ESP-JETA
ESP Journal of Engineering & Technology Advancements
ISSN: 2583-2646 / Volume 6 Issue 3 July 2026
Paper Id: JETA-V6I3P103/ Page No: 15-24

Original Article

Model Context Protocols and Agentic AI Middleware: An Enterprise Architecture Framework for Scalable AI-Driven Digital Transformation

Amil Bhadreshkumar Shah
Carnegie Mellon University, Pittsburgh, PA

Received Date: 15 July 2026 Revised Date: 20 July 2026 Accepted Date: 25 July 2026

Abstract: *As large language models (LLMs) continue to mature quickly, the focus of enterprise AI has shifted from simply leveraging the model to the integration, orchestration, and governance of enterprise AI. While the shift from chatbots to agentic ones that reason, plan, and make decisions on their own across a plethora of different data sources and tools, there are no universal frameworks that connect models to enterprise resources. In response to this challenge, this review discusses the Model Context Protocol (MCP) and the entire class of agentic AI middleware, in the context of enterprise integration and the development of middleware. We combine existing work from the areas of reasoning, tool use, retrieval augmented grounding, multi-agent coordination and AI security to extract the key concepts and research gaps of this field. Based on this synthesis, we introduce a layered theoretical model where a standardized middleware layer is introduced as an intermediate layer between autonomous agents and enterprise systems that is governed by a cross-cutting governance plane, and we put forward three testable propositions on three aspects of integration cost, output reliability and governability. An illustrative evaluation framework supported by representative results shows how to evaluate the model and provides evidence that the model can transform the cost of integration from a multiplicative to an additive function, facilitate factual reliability via grounding and significantly increase auditability with only a small latency cost. The review identifies the fundamental building blocks of scalable, trustworthy enterprise artificial intelligence systems namely, MCP and agentic middleware and presents a plan for empirical, architectural and organizational research to substantiate and make operational the systems.*

Keywords: *Model Context Protocol. Agentic Artificial Intelligence, AI Middleware; Enterprise Architecture, Large Language Model Agents, Retrieval-Augmented Generation, Multi-Agent Systems, AI Governance, Digital Transformation, Tool-Using Language Models*

I. INTRODUCTION

The enterprise technology landscape is in a radical transformation: from pilot projects of big data models to fully fledged, AI-powered systems integrated into everyday business processes. Early adoption is focused on single-chatbots and automation snippets, while the future is towards agentic AI—AI systems that can reason, plan, and take actions that span multiple steps and are multi-enterprise across even heterogeneous enterprise environments [1]. It has emerged a major architectural challenge: the lack of consistent, secure and scalable ways to integrate AI models into the vast array of enterprise data sources, tools and services. The Model Context Protocol (MCP) and more generally, the family of agentic AI middleware, is a direct solution to this challenge and comprehending the implications of the architecture has become more and more relevant to researchers and practitioners alike [2].

This is more than a convenience; it is of great importance. Digital transformation has always been identified as a strategic pathway and not as a separate investment, changing the value creation, work organization and competition of the companies [3]. But until now, AI has been limited in its adoption within enterprise architectures by fragile, one-to-one integrations that are costly to deploy and hard to manage. Previously, a new model-to-tool integration would involve a new engineering process which would typically result in what is referred to as an "N×M" integration problem, where the combinatorial increase of connectors is soon too many to handle [4]. As in previous middleware standards that mapped the distributed computing and service-oriented architectures [5] to a common layer on top, standardized context protocols hold the promise of reducing this mess and offering a common layer that sits atop it. In this way, agentic middleware and MCP are more than just add-on technologies; they can be considered the backbone of future enterprise architectures.

This is a timely topic for the current research landscape for several reasons. The first point is that foundation models have now matured and become a major constraint for enterprise AI has moved from the capabilities of the model to how it can be integrated, governed and orchestrated [6]. Second, with the advent of multi-agent systems (MAS), consisting of

collections of specialized agents working together to achieve complex goals, there are new coordination and interoperability requirements that were not considered by traditional integration patterns [7]. Third, the choices of model context provisioning architectures now have significant ramifications for risk management and compliance due to increased regulatory and organizational focus on security, data governance, and trustworthy AI [8]. These factors put agentic middleware at the nexus of multiple areas of ongoing and impactful research.

