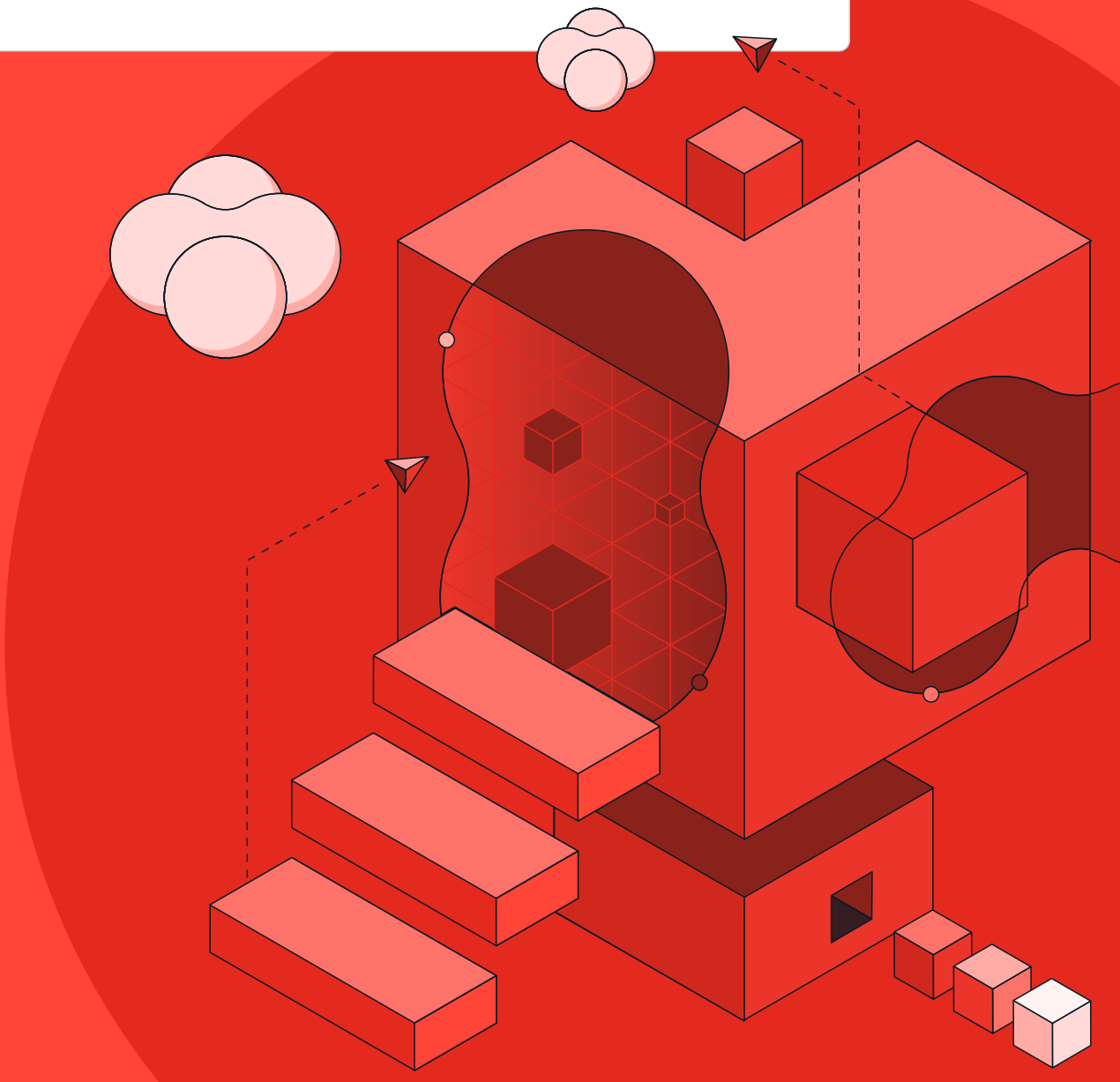
Redis

2026 EDITIION

REPORT

# The state of context engineering



© 2026 Redis

Executive summary 03

What you build today becomes your advantage tomorrow 04

Patterns of context maturity 06

The structural gaps that limit what you can build 08

The opportunity ahead of you (& the opportunity cost of missing it) 11

The window is closing 12

About this research 12

# Executive summary

Almost everyone building with AI now agrees on one thing: the models are no longer the constraint. The hard part is context. These are the data, records, memory, and live state that an agent needs to perform real, production-ready, high-stakes work.

Ask enterprise leaders, however, whether context matters, and they overwhelmingly say yes. Ask whether they have built the systems to supply it, and the honest answer is almost always no. Some organizations are trying, but very few meet the ideals everyone agrees are worth meeting.

That gap is the subject of this report. We surveyed IT and AI infrastructure leaders about how they build, feed, and govern context for production agents. Three findings run through everything that follows.

First, people agree that the problem is real. Leaders have stopped blaming models for disappointing agent behavior and started blaming context. They increasingly see broken or insufficient context as the main reason agents fail, and they rank fresh, navigable context above bigger models as the path to reliability.

