Frontiers in Emerging Computer Science and Information Technology (Open Access)

Volume 01, Issue 02, September 2024,

PageNo.75-85

Engineering Autonomous Multi-Agent Software Systems: Implementing Hybrid Architectures, Interaction Protocols, and Execution Loops

Mykhailo Nykoliuk
Full-Stack Engineer
Kyiv, Ukraine

RECEIVED - 11-07-2024, **RECEIVED REVISED VERSION** - 13-08-2024, **ACCEPTED**- 22-08-2024, **PUBLISHED**- 06-09-2024

Abstract

The paper examines the engineering transition from static software paradigms to autonomous agentic architectures (Software Engineering 3.0). Instead of focusing on organizational theory, the analysis concentrates on the technical implementation of functional subject attributes in software agents: goal planning, causal reasoning, and standardized execution. Key architectural patterns for deploying distributed multi-agent systems (MAS) are synthesized, specifically focusing on the integration of generative models with Case-Based Reasoning for strictly typed decision reproducibility. The study details the technical requirements for Agent-to-Agent (A2A) communication protocols and the establishment of stable behavioral contracts between autonomous entities. Implementation challenges are addressed through the lens of data engineering, specifically context memory management and vector data integration within existing IT landscapes. Furthermore, the paper structures the technical aspects of Artificial Intelligence Trust, Risk, and Security Management (AI TRiSM), defining methods for behavioral control, immutable logging, and decision traceability in high-load environments. The efficiency analysis is reframed from general business ROI to specific engineering metrics, correlating system performance with the costs of development, integration, and computational maintenance.

Keywords: *Software Engineering 3.0, Multi-agent systems implementation, Agent-to-Agent protocols, Hybrid software architectures, AI TRiSM, System integration, Autonomous execution loops.*

Introduction

The contemporary corporate landscape is undergoing a technological rupture comparable in scale to the Industrial Revolution [1]. Whereas the preceding decades were dominated by data digitization and process automation grounded in rigidly formalized algorithms, the 2023–2024 period registers a transition to the logic of autonomous action. Generative artificial intelligence, which became broadly accessible in 2023, established a new level of performance in the synthesis of text, code, and media formats; functionally, however, it remained predominantly an auxiliary mechanism, that is, an instrument for augmenting human productivity [1]. By 2024, the central object of attention becomes agentic artificial intelligence: systems capable of perceiving context, performing reasoning, and acting independently in dynamic environments.

The shift from models oriented toward content generation to systems that realize autonomous goal attainment inevitably requires a reassembly of business operations at the level of principles and governance loops. Agentic solutions differ fundamentally in that they translate abstract business intentions into sequences of executable steps without the need for continuous manual oversight. On this basis, a class of multi-agent systems is taking shape, in which several specialized agents, assigned to distinct roles (for example, researcher, developer, reviewer), coordinate actions in hierarchical or distributed configurations and jointly solve tasks that exceed the capabilities of a single model.

The relevance of the study is determined by the gap between high awareness and the practical maturity of deployment: although 88% of organizations regularly use

artificial intelligence, a substantial share of companies remains in a piloting loop without moving to stable scaling [2]. According to 2024 data, only about 23% of organizations have begun expanding agentic systems, and in certain functional domains the scaling rate rarely exceeds 10% [2]. The key difficulty is associated not so much with limitations of computing resources as with the complexity of embedding autonomous agents into existing business processes, with the necessity of ensuring trust and controllability through Artificial Intelligence Trust, Risk, and Security Management platforms, and with the development of new interface forms of interaction, including the agentic user interface [3, 8].

The purpose of this paper is to analyze how the transition from static generative artificial intelligence tools to autonomous multi-agent systems and hybrid generative architectures transforms corporate operating models, process architecture, and governance loops.

The author's hypothesis is grounded in the assumption that the scalable value of agentic solutions arises from closing a coherent perception–memory–planning/action–execution cycle, from standardizing interactions through Agent-to-Agent and agentic user interface mechanisms, and from governing trust and risks within the logic of Artificial Intelligence Trust, Risk, and Security Management, rather than from the sheer power of large language models as such.

The scientific novelty of the study lies in the fact that, within the framework of this paper, an integral description framework for enterprise large language model multi-agent systems is proposed for the first time as a managed operating loop with measures against uncertainty cascading and an applied return on investment model that accounts for costs of development, integration, maintenance, and governance control.

Materials and Methods

The methodological foundation of the study is constructed as a comprehensive systems analysis that integrates qualitative and quantitative approaches to assessing the dynamics of technological development. The empirical and theoretical base was developed using several types of sources, ensuring representative coverage of both scholarly discourse and practices of industrial deployment.

