



Closing the Enterprise Context Gap



How Governed Business Context Makes AI More Accurate, Explainable, and Actionable

Enterprise artificial intelligence is undergoing a fundamental transformation, moving from conversational chatbots to autonomous agents that make decisions and execute operational workflows. Yet, many enterprise agent deployments stall before reaching production. These failures rarely stem from inadequate foundation models; they run into an architectural wall: language models operate in a business context vacuum.

Attempting to resolve this structural data challenge through basic vector search, isolated metadata catalogs, or extreme prompt engineering is tempting. These are often expensive dead ends. Point tools fail to supply governed, deterministic meaning at runtime. Futurum's 1H 2026 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey shows enterprise agentic AI adoption leveling off at 47%, with more than a quarter of technology leaders citing data integration difficulties as their primary bottleneck.

Breaking through this plateau requires an active, neutral context and execution foundation. By linking distributed data stores, embedding domain-specific semantic models, enforcing runtime governance, and establishing an operational memory loop, this foundation gives autonomous agents the governed business context required not only to understand enterprise data, but to reason, execute, and act reliably across hybrid and multi-cloud environments.

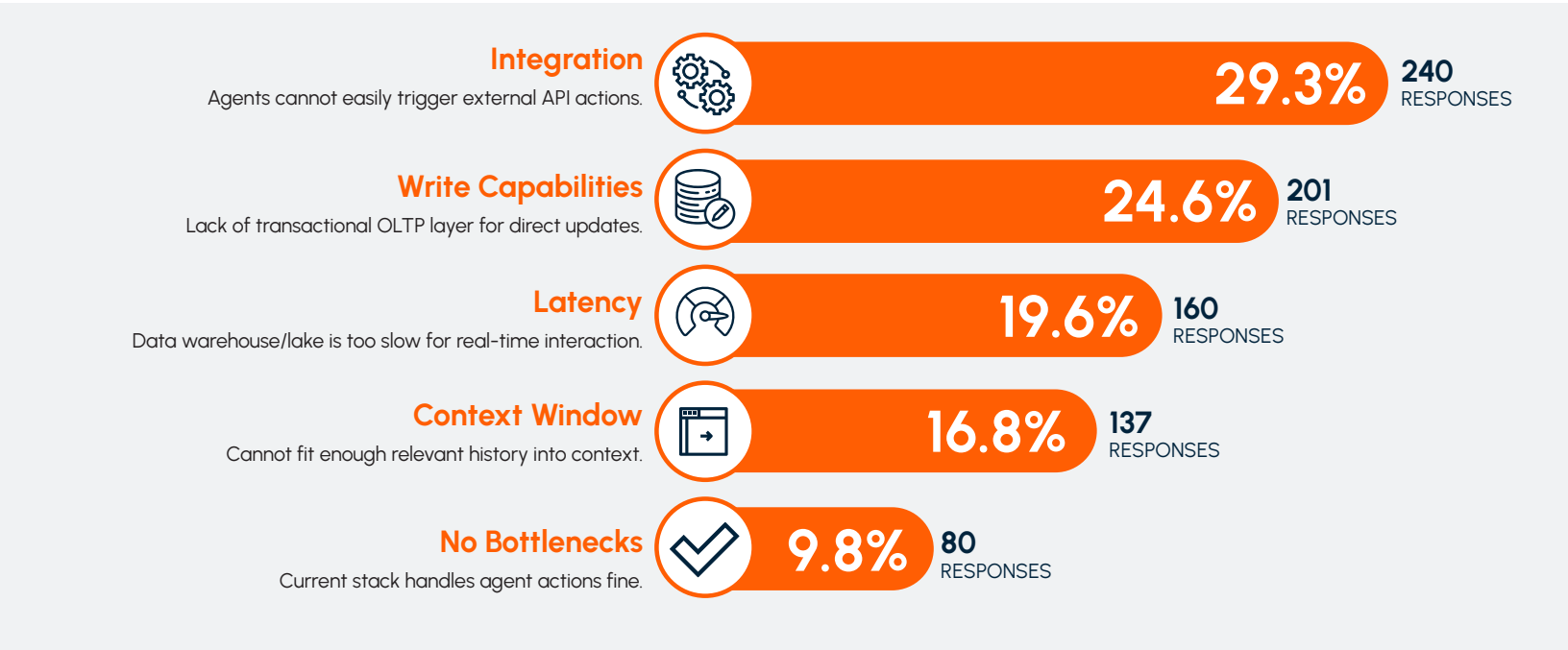
The Reality Check

Why Autonomous AI Stalls at the Production Threshold

Today's enterprise engineering teams are moving past retrieval-augmented generation (RAG) toward agentic architectures that plan analytical workflows, query operational stores, and update records across systems. While reasoning models show linguistic fluency, they break down when dropped into complex enterprise data environments.

The primary point of failure is the lack of a verified, runtime business context. When an autonomous agent queries an enterprise warehouse, it frequently encounters cryptic column names, denormalized transactional tables, and fragmented document stores. A model cannot know whether a metric labeled `churn_rate` reflects 30 days of inactivity, a formal contract cancellation, or a delayed payment cycle. Without clear semantic understanding and explicit, deterministic business rules, agents generate syntactically clean SQL queries that produce inaccurate business results.

Figure 1: Architectural Bottlenecks for Autonomous Agent Deployment



Enterprise decision-makers identify system integration and the absence of transactional write capabilities as the primary architectural obstacles preventing autonomous agents from acting safely on enterprise data.