Second, almost no one has fixed it. Belief has raced ahead of practice. Most organizations are still assembling context by hand—often one project at a time—with no shared infrastructure, no freshness guarantees, and little governance. The distance between a bold but improvisational approach and a truly mature context engineering system is wide, and most teams are only just getting started.

Third, that gap is an opportunity to leave your competition behind. Organizations that treat context as infrastructure, while the standards are still forming, will build an advantage that grows with use and becomes harder to copy over time. The ones that keep improvising will inherit fragmentation and expensive re-platforming instead. Context is an appreciating asset, and when it comes to AI, all organizations are young. Invest now, and the returns compound over time.

**The numbers that matter**

94%

**of organizations say compounding system intelligence is essential for production-grade maturity.** Only 4% have reached the stage where that actually happens.

69%

**of the most mature organizations still feed critical decisions with stale data.** Even the leaders have unfinished work.

60%

**of organizations are, by their own account, ineffective at governing who and what can reach context.** Only 58% are confident their context is well governed.

69%

**of enterprises now consider an agent that can't navigate across systems more dangerous than one that hallucinates.**

The rest of this report walks through what that data means, what is holding organizations back, and what the few pioneering leaders are doing differently.

## What you build today becomes your advantage tomorrow

Every major shift in enterprise software includes a window when the shape of the next paradigm is still malleable. It's when the cutting edge is still sharp, and before the standards, categories, best practices, and reference designs have hardened.

Context engineering is in that window right now. The decisions organizations make over the next year will shape what their AI systems can do over the next five years or longer.

Context engineering is the practice of deciding what an agent should see at each step of a task, including which facts, records, past interactions, and from which systems to retrieve them all. Then, it delivers that information in a form the agent can use, a form current enough to trust and fast enough for production. That's what most organizations understand, but few can put into practice.

### A new discipline is taking shape

One of the clearest signs that a category is forming is when the analysts, standards bodies, and service firms all move in the same direction at once. These are concrete business decisions, not marketing moves.

In July 2025, Gartner told AI leaders that "context engineering is in, and prompt engineering is out," and advised them to build context-aware architectures. When Gartner names a successor discipline, budgets and org charts tend to follow.

In August 2025, Cognizant announced it would train and deploy 1,000 context engineers over the following year. When a firm of that size creates a job title and hires a thousand people into it, the discipline has a labor market and a role to describe the work they do.

The plumbing is standardizing, too. Anthropic introduced MCP in November 2024, and in December 2025, Anthropic donated it to the Agentic AI Foundation, a new fund under the Linux Foundation. A protocol operating under neutral, open governance is what turns a merely popular idea into shared infrastructure that enterprises are willing to build on.

**For an agent to be useful in production, the context it runs on has to meet four requirements:**

**Navigable**

An agent must be able to move across related records and entities rather than grabbing from a bag of loose text.

**Fast**

A single task can involve dozens of retrievals, and slow context breaks the workflow.

**Fresh**

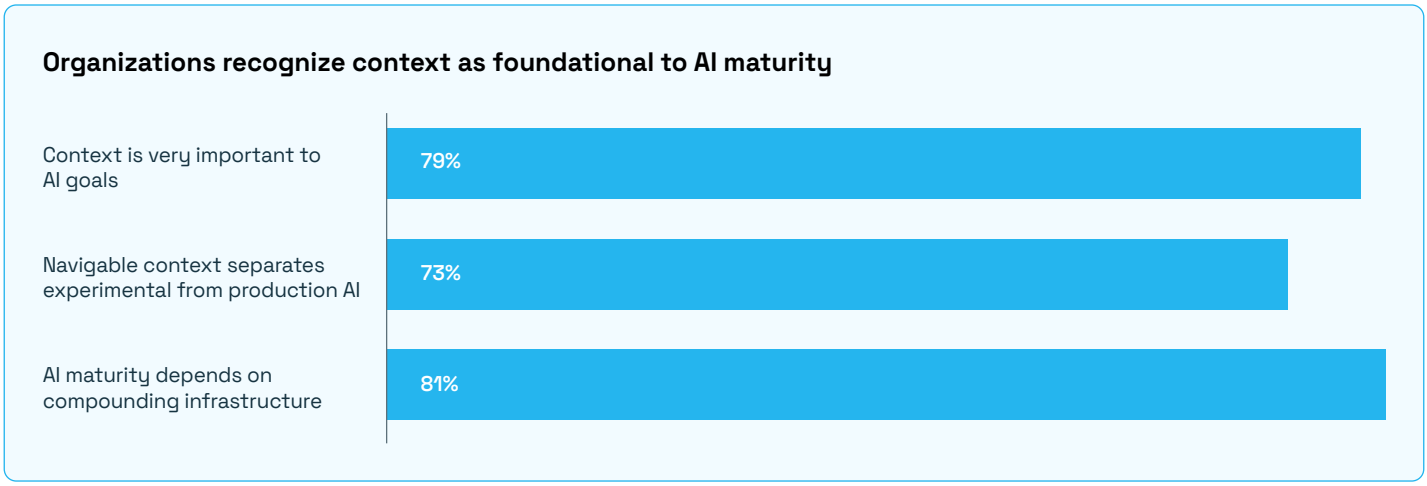
An agent acting on stale data will act confidently and wrongly.