The corpus of academic materials includes recent peer-reviewed publications selected from IEEE Xplore, the ACM Digital Library, and Springer, with a focus on architectures of large language model based multi-agent systems, the problem field of trusted artificial intelligence and interpretability, and explainable artificial intelligence, as well as instruments for the formal verification of autonomous agents [10]. The scholarly findings were compared with industry reports and forward-looking reviews presented in Gartner analytics, including Hype Cycle for Artificial Intelligence and Top Strategic Technology Trends, in materials from the McKinsey Global Survey 2024, and in specialized market studies prepared by International Business Machines and Salesforce [2]. The technological component of the review was supplemented by an analysis of technical documentation and protocol specifications, including the open interaction standards Google Agent-to-Agent, Agent-User Interaction, and the Vercel Artificial Intelligence Software Development Kit, as well as an examination of Supabase architectural solutions for working with vector data and contextual memory [15]. To validate conclusions related to practical effectiveness, empirical data from development repositories were used: the GitHub Octoverse report was employed, as well as the AIDev dataset containing approximately one million agent-generated pull requests, which made it possible to assess the actual productivity of autonomous coding agents using real engineering artifacts [4].

The research process was structured in three stages. At the first stage, data collection, normalization, and clustering were carried out across key thematic areas: multi-agent systems, hybrid architectures, security, and indicators of economic effectiveness, return on investment. At the second stage, results were synthesized to identify causal relationships between the adoption of specific technological patterns and changes in business metrics, including parameters of operational speed, process execution quality, and governance resilience. At the third stage, conclusions and applied recommendations for the corporate sector were formulated, oriented toward overcoming scaling barriers and reducing the risks of operating autonomous systems. A priority remained the alignment of theoretical models of autonomy with empirical deployment effects in practical domains, including finance, healthcare, and logistics [6].

Results and Discussion

Multi-agent systems in the corporate context are regarded as a current apex of artificial intelligence evolution, reflecting a shift in research and applied focus from Internet artificial intelligence to embodied artificial intelligence and to practices of distributed intelligence [24, 25]. Whereas in classical multi-agent systems the logic of agent behavior was predominantly specified by deterministic rules and pre-specified scenarios, contemporary configurations based on large language model based multi-agent systems use natural language as a universal operational layer through which coordination, planning, meaning alignment, and knowledge exchange are realized [12, 13].

The architectural profile of such systems is defined by distributed computation, the locality of each agent's perception, and a fundamentally decentralized character of control, which reduces dependence on a single governing component and increases adaptability in dynamic environments. At the same time, the classical properties of intelligent agents, established in the works of Wooldridge and Jennings, by 2024 have acquired a more applied and technologically saturated content. Autonomy is interpreted as an agent's capacity to maintain control over its own internal state, memory, and action selection without constant direct human intervention, which is especially significant when performing multi-step tasks under conditions of incomplete certainty. Social ability in contemporary large language model based multi-agent systems is realized not only through formalized communication languages and standards, including Agent Communication Language and the Foundation for Intelligent Physical Agents, but also through natural language, which functions as an interface for aligning intentions and distributing roles among agents [13]. Reactivity and proactivity cease to be symmetric characteristics of response and initiative: the former is fixed as a mechanism of continuous calibration of behavior based on environmental signals, whereas the latter is fixed as the capacity to construct sequences of actions oriented toward achieving stable long-term goals, including dependency management and the prevention of deviations from the target trajectory [13, 14].

The growth of business interest is confirmed by measurable indicators: the number of Gartner inquiries related to multi-agent systems increased by 1445% in the interval from 2024 [9]. The forecasting framework also points to structural changes in the composition of agent ensembles: by 2027, it is expected that approximately 70% of multi-agent systems will be formed from narrowly specialized agents, which can potentially increase accuracy through expert specialization but simultaneously intensifies coordination complexity, interaction protocol requirements, and the load on orchestration mechanisms in a distributed environment [9].

Within Table 1 presented below, existing architectural principles as well as coordination mechanisms will be described.

Table 1. Architectural principles and coordination mechanisms (compiled by the author based on [9]).

