

DELIVERING PREDICTABLE OUTCOMES IN THE AI ERA



Enterprise IT has spent decades engineering predictability into its systems. Transactions commit or roll back, data pipelines land on time, and a workload that ran last night runs the same way tonight. That consistency is what lets a bank clear a payment and a hospital admit a patient. Predictability is the product.

AI is the first technology we've deployed at enterprise scale that behaves differently. Give a model the same input twice and it may reason its way to two different, equally defensible answers. That property is precisely what makes AI valuable on problems deterministic software was never built to solve, and it's also what makes AI hard to operate inside businesses that have organized themselves around certainty.

The capability question is settled. What enterprises are working out is how to embed non-deterministic AI inside processes built on certainty. An agent belongs in production when the work it was given completes, the evidence behind its decisions is visible, and a person can still intervene. Agentic AI governance is what now stands between a promising pilot and a production deployment.

Enterprise AI stalls on complexity

IDC projects its Global DataSphere will surpass 700 zettabytes by 2030. Google alone now

processes more than 3.2 quadrillion AI tokens a month, up from 9.7 trillion two years earlier. Nearly three-quarters of organizations operate hybrid cloud environments. And our own research finds 94 percent of organizations still treating the mainframe as a long-term platform or a home for new workloads.

Enterprises have managed the data platform, the hybrid estate, and the mainframe as separate programs, each with its own owners and controls. An AI agent placed inside a business process crosses all three domains in a single execution path. A process that once touched three systems now spans enterprise applications, data pipelines, public and private cloud, the mainframe, and a growing population of AI agents.

Our customers cite four primary concerns about scaling AI: data quality and fragmentation, the operational demands of new AI workloads, tool sprawl across hybrid environments, and the gaps between the mainframe and everything around it. All four concerns predate agentic AI, and agents compound each one by multiplying connections faster than organizations can govern them.

Orchestration restores predictability

An AI agent determines its course of action each time it runs, so two attempts at the same incident can follow unique paths to a diagnosis. But the agent doesn't execute its own proposals. Orchestration decides whether to permit the action, sequences it against work in flight, and records the result, and agentic AI governance enforces identity, scope, approval, and audit at that moment. A policy document works only if someone reads and follows it. Policy enforced in the execution path applies to every agent, every time, regardless of which team built it.

Enterprises already own orchestration built for their most critical workloads. Extending it to AI is faster and less expensive than standing up governance separately inside every AI project.

Any capable team can now assemble an agent that retrieves information, calls tools, and finishes a defined task. The difficulty sits around the agent, in giving it access to live operational context, a supported way to act, and an owner when conditions change. A customer's own agent can run inside those same controls, and it should.

Three outcomes determine whether that extension succeeds: scalability, so automation grows with the business; operational resilience, so disruption stays contained and recoverable; and trust and confidence, so teams can explain and defend what their systems did. The work we're announcing across our portfolio maps to these outcomes.

Governance must hold across every workload

Consider loan origination. Intake, processing, underwriting, decisioning, booking, funding, and servicing have long been orchestrated across a chain of enterprise systems. In the agentic version of that same process, several of those stages are handled by AI agents and AI orchestrators sitting alongside custom applications, real-time & batch data exchange, and traditional systems. The business process stayed the same while the mix of systems executing it changed. An agent in that chain matters for its effect on whether the process finishes, along with the dependencies, exceptions, and service levels around it.

That's co-orchestration of mixed workloads: deterministic tasks and non-deterministic agents coordinated as one process, with dependencies, service levels, and recovery applied uniformly

across both. Without it, agents run outside the system that governs everything else, the same problem enterprises solved when they consolidated unsanctioned scripts and schedulers.

Teams grant agents authority in proportion to what they can verify and reverse, so scope must be concrete. An agent with a defined task and a defined blast radius earns trust faster than one with broad latitude, and the boundary matters more than the capability.

In practice that means starting with a single task, measuring it against how the work runs today, and widening an agent's remit when the results hold up.

