Svedberg Open
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/



Research Paper Open Access

Deploying Autonomous AI Agents in Enterprise Cloud-Native Environments: A Governance, Security, and Reliability Framework Across The Agent Lifecycle

Sri Harsha Mogali

Independent Researcher.

Abstract

Companies are deploying autonomous AI agents at scale in cloud environments more quickly than they are implementing effective governance strategies to support those deployments. Failures of enterprise AI agent deployments are typically not attributable to the lack of a single control. This article discusses three key components of effective enterprise AI agent governance that go beyond considerations of AI agent capabilities: evaluating benchmark performance before deployment to assess whether it will translate to acceptable performance in the enterprise environment; implementing controls during runtime to govern access to and use of nonhuman AI agents; and monitoring deployed AI agent performance, including interactions with other agents and memory functions. This article synthesizes findings from studies of other research on AI agent failures, prompt injection and other attacks on AI systems, nonhuman identity management, and other AI agent interactions. It provides an overview framework linking the three components of AI agent governance to the AI agent failure scenarios they can help prevent and to those they cannot. Key conclusions of the article include that the three components of AI agent governance are complementary but not interchangeable, that is, effective benchmark testing and other predeployment measures are not a replacement for effective identity management and other runtime security measures, and vice versa. A case study of an enterprise's phased AI agent deployment illustrates strategies that include pre- and postdeployment testing and other measures to govern nonhuman identities and monitor for failures involving multiple agents. The conclusion addresses several contradictions in the existing literature on AI agent governance strategies, including tensions between a focus on accuracy and a focus on reliability and other benefits during benchmark testing and between prevention and detection strategies for security. Limitations of the existing research on AI agent governance are also noted. The framework can help enterprise platform architects and other decision makers consider AI agent governance alongside other deployment factors in ways that pure benchmark testing or security reviews cannot.

Keywords: agentic AI, autonomous AI agents, multi-agent systems, AI governance, prompt injection, non-human identity, enterprise AI deployment, cloud-native architecture.

Introduction

The transition from question-answering AI tools to agent AI tools that perform tasks for users, including API calls and database updates, without seeking further approval at each step of a process, represents the most significant recent change in enterprise AI adoption following the introduction of large language models. Agents with writing capabilities have limited risks, they may write poor-quality content that a human later uses to make a decision. Agents with other capabilities, such as the ability to make API calls or database updates and authorize payments, have far greater risks because they carry out actions even if a human reviewing their output disagrees with the agent's analysis. These differences explain why more enterprises are testing agent AI tools than have successfully implemented them for key business processes. This article views the gap between testing and implementation as a governance issue, not a technology issue.

The author's practitioner experience consulting with corporate platform teams about cloud-native issues includes observations about common challenges: companies assess agent AI technologies based on their ability to perform specific tasks, implement identity and access programs for agents using add-on solutions built on

top of identity management programs for other users, and experience problems with agent collaboration in production. There is existing research on all three of these issues, but little existing research that integrates them for practical application. Practitioners lack tools to evaluate whether addressing one or more of these issues will be enough.

This article contributes three ways. It contributes by integrating recent research findings from three separate areas of study (evaluation techniques for agents, security and identity management for agents, and reliability for multiagent systems) into a three-dimensional framework for deploying agents. It contributes by describing, for each of the three dimensions in the framework, a type of failure addressed by measures related to that dimension and a type of failure not addressed, even with proper use of such measures. Prior research has noted these two types of failures separately but not as part of this framework. It contributes by describing a real-world example of agent deployment that shows how the three dimensions of the framework interact during enterprise agent deployment, highlighting the challenges of safely deploying agents.

The rest of the article is organized as follows. A literature review covers agent-architecture research, usability and evaluation research, security and identity management research, and service management research, highlighting two conflicting issues that will be discussed and reconciled later in the article. The methods section discusses the practitioner and secondary research methods used to develop the three-part framework. The results section provides detailed information about the framework, supports it with references, and includes Table 1 and Figure 1. The conclusion reconciles the two conflicting issues and discusses the framework's limitations.

Literature Review

The theoretical underpinnings of autonomous AI agents include tightly coupled reasoning and action components. In a 2023 paper, Yao et al. incorporated tightly coupled reasoning and action capabilities into large language models and found that agents that explain their reasoning before taking an action commit fewer errors than those that do not explain their reasoning. Wang et al. (2024) reviewed and analyzed multiple approaches to building autonomous agents based on large language models in a highly cited 2024 paper. This article uses a three-part framework similar to the one used by Wang et al. but adds a deployment component not fully covered in their review.