MAS Component	Function Within the System	Technological Implementation
Agent Profile	Specification of the agent's role, parameters, and defining characteristics	Prompt engineering, domain-specific adaptation
Perception	Acquisition and aggregation of information from external sources	Application programming interfaces, data parsing, sensor-based systems
Self-action	Planning, deliberation, and inferential reasoning	Chains of thought (CoT), ReAct mechanisms

Mutual interaction	Communication, coordination, and information exchange among agents	Agent-to-agent protocols (A2A), shared data buses
Memory management	Preservation and organization of contextual information and knowledge	Vector databases, Zettelkasten method (A-Mem)

The presented decomposition of multi-agent systems by components (profile, perception, self-action, mutual interaction, memory management) demonstrates that the practical robustness of large language model based multi-agent systems is determined not by the model's isolated intelligence, but by the coherence of the perception–memory–planning–execution loop and by the quality of inter-agent communications. At the level of operating architecture, this means that errors and uncertainty arise not only within the large language model, but also at the boundaries: during context retrieval, message transmission, role alignment, and the fixation of artifacts in memory. To illustrate how multi-agent system components, close into a manageable enterprise loop with control and verification points, a generalized schematic of large language model based multi-agent system architecture in an enterprise environment is provided below.

Figure 1 captures a typical reference architecture of a large language model based multi-agent system as an operating loop in which the large language model functions as a reasoning core, while enterprise systems and protocols serve as the environment of action and constraint.

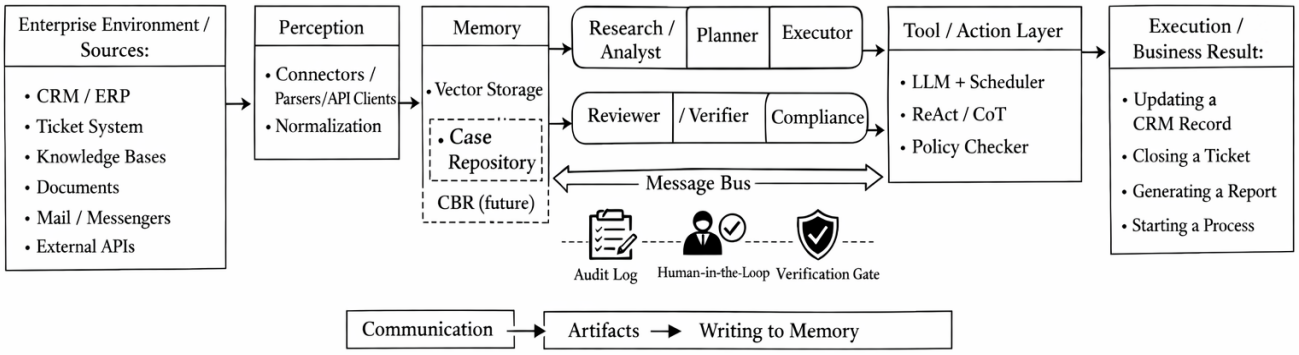

Fig. 1. Reference architecture of a large language model based multi-agent system in a corporate environment: The Perception–Memory–Planning/Action–Execution loop and inter-agent coordination (compiled by the author based on [9, 11]).

Despite high adaptability, large language model based multi-agent systems are subject to a risk that may be characterized as uncertainty cascading: the probabilistic nature of large language models creates conditions under which a local error by an individual agent can be transmitted through the interaction network and accumulate, forming a collective deviation from the correct solution [11]. In academic discourse, this effect is captured by the concepts of knowledge drift and misinformed perspective propagation, emphasizing that distortions arise not only at the level of single inferences but also in the logic by which interpretations spread among agents. As countermeasures, scholars propose the use of rigorous probabilistic frameworks for the quantitative assessment of uncertainty in inter-agent communications, as well as the introduction of formal verification mechanisms aimed at constraining the spread of erroneous claims and increasing the reliability of resulting reasoning trajectories [11, 16].

A radical restructuring of operating models in the corporate environment logically leads to demand for hybrid generative architectures that combine the stability of structured repositories and registries with the productive potential of generative models. In this context,

the synergy of Case-Based Reasoning and generative artificial intelligence is considered particularly promising, as it makes it possible to unite evidential grounding, reliance on precedents, and an expanded capacity for semantic generalization [26]. Classical Case-Based Reasoning systems are built around a repository of validated cases, which increases the reproducibility of decisions but often constrains variability and sensitivity to context. Generative artificial intelligence, by contrast, enables the extraction of tacit knowledge from unstructured materials, including interviews, reports, and video recordings, with subsequent transformation into formalized representations, including ontologies suitable for enriching and updating Case-Based Reasoning bases [26]. In this way, generative components perform the function of a connective layer between human expertise and explainable artificial intelligence systems, translating fragmented knowledge into manageable and verifiable structures [20, 26].

