline as they implement and refine new AI-assisted functions on that platform.

Context Fragmentation Is a Structural Barrier

These operational issues typically stem from a structural one: Enterprise context is fragmented. Core data is distributed across data warehouses, data lakes, line-of-business applications, SaaS platforms, and unstructured repositories. Policies and controls live in countless documents, access rules, and manual procedures. Worse, business meaning — often including the definitions of key terms — is often implicit and difficult for machines to interpret. These gaps reveal the truth that existing architectural designs were never intended to support real-time, autonomous AI, especially as it extends into edge environments and sovereign regions.

It's not that no progress on AI is being made. Much of the progress is happening at the workflow level, where employees are already weaving AI into their day-to-day tasks as fast as they can in the areas they control. Unfortunately, this also means that employees are stitching together AI-enabled ways of working faster than organizations are redesigning the systems that support them. In terms of architecture, teams are adding model access, copilots, vector layers, and frameworks faster than the enterprise can build a shared knowledge layer to govern them. The result is shadow AI at the workflow level and shadow architecture at the platform level.

The downsides are increasingly hard to ignore. Every agent that rebuilds its own context assumptions is a governance liability. Every duplicate retrieval pipeline is a cost sink. Every hallucination traced to stale context is a remediation expense. Enterprises that don't address context fragmentation now are accumulating the next generation of technical debt at AI speed.



Data Stacks for Humans Don't Meet the Needs of Agentic AI

Most enterprise data systems were designed for people, not machines. Traditional warehouses, lakes, and BI platforms excel at producing dashboards and batch analytics for human decision-makers, but they cover only part of what autonomous operations require. Business rules and exceptions are often buried in spreadsheets, e-mail threads, ticketing systems, or scripts owned by a small set of experts. As a result, they're not available as structured, machine-readable context that agents need for continuous execution.

Agents require more than raw data. Autonomous systems depend on enterprise knowledge, including governed data, operating models, and accumulated experience enriched with semantics, relationships, lineage, policies, and constraints, so they can sense, decide, and act within enterprise guardrails. This is increasingly how platform vendors are describing the next stage of enterprise AI. One recent example comes from

Teradata, which defines autonomous knowledge as turning that governed context into repeatable understanding, decisions, and actions.

In practice, that requires explicit definitions of entities and relationships, consistent policy enforcement, context stores that combine structured and unstructured sources, and decision traces that record how outputs were produced. Business context, meaning, relationships, and lineage are embedded in the platform so AI systems can reason and act reliably within enterprise policies and controls.

Consider this contrast: A personal AI assistant can use a general-purpose model with minimal enterprise grounding to do something straightforward like summarizing a report or gathering background information. This is quite different from an operational agent that can execute and adjust business workflows, enforce policies, coordinate across systems, or recommend a decision and explain why. Agents require fundamentally different grounding, and organizations that do not achieve that grounding run the risk that they will successfully roll out only personal AI tools while the harder organizational use cases remain stranded at the pilot stage.

THE CONTEXT ENGINE: ENTERPRISE-GRADE KNOWLEDGE FOR AI

As agents begin to operate continuously across systems, the absence of a shared knowledge layer becomes a bigger problem. Without it, scaling AI tends to create fragmentation and entropy. Every new agent risks becoming another isolated island of logic, and every local context decision adds more complexity to govern and explain.

We must reconcile the push toward autonomous AI operations with the growing recognition that access to contextual knowledge, rather than model capability, is the limiting factor. “Autonomous AI + Knowledge” is the category frame that captures this reality. It describes systems in which AI operates continuously within a governed enterprise understanding. The context engine is the architectural pattern that makes it operational.

