The Agentic Future Won't Be Monolithic

Strategies for Enterprises to Manage Complexity, Security, and Lock-In

Executive Summary

Drawing on insights from more than 800 decision-makers and purchasing influencers worldwide, this whitepaper examines the rapidly evolving state of agent adoption. While building agents is now a strategic priority for most organizations, complexity is emerging as a primary barrier to scaling. Vendor lock-in concerns are also widespread, shaping architectural decisions across industries.

Security and Complexity Are the Top Barriers

40% of respondents cite security as the #1 challenge in scaling agentic AI, with 45% struggling to ensure tools are secure and enterprise-ready. Technical complexity compounds the challenge. One in three organizations (33%) report orchestration difficulties as multi-model and multi-cloud environments proliferate (79% of organizations run agents across two or more environments).

Lock-in Concerns Are Real

Seventy-six percent of global respondents report active concerns about vendor lock-in—rising to 88% in France, 83% in Japan, and 82% in the UK. And these aren't theoretical anxieties. They center on the very layers of the stack that power agentic systems today.

Containerization Remains Foundational

94% use containers for agent development or production, and 98% follow the same cloud-native workflows as traditional software, establishing containers as the proven substrate for agentic AI infrastructure.

In this report, we break down the key findings from our research: how teams are architecting their agentic workloads today, the operational and security challenges they face, and what it takes to design a scalable, future-ready agent infrastructure.

The conclusion is clear: scaling agents doesn't require reinvention, it requires standardization. Open, interoperable and portable infrastructure must become the foundation, with secure-by-default runtimes, consistent orchestration and policy enforcement, and the flexibility to operate seamlessly across models, tools, frameworks, and agents. Teams that invest now in a trust layer built on container principles of isolation, portability and simplicity can move beyond point productivity gains to sustainable enterprise-wide outcomes while reducing vendor lock-in risk.

The Roadblocks to Scale: Security, Complexity, and Enterprise Readiness

As organizations move from experimenting with AI agents to scaling them in production, the challenges shift from feasibility to enterprise reality. The barriers are no longer conceptual; they define how far and how fast agentic systems can grow. Across the global sample, two obstacles dominate this next phase: security and technical complexity, each amplified by the growing diversity of models, tools, and deployment environments.

Technical Complexity: The Expanding Challenge

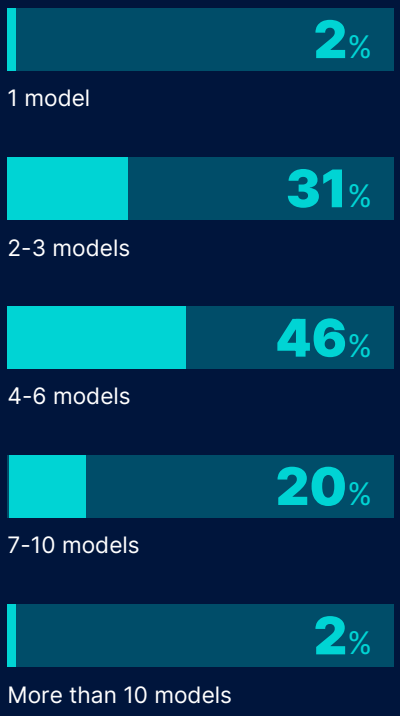
While security defines whether agents can scale, technical complexity determines how easily they do.

33% **One in three organizations (33%) cite technical complexity as a top barrier, encompassing orchestration, model diversity, and infrastructure fragmentation.**