The practical relevance of hybrid architectures is evident in industry scenarios where simultaneous compliance with regulatory constraints, decision controllability, and high-speed processing of complex cases are required. In the banking sector, such models are applied to optimize credit processes: the integration of enterprise architecture and artificial intelligence is used both to model scenarios of how changes in credit policy affect return on investment within the strategic loop and to deploy scalable intelligent applications at the applied level through modular microservices and containerization, including Docker and Kubernetes, which support resilient client interaction [21, 22].

At the level of software development practices, 2024 is described as a transition from an artificial intelligence assisted model, Software Engineering 2.0, to agentic engineering, Software Engineering 3.0, in which agents cease to be limited to generating isolated code fragments and begin to be treated as task executors oriented toward goal attainment and spanning the full cycle of engineering activity [4]. Within the concept of Structured Agentic Software Engineering, a shift is recorded from exclusively human teams to hybrid Human–Agent configurations, in which the human function is interpreted as mentorship and managerial calibration of agent behavior, the Agent Coach role, and a reorientation from improvisational prompt engineering, vibe coding, to repeatable and formalized engineering activities [4]. An important element becomes the replacement of informal assignments with machine-readable artifacts, including BriefingScripts and Consultation Request Packs, which increases requirement traceability and result reproducibility. Complementing this logic is the development of specialized work environments, such as the Agent Command Environment, which makes it possible to orchestrate coordinated work of multiple agents within a single governance loop [4, 5].

Technological choice in agentic ecosystems also demonstrates pronounced shifts: TypeScript in 2024 became the dominant language on GitHub, surpassing Python and JavaScript, which correlates with demand for determinism and security enabled by strict typing when agents generate and modify code [19]. It is noted that 60–70% of startups in the agentic artificial intelligence domain choose TypeScript to build the execution layer, whereas Python retains priority predominantly in the model training loop.20 Ecosystems based on Next.js and the Vercel Artificial Intelligence Software Development Kit support the creation of interface solutions that permit real-time updates and coupling with agentic behavior, including the use of protocols of the Agent-User Interaction class [17].

Against the background of accelerating technological progress, an asymmetry between experimentation and scaling persists: McKinsey data record regular use of artificial intelligence in 88% of companies, while only 33% of organizations expand deployment programs to the enterprise-wide level [2]. The most pronounced gap appears in the distribution of outcomes: approximately 6% of organizations fall into the high performer’s category, achieving artificial intelligence contribution to earnings before interest and taxes at a level greater than 5% [2]. These companies are characterized by concentration of effort not on algorithms as such but on transforming people, processes, and organizational culture, which account for approximately 70% of the focus, whereas the algorithmic component itself accounts for about 10%, underscoring the primacy of managerial and process restructuring relative to model selection [32].

Table 2 describes features of business transformation as well as performance indicators, return on investment.

Table 2. Business transformation and performance indicators, return on investment (compiled by the author based on [27, 33, 34]).

Industry	Primary Use Case	Implementation Outcome
Technical support	Integration of a ticketing system with an artificial intelligence chatbot	Reduction of incident resolution time by 74.17% (from 120 to 31 minutes)
Recruiting	Automation of candidate sourcing and selection	Reduction of the hiring cycle by 40%, cost reduction by 30%
Logistics	Predictive inventory management	Increase in inventory accuracy to 95%, savings of 5 million USD per year
Aviation	Optimization of fuel consumption	Annual savings of up to 1 billion USD for large fleets
Cybersecurity	Multi-agent fraud detection	Increase in accuracy from 87% to 96%, reduction of false-positive triggers by 65%

The sectoral effect data in Table 2 demonstrate that the operational benefit of agentic systems is expressed primarily in metrics of speed, quality, including accuracy and the reduction of false positives, and cost, including savings and the reallocation of labor effort. However, comparison of cases shows that the maximum effect is achieved not through a point deployment of a model but through restructuring the full work execution loop: from task formulation and data access to risk control and the institutionalization of results within processes. Consequently, the interpretation of return on investment for agentic solutions must take into account the structure of costs, including integration, maintenance, control, and security, and the structure of effects, including acceleration, error reduction, throughput growth, and compliance. For a clear comparison of value drivers and cost factors, a conceptual return on investment model for agentic deployments is presented below [31, 36].

Figure 2 summarizes the two-sided nature of the effect: business value emerges at the level of processes and people, whereas the key costs are concentrated in data, integration, and the governance loop, Artificial Intelligence Trust, Risk, and Security Management.