**Compounding**

The system must become more useful the more it is used, rather than having to start from scratch every time.

Organizations that mature along these four axes are the most likely to succeed and to turn pilots into value-generating assets.

## Why the timing matters

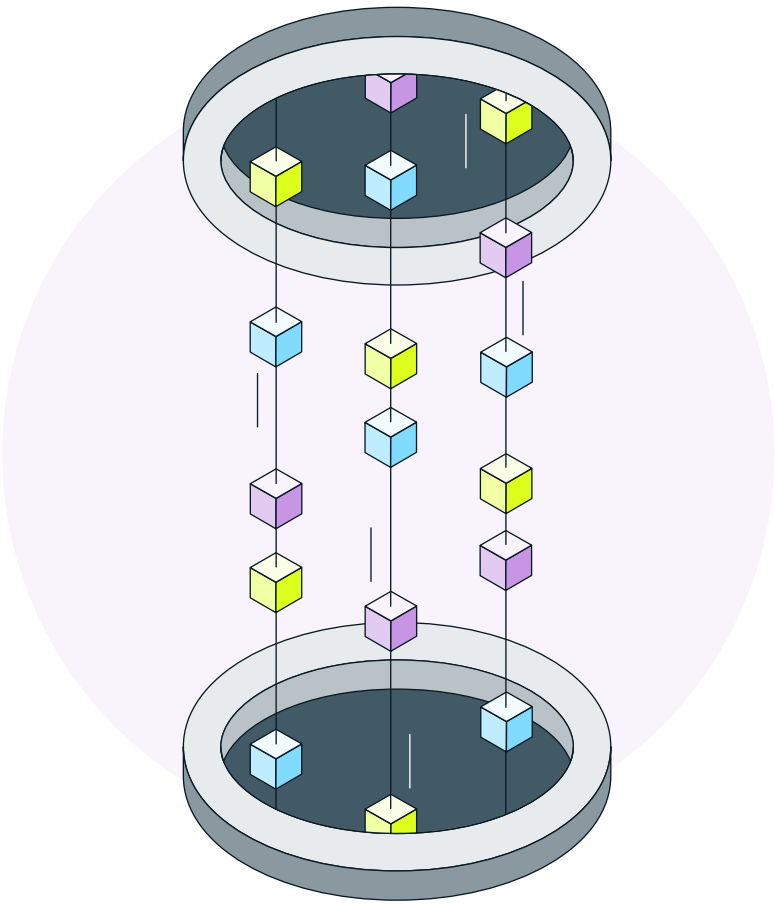



Standards are cheap to adopt when they are still forming and expensive to retrofit once your systems have grown in other ways. This is the case for acting now rather than waiting for the category to settle.

The survey shows this belief is already there, even if the follow-through isn't.

Nearly four in five organizations (79%) say context is very important to their data and AI goals, and another 18% call it somewhat important. That leaves almost no one who thinks it doesn't matter. 73% agree that navigable context systems are becoming the line that separates experimental AI from production AI. And 81% agree that the next phase of AI maturity will be defined by infrastructure that gains value as knowledge accumulates over time.

Those three numbers alone show the headwinds that are now blowing. Leaders believe context is the deciding factor; they believe it separates demos from production; and they believe lasting advantages will come from systems that build on themselves. But the systems aren't yet built. That's the tension we're here to explore.



## Patterns of context maturity

The importance of context isn't a view only held by early adopters. **73% agree that AI agents fail more often due to broken context than to broken models, and 83% believe fresh context matters more than adding more model parameters for enterprise reliability.**

In parallel, the way most organizations think about failure has also changed. **69% find their agents unreliable at navigating relationships across systems**, and **69% say an agent that cannot navigate creates more operational risk than one that hallucinates.**

As beliefs shift, implementations diverge. In our research, we saw organizations maturing as they adopted new approaches to context, and others struggling as they continued to improvise.

### Organizations are diverging

In our research, a pattern emerged that informs what constitutes good context engineering and what separates early approaches from mature systems. The pattern breaks out into five stages:

**The 5 stages of assembling context**

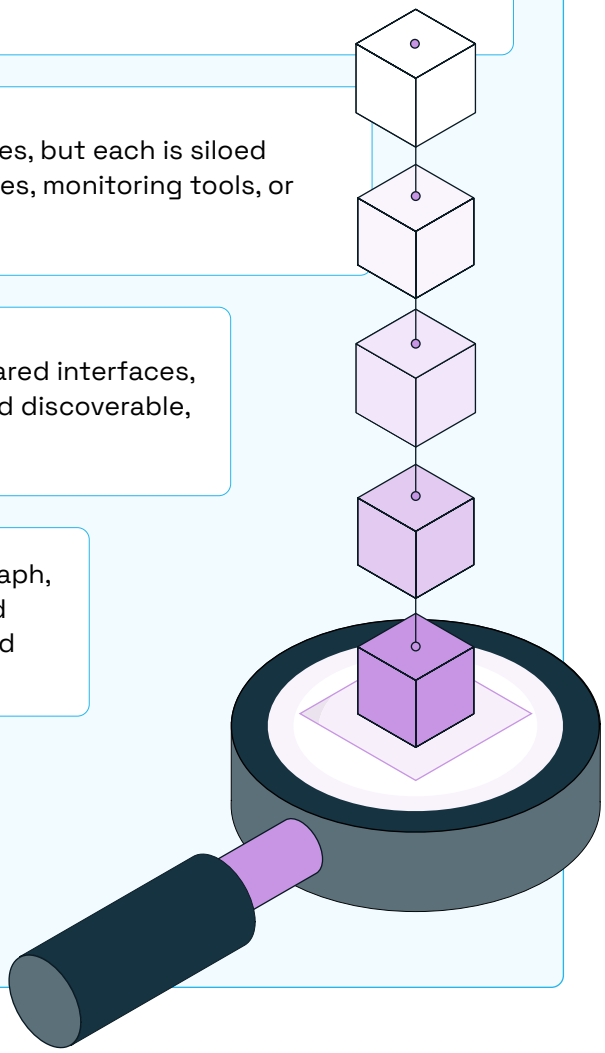
**Ad hoc:** Individual teams handle context manually via one-off prompts and informal data pulls. There is little to no shared infrastructure and governance, resulting in inconsistent outputs.

**Exploratory:** Some teams have RAG pipelines and vector stores, but each is siloed and built differently. There are few to no freshness guarantees, monitoring tools, or governance systems.

**Standardizing:** Teams begin wrapping context sources in shared interfaces, such as MCP servers. Agent capabilities become reusable and discoverable, and basic freshness and monitoring start to form.

**Managed:** There is a defined semantic model and context graph, monitored service levels for latency and freshness, governed access, captured decision traces, and memory that is scoped and retention-managed.

**Compounding:** Context improves continuously with use. Context is run as a product, with an owner and a roadmap, and teams can onboard new agent use cases in days because the foundation already exists.



Put that end-state against the on-the-ground data, and the gap between what leaders want and what they have is hard to miss.

**79% say context is critical**, yet **81% remain in the two earliest stages**—ad hoc or exploratory—leaving most organizations without true data freshness, monitoring, or governance.

And the destination almost everyone wants is the one almost no one has reached: **94% say compounding intelligence is essential for production-grade maturity**, but **only 4% have reached the compounding stage**.

This problem won't be solved by patience. Organizations aren't just "behind," such that waiting will eventually and inevitably deliver maturity. Teams are adopting agents faster than they are building the systems to run them. **67% agree they are still in the prototype phase** for agents, despite pressure from executives to scale.

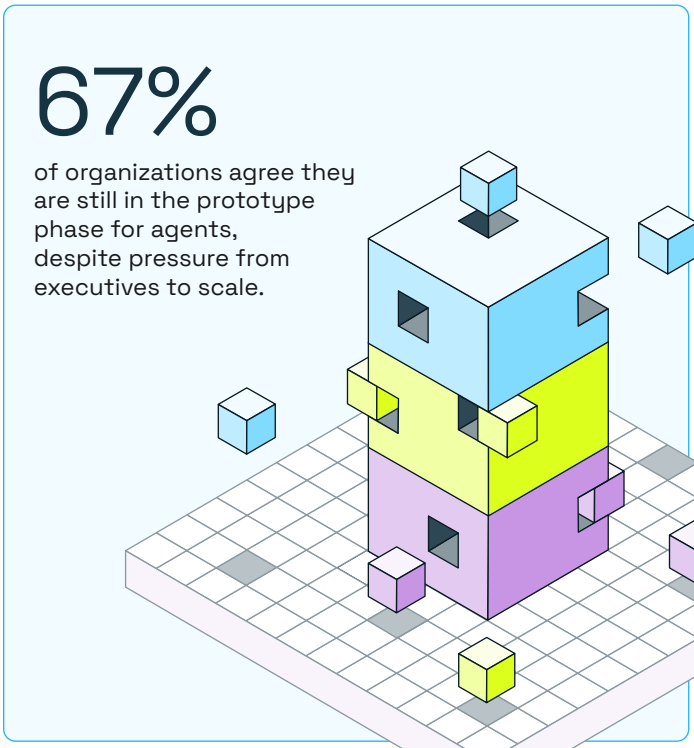
Despite that pressure, they aren't sure how ready they are: only **42% say they accurately measure their readiness for production agents**, while **39% are neutral or unsure**. Being unsure whether you are ready is an answer in itself.

## Improvising is more expensive than it looks

Improvisation feels freeing, and when LLMs first emerged, it was often the best approach. Each team wired up what it needed, shipped something, and learned—sometimes iterating and sometimes moving on. Over time, however, the cost of experimentation flips as ongoing costs overtake actual results.