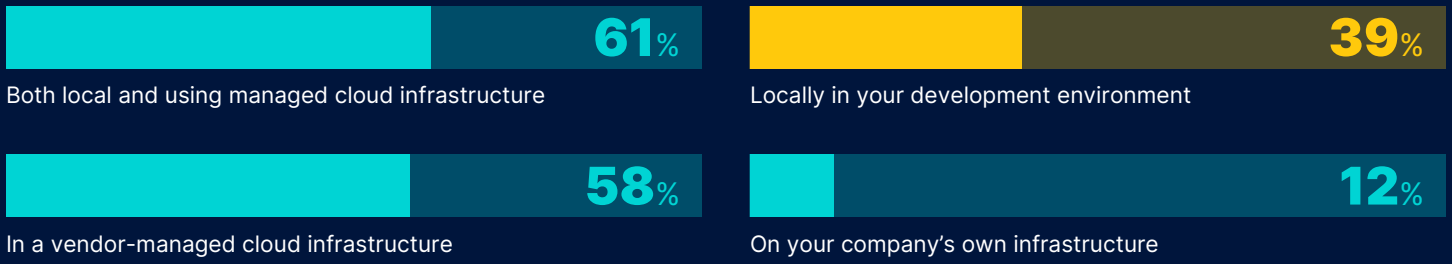
Complexity Manifests in Several Dimensions

1. **Multi-modal ecosystems:** Nearly two-thirds of organizations (61%) combine cloud-hosted and local models. This strategy increases integration effort and performance tuning complexity, but it is also intentional. Complexity grows further within the model layer itself: 46% of organizations report using between four and six models within their agents, while only 2% rely on a single model. Enterprises are adopting multi-model and multi-cloud architectures to give teams greater control over performance, customization, privacy, and compliance, reflecting the practical, use-case-driven nature of today's agentic ecosystems.

Number of Models Actively Used Within Agent



Where AI Agent Models Are Executed

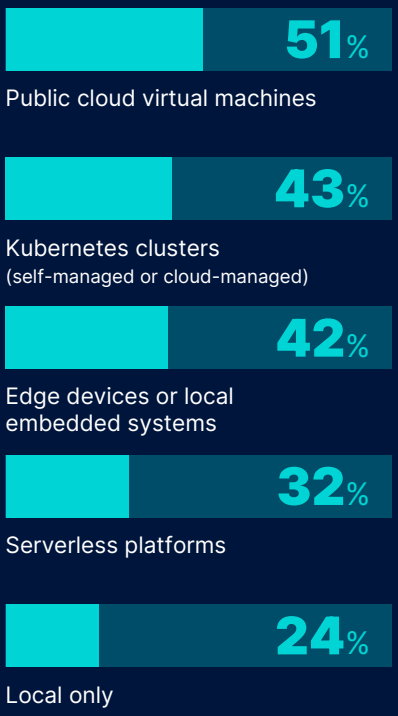


Complexity Manifests in Several Dimensions Continued

2. **Hybrid and multi-cloud deployments:** To maintain flexibility, 79% of respondents now operate agents across two or more environments–51% in public clouds, 40% on-premises, and 32% on serverless platforms. This approach enables greater control over performance, privacy, and compliance but also multiplies orchestration and governance demands, adding to the overall complexity of agent operations.
3. **Orchestration and workflow management:** Coordinating multiple models, tools, and frameworks is consistently identified as one of the hardest aspects of building agents. Ensuring reliability across heterogeneous systems requires new orchestration patterns, observability layers, and runtime policies.
4. **Lack of standardization:** With no universal framework for packaging or sharing agents, teams are forced to create custom processes, increasing maintenance costs and slowing deployment.

Coordinating across this expanding ecosystem is already stretching existing workflows beyond their limits. Nearly half of global respondents (48%) cite operational complexity in coordinating multiple components as their top challenge, while 43% point to increased security exposure stemming from orchestration sprawl. In certain markets, the burden is even heavier: 65% of organizations in India, 55% in Germany, and 53% in Singapore identify orchestration as the most acute pain point in their agent development pipeline. Yet orchestration is only one facet of a broader problem: integrating diverse models, t ools, frameworks, and deployment environments adds further layers of difficulty that test even the most mature cloud-native operations. But beneath that complexity lies an even more fundamental question: who governs it all?

Typical Development and Run Platforms



From Complexity to Governance: The Price of Freedom

As organizations diversify across models, tools, and deployment environments to gain flexibility, that same complexity is giving rise to a **new kind of challenge: governance.** Each additional model, orchestration tool, or cloud platform adds autonomy but also enlarges the surface area.