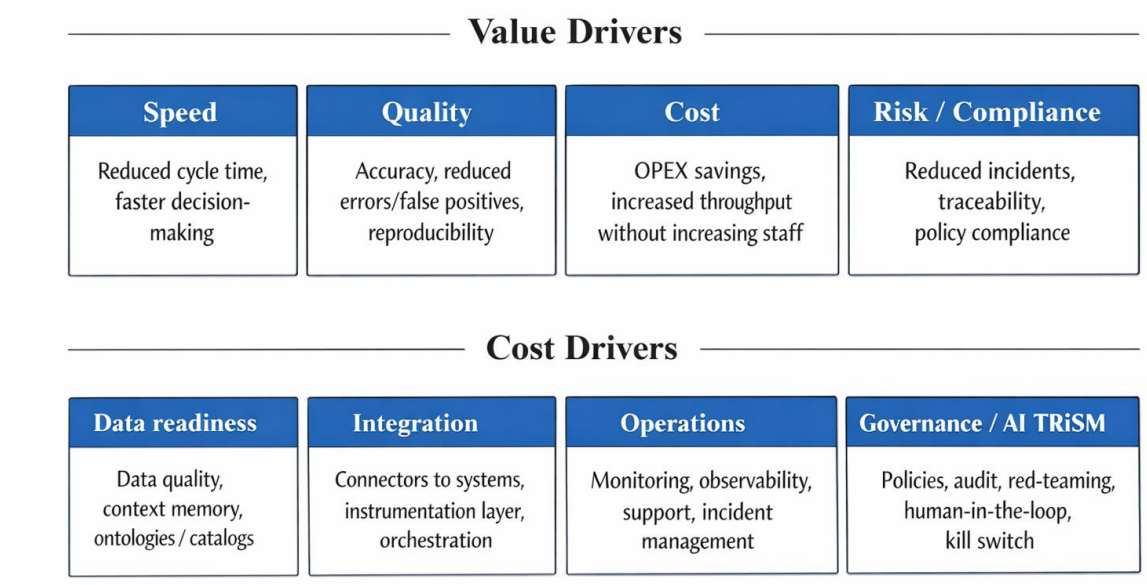

Fig. 2. Conceptual return on investment model for agentic systems: value drivers, cost structure, and control loops, Artificial Intelligence Trust, Risk, and Security Management, across the deployment life cycle (compiled by the author based on [27, 33, 34]).

Forecast estimates indicate a substantial macroeconomic effect from large-scale deployment of agentic systems and robotic solutions: according to McKinsey calculations, by 2030 the relevant technologies are capable of generating approximately 2.9 trillion dollars in economic value annually in the United States alone [35, 37]. At the same time, a transformation of the structure of demand for competencies is observed: over two years, the need for AI fluency skills, interpreted as the ability to confidently operate artificial intelligence tools and embed them into work loops, increased sevenfold. Labor market dynamics are described not by the logic of direct displacement of human functions but by a redistribution of cognitive effort: the share of time spent on document preparation and routine formatting declines, and the importance of posing correct questions, contextualizing tasks, and interpreting results produced by artificial intelligence systems increases [30, 35].

Changes in the character of work and the strengthening of agent autonomy have prompted a revision of approaches to interface design. Classical graphical user interfaces historically formed around clicks, navigational patterns, and fixed screen flows, whereas by 2024 a need has emerged for agentic interfaces that allow dynamic reconfiguration of the user interface depending on the dialogue context and the current state of the task [18]. Within this direction, the Google A2UI project, Agent-to-User Interaction, is considered an attempt to standardize the representation of updatable interfaces generated by agents through a set of principles that constrain arbitrariness and increase controllability [15]. The declarative approach assumes that the agent transmits a JSON description of a component tree, after which the client side performs rendering via native libraries and frameworks, including Lit, Angular, and Flutter [15, 23]. Security is achieved by refusing to execute arbitrary code on the client side: the agent operates within a catalog of pre-validated components defined by the host application, which reduces the risk of injecting malicious logic. Interface iterativity implies incremental changes during interaction, allowing the presentation to be adapted to new requirements and refinements as the dialogue develops. A complementary element is the Agent-User Interaction protocol, which links the agentic backend and frontend, ensuring multimodality, including text, voice, and video, and supporting the human in the loop principle: the possibility of operative intervention, correction, or confirmation of agent actions in real time is preserved, which increases the controllability of autonomous behavior [15, 18].

