

AI Agents Are Entering Their Rebuild Era as Enterprises Confront the Reliability Problem



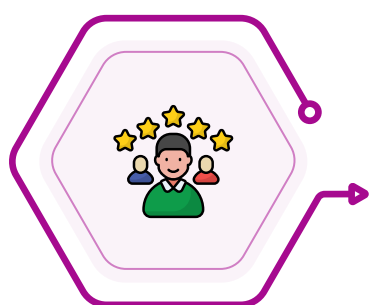
The first wave of enterprise AI agents was about speed. Get something working, get it in front of users, and prove the technology could do real work. That phase is ending. As agents move out of demos and into day-to-day production, a less glamorous question has taken over the conversation: can you actually depend on them? For a growing number of organizations, the honest answer has been no, at least not yet, and that realization is pushing the entire field into a rebuilding phase.



The lesson many teams are learning the hard way is that the raw capability of the underlying language model is not what decides whether an agent succeeds in production. A model can be brilliant in isolation and still fail as part of a live system. What separates a reliable agent from a fragile one is everything around the model: whether the workflow can survive a crash, hold onto its state, recover when something downstream breaks, keep inference costs under control, and coordinate cleanly across the APIs, tools, and internal systems it has to touch.

Why Version 2.0 Is Becoming the Norm

After rushing first-generation agents into service, companies are now circling back to rebuild them on sturdier foundations, this time designing around orchestration, observability, governance, and recovery rather than bolting those concerns on later. That pattern came through clearly in comments from Preeti Somal, Senior Vice President of Engineering at Temporal Technologies, speaking at a recent AI Impact Series event in New York.



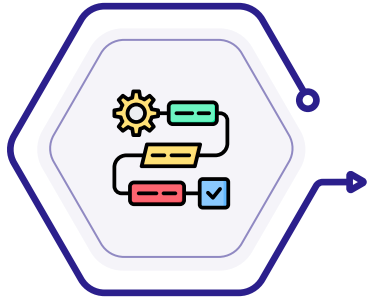
"We do have a lot of customers that come to us where they're building version 2.0 of the same agent," Somal said. "They had to move really fast, but they didn't take care of the plumbing. Things crash and burn, and then they're back to rebuilding with the reliable foundation."

For more content like and follow me:



@bertblevins

INCGPT.COM



Temporal is a workflow orchestration company whose infrastructure predates the current agentic AI boom, which gives it a useful vantage point. The way Somal frames it, the broader enterprise is arriving at a single conclusion: production AI cannot run on hope. It needs durable execution, real state management, visibility into what a workflow is doing, and dependable mechanisms to recover when a model or a connected system fails.

Agentic AI Did Not Invent These Problems — It Amplified Them

None of the underlying engineering challenges here are new. Durability, state, failure recovery, and coordination have shadowed distributed systems for decades. What is new is the intensity. As Somal put it, the patterns themselves are familiar, and AI simply supercharges them.

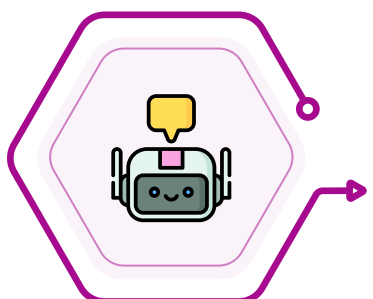


The reason agents strain these old fault lines so badly is their shape. An agentic system is rarely a single quick call. It is often a long-running, multi-step process that stretches across several services, models, APIs, and tools, sometimes unfolding over hours or days. A single workflow might invoke multiple large language models, pull from retrieval systems, fire off actions in external applications, and carry state the entire way through. Every one of those touchpoints is a place where things can go wrong, and the failure modes only reveal themselves once the agent is actually running in the wild.

“People will write agents but haven’t thought about what happens if the agent crashes,” Somal said. “Am I going to need to run the entire agent flow again?”

When a Crash Costs Real Money

That question is not academic, especially for organizations watching their AI budgets. With traditional software, restarting a failed process is mostly an inconvenience. With agents, it can be expensive. If a long workflow dies partway through and has to start over from the beginning, every model call it already made gets paid for again. The result is multiplied inference costs, added latency, and a degraded experience for the customer waiting on the other end.



Reliable agent design treats this as a first-order concern. The goal is to make workflows resumable, so a crash means picking up from the last good checkpoint rather than throwing away all the work and money already spent. That is the difference between an agent that merely demonstrates well and one that an enterprise can afford to run at scale.

A Familiar Pattern From the Cloud Era

There is precedent for this moment. Somal likened it to an earlier stage of enterprise cloud adoption, when organizations rushed to migrate workloads before stopping to ask whether their underlying architectures needed to be redesigned for the new environment. Lifting and shifting got systems into the cloud quickly, but the real benefits, and the real reliability, only arrived once teams went back and rebuilt for how the cloud actually worked.





Agentic AI appears to be repeating that arc, just faster. The initial enthusiasm produced a lot of agents that technically functioned but were never engineered to endure. Now comes the less exciting, more durable work of getting the plumbing right.

What a Reliable Agent Foundation Looks Like

The emerging consensus is that production-grade agents need a set of capabilities baked in from the start rather than discovered after the first outage:



Durable execution, so a workflow can survive crashes and resume instead of restarting from scratch.



State management, so the agent remembers where it is and what it has already done across long-running tasks.



Observability, so teams can see what an agent is actually doing at each step and diagnose problems when they occur.



Governance, so an agent's actions stay within defined, auditable boundaries.



Recovery mechanisms, so the system fails gracefully and bounces back when a model or a downstream service goes down.

The Takeaway

The agent reliability problem is not a verdict against the technology. It is a sign of maturation. Enterprises have proven that agents can do meaningful work; now they are confronting what it takes to make that work dependable, repeatable, and cost-effective in production. The rebuild era is less about chasing smarter models and more about engineering the foundation those models run on. The organizations that invest in that foundation now are the ones whose agents will still be standing after the hype settles.

