SINGULARICS

White Paper

The Intention Graph

Why Every AI-Augmented Enterprise Needs a System of Record for Intent

Alexander S. Petty
SINGULARICS | singularics.ai

For permissions, bulk copies, or licensing inquiries:
hello@singularics.ai

First published June 2026.

singularics.ai

Contents

Abstract

Strategy lives in slides. Execution lives in tickets. AI agents answer to whoever prompted them last. Between these three systems there is no shared record of what the organization intends, no structural connection between a board-level outcome and the work being done to achieve it, and no governed mechanism for ensuring that AI execution serves declared intent. This paper makes the case for the organizational intention graph: a live, governed system of record that connects strategic outcomes to deliverable artifacts, governs AI agent execution through structural admissibility, tracks the governance cycle at every organizational level, conserves context for AI operations, and produces an audit trail as a structural byproduct of the work itself. Drawing on enterprise AI adoption data, agent governance research, token economics analysis, and evidence from a graph-governed context resolution experiment on real issue-tracker data, we show that the intention graph addresses five problems that current enterprise tooling cannot solve: organizational coherence, continuous governance, context conservation, audit integrity, and cost attribution. The paper is written for executives evaluating AI infrastructure, portfolio leaders managing AI-augmented delivery, and technologists building the systems that connect strategy to execution.

The State of Enterprise AI in 2026

Enterprise AI adoption has crossed the threshold from experimental to operational. McKinsey's 2025 State of AI report found that 72% of organizations have adopted AI in at least one business function [1]. Gartner forecasts that 40% of enterprise applications will embed task-specific AI agents by end of 2026, up from under 5% in 2025 [2]. The global AI agents market has reached $10.9–12.1 billion with a 44–46% CAGR projected through 2030 [3].

The spending follows the adoption. Model API spending doubled from $3.5 billion to $8.4 billion between late 2024 and mid-2025 [4]. Average monthly enterprise AI spend jumped from $63,000 to $85,500, a 36% increase in a single year [4]. AI is now the fastest-growing expense in corporate technology budgets, with some firms reporting that it consumes up to half of their IT spend.

But adoption and spending are not the same as value. Only 11–14% of enterprise AI agent pilots have reached production at scale, with 86–89% failing to realize durable value [5]. The failure is not technological. The models work. The agents execute. The failure is organizational: governance breakdowns, integration complexity, and the absence of a structural connection between what the organization intends and what the agents build.

86–89% of enterprise AI agent pilots fail to reach production at scale. The models work. The organizational infrastructure does not.

Five Problems No Current Tool Solves

Current enterprise tooling was built for a world where humans did all the execution. Project management tools track human work. Portfolio tools aggregate human-reported status. ALM systems manage human-authored artifacts. None of them were designed for a world where AI agents execute thousands of operations per day, each of which needs organizational context, governance, and traceability.

Five problems emerge.

1. Organizational Incoherence

Strategy lives in slide decks. Execution lives in Jira tickets. Between them there is no structural connection. A board approves a strategic outcome. That outcome is translated (with loss) into an initiative. The initiative is decomposed (with further loss) into epics. The epics are broken into stories. By the time work reaches a team, the connection to the original strategic intent is tenuous at best, absent at worst. Nobody in the building can trace a piece of daily work back to the board-level outcome it is supposed to serve.

AI makes this worse, not better. AI agents execute at speed. When the intent is clear, that speed is an advantage. When the intent is unclear or disconnected, that speed amplifies the incoherence. The agents build faster in a direction nobody verified was right.

McKinsey's 2024 research found that large enterprises spend 20–35% of their capacity on coordination activities rather than value delivery [6]. That coordination overhead exists because there is no structural mechanism for maintaining alignment. The ceremonies (PI Planning, portfolio reviews, status meetings) are the workaround for the absence of a system of record for intent.

2. Ungoverned AI Execution

72% of organizations are using or planning to use agentic AI, while 65% say their AI adoption is moving faster than their ability to fully understand it [7]. A 2026 Gravitee survey found that only 24.4% of organizations have full visibility into which AI agents are communicating with each other [8]. More than half of all agents run without any security oversight or logging.