The growth of agent autonomy simultaneously expands the attack surface and complicates governance loops, forming new threat vectors, including prompt injections, token compromise, model poisoning, and unauthorized data movements between systems [38]. In this regard, the concept of Artificial Intelligence Trust, Risk, and Security

Management, advanced by Gartner, is consolidated as a framework in which security is interpreted as an aggregate of trust, risk management, and protection measures distributed across multiple control layers [3]. Forecasts indicate that by 2028 more than half of enterprises will use specialized artificial intelligence security platforms to counter risks associated with rogue agents, that is, autonomous components operating outside permissible behavior policies [6]. In applied terms, key 2024 measures include machine-to-machine authentication using cryptographic attestations, which makes it possible to unambiguously identify each agent in the corporate environment [39]. Additionally, the practice of automatically assigning agents unique identities in corporate access directories through Entra Agent ID is highlighted, which simplifies permission management and the segregation of authority [40]. A significant element of control is immutable audit logs, in which each agent action must record a timestamp, identifier, requested action, applied policy, and execution result, ensuring traceability and an evidentiary basis for investigations and compliance [29, 38].

The governance loop in agentic systems is not exhausted by technical guardrails and access regulations, because autonomy requires aligning behavior with the organization's normative and value foundations. Within this logic, the need for value embedding is emphasized, implying the formation of constraints and decision criteria that prevent unacceptable deviations even as context changes. A practically significant task becomes identifying and preventing goal drift, when an agentic system gradually shifts priorities toward locally optimal but strategically undesirable actions; an additional mechanism regarded as critically important is emergency shutdown tooling, which allows autonomous components to be stopped when behavior falls outside allowable parameters [28, 41].

In discussing scaling barriers, a 2024 paradox is recorded: artificial intelligence is widely applied as an instrument for increasing individual productivity, yet it rarely becomes a full-fledged infrastructural foundation of enterprises [7]. Organizations often get stuck in an endless chain of Proof of Concept efforts, which is associated with three systemic causes: data fragmentation, the persistence of workflows not redesigned for artificial intelligence capabilities, and the absence of clearly articulated scaling priorities that would move initiatives from an experimentation mode to a transformation program mode. In response to these constraints, market leaders rely on the 10-20-70 principle, in which the main share of effort is directed to people and processes, while the technological component is treated as necessary but not determinative for success [7, 32]. Within such a governance logic, agentic systems are most appropriately interpreted as a dynamic intermediate workforce capable of reconfiguring connectivity between systems and functions, rather than as improved chatbots mechanically inserted into prior loops without changing organizational architecture

Conclusion

The review of the transformation of operating models under the influence of multi-agent systems and hybrid generative architectures in 2023–2024 makes it possible to formulate a set of synthesizing propositions.

The technological trajectory has convincingly shifted toward autonomy as a foundational property of corporate artificial intelligence systems. The diffusion of Agent-to-Agent protocols and Agentic user interface approaches forms infrastructural preconditions for an Internet of agents in which digital entities created by different vendors gain the ability to interact in an aligned and seamless manner to achieve user goals. A substantial accompanying shift has been the consolidation of the tooling stack around TypeScript, which within the agentic solutions loop has been established as a de facto standard due to the combination of strict typing, predictable execution, and an elevated level of code security under conditions of partial generation by autonomous components.

In software engineering, a transition to the Software Engineering 3.0 paradigm has been recorded, in which agentic systems cease to be auxiliary means for accelerated code writing and acquire the status of functionally responsible virtual colleagues spanning architectural design, testing, and deployment. However, this shift is not reducible to a multiplicative increase in development speed, potentially estimated as a 100–1000-fold increase, because controllability and trust in results become critically important. Under these conditions, forms of structured governance of agentic development are in demand within the logic of Structured Agentic Software Engineering, ensuring reproducibility, traceability of engineering decisions, and the reduction of

risk of uncontrolled deviations.

The empirical picture of business effects remains markedly heterogeneous. Despite the existence of cases demonstrating a 15–30% reduction in operating costs and a 40–70% acceleration of processes, most organizations encounter difficulties in moving from demonstrative initiatives to scaling. In 2024, the determining success factor is the ability to overcome the inertia of pilot projects and embed artificial intelligence into the operational fabric of the enterprise, which presupposes not only technical integration but also the returning of processes, roles, and accountability. Within this logic, cultural and organizational transformation plays a key role, accounting for approximately 70% of total effort, whereas the technological component functions as a necessary but insufficient condition for outcomes.

Security and ethical alignment issues have acquired the status of priority constraints that determine the bounds of permissible autonomy. The concept of Artificial Intelligence Trust, Risk, and Security Management and identity and access control instruments, including Entra Agent ID, become critical elements of risk governance associated with autonomous artificial intelligence action. The transparency of reasoning and decisions of agentic systems, as well as the preservation of human oversight mechanisms, are treated as mandatory conditions for deployment in mission-critical domains, where unauthorized actions, uncontrolled data propagation, and non-traceable governance consequences are unacceptable.

