

# THE COMMODITIZATION OF INTELLIGENCE

Why Context Becomes the Strategic Layer in the Age of LLMs

---

Robert Ford · Toolsie · May 2026

## Abstract

The first phase of the generative AI era was defined by capability shock. During this phase, competitive advantage centered on model capability: larger training runs, proprietary architectures, frontier-scale compute, and access to increasingly scarce technical talent.

This paper argues that the industry is entering a different phase. As high-quality reasoning capabilities commoditize through open-weight models, provider competition, and declining inference costs, raw intelligence alone becomes a less durable source of differentiation. Competitive advantage begins shifting upward into **context infrastructure**: the systems responsible for preserving operational continuity, institutional memory, constraints, workflow state, and cumulative understanding across interactions.

The central claim is not that models cease to matter. Rather, once baseline reasoning quality crosses a practical threshold, the dominant constraint shifts from intelligence generation to contextual coherence. Durable advantage accrues to systems that reliably *preserve, govern, and compound* understanding over time.

**This paper makes three distinct claims with different epistemic standing:**

Claim	Domain	Confidence
Continuity failures are real and systematically underestimated in sustained AI deployment.	<b>Operational</b>	<i>Well-supported</i>
As baseline reasoning commoditizes, context infrastructure emerges as the primary differentiating layer.	<b>Structural</b>	<i>Directionally plausible</i>
Organizationally-embedded context systems create durable competitive advantage resistant to commoditization.	<b>Economic</b>	<i>Speculative; logically supported</i>

The paper argues all three are likely. It does not argue they are equally proven.

---

## 1. Introduction — The Wrong Scarcity

For much of the generative AI boom, the dominant assumption was straightforward: intelligence itself would remain scarce. The industry organized accordingly. Competitive advantage concentrated around larger training runs, proprietary model architectures, GPU access, and the capital required to sustain frontier-scale development.

That assumption is already beginning to weaken. Over a remarkably short period, high-quality reasoning capabilities have become increasingly interchangeable. Open-weight models continue narrowing

performance gaps. Model routing layers increasingly abstract provider choice away from end users. In many practical business contexts, users cannot reliably distinguish which frontier model produced a given output.

This does not imply that frontier capability no longer matters. It does. But it changes where constraints emerge. For a growing category of operational work, the limiting factor is no longer whether a system can generate competent reasoning in isolation. The limiting factor is whether the system understands enough situational context to reason correctly, consistently, and cumulatively over time.

*The intelligence is increasingly available. Continuity is not.*

## 2. The Commoditization Curve of Intelligence

### 2.1 Infrastructure Commoditization Patterns

Technological history repeatedly follows a recognizable pattern. Capabilities initially emerge as scarce, expensive, and highly differentiated. Over time, competition, standardization, and scale compress those advantages into increasingly accessible infrastructure layers. Electricity, cloud computing, storage, bandwidth, and GPS all followed this trajectory.

Generative AI appears increasingly likely to follow a similar path. The first phase rewarded frontier capability creation. As the ecosystem matures, two parallel forces reduce the durability of capability-based differentiation: *performance convergence* and *abstraction layers*. The strategic question gradually changes from *which model is smartest* to *which system produces the most operationally reliable outcomes over time*.