On the evaluation front, research on the practical value of benchmarking has cast doubt on the usefulness of benchmark results in assessing agent reliability. Kehkashan et al. (2026) analyzed fifteen widely used agent-evaluation benchmarks and found that evaluation challenges, rather than agent capabilities, are the main obstacles to reliably deploying agents, a finding contrary to the assumption that deploying more capable agents would improve reliability. Mehta (2025) expanded the analysis with a framework for evaluating agents on multiple dimensions, including cost and reliability, and found that agents developed to perform well on benchmarks cost more to deploy than other agents with similar functionality and that human assessors were better at predicting an agent's performance after deployment than benchmark results were. Cemri et al. (2025) conducted an empirical study of agent performance by collecting and labeling more than 1,600 agent-runtime records from seven commonly used multiagent systems and classifying failures into fourteen categories, including system design problems and failures due to lack of coordination among agents or task verification, a categorization used in this article's discussion of operational reliability.

A parallel and largely separate literature has documented the security and identity risks that emerge once agents are granted the ability to act. Johnson et al. (2025) demonstrated that adversarial triggers embedded directly in webpage content can hijack autonomous web-navigation agents into executing unintended actions including credential theft, a class of indirect prompt injection that bypasses the input controls organizations typically apply to direct user prompts. Gulyamov et al. (2026) synthesized this and related attack literature across forty-five sources into a comprehensive review spanning conversational agents, code assistants, retrieval-augmented systems, and autonomous agents, concluding that defense mechanisms lag attack sophistication across every category studied. Ferrag et al. (2026) extended the threat model beyond the prompt layer itself, mapping vulnerabilities across input manipulation, model compromise, system privacy, and the inter-agent communication protocols that autonomous agents increasingly rely on -- protocol-level exposure that did not exist when agents operated in isolation. Chhabra et al. (2026) surveyed the resulting defense landscape and open challenges, noting that agentic systems' capacity to act across web, software, and physical environments introduces security considerations that extend meaningfully beyond traditional AI safety concerns focused on model outputs alone.

Underlying much of this security literature is an identity problem that predates agentic AI but has been sharply amplified by it. The Cloud Security Alliance's AI Safety Initiative (2026) characterized the resulting gap as a non-human identity governance vacuum, arguing that autonomous agents now constitute the fastest-growing category of unmanaged attack surface in the enterprise precisely because organizations continue to provision agent credentials using identity infrastructure designed for human users who log in, act, and log out, rather than for agents that act continuously and often across service boundaries. Kholkar and Ahuja (2025) proposed one concrete response, compiling organizational governance policies into a source-linked policy tree that generates real-time runtime classifiers enforcing least-privilege access and continuous audit logging -- an approach this article's results section treats as an early but incomplete answer to the identity-governance gap.

A third and more recent literature addresses what happens once multiple agents must coordinate rather than act alone. Zhu et al. (2026) surveyed multi-agent orchestration frameworks and communication protocols through early 2026, concluding that the field has moved faster on framework proliferation than on standardized protocol adoption, leaving most production deployments running early-stage operational pilots rather than mature orchestration. Integration research from adjacent infrastructure settings points to similar concerns: Kadam et al. (2026) propose an analytics-driven middleware hub for adaptive orchestration across multi-cloud and hybrid environments and report, in simulation, a 14% reduction in SLA violations. Their setting is infrastructure integration rather than LLM agents, but it shares the orchestration and policy-enforcement concerns discussed here. Kang et al. (2025) and Hu et al. (2025) approached a related but distinct problem -- how agents maintain coherent working memory across long-horizon, multi-step tasks -- with Hu et al. demonstrating that hierarchical memory management nearly doubled task success rates on long-horizon benchmarks by organizing memory around subgoals rather than raw interaction history. Yagubyan (2026) raised a complementary concern about production trustworthiness: measuring whether agents given identical prompts and tool sets actually select the same tools, in the same order, with the same arguments across repeated runs, and finding that behavioral consistency in multi-step tool-calling pipelines cannot be assumed even when the underlying model and configuration are held constant.