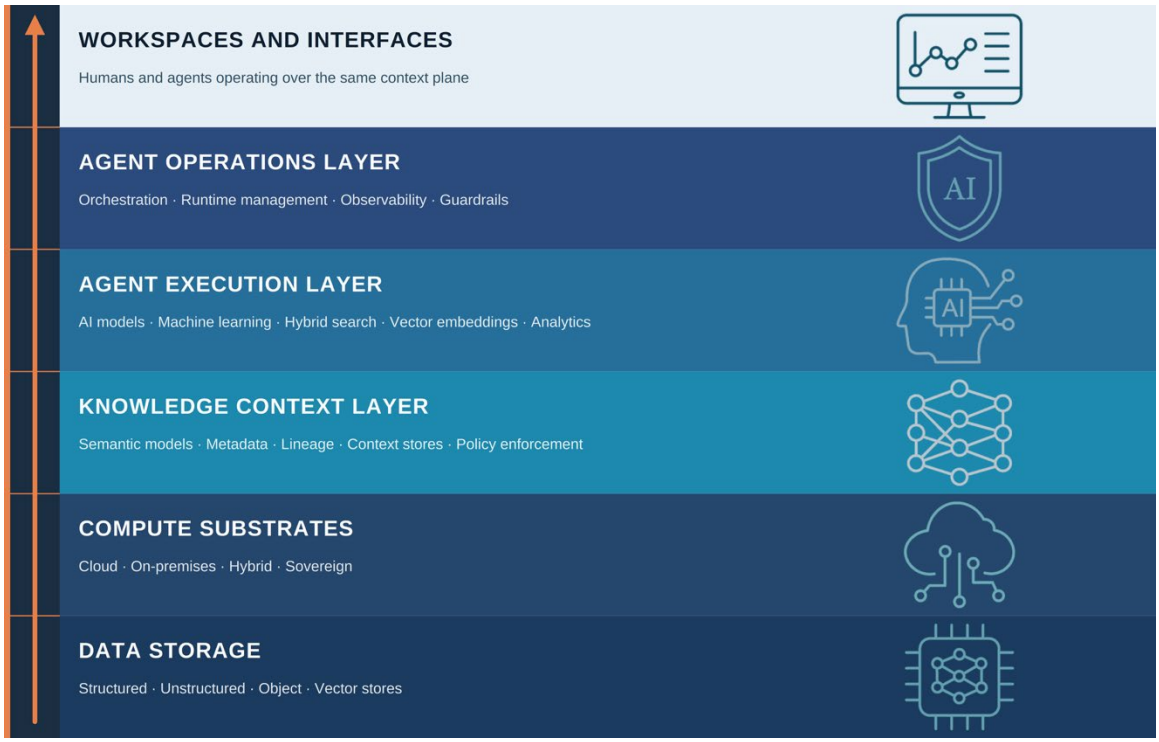
Autonomous AI + Knowledge is a broader frame than a model platform, a data platform, or an agent framework on its own. The value of this framing is that it clarifies what must come together. Knowledge provides the grounded business context, meaning, relationships, lineage, and policies that allow AI systems to reason and act reliably. Autonomy provides the continuous optimization, adaptation, and execution needed to move from isolated outputs to real systems of action. Teradata’s platform overview is useful here because it explicitly frames Autonomous AI + Knowledge as a convergence

of data, AI, and execution layers. Together, these elements point to an enterprise architecture where AI is intertwined with governed knowledge rather than layered loosely on top of data.

A Context Engine Enables Knowledge Synthesis

A context engine is not a feature, an interface, or a standalone product category. Rather, it is a knowledge orchestration layer — a runtime service that combines data freshness, semantic grounding, policy enforcement, and access control so that agents always operate against a consistent, governed understanding of the enterprise.

Figure 1: A Context Engine



A context engine unifies data, knowledge, agent execution, and operations so humans and agents share the same governed context in solo, team, and multi-agent scenarios.
(Credit: Moor Insights & Strategy)

The foundation is a unified data and compute fabric across cloud, on-premises, hybrid, and sovereign environments. On top of that is a knowledge context layer comprising semantic models, metadata, lineage, context stores, and policy enforcement. Above that is an agent execution layer where knowledge synthesis occurs, leveraging machine learning models, large language models, hybrid search, vector embeddings, and

analytics. In Teradata's case, this maps to in-database AI and ML capabilities over governed data. The uppermost agent operations layer includes orchestration, runtime management, observability, and guardrails. Finally, workspaces and interfaces allow humans and agents to operate over the same context plane. Several platform vendors, including Teradata, are beginning to use this pattern to organize their data, AI, and execution capabilities.