Source: Futurum Research. 1H 2026 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey.
Question: "What is your biggest architectural bottleneck when building autonomous agents that need to take action (read/write data)?" Base: n=818 qualified decision-makers.

Survey data reflects this infrastructure disconnect (Figure 1). Two hurdles together account for more than half: integration friction (29%) and the lack of safe transactional write-back capabilities (25%). Scaling autonomous AI is not a prompt-crafting exercise; it requires embedding structural enterprise context and transactional boundaries into the execution path.

The Failure of Point Solutions & The Real Economics of Context

Why Prompt Stuffer Hacks Fail and What They Cost

Enterprises bridging this context gap stitch together a patchwork of point tools, only to discover each introduces steep operational compromises (Figure 2).

Vector RAG operates on probabilistic similarity across unstructured text, blind to structured schema relationships, relational joins, and precise metric definitions. Data catalogs store passive metadata, dictionaries, and ownership tags, but operate offline and can't enforce business logic or access policies at query time. Standalone ontologies and knowledge graphs map conceptual relationships well on paper; they run isolated from the data engines that do the computing.

This architectural fragmentation triggers acute "agent amnesia." Human operators make daily decisions using unwritten exception rules, playbooks, and tribal knowledge that remain trapped in human heads or static wikis, so agents lack session continuity. They cannot retain state across multi-turn workflows or adapt to evolving business conditions, leading to repeated user corrections and brittle execution paths.

Figure 2: Operational Compromises

Friction Point	Architectural Mechanism	Economic Impact
Token Overhead	Stuffing system prompts with schemas and raw documentation	Escalating API inference costs and severe query latency
Redundant Engineering	Hand-coding bespoke semantic mappings per agent	Suppressed developer velocity and mounting technical debt
Retrieval Noise	Probabilistic similarity matching without ontologies	Hallucinated queries, excessive retries, compute waste
Security Blindspots	Static access policies detached from the execution plane	Data leakage risks and non-compliance with regulations

The financial toll of these workarounds is substantial. Stuffing context windows with thousands of lines of raw data definition language (DDL), API documentation, and static instructions taxes every inference call, driving up API spend and latency. When context is ambiguous, agents enter trial-and-error retry loops, multiplying compute consumption.