Every agent action without a governance record is an audit gap. The agent may have followed its instructions precisely and still produced the wrong thing, because the instructions did not capture the intent. Without structural governance, there is no mechanism for granting permission before execution or verifying acceptance after delivery. The agent executes. Nobody checked before. Nobody verified after. The audit trail does not exist.

KPMG's 2026 survey found that 75% of large-enterprise leaders cite security, compliance, and auditability as the most critical requirements for agent deployment [9]. Yet 72% of S&P 500 companies disclosed at least one material AI risk in 2025, while only 26% have comprehensive AI governance policies in place [7].

3. Context Waste

Every AI agent interaction begins by re-establishing context. Organizational structure, strategic intent, project history, code architecture, prior decisions, acceptance criteria, domain constraints. Most of this context is stable between interactions. Nearly all of it is re-ingested from scratch.

Enterprise AI token consumption has increased 13× since January 2025 [4]. Gartner's March 2026 analysis found that agentic models require between 5 and 30 times more tokens per task than a standard generative AI chatbot [10]. The paradox: per-token prices have dropped approximately 80% between early 2025 and early 2026 [4], yet total enterprise AI spending continues to surge because consumption has outpaced the price decline.

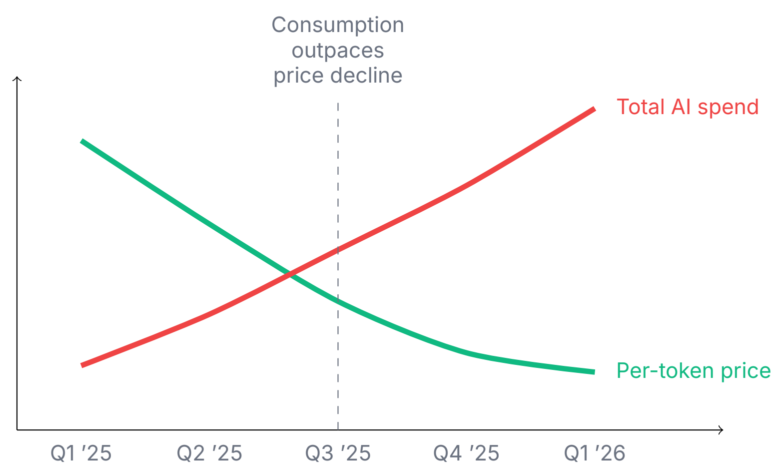

Figure 1: The token economics paradox. Per-token prices have dropped 80% since early 2025, but total enterprise AI spending has more than doubled because token consumption has increased 13× [4,10]. The majority of that consumption is context re-establishment, not execution.

The assumption that token prices would continue falling has not held. In mid-2026, frontier model providers reversed course. GPT-5.5 launched with a 2× input price increase, and real-world cost increases ranged from 49% to 92% depending on prompt length [21]. One analysis described "volatility of 50% to 90% inside 48 hours" with "no precedent in enterprise commodity infrastructure" [22]. The average enterprise AI budget has grown from $1.2 million per year in 2024 to $7 million in 2026 [23]. Some Fortune 500 companies report monthly inference bills in the tens of millions [23]. A healthcare enterprise consumed 1 trillion tokens over six months, translating into more than $6 million in unplanned costs before the finance team understood what was driving it [23].

Enterprises are now dependent on AI for productivity in a market where the unit cost of that productivity is volatile and controlled by providers they have no leverage over. This is the pricing dependency risk that no enterprise technology budget has faced before. AWS never doubled its per-unit price inside a product cycle. AI providers have.

The structural response is not to reduce usage (that sacrifices the value of AI). It is to reduce the token waste that makes usage expensive. Context re-establishment is the dominant source of that waste. An organization that resolves context structurally rather than re-ingesting it on every operation is insulated from pricing volatility because the volume of tokens consumed per operation is bounded by graph structure, not by corpus size.