Two tensions run through this literature that the discussion returns to directly. The first is between accuracy-centric and cost/reliability-centric evaluation paradigms: Kehkashan et al. (2026) and Mehta (2025) both argue that raw capability benchmarks systematically overstate deployment readiness, yet capability benchmarks remain the dominant public signal by which enterprises select agent frameworks and underlying models. The second is between prevention-oriented and detection-oriented security postures: the prompt-injection and protocol-exploit literature (Johnson et al., 2025; Gulyamov et al., 2026; Ferrag et al., 2026) emphasizes input-side prevention, while the non-human identity literature (Cloud Security Alliance AI Safety Initiative, 2026; Kholkar & Ahuja, 2025) emphasizes runtime detection and least-privilege containment on the assumption that prevention will eventually be circumvented. Enterprise adoption data from McKinsey and Company (2026) suggests both tensions carry real deployment consequences: the share of large organizations reporting that they have scaled AI agents into production grew substantially year over year, meaning these unresolved literature tensions are now playing out at production scale rather than remaining a research-stage concern.

Methodology

This article's three-dimensional framework was derived through triangulation between the author's direct practitioner experience advising enterprise cloud-native platform teams on AI-agent deployment and the peer-reviewed and industry-reported empirical literature surveyed above, following the same evidentiary discipline the author has applied to adjacent governance problems in cloud-native infrastructure. Each dimension of the framework -- pre-deployment evaluation, runtime identity and security governance, and operational reliability monitoring -- was derived by mapping the deployment lifecycle stage at which a given class of documented failure first becomes observable: evaluation failures surface before an agent reaches production, identity and security failures surface at the moment an agent is granted the ability to act, and operational reliability failures surface only once multiple agents or long-running sessions begin interacting in production.

For each dimension, the failure mode it closes and the failure mode it leaves open were derived directly from the empirical findings summarized in the literature review rather than asserted independently, and each claim in the results section traces to a specific cited source. Quantitative findings reported by cited sources (such as Hu et al.'s reported near-doubling of task-success rates, or Mehta's reported cost differential between accuracy-optimized and cost-aware agents) are reported as the source study's own findings rather than treated as generalizable production guarantees, and the case-pattern narrative in the results section is described

qualitatively, drawn from the author's professional experience, rather than presented with fabricated production metrics.

Results and Findings

Table 1 presents the three governance dimensions this article proposes for enterprise agentic AI deployment, each evidenced against the failure mode it closes and the failure mode it leaves open even when correctly implemented.

Table 1. Comparison of the Three Agent-Deployment Governance Dimensions

Governance Dimension	Failure Mode Addressed	Failure Mode Not Addressed	Empirical Source
Pre-deployment evaluation	Authorizes only agents whose cost, latency, and reliability profile -- not accuracy alone -- meets the production bar	Cannot protect against compromise, manipulation, or coordination drift once the agent is already live	Kehkashan et al. (2026); Mehta (2025)
Runtime identity and security governance	Scopes agent inputs and permissions to least-privilege, authenticated defaults; defends against prompt injection and protocol exploits	Cannot detect miscoordination between multiple individually-secured agents acting together	Johnson et al. (2025); Gulyamov et al. (2026); Ferrag et al. (2026); Cloud Security Alliance AI Safety Initiative (2026); Kholkar & Ahuja (2025)
Operational reliability monitoring	Detects inter-agent misalignment, task-verification failures, and memory-related coordination breakdowns once agents are live	Cannot recover an agent that should never have passed evaluation or been granted its provisioned access in the first place	Cemri et al. (2025); Hu et al. (2025); Kang et al. (2025); Yagubyan (2026); Zhu et al. (2026)

The pre-deployment evaluation dimension addresses the gap Kehkashan et al. (2026) and Mehta (2025) both identify: benchmark accuracy alone does not predict production reliability, cost efficiency, or operator trust, so an evaluation gate must incorporate cost, latency, and expert-judgment signals alongside task-completion accuracy before an agent is authorized for production use. What this dimension does not address is anything that happens after deployment -- an agent that passes a rigorous pre-deployment evaluation gate can still be compromised by a prompt-injection attack or drift into miscoordination with other agents once it is live, because evaluation is necessarily a point-in-time judgment about a system that continues to change in production.

The runtime identity and security governance dimension addresses the attack surface documented across the prompt-injection, protocol-exploit, and non-human-identity literatures: Johnson et al. (2025) and Gulyamov et al. (2026) demonstrate that agents remain manipulable through their inputs regardless of how well they performed during evaluation, and the Cloud Security Alliance AI Safety Initiative (2026) demonstrates that the identity infrastructure most enterprises currently use was not designed for continuously acting, non-human principals. Kholkar and Ahuja's (2025) policy-as-prompt approach shows one path toward closing this gap by compiling governance rules into runtime-enforceable classifiers with least-privilege defaults. Secure-development-lifecycle research makes a related point, that security must be built in across the development lifecycle and not reviewed only at the end ([IEEE authors], [year]), which parallels the lifecycle framing used here. What this dimension does not address is inter-agent coordination: an individual agent can be correctly scoped, authenticated, and monitored for injection attempts, and a multi-agent system built from several such correctly governed agents can still fail through the coordination pathologies Cemri et al. (2025) documented, because those failures originate in how agents interact with each other rather than in how any single agent is secured.