Recognizing prompt stuffing is unsustainable, 44.5% of enterprise technology leaders plan to invest more in dedicated semantic layers over the next 24 months, seeking deterministic business definitions over statistical guesswork. The economics case is direct: every time an agent must infer a complex metric definition without governed context and produces an incorrect query, it retries — feeding large outputs back into its context window, adjusting incrementally, and accumulating inference cost with each attempt. This pattern compounds across agentic workflows. Moving that work from non-deterministic agent reasoning into governed, deterministic query and execution paths — what might be called deterministic query economics — reduces inference overhead, shortens task completion, and improves reliability without per-use-case tuning.¹

¹ Futurum Research.IH 2026 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey

The Architecture

How Governed Context Powers Autonomous Action

Tera Context Engine directly targets these ills, serving as the context foundation of the Teradata Autonomous Knowledge Platform. It converts disconnected data stores into reusable, AI-ready data products and works in concert with Tera Harness, which coordinates intelligent execution, and Tera Agents, which apply specialized enterprise expertise. Together, these capabilities connect governed enterprise context to intelligent execution and action. Rather than treating context as a static text prompt, Tera Context Engine structures enterprise meaning across four operational tiers (Figure 3):

Figure 3: Tera Context Engine Operational Tiers



- Semantic Foundation:** This tier establishes explicit metric definitions, business entities, and ontologies across structured stores. It absorbs existing business logic by connecting with data transformation and BI tools such as dbt and Looker, anchoring agents to verified calculations and consistent KPIs (key performance indicators).
- Organizational Knowledge:** Operational reality rarely lives in database schemas. This layer captures internal procedures, compliance constraints, and human exception playbooks, turning tribal practices and reference documentation into structured parameters that guide agent reasoning.
- Operational Memory:** Agents require persistent, multi-session state over time. This layer logs disambiguation histories, user preferences, and execution feedback, allowing the system to learn from past interventions and maintain state across complex workflows.
- Governance & Trust:** Security must be enforced at runtime. This tier applies row-level security, role-based entitlements, data quality scores, and blast-radius warnings to agent execution paths. Every query and action remains auditable and explainable.

Underpinning the Semantic Foundation tier is a neurosymbolic approach that separates Tera Context Engine from conventional RAG-based alternatives. Rather than requiring organizations to construct business meaning from scratch or rely solely on probabilistic inference, the engine ships with Industry Knowledge Models: expert-authored conceptual schemas, analytical star schemas, and data-quality rules built from decades of Teradata domain engagement across financial services, healthcare, telecommunications, manufacturing, and retail. These models provide a pre-governed canonical foundation that maps extracted enterprise data through governed graph traversal rather than relying solely on zero-shot synthesis — creating more deterministic execution paths with less inference, faster time to value, and fewer opportunities for agent error.

Figure 4: Tera Context Engine Approach

Capability	Legacy Approach (Vector RAG/Point Tools)	Tera Context Engine Approach
Knowledge Foundation	Built per project through manual mapping or model inference	Pre-governed Industry Knowledge Models with domain semantics, column-level lineage, and bidirectional context writeback
Data Grounding	Probabilistic similarity via vector embeddings	Deterministic semantic mapping and structured ontologies
Business Logic	Hardcoded into prompt strings or isolated reports	Centralized, reusable data products (dbt, Looker)
Execution Memory	Ephemeral, stateless execution resetting per run	Stateful operational memory with active feedback loops
Governance Plane	Static metadata catalogs evaluated offline	Active runtime enforcement with row-level security
Tool Integration	Brittle, point-to-point bespoke scripts	Standardized connectivity via Model Context Protocol

At runtime, Tera Context Engine coordinates these tiers through Context Playbooks, an Agent Skills repository, and a Decision Traces Graph. Using progressive context disclosure, the engine injects only the semantic definitions and tool schemas a given sub-task requires. This eliminates retrieval noise, cuts token overhead, and keeps models grounded within safe parameters (Figure 4).

