

WHY ORCHESTRATION—NOT MORE AGENTS—IS THE KEY TO SCALING ENTERPRISE AI



The multi-agent AI era isn't coming—it's already here. According to [Deloitte](#), 75% of organizations are investing in AI agents, driving a surge in enterprise adoption. And according to [IDC](#), this isn't incremental—it's a structural shift, with agentic AI-driven investment expected to reach \$1.3 trillion by 2029.

On the surface, more agents should mean more value.

It doesn't.

The very force accelerating AI adoption—agent proliferation—may ultimately constrain its impact. As Deloitte notes, once organizations move toward multiagent systems, orchestration becomes essential to unlocking their full potential. Yet many enterprises still frame the problem as one of intelligence: bigger models, smarter agents, more autonomy.

That framing is incomplete. The real challenge facing enterprises today isn't intelligence—it's execution.

This article explores why orchestration, not additional agents, is the critical missing layer in enterprise AI. It explains how agent sprawl creates complexity, why agentonly orchestration falls short, and why enterprises must treat orchestration as a control plane—coordinating agents, workflows, data pipelines, and legacy systems—to reliably translate AI into real business outcomes.

Five realities enterprises must confront

The challenge isn't whether agents will be adopted. That's already happening. The real question is whether enterprises are prepared for what comes next.

1. Agent sprawl is inevitable

As agents deliver value, organizations will deploy more of them—quickly, on a multitude of platforms. What starts as a targeted approach becomes a distributed ecosystem of autonomous components. Left unchecked, this fragmentation creates a coordination problem—multiple agents making decisions across disconnected environments with no shared understanding of timing, dependencies, or outcomes. It's a bit like the "Not Hot Dog" app from the series [Silicon Valley](#)—a model that could perfectly identify a hot dog and confidently label everything else as "not hot dog." Technically impressive. Practically useless beyond a very narrow context

2. Orchestrating agents isn't enough

The instinctive response is to orchestrate the agents themselves. But that only solves part of the problem.

Agents don't operate in isolation—they plug into larger business processes. Financial close, trade reconciliation, inventory replenishment, even data pipelines that power inference, RAG, and BI all remain multi-step workflows. Agents may automate decisions within them, but they don't run them end-to-end.

Which means orchestration can't be designed around agents alone. It has to coordinate agents alongside scripts, APIs, batch jobs, and serverless functions that make up the rest of the process.

Otherwise, you're not eliminating complexity—you're creating another orchestration silo that still has to be connected to everything else.

3. AI-ready data doesn't solve itself

Another emerging lesson is that agents are only as good as the data they consume. As enterprises invest heavily in models and agents, many discover that the real bottleneck is data readiness. Fragmented, outdated, or poorly governed data leads to unreliable outputs. What is new, however, is orchestration's role in resolving that bottleneck. Preparing AI-ready data requires coordinating data pipelines, application workflows, and event triggers across the enterprise. The intelligence layer depends on that foundation.

4. The enterprise is more hybrid than ever

Despite the hype around new technologies, most enterprises operate in deeply hybrid environments. Mainframes remain the lifeblood of many of the world's largest companies—powering core transactions and systems of record—while cloud-native platforms and microservices drive new digital experiences and AI innovation. Modern data tools interact with both, and critical processes now span generations of infrastructure. These systems aren't disappearing anytime soon. The challenge isn't replacing them—it's ensuring that new AI-driven capabilities work alongside them. That's where orchestration across the entire stack becomes essential.

5. Reliability still defines success

In the race to deploy the newest AI tools, it's easy to overlook something fundamental: reliability. Enterprise workflow orchestration has long been judged by a simple standard—it just works. Think of it like a Swiss watch: precise, dependable, and trusted to run critical operations. AI systems must meet that same bar. Autonomy is powerful, but enterprises won't accept fragile automation in mission-critical environments. The orchestration layer must ensure workflows remain predictable, auditable, and resilient—even as intelligence becomes more distributed.

The path from complexity to simplicity

Most enterprise problems aren't glamorous. It's easy to get excited about frontier models, GPUs, and agents that can reason and act—that's where the headlines are. But the problems haven't changed: How do we accelerate financial close? How do we detect and prevent fraud before it impacts customers? How do we execute trades reliably at market speed? How do we keep shelves stocked and orders fulfilled? How do we ensure critical healthcare data is available when it's needed most?

Simple in nature. Relentless in execution.

Delivering those outcomes means coordinating workflows across agents and traditional systems like ERPs, CRMs, data lakes that span everything from multiple clouds to mainframe systems—all while meeting SLAs, audit, traceability, and explainability requirements. There's nothing flashy about that. But without it, AI stays in the lab and never graduates to production environments, which is where systems deliver business value.

This is why orchestration isn't just a tool—it's a strategy. A control plane for execution.

The goal isn't more agents—it's better outcomes. Achieving that requires something often overlooked: simplicity. Orchestration is what turns complexity into simplicity. As Leonardo da Vinci put it, "*Simplicity is the ultimate sophistication.*"