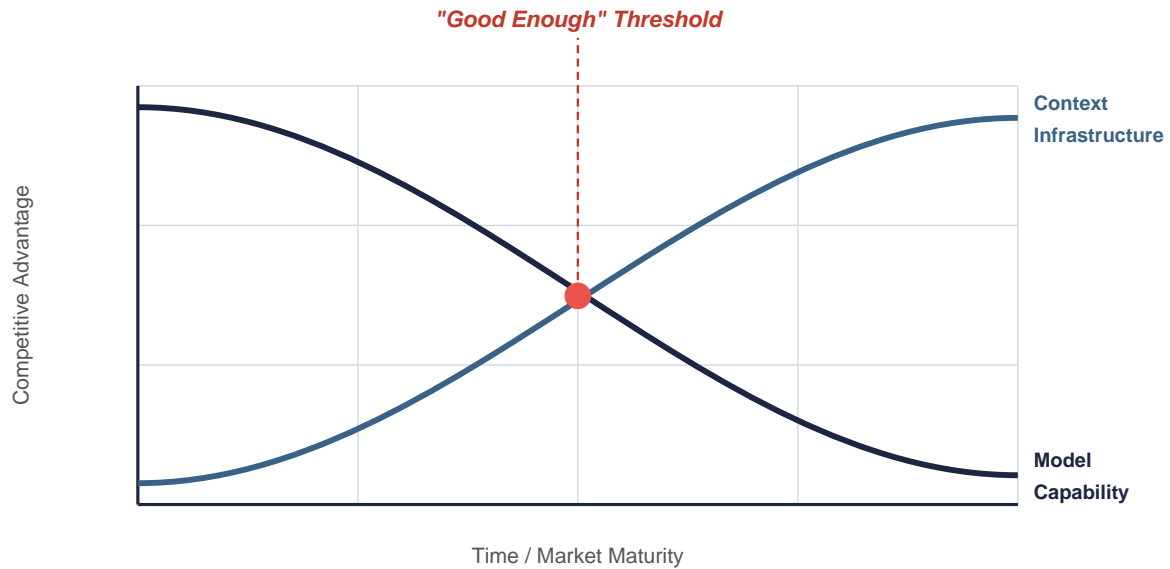


Figure 1. As model capability commoditizes, context infrastructure emerges as the primary source of competitive differentiation.

## 2.2 The "Good Enough" Threshold

Not all capability gaps matter equally. A frontier model may outperform another system on advanced reasoning benchmarks while producing little meaningful operational advantage within a constrained workflow containing strong contextual grounding, domain constraints, retrieval systems, and structured decision frameworks.

In practice, many business processes do not fail because the model lacks abstract intelligence. They fail because the system lacks sufficient situational understanding: missing organizational history, forgotten prior decisions, absent workflow state, inconsistent policy application. As baseline reasoning quality improves across the market, the relative importance of contextual failures increases. The result is a gradual inversion: *intelligence remains necessary, but context increasingly determines usefulness.*

## 3. Stateless Intelligence as a Structural Limitation

*A note before the analysis: the failure mode described in this section is not hypothetical. It is common enough to be observed across independent deployment environments. The following scenario is composite but representative.*

*A small clinical practice deploys an AI assistant to support care coordination — intake summaries, transition planning, follow-up documentation. Early sessions perform well. Six weeks in, a staff member opens a new session for a patient previously discussed in two earlier conversations. In that second session, the patient's daughter had made clear she was not able to provide home support — a constraint relevant to any discharge recommendation. The*

*constraint was noted. It was not loaded into the new conversation.*

*The system, working from the intake information available in the current session, recommends home discharge as the strongest option. The recommendation is well-reasoned, contextually appropriate to the information it has, and wrong. Not because the model lacked intelligence. Because the system lacked operational memory.*

*The care team catches it. This time. The underlying problem — a deployment architecture that treats every session as a fresh start — remains.*

### 3.1 The Hidden Cost of Context Reconstruction

Most current AI interaction models remain fundamentally stateless. A session begins. Context is reconstructed manually. The model produces outputs. The session ends. Operational understanding largely disappears. This creates a recurring burden: *perpetual reorientation*. Organizations do not merely pay for inference — they pay for continuous reorientation of systems that cannot remember where they were.

### 3.2 Stateless Systems and Decision Drift

Stateless interaction models create a secondary problem: *decision drift*. Without persistent operational memory, systems repeatedly renegotiate assumptions that should remain stable. Architecture decisions get re-litigated. Writing tone becomes inconsistent. Policy exceptions are forgotten. The result is not merely inefficiency — it is organizational instability that compounds invisibly.

What is specifically new here is not the problem. Institutional fragmentation and decision drift predate AI by decades. What changes when probabilistic AI systems participate operationally is the rate and scale at which these problems manifest, and the degree to which they become structurally embedded in automated workflows.

---

## 4. The Context Stack

Discussion around AI context frequently collapses multiple distinct concepts into a single category. This paper proposes a layered model — the **Context Stack** — organized by a single distinguishing principle: the *decay rate* of each layer's operational value.



Figure 2. The Context Stack: five layers organized by persistence. Each layer presents a distinct engineering and governance challenge.

#### 4.1 Session Context

The immediate information available within a single interaction window. Highly ephemeral — once the interaction ends, most of its operational value disappears unless explicitly preserved elsewhere.

#### 4.2 Workflow Context

The operational state of ongoing work: current objectives, active tasks, project stage, unresolved blockers, open decisions, and recent progress. Spans multiple interactions and enables *work resumption* rather than simple conversation continuation.

#### 4.3 Operational Context

The persistent rules and constraints governing how work should be performed: organizational policies, architectural standards, regulatory requirements, brand voice, approval structures. Functions less like memory and more like institutional discipline.

#### 4.4 Institutional Context

Accumulated organizational knowledge distributed across time, teams, systems, and decisions. The problem is rarely knowledge absence — it is *contextual fragmentation*.

#### 4.5 Behavioral Context

Patterns in how individuals or organizations operate over time: decision tendencies, communication preferences, strategic priorities, risk tolerance. The most sensitive and potentially powerful layer of the stack, and the one requiring the most careful governance.

## 5. From Intelligence Systems to Continuity Systems

Most current AI products remain fundamentally designed as *intelligence surfaces*. A user asks a question. The model generates an answer. The interaction concludes. Even sophisticated implementations frequently preserve this core interaction pattern.