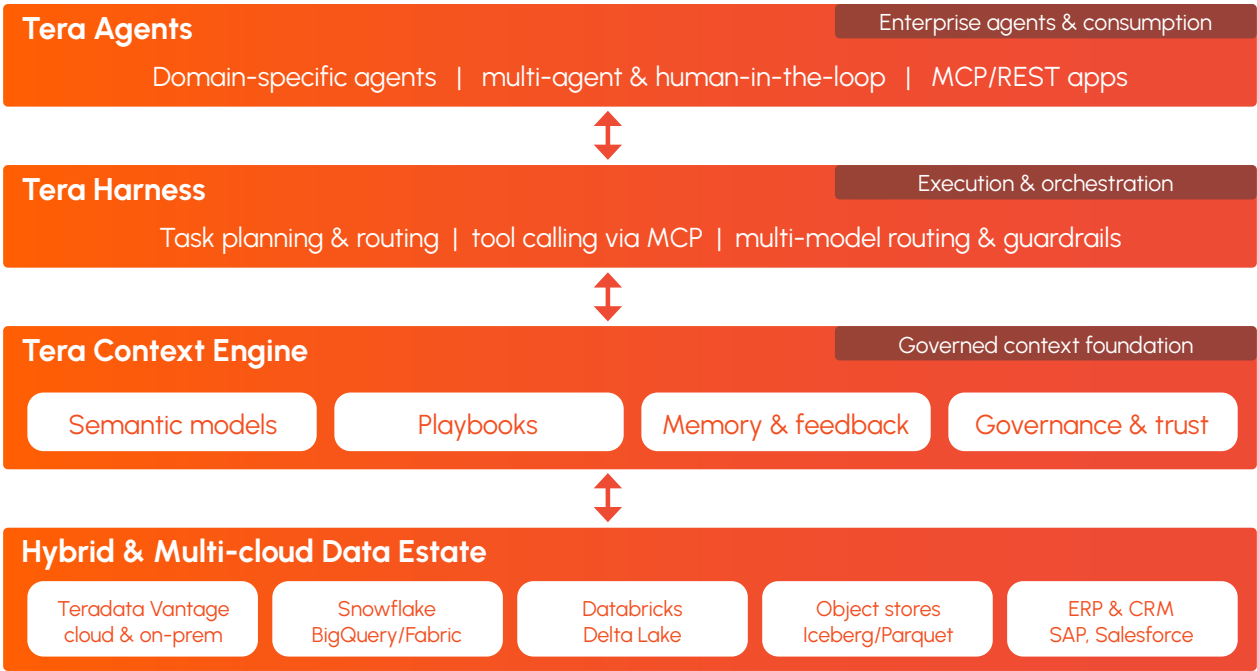


Open, Connected, and Platform-Neutral

Why Context Must Bridge the Entire Data Estate

Enterprise data does not live in a single repository. Modern environments are distributed, spanning systems such as Snowflake, Databricks, Google BigQuery, object storage, on-premises warehouses, and transactional ERPs. Building a context layer cannot require ripping and replacing these assets or centralizing all enterprise data into a single vendor's database.

Figure 5: Tera Context Engine Layer



As part of the Teradata Autonomous Knowledge Platform, Tera Context Engine provides an open context layer across this hybrid landscape, while Tera Harness coordinates intelligent execution across enterprise systems, models, and tools. Together with specialized Tera Agents, they allow organizations to govern, ground, and operationalize autonomous AI workflows without requiring expensive, multi-year data migration projects (Figure 5).

Interoperability relies on open microservices, REST APIs, and adoption of the Model Context Protocol (MCP). MCP provides an open standard for AI agents to discover schemas, call tools, and execute governed queries across external systems. Futurum market data indicates the Semantic Layer category is expanding at roughly 30% annually, with one in five enterprise agentic deployments utilizing MCP in production.^{1,2} Standardizing on open interfaces ensures organizations can swap underlying foundation models and agent orchestrators without rebuilding their business logic and governance rules.