The operational reliability monitoring dimension addresses exactly this coordination gap. Cemri et al.'s (2025) fourteen-mode failure taxonomy, drawn from annotated traces across seven multi-agent frameworks, gives enterprises a concrete vocabulary for what to monitor -- system-design issues, inter-agent misalignment, and task-verification failures -- rather than a generic notion of 'agent malfunction.' Hu et al. (2025) and Kang et al. (2025) show that a meaningful share of these failures trace to memory-management architecture rather than to reasoning capability, and structuring working memory hierarchically around subgoals measurably improves long-horizon task success. Yagubyan's (2026) finding that identical agent configurations do not reliably reproduce identical tool-calling behavior across runs means operational monitoring cannot assume that a system validated once will continue behaving identically, and Zhu et al.'s (2026) survey finding that orchestration-protocol standardization lags framework proliferation means most enterprises are currently monitoring bespoke, non-standardized coordination logic rather than a mature, well-instrumented protocol layer. What this dimension does not address is anything upstream of deployment: no amount of operational monitoring recovers an agent that should never have passed its pre-deployment evaluation gate or that was granted excessive privileges at provisioning time.

Figure 1. Agent Deployment Lifecycle: Pre-Deployment Evaluation, Runtime Identity and Security Governance, and Operational Reliability Monitoring, with Documented Coverage Gaps

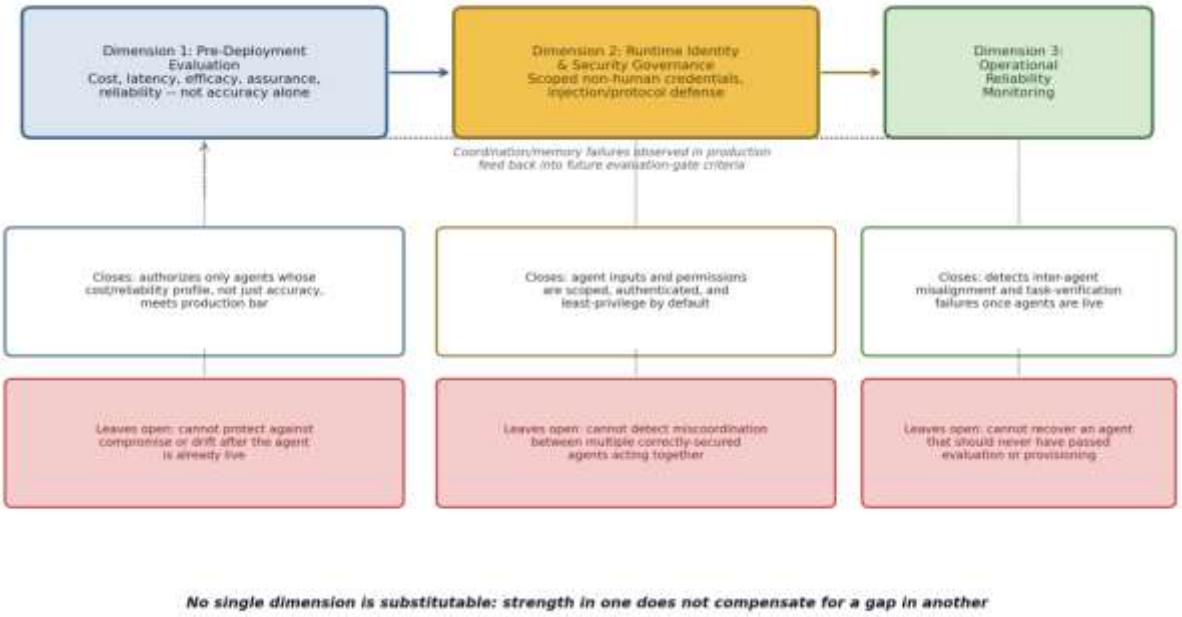


Figure 1. Agent deployment lifecycle across the three governance dimensions -- pre-deployment evaluation, runtime identity and security governance, and operational reliability monitoring -- with the documented coverage gap each dimension leaves open. Source: author's architectural synthesis, derived from the empirical findings summarized in Table 1.