The hidden driver is context. In lab testing on real public issue-tracker data, we found that full-injection context resolution consumed an average of 201,037 tokens per operation [11]. Naive semantic retrieval reduced that to 6,990 tokens but recovered only 29.3% of required context and missed 185 governance records. Graph-governed context resolution used 8,039 tokens, recovered 100% of required context, and missed zero governance records. That is 4% of the full-injection cost with complete recall.

4% of full-injection token cost. 100% of required context. Zero missed governance records. Graph-governed resolution on 1,042 nodes of real Kubernetes issue-tracker data [11].

The implication: the majority of enterprise token spend is not execution. It is context re-establishment. The organization is paying to teach its AI what it already knows, over and over, at machine speed. A structural context mechanism (not a semantic retrieval mechanism) eliminates that waste.

4. Audit Gaps

In regulated industries (financial services, healthcare, government), every AI agent action requires a governance record. Who authorized the work. What constraints applied. What evidence supports the outcome. Who accepted the deliverable and against what criteria.

Current systems do not produce this record structurally. The audit trail must be reconstructed after the fact from git logs, Slack threads, email chains, and institutional memory. That reconstruction is incomplete, expensive, and non-authoritative.

NIST AI RMF 1.1, released March 2026, is the practical standard for US federal and enterprise AI governance [12]. ISO 42001 is the international AI Management System standard that enterprise procurement teams verify against [12]. Both require explainability, auditability, and traceability. Neither can be satisfied by a system that produces no structural governance record.

5. Cost Without Attribution

AI spend rolls up as a line item on the technology budget. Nobody can attribute token cost to the organizational intent it served. Nobody can answer the question: what did we spend on this strategic outcome? Did the AI execution that served this initiative cost more or less than the value it produced? What is the cost-per-outcome?

Without cost attribution at the intent level, AI economics are ungovernable. The CFO sees a growing bill with no way to evaluate whether the spend is producing proportional value. Usage limits become the default governance mechanism, throttling value along with waste.

The Intention Graph

The intention graph is a live, governed system of record for organizational intent. It connects strategic outcomes to deliverable artifacts through governed decomposition, tracks the governance cycle at every organizational level, provides structural context for AI operations, and produces an audit trail as a byproduct of the work itself.

What the Graph Contains

Every node in the intention graph represents an organizational intent. An intent is either an outcome (a state change the organization wants to achieve) or an output (something built or delivered to bring about an outcome). Outcomes come first. Outputs exist to serve them.

Each node carries:

- **Declaration.** What should become true (outcome) or what should be produced (output).
- **Success conditions.** Measurable criteria that define what done means. For outcomes, conditions describe the state change. For outputs, conditions describe acceptance criteria.
- **Ownership.** A human accountable for whether the conditions are met.
- **Governance state.** Where this intent sits in the governance cycle (declared, proposed, approved, executing, delivered, verified).
- **Decision history.** Every permission and acceptance decision recorded with who, when, evidence, and rationale.
- **Context pointers.** Typed references to external resources (repositories, documents, prior deliverables) with metadata sufficient to evaluate relevance without retrieving content.

Edges in the graph represent typed relationships: *parent-of*, *serves*, *depends-on*, *constrains*. Every output connects to the outcome it serves. Every piece of work traces back to the strategic intent that authorized it.

What the Graph Provides

The intention graph is not a project management tool. It does not replace Jira, Azure DevOps, or GitHub. It sits above them as the meaning layer, the structural connection between strategy and execution that those tools were never designed to provide.

Function	Without the graph	With the graph
Strategic alignment	Quarterly review of slide decks	Continuous, structurally computed alignment scoring across every piece of work
Governance	Periodic ceremonies (sprint review, PI Planning, steering committee)	Continuous governance cycle running at every level at the cadence the work demands
Agent context	Full corpus injection or semantic retrieval (RAG)	Structural context from the graph neighborhood, 4% of full-injection cost
Audit trail	Reconstructed after the fact from logs, Slack, email	Structural byproduct of the governance cycle, recorded as it happens
Cost attribution	Aggregate technology budget with no intent-level breakdown	Token spend attributed to the intent it served, rolling up through the graph

Table 1: What changes with the intention graph. Each row addresses one of the five problems identified in Section 2.