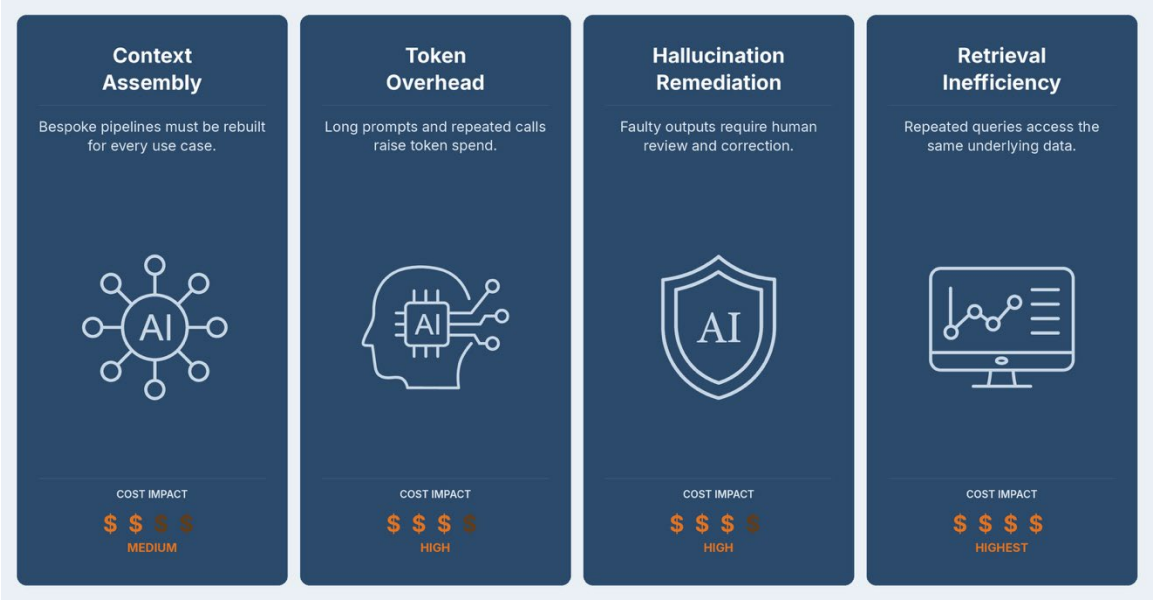
WHY CONTEXT STANDARDIZATION MATTERS

As we have seen, autonomous AI requires common context and common control. A context engine should coordinate actions to make behavior more predictable and governance more scalable across agents and applications.

This functionality goes beyond metadata catalogs —which are optimized for human discovery and governance reporting — by making knowledge machine-consumable in real time. It likewise improves on vector databases by integrating vector search with relational data and governance controls, and it improves on stateless, session-scoped RAG architectures by providing persistent, enterprise-scoped knowledge with provenance. In practice, that means that context accumulates and can be governed consistently across every agent and every query.

The cohesive design of a context engine also circumvents the problems that many enterprises face when they assemble a patchwork of agent frameworks, standalone vector databases, cloud-native AI services, and traditional data platforms. Besides solving for enterprise context, a context engine avoids other challenges including cloud vendor lock-in, cost volatility, and the "integration tax" that comes when teams must stitch together multiple control planes, duplicate context across tools, monitor activity in disconnected dashboards, and try to reconcile policies after the fact.

FIGURE 2: AI CONTEXT GAPS



Where AI context gaps turn into cost. Note that the cost impact of token overhead can be low for single-turn AI jobs, but it tends to be high for agentic AI.
(Credit: Moor Insights & Strategy)

THE TOKENOMICS OF CONTEXT: WHAT RUNNING AI WITHOUT A KNOWLEDGE LAYER COSTS

For enterprise decision-makers, the context problem also has a direct financial signature. These costs show up in token spend, engineering overhead, and remediation work as AI usage scales. Without a persistent, semantically grounded context store, agents must reassemble enterprise knowledge from scratch at inference time — with longer prompts, more retrieval calls, and higher token counts per query. Engineering teams end up building bespoke context pipelines for every use case, each with its own development cycle, maintenance overhead, and ongoing data refreshes. Meanwhile, agents must repeatedly query source systems for the same underlying data because nothing maintains a shared state or freshness.