Most of what organizations currently call context engineering is improvisation. **81% still handle context through ad hoc (43%) or exploratory (38%) approaches**, with little to no shared infrastructure, governance, or consistent results. **A third (33%) agree or strongly agree that they are still stitching AI infrastructure together from disconnected point solutions.**

Every month spent this way adds another custom integration, another undocumented pipeline, another store of memory that only one team or person understands. None of it was designed to fit together, so none of it will. Organizations aren't on track to have the context engineering system they dream of and, worse, are spinning up a complex web of technologies that will eventually require slow, costly unwinding.




For the organizations at the earliest stage, the picture is worse than their leaders tend to assume. Among Stage 1 organizations:

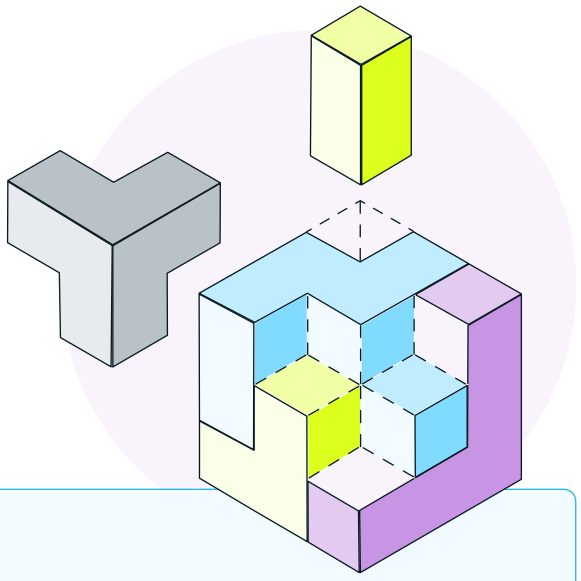
- **98% say context is very important to their AI and data goals, and only 2% report that their agents can reliably navigate across systems.**
- **95% probably or definitely lack a reliable semantic data model.**
- **94% probably lack a well-defined context graph.**

And when these organizations try to ship, the timeline slips: **56% say reaching production took much longer than they expected.**

The belief is total, and the energy is high, but the foundation is nearly absent, which is exactly the combination that produces expensive surprises.

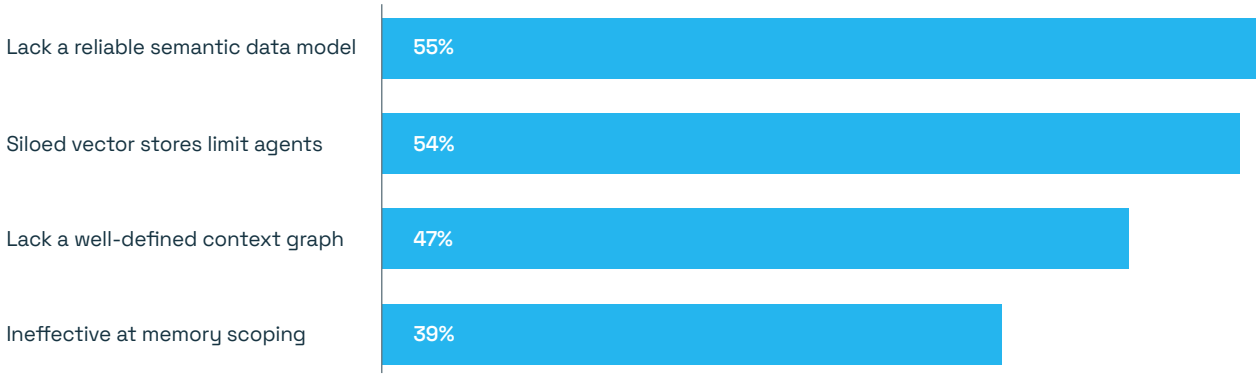
## The structural gaps that limit what you can build

The gaps holding organizations back are not the kind they can close by adding a tool or tuning a setting. They are structural, and like structural support beams in a building, the tallest heights cannot be reached until they're firmly set. Until then, there's a strict limit on what model quality or prompt cleverness can achieve.



### Opacity makes it hard to get started

**Foundational data gaps remain**




Agents fail to navigate for a reason that shows up before any agent is even deployed: the business and its data were never modeled in a way an agent could traverse. The opacity is present from the start.

The base layer is missing for about half the field. **55% lack a reliable semantic data model—a defined, agent-readable model of their business entities, attributes, and relationships. 47% lack a well-defined context graph that connects those entities.** Without them, an agent cannot move from an account to its open opportunities or from a support ticket to the underlying customer state, without hand-holding each time. Agents can match similar-looking text, but they cannot follow the relationships that convey the actual answer.