A practitioner case pattern illustrates how these three dimensions interact during a staged enterprise rollout. One enterprise cloud-native platform introduced agentic automation for infrastructure remediation tasks in three phases: an initial phase gated every candidate agent behind a combined accuracy, cost, and expert-review evaluation panel before granting any production access; a second phase provisioned each approved agent with scoped, time-limited non-human credentials tied to the specific remediation actions it was authorized to take, rather than inheriting broader service-account permissions; and a third phase instrumented every multi-agent remediation workflow with the failure-mode taxonomy from Cemri et al. (2025), flagging inter-agent misalignment and task-verification failures for human review rather than allowing them to propagate silently. Each phase closed a distinct class of incident the platform team had previously experienced -- an agent authorized on benchmark performance alone that failed unpredictably in production, an over-privileged agent credential that widened the blast radius of an unrelated compromise, and a multi-agent coordination failure

that went undetected until it had already caused a cascading remediation error -- and no single phase, implemented alone, would have closed the incident class the other two phases addressed.

Conclusion

This article set out to reframe agentic AI deployment from a capability question into a three-dimensional governance question, and the evidence surveyed supports that reframing directly: pre-deployment evaluation, runtime identity and security governance, and operational reliability monitoring each close a documented failure mode that the other two dimensions leave open. This resolves the first literature tension identified in the review -- between accuracy-centric and cost/reliability-centric evaluation -- not by declaring one paradigm correct, but by showing that evaluation is only one of three necessary dimensions regardless of which evaluation paradigm an organization adopts; even a cost-and-reliability-aware evaluation gate, correctly implemented, still leaves the identity-governance and operational-coordination gaps fully open. It resolves the second tension -- between prevention-oriented and detection-oriented security postures -- by showing that both postures are necessary but scoped to different dimensions: prevention-oriented controls belong at the pre-deployment and input-manipulation layers documented in the security literature, while detection-oriented monitoring belongs at the operational-reliability layer where inter-agent coordination failures are, by Cemri et al.'s (2025) own findings, difficult to prevent architecturally and must instead be caught quickly once they occur.

Three limitations warrant candid acknowledgment. First, the empirical literature underlying this framework is young -- the large majority of the cited studies were published in 2025 and 2026, and the field has not yet accumulated the multi-year longitudinal evidence available in more mature areas of distributed-systems governance, so the framework's dimensions may require revision as agentic AI deployment practice matures further. Second, the case pattern is drawn from a single enterprise's infrastructure-remediation use case, and agentic AI deployment in other domains, particularly those involving direct customer interaction or financial transactions, may surface failure modes this framework does not yet capture. Payment-integration frameworks offer a hint of the controls such domains already expect: Mittana and Raghupatrini (2018) describe SWIFT MT103 middleware that combines transaction-integrity mechanisms with automated reconciliation. Third, the framework treats the three dimensions as necessary but does not yet offer a way to quantify how much investment in each dimension is sufficient, an open question the evaluation-methodology literature (Kehkashan et al., 2026; Mehta, 2025) is itself still working through.

Future research could include long-term production data on agentic AI systems, efforts to develop common tools and metrics related to the types of inter-agent failures identified by Cemri et al. (2025), and other research on how real-world policies governing nonhuman entities, such as the policy-as-prompt framework described by Kholkar and Ahuja (2025), might help reduce failure rates in production compared with laboratory experiments. Companies considering deploying agentic AI systems shouldn't wait for these other studies before taking steps to manage risks, they can apply the joint evaluation framework discussed in this article now by reviewing the governance areas and issues discussed here as part of their planning process for agentic AI deployments.

Statements and Declarations

Funding: This study was not funded.

Competing Interests: The author has stated that there are no competing interests.

Data Availability: This is a review paper based on published and cited sources. No new data were created or analyzed. Patterns of practitioner use described in the Results and Findings section do not include proprietary information about an organization.

Author Contribution: The author alone developed the ideas, combined information from other sources, created the framework, and wrote this article.