As directions for the practical application of the results obtained, it is appropriate to highlight an orientation toward forming artificial intelligence ready data as a necessary foundation for the robust operation of agentic systems, a transition to modular agentic platforms and open protocols for the sake of interoperability, and investments in workforce reskilling with a shift in emphasis from performing routine operations to orchestrating and mentoring artificial intelligence agents. An additional maturity requirement is the deployment of continuous monitoring and observability, ensuring real-time control of autonomous loop behavior and early detection of goal drift, coordination errors, and violations of security policies.

Further competitiveness in 2026 and subsequent periods will be determined not so much by access to the most powerful model as by the ability to form an effective and trusted ecosystem of collaborative work between people and multiple specialized artificial intelligence agents unified by shared interaction protocols, governance mechanisms, and transparent accountability loops.

References

1. McKinsey & Company. (2024, May 30). The state of AI in early 2024: Gen AI adoption spikes and starts to generate value. Retrieved from: https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-2024 (date accessed: June 6, 2024).

2. Stanford University Human-Centered Artificial Intelligence. (2024). Artificial Intelligence Index Report 2024. Retrieved from: https://hai.stanford.edu/ai-index/2024-ai-index-report (date accessed: June 10, 2024).

3. Deloitte. (2024). The state of generative AI in the enterprise: Q1 report. Retrieved from: https://www.deloitte.com/ce/en/services/consulting/research/state-of-generative-ai-in-enterprise.html (date accessed: June 14, 2024).

4. Feuerriegel, S., Hartmann, J., Janiesch, C., & Zschech, P. (2024). Generative AI. Business & Information Systems Engineering, 66(1), 111–126. https://doi.org/10.1007/s12599-023-00834-7

5. Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., et al. (2023). The rise and potential of large language model based agents: A survey. arXiv. https://doi.org/10.48550/arXiv.2309.07864

6. Kanbach, D. K., Heiduk, L., Blueher, G., Schreiter, M., & Lahmann, A. (2024). The GenAI is out of the bottle: Generative artificial intelligence from a business model innovation perspective. Review of Managerial Science, 18(4), 1189–1220. https://doi.org/10.1007/s11846-023-00696-z

7. Benaich, N., & Hogarth, I. (2023). State of AI Report 2023. Retrieved from: https://www.stateof.ai/2023 (date accessed: March 18, 2024).

8. OpenAI. (2024, January 10). Introducing ChatGPT Team. Retrieved from:

https://openai.com/index/introducing-chatgpt-team/ (date accessed: January 18, 2024).

9. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., & Wang, C. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. arXiv. https://doi.org/10.48550/arXiv.2308.08155

10. Chen, W., Su, Y., Zuo, J., Yang, C., Yuan, C., Chan, C.-M., et al. (2023). AgentVerse: Facilitating multi-agent collaboration and exploring emergent behaviors. arXiv. https://doi.org/10.48550/arXiv.2308.10848

11. Guo, T., Chen, X., Wang, Y., Chang, R., Pei, S., Chawla, N. V., Wiest, O., & Zhang, X. (2024). Large language model based multi-agents: A survey of progress and challenges. arXiv. https://doi.org/10.48550/arXiv.2402.01680

12. Li, G., Hammoud, H. A. A. K., Itani, H., Khizbullin, D., & Ghanem, B. (2023). CAMEL: Communicative agents for "mind" exploration of large language model society. arXiv. https://doi.org/10.48550/arXiv.2303.17760

13. Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, C., et al. (2023). MetaGPT: Meta programming for a multi-agent collaborative framework. arXiv. https://doi.org/10.48550/arXiv.2308.00352

14. Microsoft. (2024, June 24). Introduction to Semantic Kernel. Retrieved from: https://learn.microsoft.com/en-us/semantic-kernel/overview/ (date accessed: June 28, 2024).

15. Figma. (n.d.). Figma AI. Retrieved from: https://www.figma.com/ai/ (date accessed: July 2, 2024).

16. Supabase. (n.d.). Architecture. Retrieved from: https://supabase.com/docs/guides/getting-started/architecture (date accessed: February 7, 2024).

17. Supabase. (n.d.). Auth architecture. Retrieved from: https://supabase.com/docs/guides/auth/architecture (date accessed: February 12, 2024).

18. TypeScript. (n.d.). The TypeScript Handbook. Retrieved from: https://www.typescriptlang.org/docs/handbook/intro.html (date accessed: March 5, 2024).

19. Ant Design. (n.d.). Components overview. Retrieved from: https://ant.design/components/overview/ (date accessed: April 16, 2024).