The operational impact is often worse than these technical inefficiencies. Faulty outputs require human review and correction, and sometimes expensive downstream remediation if bad agentic decisions get implemented. This is only compounded in regulated environments, where a hallucinated decision trace or a policy misapplication could carry stiff compliance consequences. With all of this in mind, we see that a

context engine is not only a vital piece of technical infrastructure, but also a cost management instrument.

What Enterprise Leaders Need to Do Next to Move from Pilots to Autonomous Operations

A context engine should reduce both cost and complexity by standardizing the environment in which agents are built and run. Shared workspaces allow teams to explore data, define context, test policies, and compose agents against a common substrate, while lifecycle tools support versioning, canary releases, rollbacks, and telemetry feedback into models and behavior. This reusable knowledge layer makes it easier for use cases to move from bespoke pilots to governed production.

Microsoft's 2026 Work Trend Index shows that durable value comes from redesigning workflows, management practices, and human-agent collaboration around AI rather than layering tools onto existing patterns (see Moor Insights & Strategy's analysis of the WTI). A context engine should provide the common fabric for that redesign, grounding agents in shared context, shared control, and shared accountability.

For enterprise leaders, the priority is not to pursue autonomy everywhere at once. It is to build the conditions that make autonomy trustworthy. That begins with assessing context maturity and then prioritizing reusable components. Data products, semantic models, policy patterns, and agent workflows that can be applied across domains create more value than bespoke pilots built in isolation. Leaders should also quantify the hidden cost of context fragmentation by auditing token consumption patterns, context assembly overhead, hallucination remediation, and retrieval redundancy across current AI workloads, which typically will reveal that the economics of a shared knowledge layer are economically preferable to piecemeal tooling.

Finally, institutionalizing AI services, AgentOps, and ModelOps as shared enterprise disciplines — complete with version control, monitoring, rollback paths, governance review, and economic oversight — should help companies move successfully from pilots to autonomous operations by embedding shared context, shared control, and shared accountability into the platform from the beginning.

Turning Autonomous AI + Knowledge into Practice

Enterprise AI is moving quickly, and the center of gravity is shifting toward governed enterprise knowledge. Autonomous AI + Knowledge ties continuous execution directly

to a shared, explainable context layer grounded in that enterprise understanding. Vendors such as Teradata are adopting this frame as they design platforms where AI operates continuously over governed enterprise knowledge, but the real decision sits with leaders: Will they accept a limited approach where AI stays scattered across tools and teams, with rising token spend and remediation costs and little impact on core operations? Or will they invest in a context engine now to turn fragmented data, policies, and workflows into a shared substrate for humans and agents?

Important Information About This Paper

Contributors

Mel Brue, Vice President and Principal Analyst, Modern Work, HRM/HCM, FinTech
Mike Leone, Vice President and Principal Analyst, Enterprise Data Platforms, Analytics, and Governance

Publisher

Patrick Moorhead, CEO, Founder and Chief Analyst at Moor Insights & Strategy

Inquiries

Contact us if you would like to discuss this report, and Moor Insights & Strategy will respond promptly.

Citations

This paper can be cited by accredited press and analysts but must be cited in-context, displaying the author's name, author's title, and "Moor Insights & Strategy." Non-press and non-analysts must receive prior written permission by Moor Insights & Strategy for any citations.

Licensing

Disclosures

Teradata commissioned this paper. Moor Insights & Strategy provides research, analysis, advising, and consulting to many high-tech companies mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

Disclaimer

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. Moor Insights & Strategy disclaims all warranties as to the accuracy, completeness, or adequacy of such information and shall have no liability for errors, omissions, or inadequacies in such information. This document consists of the opinions of Moor Insights & Strategy and should not be construed as statements of fact. The opinions expressed herein are subject to change without notice.

Moor Insights & Strategy provides forecasts and forward-looking statements as directional indicators and not as precise predictions of future events. While our forecasts and forward-looking statements represent our current judgment on what the future holds, they are subject to risks and uncertainties that could cause actual results to differ materially. You are cautioned not to place undue reliance on these forecasts and forward-looking statements, which reflect our opinions only as of the date of publication for this document. Please keep in mind that we are not obligating ourselves to revise or publicly release the results of any revision to these forecasts and forward-looking statements in light of new information or future events.