Organizational Coherence

The intention graph makes alignment a continuously computed system property rather than a quarterly conversation.

Every piece of work in the graph traces to the outcome it serves through typed edges. Alignment is not an assertion in a status report. It is a structural property that can be verified by traversing the graph. If a piece of work does not connect to a declared outcome, it is visible as an orphan. If a declared outcome has no work connected to it, it is visible as uncovered. If capacity is concentrated on low-priority outcomes while high-priority outcomes are starved, the graph makes that visible in real time.

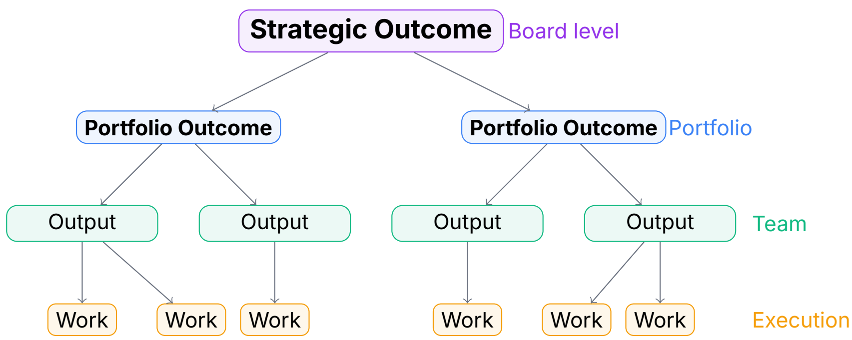

Figure 2: The intention cascade. Every piece of work traces to the outcome it serves through the graph. Alignment is a structural property, not a status report. Orphaned work (not connected to an outcome) and uncovered outcomes (no work serving them) are visible in real time.

The graph does not create alignment. It reveals where alignment exists and where it does not. The human judgment required to restore alignment (reprioritize, redirect, cancel, fund) is the same judgment it always was. The graph makes that judgment possible by providing the structural visibility that slide decks and ticket systems cannot.

Continuous Governance

The intention graph enables continuous governance: the practice of injecting human judgment into a continuously executing system at the structurally right places, at the cadence the work demands, at every level of the organization.

The governance cycle has five steps:

1. **Intent is defined.** A human declares what should become true or what should be produced.
2. **AI proposes.** The system evaluates the intent, proposes decomposition, surfaces conflicts.
3. **Humans judge.** A human at the appropriate level makes the permission decision.
4. **AI executes.** Agents execute approved work under the constraints of the graph.
5. **Outcomes are verified.** A human verifies delivered work against declared conditions.

Two of these steps are human gates that cannot be automated: step 3 (permission) and step 5 (acceptance). An agent may propose and an agent may execute. An agent may never grant permission and may never accept its own work.

The cycle is fractal. It runs identically at every organizational level.

Level	Cadence	Horizon
Board / Investors	Monthly	6–18 months
C-Suite	Weekly	3–6 months
VP / GM	Daily	2–6 weeks
Director	Multiple per day	Days–2 weeks
Manager / PO	3–5 per day	Hours–days
Practitioner	Continuous	Hours

Table 2: The governance cycle runs at every level. Same five steps. Same two human gates. Different cadence. The structure is invariant. The frequency scales with the work.

Continuous governance follows this same structure at every level. The drift rate rises when judgment cycles lag behind execution, while the invariant define–propose–judge–execute–verify loop keeps authority and evidence attached to the work.

Context Conservation

The intention graph is not just the governance layer. It is the context layer.

When an AI agent operates on a node in the graph, the graph already knows what context is relevant. The node's parent outcome (why this work exists), sibling outputs (what adjacent work is happening), conditions (what done looks like), ownership (who is accountable), decision history (what was approved and why), governance state (where it sits in the cycle). All of this is reachable by traversing edges in the graph. No search. No embedding. No re-ingestion.