Memory has the same problem. 54% report that siloed vector stores significantly or moderately limit the effectiveness of their agents. **Similarly, 39% rate their organization as ineffective or very ineffective at scoping memory for agents, meaning they struggle to define what should persist, for how long, and for whom.**

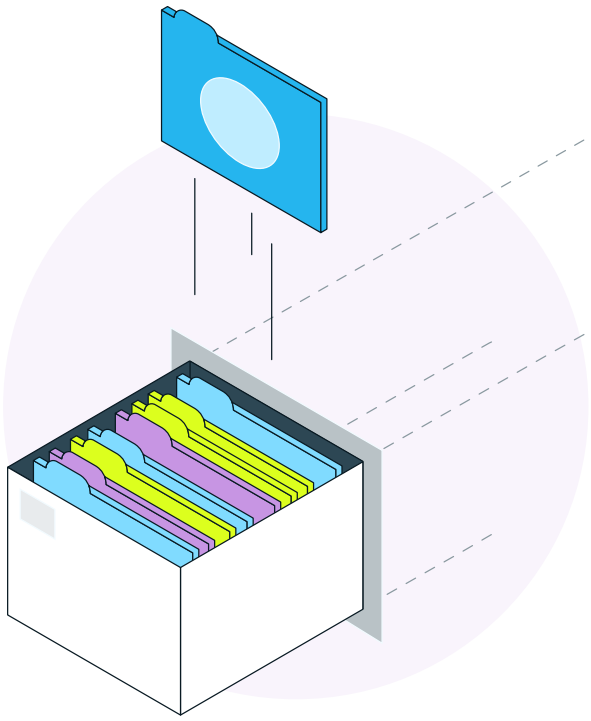
## Old & slow context can't be trusted

Even when context exists and is accessible, it's often too outdated or too slow to rely on when an agent is about to act. And the cost of acting from bad context is much higher than the cost of merely answering with it.

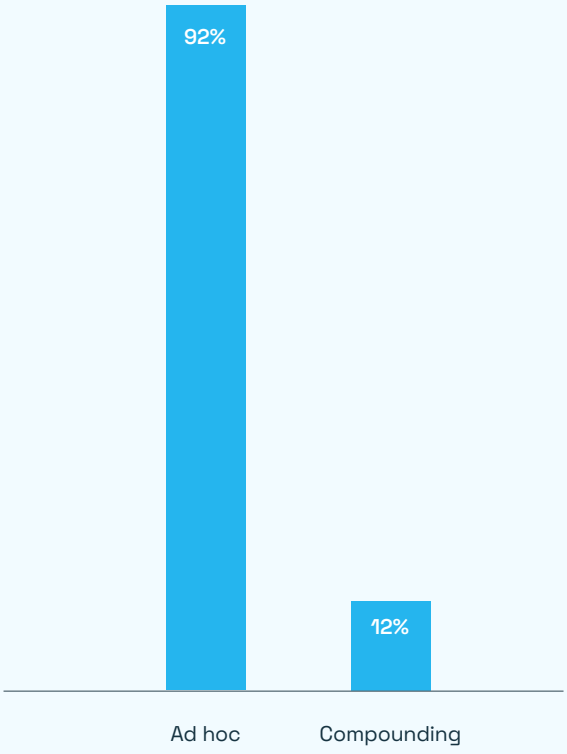
Latency is where mature and immature organizations most sharply separate. Among ad hoc organizations, 92% see significant or moderate latency risk in multi-agent workflows. Among compounding organizations, only 12% do.

One cause of this divergence is the same fragmentation that hurts navigation: **54% say siloed vector stores significantly or moderately limit how well their agents perform, because pulling context from several disconnected systems adds a delay at every hop, and those delays stack up over long tasks.**

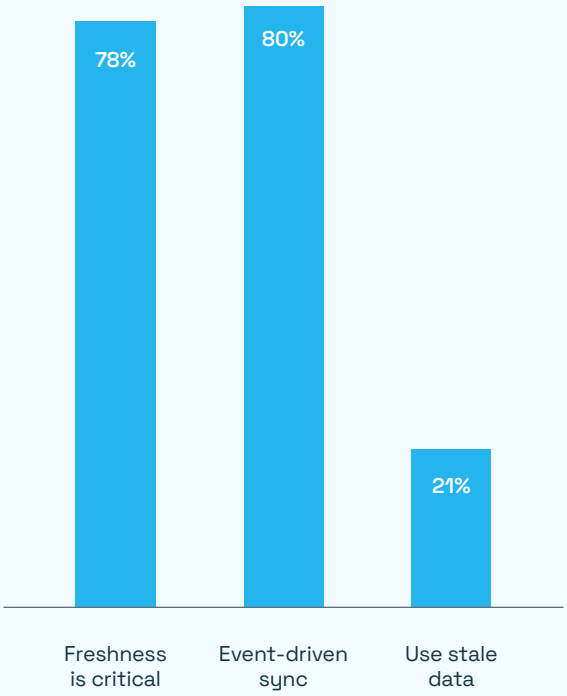
Freshness follows the same pattern. 78% say proving context is fresh before inference is very important. 80% agree that event-driven synchronization (updating context when source data changes, rather than in nightly batches) is becoming the standard for enterprise AI. Yet 21% admit their systems frequently or usually run critical decisions on stale or batch-synchronized data.