This design assumption increasingly collides with the realities of sustained professional work. Most meaningful work does not occur as isolated prompts. It unfolds across extended sequences of decisions, revisions, constraints, and accumulated understanding.

This distinction marks the transition from **intelligence systems** to **continuity systems**. Intelligence systems optimize for response quality within isolated interactions. Continuity systems optimize for cumulative coherence across extended operational lifecycles.

*Because once intelligence becomes abundant, continuity becomes the scarce resource.*

---

## 6. The Economic Shift — From Intelligence to Coordination

### 6.1 Intelligence as Infrastructure

Infrastructure technologies typically follow a recognizable economic trajectory. Electricity, cloud computing, storage, and GPS all followed variations of this path. The companies that captured durable value were not primarily those that owned the infrastructure layer after it commoditized. They were those that built the most operationally valuable systems on top of it.

### 6.2 The Shift From Generation to Governance

As intelligence becomes more abundant, the operational challenge shifts from generation to governance. Generating plausible outputs is no longer the primary difficulty. The difficulty is ensuring that outputs remain contextually coherent, operationally aligned, historically informed, and organizationally trustworthy across time.

### 6.3 Why Context Systems Create Durable Advantage

A legitimate objection: context systems may commoditize as rapidly as models did. The answer depends on understanding what context systems actually accumulate. Unlike model capability — largely replicable given sufficient compute and data — operational context is organizationally specific, domain-embedded, and migration-resistant.

A mature context system contains: the accumulated decision history of an organization, its encoded operational constraints, its workflow patterns, its institutional knowledge — all structured for machine consumption. This *compounds with every interaction* and creates asymmetric switching costs that increase over time.

### 6.4 Context Systems as the Emerging Application Layer

Applications increasingly differentiate not through isolated intelligence generation but through contextual accumulation, operational coherence, and continuity preservation. The most valuable systems may be those that most effectively compound understanding over time.

## 6.5 The Incumbent Absorption Problem

The most serious challenge is not that context systems commoditize in isolation. It is that incumbent platforms may absorb continuity capabilities before independent context systems reach strategic scale.

Incumbents — legacy or AI-native — build continuity within the boundaries of systems they already own. Their context governance is necessarily *platform-bounded*: shaped by existing product incentives and bounded by permission and data silos between competing platforms. A Microsoft context layer governs Microsoft environments. A Salesforce Einstein layer governs Salesforce workflows. An organization running workflows across both receives **platform-bounded continuity** on each side — and a gap in between.

This creates a structural opening that platform generalization cannot close. The most defensible positions for independent context systems are precisely those where operational continuity must span multiple incumbent platforms, cross-functional domains, or consequential decision environments not owned by any single vendor: regulated healthcare workflows, multi-system nonprofit operations, Medicare and insurance eligibility decisions, legal and compliance processes that cross CRM, document, and communication boundaries.

---

## 7. Enterprise Implications

Most organizations do not primarily suffer from an absence of information. They suffer from fragmentation. Institutional knowledge exists but remains distributed across documents, meetings, chat systems, disconnected software, and tribal memory.

### 7.1 The Failure of Stateless Deployment Models

Many organizations initially approached AI deployment as an interface problem: deploy a capable model, provide access, and productivity improvements would follow. In practice, organizations frequently encounter diminishing returns — not because the model lacks intelligence, but because the deployment lacks continuity. The result is *perpetual reorientation at organizational scale*.

### 7.2 AI-Native Operational Design

Meaningful transformation may require more than attaching AI interfaces to existing workflows. In AI-native operational environments, decisions become durable artifacts. Workflow state persists across sessions. Institutional memory becomes machine-accessible. Constraints become structurally encoded rather than culturally assumed.

### 7.3 The Governance Problem

Persistent context systems introduce new categories of governance complexity. As systems accumulate organizational memory, behavioral patterns, and strategic assumptions, questions emerge regarding privacy, access control, retention policies, explainability, and contextual corruption.

Persistent systems can amplify errors as effectively as they preserve useful continuity. Incorrect assumptions may calcify. Outdated strategies may persist beyond relevance. A continuity system that faithfully preserves a flawed strategic assumption across two years of organizational decision-making is not a neutral failure — it is an amplifier of that flaw at scale.

*Context without governance is not an asset. It is a liability that compounds.*

#### 7.4 The Evaluation Problem

The field does not yet have reliable methods for measuring continuity quality. Decision drift is hard to detect systematically — it typically surfaces through operational failure rather than dashboard monitoring. Organizations deploying persistent context infrastructure should define in advance what degradation looks like. Without that operational definition, governance becomes aspirational rather than functional.

## 8. Counterarguments and Limits

The argument presented in this paper is directional rather than absolute. Several important counterarguments complicate the thesis.