Why RAG Is Not Enough

Retrieval-augmented generation retrieves content that looks semantically similar to the query. It has no awareness of organizational structure. A semantically similar chunk may be organizationally irrelevant (a similar description from a different portfolio). A structurally essential chunk may be semantically distant (a governance policy that constrains the current intent but uses different terminology).

RAG retrieves content. The intention graph governs context. The distinction is structural relevance versus semantic similarity. Structural relevance is determined by the graph's topology (what is connected to what, through what relationships, at what organizational level). Semantic similarity is determined by surface word patterns.

Figure 3: Context resolution comparison on real Kubernetes issue-tracker data (1,042 nodes, 1,784 edges, 170 work items, 61 operations) [11]. Graph-governed resolution achieves complete recall at 4% of full-injection cost. Naive retrieval is cheaper but misses 70% of required context and 185 governance records.

Token Economics at Scale

The context conservation properties of the graph produce three economic effects.

Cost independence from scale. Token cost per operation depends on the graph neighborhood of the target node, not on the total size of the organizational corpus. Adding a new department with a million tokens of documentation does not increase the context cost of operations in other departments.

Compounding efficiency. The graph enriches with use. Every governance decision, condition update, observation, and feedback appends to the structural context at graph nodes. Subsequent operations on the same subgraph benefit from this accumulated context without re-ingestion. The marginal context cost of each operation decreases over time.

Hyperscaler risk mitigation. Organizations whose token cost is dominated by context re-establishment are exposed to provider pricing changes. A 20% price increase on a cost base that is 80% context waste is catastrophic. The graph eliminates the waste, reducing exposure to pricing changes by an order of magnitude.

Context conservation follows from subgraph locality: an agent receives the smallest relevant neighborhood, reuses durable pointers to prior decisions, and adds new evidence to the same structure. Relevance stays bounded while the reusable context around the work compounds.

The Audit Trail as Structural Byproduct

The intention graph produces an audit trail not as a separate logging system but as a structural byproduct of the governance cycle. Every permission decision (step 3) and every acceptance decision (step 5) is recorded in the graph with:

- Who decided (actor attribution)
- When (timestamp)
- On what node (the intent being governed)
- Against what criteria (the declared conditions)
- With what evidence (the artifacts reviewed)
- With what rationale (optional human annotation)

Every agent execution (step 4) is recorded with:

- What agent acted
- What it was authorized to do (the intent it served)
- What it produced
- What tokens it consumed (cost attribution)
- Whether it stayed within scope (boundary condition evaluation)

This record is append-only. No governance state can be deleted or overwritten. The event ledger is the source of the audit trail, not a reconstruction. When a regulator asks why a piece of work was done, who authorized it, and what evidence supports the outcome, the answer is in the graph. It was recorded when it happened, not assembled when it was needed.

Structural, not reconstructed.

Every governance decision, every agent action, every verification recorded in the event ledger as it happens. The audit trail is the governance cycle's exhaust, not a separate system.

Cost Attribution at the Intent Level

The intention graph enables a new kind of financial governance: cost-per-outcome computed in real time.

Every AI operation in the system is bound to the graph node it serves. The token cost of that operation is attributed to that node. Token costs roll up through the graph hierarchy. A strategic outcome accumulates the token costs of all the portfolio outcomes, team outputs, and practitioner work items beneath it.

This means the organization can answer questions that are currently unanswerable:

- What did we spend on this strategic initiative?
- Which outcomes are consuming disproportionate AI resources relative to their priority?
- What is the token cost per verified outcome?
- Where is AI spend producing value and where is it producing activity?

This is not a reporting feature bolted onto a ticket system. It is a structural property of the graph. The cost attribution is a consequence of the governance cycle: because every operation traces to an intent, the cost traces with it.

Integration, Not Replacement

The intention graph does not replace existing enterprise tools. It integrates with them as the governance and meaning layer above them.