2 Futurum Research. 1H 2026 Data Intelligence Market Forecast.



Pragmatic Next Steps for Technology Leaders

Actionable Guidance to Move from Pilots to Production

To move autonomous agents from brittle proofs-of-concept to reliable production infrastructure, engineering and data leaders should focus on the following five architectural steps:



Decouple Business Logic from Prompts: Pull metric formulas, filter logic, and business rules out of prompt templates and Python wrappers. Consolidate them into a semantic layer so agents consume the same governed definitions used by production analytics dashboards.



Activate Existing Metadata and Domain Knowledge: Avoid starting from scratch. Ingest dbt transformation models, Looker definitions, and catalog metadata into an active runtime layer, and augment them with pre-built industry knowledge models that provide governed domain semantics, relationships, and data-quality rules.



Enforce Governance at the Execution Plane: Ensure access controls, row-level security, and audit logging are dynamically attached to data flowing into LLM calls and tool executions, preventing agents from bypassing enterprise access policies.



Implement Stateful Operational Memory: Replace stateless, single-turn agent pipelines with a governed memory store that captures human corrections, execution traces, and session state to refine accuracy.



Track Unit Economics and Execution Quality: Establish metrics for AI operational efficiency, measuring token consumption per completed task, query retry rates, latency, and transactional error rates. These measures expose the economic impact of probabilistic retry loops and quantify the value of moving repeatable work into governed, deterministic execution paths.



The Strategic Bottom Line

The bottleneck holding back enterprise AI goes beyond a lack of reasoning capabilities and is instead rooted in an architectural context deficit. As enterprises transition from passive chat interfaces to agents that read, analyze, and write back to enterprise systems, relying on probabilistic prompt hacks becomes an unsustainable operational liability.

Long-term success requires treating enterprise context as a core infrastructure tier — one that is deterministic, continuously governed, and open across multi-cloud environments. Moving business meaning and query construction out of probabilistic reasoning loops and into governed execution paths also changes the economics of enterprise AI by reducing repeated inference, unnecessary token consumption, and costly retry cycles. By unifying semantic models, industry knowledge, organizational knowledge, operational memory, and runtime governance — and connecting that context to intelligent execution — solutions such as the Teradata Autonomous Knowledge Platform bridge the gap between static enterprise data and reliable autonomous action. Technology leaders who establish this foundation today can move beyond the pilot plateau and build autonomous systems that deliver measurable, scalable business value.

Important Information About This Report

AUTHORS

Brad Shimmin

Vice President & Practice Lead, Data Intelligence,
Analytics, & Infrastructure | The Futurum Group

PUBLISHER

Futurum Research

INQUIRIES

Contact us if you would like to discuss this report and
The Futurum Group will respond promptly.

CITATIONS

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

LICENSING

This document, including any supporting materials, is
owned by The Futurum Group. This publication may not be
reproduced, distributed, or shared in any form without the
prior written permission of The Futurum Group.

DISCLOSURES

The Futurum Group provides research, analysis, advising,
and consulting to many high-tech companies, including those
mentioned in this paper. No employees at the firm hold any
equity positions with any companies cited in this document.

The Teradata logo is displayed in a bold, orange, sans-serif font.

ABOUT TERADATA

Teradata Autonomous Knowledge Platform activates
enterprise intelligence by unifying data, knowledge,
and business context, for real-time business outcomes.
Enterprises can combine domain expertise with
enterprise context for impact when it matters,
connecting and scaling across any environment.

The Futurum logo is displayed in a bold, black, sans-serif font with a registered trademark symbol.

ABOUT THE FUTURUM GROUP

The Futurum Group is an independent research, analysis,
and advisory firm, focused on digital innovation and market-
disrupting technologies and trends. Every day our analysts,
researchers, and advisors help business leaders from around
the world anticipate tectonic shifts in their industries and
leverage disruptive innovation to either gain or maintain a
competitive advantage in their markets.

The Futurum logo is displayed in a bold, white, sans-serif font with a registered trademark symbol.

CONTACT INFORMATION: The Futurum Group LLC | [futurumgroup.com](https://www.futurumgroup.com) | (833) 722-5337