20. OWASP Foundation. (2024). Top 10 for LLMs and Gen AI Apps 2023–24. Retrieved from: https://genai.owasp.org/llm-top-10-2023-24/ (date accessed: April 18, 2024).

21. Vandevenne, N., Van Riel, J., & Poels, G. (2023). Green enterprise architecture (GREAN): Leveraging EA for environmentally sustainable digital transformation. Sustainability, 15(19), 14342. https://doi.org/10.3390/su151914342

22. van de Wetering, R. (2022). The role of enterprise architecture-driven dynamic capabilities and operational digital ambidexterity in driving business value under the COVID-19 shock. Heliyon, 8(11), e11484. https://doi.org/10.1016/j.heliyon.2022.e11484

23. Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. https://doi.org/10.1145/3586183.3606763

24. Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., & Anandkumar, A. (2023). Voyager: An open-ended embodied agent with large language models. arXiv. https://doi.org/10.48550/arXiv.2305.16291

25. Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. arXiv. https://doi.org/10.48550/arXiv.2303.11366

26. Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., & Mordatch, I. (2023). Improving factuality and reasoning in language models through multiagent debate. arXiv. https://doi.org/10.48550/arXiv.2305.14325

27. Weisz, J. D., He, J., Muller, M., Hoefer, G., Miles, R., & Geyer, W. (2024). Design principles for generative AI applications. In Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems. https://doi.org/10.1145/3613904.3642466

28. Chaudhry, B. M. (2024). Concerns and challenges of AI tools in the UI/UX design process: A cross-sectional survey. In Extended Abstracts of the CHI Conference on Human Factors in Computing Systems. https://doi.org/10.1145/3613905.3650878

29. Qian, C., Liu, W., Liu, H., Chen, N., Dang, Y., Li, J., Yang, C., Chen, W., Su, Y., Cong, X., Xu, J., Li, D., Liu, Z., & Sun, M. (2023). ChatDev: Communicative agents for software development. arXiv. https://doi.org/10.48550/arXiv.2307.07924

30. Yang, J., Jimenez, C. E., Wettig, A., Lieret, K., Yao, S., Narasimhan, K., & Press, O. (2024). SWE-agent: Agent-computer interfaces enable automated software engineering. arXiv. https://doi.org/10.48550/arXiv.2405.15793

31. Huang, D., Zhang, J. M., Luck, M., Bu, Q., Qing, Y., & Cui, H. (2023). AgentCoder: Multi-agent-based code generation with iterative testing and optimisation. arXiv. https://doi.org/10.48550/arXiv.2312.13010

32. Nielsen Norman Group. (2023, January 29). CASTLE framework for productivity/workplace applications. Retrieved from: https://www.nngroup.com/articles/castle-framework/ (date accessed: January 31, 2024).

33. Nielsen Norman Group. (1994, April 24). 10 usability heuristics for user interface design. Retrieved from: https://www.nngroup.com/articles/ten-usability-heuristics/ (date accessed: February 2, 2024).

34. Figma. (n.d.). Anticipation, experimentation and AI: Design trend report. Retrieved from: https://www.figma.com/reports/ai-design-trends-2024/ (date accessed: July 5, 2024).

35. McKinsey Global Institute. (2023, June 14). The economic potential of generative AI: The next productivity frontier. Retrieved from: https://www.mckinsey.com/capabilities/tech-and-ai/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier (date accessed: May 6, 2024).

36. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., et al. (2024). A survey on large language model based autonomous agents. Frontiers of Computer Science, 18, 186345. https://doi.org/10.1007/s11704-024-40231-1

37. Kar, A. K., Varsha, P. S., & Rajan, S. (2023). Unravelling the impact of generative artificial intelligence (GAI) in industrial applications: A review of scientific and grey literature. Global Journal of Flexible Systems Management, 24(4), 659–689. https://doi.org/10.1007/s40171-023-00356-x

38. Microsoft Research. (n.d.). AutoGen. Retrieved from: https://www.microsoft.com/en-us/research/project/autogen/(date accessed: June 20, 2024).

39. Singh, K., Chatterjee, S., & Mariani, M. (2024). Applications of generative AI and future organizational performance: The mediating role of explorative and exploitative innovation and the moderating role of ethical dilemmas and environmental dynamism. Technovation, 133, 103021. https://doi.org/10.1016/j.technovation.2024.103021

40. OpenAI. (2024, January 10). ChatGPT release notes. Retrieved from: https://help.openai.com/en/articles/6825453-chatgpt-release-notes (date accessed: January 20, 2024).

41. National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0) [PDF]. Retrieved from: https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf (date accessed: February 20, 2024).