**Latency risk drops with AI maturity**




**Organizations value fresh context, but many still rely on stale data**



## Governance & fragmentation are downstream from the same runtime problems

Many organizations treat governance as a tooling gap and fragmentation as an architecture gap, but they are two sides of the same problem: you can't govern what you can't assemble.

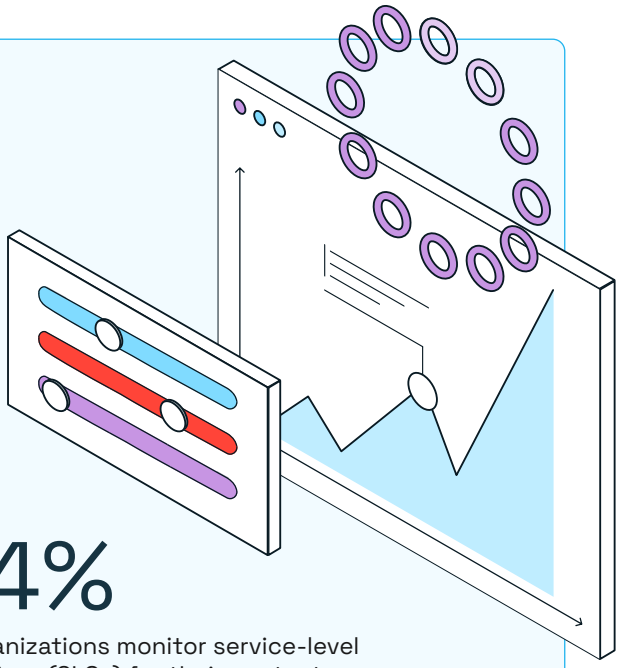
Governance is the barrier leaders name the most often, and it comes with a telling contradiction. **37% consider governance tooling the single biggest barrier to production-ready maturity**, the top answer overall. **65% rank it as essential for reaching production-grade context**, second only to compounding intelligence.

The contradiction is in the self-assessment: 60% rate their organization as ineffective or very ineffective at governing access to AI context sources, but **58% are confident or very confident that their context systems are governed effectively.**

Confidence has detached from capability, which is dangerous for a problem like governance.

The fragmentation component of the problem shows up right next to governance in the ranking of barriers. **22% name persistent memory as the second biggest barrier to production-ready maturity, and 19% name a unified runtime architecture as the third.**

These are the systems that enable organizations to assemble the right context in one place, the moment an agent runs. Their absence is why governance is so hard to even begin to build.



84%

of organizations monitor service-level objectives (SLOs) for their context systems inconsistently.

There is no single place to enforce rules, so even well-thought-out policies tend to fail.

The same absence shows up in operations. **84% monitor service-level objectives for their context systems inconsistently, and 78% capture decision traces inconsistently**, so when an agent does something wrong, there is often no record of what context it saw or why it acted.

A system you cannot observe is a system you cannot govern, and a system you cannot assemble is one you cannot observe.

**Top barriers to production-ready maturity**

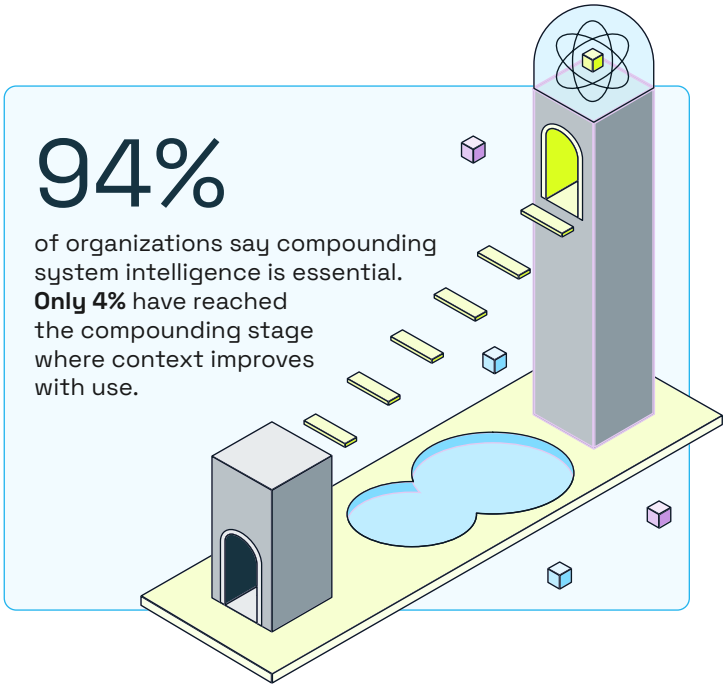
## The opportunity ahead of you (& the opportunity cost of missing it)

The thread running through all of this data is this: the capability nearly everyone says matters most is the one almost no one has built. Read one way, it's a problem. Read another way, it's an opening. The advantage of context engineering compounds as you build it, deepening the moat against competitors who have not yet started.