Tool	What it does	What the graph adds
Jira, Azure DevOps	Track tickets and sprints	Structural connection from tickets to outcomes. Alignment scoring. Governance state.
GitHub, GitLab	Manage code and CI/CD	Traceability from commits to the intent they serve. Agent execution governance.
Cursor, Copilot, Claude Code	AI coding assistants	Structured context from the graph via MCP. Acceptance criteria alongside the code.
Apptio, financial tools	Track technology budgets	Token cost attributed to outcomes, not just departments. Cost-per-outcome rollup.
Slack, Teams	Communication	Feedback routed into the governance cycle as governed intent, not lost in message history.

Table 3: The intention graph integrates with existing tools. It does not replace them. It adds the governance, alignment, and context layer they were not designed to provide.

The integration mechanism is the Model Context Protocol (MCP). The intention graph exposes an MCP server that any AI coding tool can read. The tools receive structured context about what to build, why, and what done looks like. The organization does not give the graph its repository. The graph gives the tools its intent.

Architecture

The intention graph is deployed as a dedicated, isolated instance per customer. No shared tenancy. No data mixing.

Property	Description
Tenant isolation	Dedicated server and database per customer. No cross-tenant data access at any layer.
Bring your own keys	OpenAI, Anthropic, or any compatible provider. Prompts never touch shared infrastructure.
SSO / OIDC	Customer's identity provider. Customer's policies. Customer's audit requirements.
Append-only event ledger	Every governance state change recorded with actor attribution. No governance state can be deleted.
Self-hosted option	Enterprise customers can deploy on their own infrastructure. Nothing leaves the perimeter.
Real-time synchronization	WebSocket event fabric. All connected clients see graph changes instantly.

Table 4: Architecture properties. The intention graph is your competitive strategy in structured form. It is protected accordingly.

The Formal Foundation

The intention graph is not assembled from product intuitions. It is built on a formally specified canonical model with mathematically defined properties.

- **Work-to-outcome lineage.** Every piece of work has a provable path to the outcome it serves. Not a label. A structural property enforced by the graph.
- **Outcome/output separation.** Outcomes (intended state changes) and outputs (deliverable artifacts) are structurally distinct. The system cannot collapse them.
- **Decomposition termination.** The intent graph cannot expand infinitely. Scope descent and acyclicity guarantee that every decomposition chain terminates.
- **Projection determinism.** Every view (the constellation, the signal board, every report) is a pure function of canonical state. Same state, same view. Always.
- **Event-ledger explainability.** Every state in the system can be historically explained by a finite trace of recorded events. The audit trail is the source, not a reconstruction.

- **Tenant-local subgraph separation.** No ordinary relation can cross tenant boundaries. The intention graph is architecturally isolated from every other customer.

These are not aspirational properties. They are guaranteed properties of the architecture, verified by the formal model. The difference shows up when the organization scales, when the auditor asks for proof, and when the system needs to explain why something happened.

Who This Is For

For the VP of Engineering

You open Jira. 400 tickets across 8 teams. You have no idea which ones connect to the objectives you committed to last quarter. Three teams are using AI agents, but you cannot tell what those agents built or whether it was authorized.

With the intention graph, every piece of work traces to a declared outcome. Agent activity is visible with cost attribution. The signal board shows what needs your judgment right now.

For the Product Owner

You write user stories and hope developers interpret them correctly. With AI agents, interpretation happens at machine speed, in directions you did not anticipate. You review output that technically matches what you wrote but is not what you meant.

With the intention graph, you define intent with conditions that capture what done actually means. AI proposes a decomposition. You approve or adjust. The agent executes and deploys a live preview. You verify against a running system, not a diff.

For the CISO

Your organization deployed AI agents across four teams last quarter. You have no authorization records, no audit trail, no way to know what those agents accessed or produced.

With the intention graph, every agent proposal, every human approval, every execution event is recorded in an append-only event ledger. The audit trail is structural. You do not reconstruct it. You query it.

For the CFO

AI is the fastest-growing line item in the technology budget. You cannot attribute any of that spend to the outcomes it produced.