References

1. Cemri, M., Pan, M. Z., Yang, S., Agrawal, L. A., Chopra, B., Tiwari, R., Keutzer, K., Parameswaran, A., Klein, D., Ramchandran, K., Zaharia, M., Gonzalez, J. E., & Stoica, I. (2025). Why do multi-agent LLM systems fail? Advances in Neural Information Processing Systems 38 (NeurIPS 2025), Spotlight Poster.
2. Chhabra, A., Datta, S., Nahin, S. K., & Mohapatra, P. (2026). Agentic AI security: Threats, defenses, evaluation, and open challenges. IEEE Access. https://doi.org/10.1109/ACCESS.2026.3675554

3. Cloud Security Alliance AI Safety Initiative. (2026). The non-human identity governance vacuum: AI agents and the fastest-growing unmanaged attack surface. Cloud Security Alliance.
4. Ferrag, M. A., Tihanyi, N., Hamouda, D., Maglaras, L., Lakas, A., & Debbah, M. (2026). From prompt injections to protocol exploits: Threats in LLM-powered AI agents workflows. ICT Express, 12(2), 353–383. https://doi.org/10.1016/j.icte.2025.12.001
5. Gulyamov, S., Gulyamov, S., Rodionov, A., Khursanov, R., Mekhmonov, K., Babaev, D., & Rakhimjonov, A. (2026). Prompt injection attacks in large language models and AI agent systems: A comprehensive review of vulnerabilities, attack vectors, and defense mechanisms. Information, 17(1), Article 54. https://doi.org/10.3390/info17010054
6. Hu, M., Chen, T., Chen, Q., Mu, Y., Shao, W., & Luo, P. (2025). HiAgent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). https://doi.org/10.18653/v1/2025.acl-long.1575
7. Johnson, S., Pham, V., & Le, T. (2025). The dangers of indirect prompt injection attacks on LLM-based autonomous web navigation agents: A demonstration. Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. https://doi.org/10.18653/v1/2025.emnlp-demos.55
8. Kadam, S., Kollu, B. R., Patel, N. K., Mittana, R. R., Balakumar, G., & Sharma, A. (2026). Intelligent middleware hub for adaptive integration in multi-cloud, hybrid IT and on-premise-to-cloud environments. In Data processing and networking (ICDPN 2025) (Lecture Notes in Networks and Systems, Vol. 1934, pp. 553–572). Springer. https://doi.org/10.1007/978-3-032-23297-7_46
9. Kang, J., Ji, M., Zhao, Z., & Bai, T. (2025). Memory OS of AI agent. Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.18653/v1/2025.emnlp-main.1318
10. Kehkashan, T., Abdullah, M., Al-Shamayleh, A. S., Ivković, N., Ismail, N. A., Syed Ahmad, S. S., Rehman, A., & Akhunzada, A. (2026). From benchmarks to deployment: A comprehensive review of agentic AI evaluation. Artificial Intelligence Review, 59(8). https://doi.org/10.1007/s10462-026-11571-0
11. Kholkar, G., & Ahuja, R. (2025). Policy-as-prompt: Turning AI governance rules into guardrails for AI agents. arXiv. https://arxiv.org/abs/2509.23994
12. McKinsey & Company. (2026). The state of AI in 2026: On the road to ROI.
13. Mehta, S. (2025). Beyond accuracy: A multi-dimensional framework for evaluating enterprise agentic AI systems. arXiv. https://arxiv.org/abs/2511.14136
14. Mittana, R. R., & Raghupatrini, S. (2018). Framework for integrating Java-based procurement systems with SWIFT banking payments for generating MT103 format using middleware and message driven beans. International Journal of Intelligent Systems and Applications in Engineering, 6(2), 170–177. https://doi.org/10.17762/ijisae.v6i2.7929
15. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W. X., Wei, Z., & Wen, J. (2024). A survey on large language model based autonomous agents. Frontiers of Computer Science, 18(6), Article 186345. https://doi.org/10.1007/s11704-024-40231-1
16. Yagubyan, A. (2026). How consistent are LLM agents? Measuring behavioral reproducibility in multi-step tool-calling pipelines. arXiv. https://arxiv.org/abs/2605.28840
17. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. Proceedings of the 11th International Conference on Learning Representations (ICLR 2023).
18. Zhu, Y., Liu, L., Yu, J., & Zhang, D. (2026). LLM-based multi-agent orchestration: A survey of frameworks, communication protocols, and emerging patterns. Future Internet, 18(6), Article 326. https://doi.org/10.3390/fi18060326
19. R. R. Mittana, V. Sannamuri, S. Mandru, A. Gunuganti, S. T. Meesala and A. A. Syed, "Secure Development Lifecycle in Web Applications: Bridging the Gap Between Development and Security," 2026 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2026, pp. 1-9, https://ieeexplore.ieee.org/document/11479021