### Compounding context is the next competitive advantage

**94% of organizations say compounding system intelligence is essential for production-grade context maturity, but only 4% have reached the compounding stage where context actually improves with use.**

A gap that wide doesn't usually last. Either the belief fades, or practice catches up to theory, and the belief here is too strong and too broadly held to fade. That means practice will catch up—eventually and unevenly—and the organizations that start first will be building on foundations their competitors are still trying to pour.




### What the leaders look like

The few organizations at the compounding stage aren't just doing more of what everyone else does or doing it better. They're clearing the structural gaps, and it shows across every dimension the rest of the field struggles with:

- They have modeled their domain: **54% of the most mature organizations have a well-defined context graph.**
- They know where they stand: **49% can measure their production readiness very accurately.**
- They are fast: only **12% of compounding organizations perceive significant latency risk.**
- And their sense of what is dangerous has fully inverted relative to the old, model-first worldview: **97% agree that a traversal failure poses greater risk than a hallucination.**

That said, gaps still exist, and pportunities remain.

### Even the leaders have unfinished work

The gap between theory and practice is wide, but that doesn't mean the few organizations that've crossed it have completed their work.

The data shows that **69% of the most mature organizations still rely on stale data for critical decisions.** The leaders are far ahead in navigation, speed, modeling, and governance, but they have not yet fully matured in freshness.

If the best organizations in the field are still working on this, no one gets to treat context engineering as a solved problem. The work is ongoing even at the frontier, which means the frontier remains open.

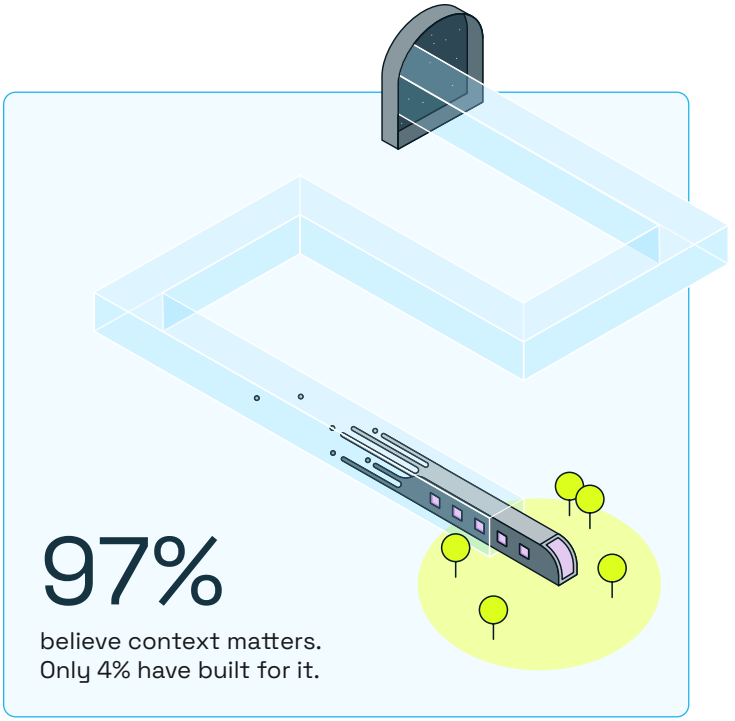
## The window is closing

Step back from the individual numbers, and one pattern remains. Belief in context engineering is close to universal, and real implementation is close to absent. 97% believe context matters. 4% have built for it.

Everything in the space between those two numbers is open ground, but it will not stay open forever. The architecture is being written now, in the standards that are forming today, and the systems teams are building on them.

Organizations that do the work will build a compounding advantage. Organizations that keep improvising will fall further and further behind.

As foundation models converge and become components you can swap in and out, the model stops being the deciding factor. Every competitor has access to the same models. What differs is what the model knows about your organization, your customer, and your workflow at the moment it acts. And what the model knows is set entirely by the quality of the context layer feeding it.




But the quality of your context layer isn't simply good or bad. It's not even a spectrum from insufficient to sufficient. To figure out how your organization is positioned, read The Context Engineering Maturity Model, a framework for building production-ready agentic systems that includes a self-assessment to help you identify growth opportunities.

## About this research

This report is based on a survey of IT and AI infrastructure leaders on how they build, feed, and govern context for production AI agents. Respondents span organizations at every stage of the maturity model described above, from ad hoc, prompt-by-prompt approaches to the small minority running compounding, self-improving context systems. Percentages reflect the share of respondents selecting each option; where questions allowed multiple selections, totals may exceed 100%.

Redis

Trust
Privacy
Terms of use
Legal notices

**Use cases**
Vector database
Feature stores
Semantic caching
NoSQL database
Leaderboards
Data deduplication
Messaging
Authentication token storage
Fast-data ingest
Redis Search
All solutions

**Industries**
Financial services
Gaming
Healthcare
Retail
All industries

**Compare**
Redis vs. Elasticache
Redis vs. Memcached
Redis vs. Memory Store
Redis vs. Redis Open Source

**Company**
Mission & values
Careers
News

**Connect**
Community
Events & webinars

**Partners**
Amazon Web Services
Google Cloud
Microsoft Azure
All partners

**Support**
Professional services
Support