The “price of freedom” is greater coordination overhead and heightened security exposure, making governance not just a policy question but an architectural one.

Leading teams are closing this gap by building governance and interoperability directly into their architectures, embedding standardized orchestration policies, controlled runtimes, and secure-by-default toolchains. These teams are redefining “enterprise readiness,” not as a certification step at the end of deployment, but as an architectural principle from the start.

Lock-In Fears Are Real: Portability as the Foundation of Resilient Agent Architectures

If orchestration is the technical bottleneck for agentic AI, vendor lock-in is the strategic one. Even as organizations invest heavily in agents, many are sounding the alarm about the fragility of their supply chains. Seventy-six percent of global respondents report active concerns about vendor lock-in—rising to 88% in France, 83% in Japan, and 82% in the UK. And these aren't theoretical anxieties. They center on the very layers of the stack that power agentic systems today.

These concerns center on the layers where inference meets infrastructure. Model hosting platforms and LLM providers (42% each) top the list of lock-in risks, closely followed by cloud providers (41%), data storage and retrieval systems (39%), and monitoring and evaluation layers (38%). Enterprises fear that today's rapid adoption could translate into long-term dependency, limiting flexibility and innovation down the road.

To mitigate that risk, organizations are diversifying rather than consolidating. They are spreading workloads across multiple models, tools, and cloud environments. Among the 61% of organizations that use both cloud-hosted and locally hosted models, the leading drivers are control (64%), data privacy (60%), and compliance (54%), with cost being far less influential (41%).

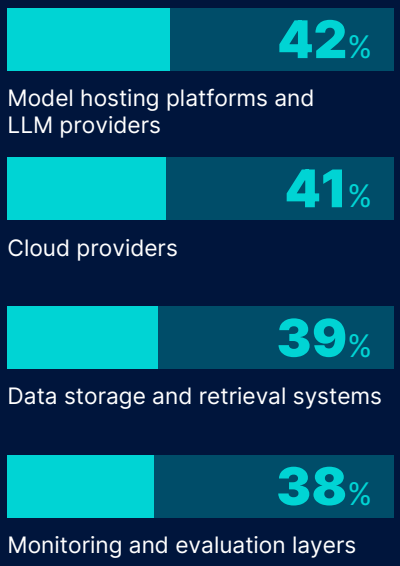
This strategy comes with trade-offs. Each additional platform, model, or runtime adds coordination overhead and security exposure, creating what can be described as "dependency management by distribution." Still, the consensus is clear: a multi-model, multi-cloud approach remains the most practical path to long-term flexibility and control.

Deployment patterns reflect this mindset. Over half (51%) of organizations run agents in the public cloud, 40% on-premises, 32% on serverless platforms, and 24% locally. In total, 79% operate across two or more environments, with common pairings such as Kubernetes clusters with public cloud VMs (48%) or hybrid on-prem/cloud setups (35%).

Amid this diversity, containers provide the connective tissue–a consistent, portable layer that enables agents to move securely between environments. They remain one of the few technologies capable of mitigating lock-in risks while maintaining governance, reproducibility, and scale.

In short, the **agentic future will not be monolithic.** It will be multi-cloud, multi-model, and multi-environment, making open standards and portable infrastructure essential to sustaining enterprise trust and flexibility.

Top Lock-In Concerns Across the Agentic AI Stack



76% OF GLOBAL RESPONDENTS

Report active concerns about vendor lock-in

Agents as the New Microservices: Containers as the Foundation of Agentic Infrastructure

As organizations navigate lock-in risks and increasing complexity, one pattern stands out: containers have become the foundational unit of agentic infrastructure. Their role is not merely theoretical or aspirational, rather it is deeply operational.

94%

Nearly all organizations surveyed (94%) already use containers in their agent development or production workflows, and the remainder plan to adopt them.