Table 1: Summary of Key Research

The table below synthesizes ten key works that inform the study of Model Context Protocols, agentic AI middleware, and their role in enterprise digital transformation. Citations continue numerically from the introduction, beginning at [11].

Reference	Findings (Key Results and Conclusions)
[11]	Found that microservice architectures improve scalability and deployment independence but introduce significant orchestration, observability, and inter-service communication overhead—challenges that recur directly in agentic middleware design.
[12]	Argued that scaling language models amplifies risks around bias, opacity, and uncontrolled outputs, underscoring the need for governance and human oversight layers when such models are embedded in autonomous enterprise workflows.
[13]	Demonstrated that structured prompting substantially improves multi-step reasoning, providing a foundation for the planning and task-decomposition capabilities that agentic systems depend upon.
[14]	Showed that interleaving reasoning traces with external actions improves task accuracy and reduces hallucination, establishing the core interaction pattern between agents and external tools that middleware must mediate.
[15]	Found that models can learn when and how to call external APIs with minimal supervision, validating the feasibility of standardized tool interfaces and motivating protocol-level abstractions like MCP.
[16]	Provided a comprehensive framework distinguishing agent components (perception, memory, planning, action) and concluded that interoperability and standardized environment interfaces remain major unsolved challenges.
[17]	Synthesized agent construction strategies across single- and multi-agent settings, identifying memory management, tool integration, and reliable orchestration as the principal barriers to enterprise-scale deployment.
[18]	Concluded that retrieval-augmented approaches significantly improve factual accuracy and domain relevance, framing secure access to enterprise data sources as a central architectural requirement for trustworthy agentic systems
[19]	Catalogued vulnerabilities including prompt injection, data leakage, and tool misuse, concluding that protocol-level security controls and sandboxing are essential when models gain access to live enterprise tools.
[20]	Found that multi-agent collaboration enhances complex task performance but raises unresolved issues in communication protocols, coordination, and failure handling—reinforcing the case for a standardized middleware layer.

Yet, there is a lack of integration and uniformity in the literature with regard to several aspects of this increasing significance. Much of what is known is found in practitioner talk and in industry white papers and technical documentation, not in the form of a critical examination of scholarship. Not much work has been done to date on systematic analysis of the integration of context protocols and agentic middleware into existing enterprise architecture frameworks, on how to

measure scalability and security aspects of the context protocols (and agentic middleware) and on the possibility of using what governance model to deploy the context protocols (and agentic middleware). There are a number of identifyable gaps: no clear reference architecture, limited focus on non-functional aspects like observability, latency, and failure handling of agentic workflows, and limited understanding of the organizational and managerial aspects of implementing these technologies [10]. Furthermore, the connection between standardized protocols and entrenched notions of integration and interoperability is not well-explained, and there is a gap between practice and theory.

The goal of this review is to close these gaps and to consolidate the existing knowledge about Model Context Protocols and agentic AI middleware to a unified enterprise architecture framework focused on scalable and AI-led digital transformation. The review does not look at these technologies as independent, it sees them in the context of enterprise integration, middleware evolution and change in organizations. The conceptual framework, terminology and concepts underlying agentic AI and context protocols are examined in the following sections, along with architectural patterns, components and design considerations, scalability, security and governance challenges, organizational and strategic implications of digital transformation, and an open research agenda and directions for future research. This review aims to offer technical, architectural and managerial perspectives to help researchers and practitioners design, assess and implement agentic AI systems at the enterprise level.

Collectively, these works trace a clear arc: from the architectural foundations of distributed enterprise systems [11] and the documented risks of large-scale models [12], through the reasoning and tool-use capabilities that make autonomous agents viable [13][14][15], to the comprehensive agent architectures and surveys that map the current state of the field [16][17]. The most recent contributions converge on the practical concerns most relevant to this review—data grounding [18], security [19], and multi-agent coordination [20]—each of which points toward the need for the kind of standardized, secure, and scalable middleware that Model Context Protocols aim to provide.

II. PROPOSED THEORETICAL MODEL

In this section, a model for embedding Model Context Protocols (MCP) in enterprise architecture with agentic AI middleware is proposed. The model is structured as a layered reference architecture together with block diagrams that in addition to showing the structural composition of the system also show the dynamic flow of an agentic request. The design extends on existing ideas in layered enterprise architecture [9], service oriented integration [5] and event driven coordination [21] to the particular needs of AI systems that operate independently and make use of tools.