### 8.1 Frontier Capability May Reassert Dominance

The paper's convergence assumption may fail. Breakthroughs in reasoning, multimodal synthesis, or autonomous execution could reintroduce substantial capability asymmetries. A discontinuous capability breakthrough could render context infrastructure substantially less important. This paper does not dismiss that possibility. It argues the current trajectory favors context systems, while acknowledging that trajectory is not destiny.

### 8.2 Larger Context Windows May Reduce the Problem

Rapidly expanding context windows may absorb some continuity challenges at the model layer. This objection is partially correct. Volume is not equivalent to coherence. An organization that loads its entire communication history into a context window has not solved its institutional memory problem — it has created a larger retrieval problem with less governance. *Context windows and context infrastructure are complementary, not competing solutions.*

### 8.3 Autonomous Agents May Alter the Dynamic

Even under a fully agentic scenario, the governance problem remains. Who defines the constraints within which agents operate? Who audits the context those agents accumulate? These are not problems agents solve for themselves — they are problems organizations must solve for agents.

### 8.4 Context Systems May Commoditize as Well

Generic continuity tooling may commoditize as rapidly as models did. The distinction that matters is between generic context infrastructure and organizationally-embedded context systems. Durable advantage does not accrue to the infrastructure. It accrues to the accumulated, governed, domain-specific context that particular organizations build within it.

---

## 9. Building Against This Thesis

*A disclosure before this section: I am building a product against the argument made in this paper. What follows is practitioner evidence, not independent proof. The constraints described here are not theoretical — they surface in live build environments, and the systems I've built to address them produce observable workflow advantages. That does not validate the thesis at market scale. It shows the thesis is worth building against.*

Over the past several years, I have been constructing AI-native workflows across creative work, product development, and social impact consulting — and progressively formalizing what I was learning into a platform called **Toolsie**. Toolsie is an ecosystem of AI-powered applications and Navigators built around a single foundational conviction: that the limiting constraint in AI deployment is not intelligence generation but contextual coherence.

The environment in which Toolsie is built is itself an implementation of the thesis. I have come to call the underlying pattern a *persistent, governed workspace architecture* — a structure consisting of four integrated components:

<b>Structured knowledge base</b> —	Domain-specific knowledge files built to a consistent standard that prioritizes high-signal structured density for LLM consumption. Governed representations of entities, relationships, constraints, and operational norms — maintained, versioned, and updated as understanding evolves.
<b>Skill system</b> —	A library of reusable, governed operational protocols — each covering a specific workflow, build pattern, or analytical process — that encode accumulated methodology rather than requiring it to be reconstructed from conversation.
<b>Governance layer</b> —	Explicit validation standards that flag outputs violating established decisions, encode constraint hierarchies, and prevent operational drift across sessions.
<b>Session architecture</b> —	Structured open and close protocols that ensure every working session begins from accumulated state rather than from zero, and ends by updating that state for the next.

These four components operate as a system. The knowledge base compounds with domain depth. The skill system compounds with build experience. The governance layer compounds with accumulated decisions. The session architecture ensures none of that compounding is lost between sessions.

None of this is straightforward to build or maintain. The governance tensions described in Section 7.3 are real and present in every product decision. These are not solved problems. They are active design challenges that surface constantly in practice.

*The goal is not the smartest system. It is the most coherent one.*

## 10. Conclusion — The New Scarcity

The first phase of the LLM era asked a foundational question: can machines produce useful reasoning? Increasingly, the answer is yes.

The emerging challenge is different, and more organizationally demanding. As high-quality intelligence becomes more abundant, the strategic bottleneck shifts toward maintaining contextual coherence across time, workflows, decisions, and organizational environments.

Organizations deploying AI in sustained operational environments increasingly encounter failures not because intelligence is absent, but because understanding fragments: decisions disappear, constraints drift, workflows reset, institutional memory fractures. The systems that prove most valuable will be those that most reliably accumulate, govern, and compound operational understanding over time — systems whose value *increases with use* rather than resetting with every session.

The strategic question therefore evolves from *how intelligent is the model* to *what does the system remember, preserve, govern, and compound*.

Frontier capability will continue to matter. But as baseline reasoning quality increasingly crosses practical thresholds for operational work, competitive advantage appears likely to migrate into the contextual architectures surrounding intelligence itself. The emerging constraint may not be intelligence generation. It may be continuity — and the organizational discipline to govern it well.

---

Version history: Draft 1 — initial thesis. Draft 2 — claim-level epistemic stratification. Draft 3 — incumbent absorption (§6.5), four-component definition, governance engineering acknowledgment. Draft 4 — AI-native incumbent layer, concrete failure vignette (§3), §9 reframed as practitioner evidence. Final — §7.4 (evaluation problem) added, conclusion corrected.