With the intention graph, token cost is attributed to the intent it served. Cost-per-outcome is computed in real time and rolls up through the organizational hierarchy. You can see where AI spend produces value and where it produces activity.

Getting Started

The intention graph is deployed through an enterprise engagement. Singularics works with your team to configure the graph around your portfolio structure, integrate with your existing tools, and establish the governance cycle that fits how your organization makes decisions.

The engagement typically begins with a single portfolio outcome. The governance cycle runs on that outcome. The graph populates through normal work. The context conservation, audit trail, and cost attribution emerge as structural byproducts. The organization sees the value before deciding to scale.

For a conversation about what the intention graph would look like in your organization, visit singularics.ai/platform or book a conversation.

References

[1] McKinsey & Company (2025). *The State of AI in 2025.* 72% adoption in at least one function. 25% adapted operating models.

[2] Gartner (2026). *AI Agent Forecast.* 40% of enterprise applications will embed task-specific AI agents by end of 2026.

[3] Multiple sources (2026). Global AI agents market $10.9–12.1B, 44–46% CAGR through 2030.

[4] NavyaAI (2026). *AI Cost Report: Token Prices vs AI Bill.* Per-token prices down 80%. API spend doubled to $8.4B. Token consumption up 13×.

[5] Agentic AI Institute (2026). *Enterprise Adoption Report.* 86–89% of AI agent pilots fail to reach production at scale.

[6] McKinsey & Company (2024). *The State of Organizations 2024.* 20–35% of enterprise capacity spent on coordination.

[7] Multiple sources (2025–2026). 72% using or planning agentic AI. 65% adoption faster than understanding. 72% of S&P 500 disclosed material AI risk. 26% have comprehensive governance.

[8] Gravitee (2026). *AI Agent Visibility Survey.* 24.4% have full agent visibility. Over half run without security oversight.

[9] KPMG (2026). *Enterprise AI Agent Survey.* 75% cite security, compliance, and auditability as most critical requirements.

[10] Gartner (2026). *Agentic AI Analysis.* Agentic models require 5–30× more tokens per task than standard chatbots.

[11] Petty, A. S. (2026). *Lab Note: Graph-Governed Context Resolution.* SINGULARICS. 4% of full-injection cost, 100% recall, zero missed governance records on Kubernetes issue-tracker data.

[12] NIST (2026). *AI Risk Management Framework 1.1.* US federal AI governance standard. ISO 42001, international AI Management System standard.

[13] Petty, A. S. (2026). *Continuous Governance.* SINGULARICS White Paper.

[14] Petty, A. S. (2026). *The Context Problem.* SINGULARICS White Paper.

[15] Brooks, F. P. (1975). *The Mythical Man-Month.* Addison-Wesley.

[16] GitHub (2024). *Research: Quantifying GitHub Copilot's Impact.* 55% faster task completion.

[17] Murali et al. (2025). *AI Productivity at Google.* arXiv:2601.19964. 21% acceleration.

[18] Atlassian (2024). *State of Teams Report.* 58% of time on work about work.

[19] Deloitte (2025). *Global Human Capital Trends.* 14% feel governance keeps pace with AI.

[20] Digital.ai (2023). *17th Annual State of Agile Report.* 53% cite cross-team coordination as primary scaling challenge.

[21] FairMind (2026). *The List Price Is Lying: Why AI Costs Decoupled From Pricing in May 2026.* GPT-5.5 launched with 2× input price increase. Real-world increases 49–92%.

[22] Investing.com (2026). *The AI Token Pricing Crisis Behind OpenAI and Anthropic's Revenue Race.* Volatility of 50–90% inside 48 hours with no precedent in enterprise commodity infrastructure.

[23] EY (2026). *Agentic AI Enterprise Token Cost.* Average AI budget \$1.2M (2024) to \$7M (2026). Fortune 500 monthly bills in tens of millions. Healthcare enterprise 1 trillion tokens, \$6M unplanned.

SINGULARICS

We help large organizations close the intent gap between strategy and execution so that when they accelerate with AI, they accelerate in the right direction.

singularics.ai | Book a conversation