As organizations ramp up agent adoption, they are extending the same cloud-native workflows that already power their application pipelines–like microservices CI/CD, and container orchestration–to support these new workflows. In fact, ninety-four percent of all teams building agents rely on containers. This approach has real advantages: it leverages familiar tooling, pipelines, and operational patterns, accelerating time-to-market and reducing overhead. Containers provide built-in portability, version control, and environmental consistency, while cloud-native architectures offer ecosystem compatibility and cost control. In this way, agent development is evolving as a natural extension of cloud-native maturity, not a departure from it.

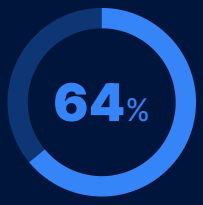
What's new is how containers are being adapted for agentic workloads. In fact, 98% or organizations report that they largely or mostly use the same development and deployment workflows for agents as they do for traditional cloud-native applications. This continuity underscores how containerization is not being reinvented for agentic AI, but extended. Teams are leveraging the same CI/CD pipelines, orchestration layers, and runtime standards that power their microservices to now support agentic workflows.

Beyond portability and rollback, teams now rely on containerization to provide sandboxed execution, version control, and predictable environments for probabilistic or autonomous behavior. These capabilities make containers the natural substrate for scaling agents securely and repeatedly.

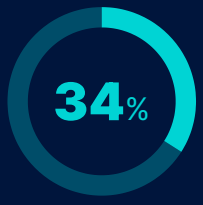
Container infrastructure is no longer just about uniform deployment across clouds. The future of agentic AI will not be a return to monoliths or black-box platforms. It will be built on the same foundations that transformed enterprise software a decade ago, and containers are once again at the heart of that transformation.

Agents and Cloud Native Software Overlap

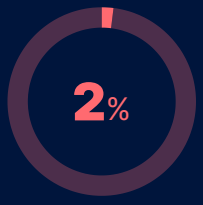
98% of organizations report that they largely or mostly use the same workflow for agents as they do for traditional cloud-native apps



YES
Our workflows remain largely the same



MOSTLY
Added some AI-specific tooling or processes



NO
Significantly changed how we build and deploy

Conclusion: Preparing for Multi-Model, Multi-Cloud Agent Infrastructure

What's stalling broader impact isn't a lack of interest or use; it's trust, complexity, and uneven pathways to scale. Security remains the dominant barrier, with organizations struggling to ensure tools are enterprise-ready, implement proper access controls, and maintain secure isolation between agents and systems. Orchestration and integration are the "silent killers," making promising pilots fragile as teams connect multiple models, environments, and clouds.

At the same time, architectural patterns are converging. Most teams are extending familiar cloud-native practices to agents: containers are foundational, hybrid and multi-cloud are normal, and many organizations blend local and hosted models for control and compliance. This portability is strategic; three in four teams worry about model and cloud lock-in, yet the same diversification introduces new coordination and governance burdens.

The path forward doesn't require reinvention so much as consolidation around a trust layer: access to trusted content and components that can be safely discovered and reused; secure-by-default runtimes; standardized orchestration and policy; and portable, auditable packaging. To earn enterprise confidence, this trusted layer should be built on the same core principles that made containers foundational: isolation, flexibility, interoperability, simplicity, and portability.

Agentic AI's near-term value is already real in internal workflows; unlocking the next wave depends on standardizing how we secure, orchestrate, and ship agents. Teams that invest now in this trust layer, on top of the container foundations they already know, will be first to scale agents from local productivity to durable, enterprise-wide outcomes.

NEXT STEPS
FOR LEADING COMPANIES

Tame complexity with standard orchestration. Favor container-centric pipelines, and unified gateways that abstract multi-model, multi-tool, and multi-cloud sprawl.

Ensure access to trusted components and content. Use verified sources for models, MCP servers, and agents to reduce security exposure and build confidence in the ecosystem.

Adopt portable packaging for agents. Move toward a container-like signed, and inspectable artifacts to make sharing safe and repeatable.

Codify security as architecture, not a check-list. Treat sandboxing, credentials, and policy enforcement as first-class concerns across agent runtime and MCP tooling.

Diversify without drifting. Use multi-model and multi-cloud approaches for flexibility while centralizing governance, observability, and rollback paths.