The same holds for the data underneath. An agent reasoning over stale or incomplete inputs will produce a defensible answer from bad evidence, and a well-coordinated process will carry that answer downstream without objecting. Knowing where data came from, whether it's fit to use, and what depends on it is an operational question now.

The mainframe raises the stakes further. It holds the transactions and the business logic the rest of the enterprise depends on, and the knowledge required to operate it sits with fewer people every year. Governance that stops at a platform boundary isn't governance. One set of policies must apply across every environment an agent touches, with a single audit path behind it, because governance built separately inside each product becomes as fragmented as the estate it was meant to control.

Three questions decide whether an agent runs

An agent proposes a fix. Three questions come before the action: how far this agent's reach extends, who extended it, and what undoing the fix would cost.

Those are questions about the system around the agent, and they're answerable only if the answers were built before the incident. An agent that can explain its diagnosis but not its authority won't get approval, and it shouldn't.

This is also why variance is tolerable. The same agent may reach the same conclusion by a different route on a different day, and that costs nothing when every route passes the same checks and leaves the same record. The route becomes a problem only when it decides what the agent was permitted to touch.

Production runs on discipline

The first phase of enterprise AI rewarded experimentation. Production asks a harder question. The number of agents an organization runs says little about whether any of them can be trusted with consequential work. What counts is whether they run inside the same discipline as everything else the business depends on, where work is sequenced correctly and a decision can still be explained months later to an auditor who was never in the room. That is what agentic AI governance does.

Models will change, and so will agent frameworks. The need for critical work to complete predictably will outlast both. A capability that cannot answer those three questions does not belong in the systems that run your business, whoever builds it. Enterprises learned once how to let software act on their behalf without watching every step, and predictability is still the product.

That is why the capabilities we are introducing sit in the execution path. Control-M's latest release advances AI-ready orchestration that connects AI agents to business workflows, available across both SaaS and self-hosted environments. Control-M's new AI Pilot capabilities bring conversational

analysis and workflow explanations into Monitoring viewpoints, so an operator can understand and resolve an issue in natural language instead of assembling that context by hand. Data Assurance for Control-M SaaS improves the quality of the data a workflow depends on, which matters here because an agent reasoning over unverified inputs will produce a confident answer from bad evidence. And Multi-Region Disaster Recovery protects mission-critical workflows against the one condition no agent can reason its way out of, which is the environment itself failing.

On the mainframe, where both the stakes and the skills problem are sharpest, BMC AMI Agentics for Ops and BMC AMI Agentics for Data put specialized agents to work investigating problems, explaining root causes, and recommending next steps, and bring the same expertise to Db2 performance and optimization. BMC AMI Assistant brings mainframe knowledge and AI assistance directly into Microsoft Teams, because evidence is only useful where the work already happens. New development capabilities go at the skills problem from the other end, generating synthetic test data from a program's own code to cut cycle time and improve quality. And enriched security context strengthens cyber resilience in the place where disruption is least affordable.

These capabilities run on the systems we have helped enterprises operate for four decades, and that is the ground any claim about dependable AI must stand on. The work now is making predictability hold while the systems inside it begin to reason. For the full release details, explore the What's New pages for Control-M and BMC AMI.

: IDC, *Market Forecast: IDC Global DataSphere Forecast, 2026–2030*, June 2026 (public abstract).
<https://my.idc.com/getdoc.jsp?containerId=US53425426>

: Google, "I/O 2026: Welcome to the agentic Gemini era," May 19, 2026.
<https://blog.google/innovation-and-ai/sundar-pichai-io-2026/>

: Flexera, *2026 State of the Cloud Report*, March 2026, p. 13.
<https://resources.flexera.com/web/pdf/Flexera-State-of-the-Cloud-Report-2026.pdf>

: BMC, *2026 Mainframe Survey*. <https://www.bmc.com/info/mainframe-survey.html>

Explore what's new:

- [What's new in Control-M](#)
- [What's new in BMC AMI](#)
- [2026 BMC Mainframe Survey](#)