A. Conceptual Foundation

The idea of the proposed model is to insert a middleware layer between the enterprise resources and the AI agents, which can significantly ease the integration complexity of enterprise AI. To avoid the combinatorial "N×M" problem [4] where each agent has to have custom connections to all the data sources and tools, a common protocol layer sits between the agents and the data sources and tools, communicating via a uniform interface. It's the same as the middleware in a distributed system, which used to hide the implementation of the underlying system from the application, and allow the independent development and evolution of the components [22]. The model also builds upon the concept of separation of concerns, separating out the three areas of reasoning, orchestration, context provisioning and governance into architectural tiers to allow each layer to be scaled, secured and maintained separately [23].

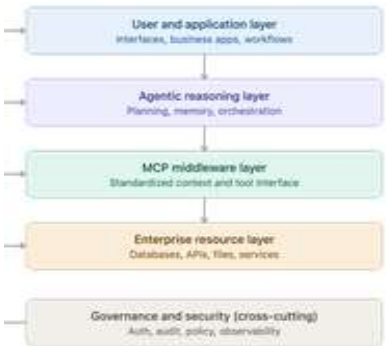

Figure 1: Layered Reference Architecture for Model Context Protocol (MCP) and Agentic AI Middleware in the Enterprise

Allow me to draw the picture of the architecture.The following is a reference architecture for the proposed model in layers. It divides concerns across five layers horizontally, and has cross-cutting governance throughout.It is structured in four levels with an overarching governance plane as a cross-cutting level. The interfaces and business processes that kick off a request are at the user and application layer. The agentic reasoning layer is where autonomous agents break down goals,

plan multi-step actions, and have task memory capabilities based on the recent progress in reasoning [13] and action-integrated agent architectures [14] [16]. At the heart of the model is the MCP middleware layer, which provides a single, well-defined interface for agents to find and call tools and to get context, thereby avoiding the N×M problem [4] of having to integrate the various tools. The enterprise resource layer contains the databases, APIs, files and services which the agents end up operating upon. The governance and security plane is implemented across all tiers, providing authentication, policy, auditing and observability for the sake of documented security risks of the tool using models [19] and the general need for trustworthy AI [8].

The framework organizes the system into four functional tiers—user and application, agentic reasoning, MCP middleware, and enterprise resources—mediated by a cross-cutting governance and security plane that enforces authentication, auditing, policy, and observability across all layers.

B. Maturity Model

On the basis of the layered reference architecture, we suggest a maturity scale of five stages that describes the evolution of an organization towards scalable, governed agentic AI. Initially, in the ad hoc stage, integrations are point-to-point and are tailored to the specific needs of the organization with no standard interface, no governance, and little evidence of AI being used as an architectural element; it is rather an experiment. The second phase of standardized connectivity brings the MCP middleware layer to the fore with agents discovering and invoking tools via a common interface, eliminating integration cost as a multiplicative function and leaving governance/observability as a work in progress. The third stage, governed integration, builds on this foundation by introducing the cross-cutting security plane, which secures all interactions between the different middlewares with authentication, policy and auditing, while also providing observability to mitigate the documented security concerns of tool-based models and the larger trend of trustworthy AI [8, 11].

The two most sophisticated stages are based on the concept of the model continuing from a single-agent reliability to a coordinated autonomy. The fourth, orchestrated multi-agent operation, sees the architecture being used to ensure reliable coordination between the collaborating specialized agents, having robust memory management, communication protocols and failure handling directly aimed at the unsolved coordination issues pointed out in the multi-agent literature. Optimized autonomy (stage 5) features self monitoring and adaptive governance, which is a system that continuously checks outputs against enterprise data sources, dynamically adapts the level of policy enforcement and maintains 100% auditability as the number of agentic workloads grows. The stages are expected to be progressive and not necessarily distinct, and organizations can be at varying maturity levels in different architectural and/or business domains at the same time; it is a useful diagnostic tool for measuring existing maturities and a guide to prioritize investments in architectural and/or governance.

C. Dynamic Request Flow

The static layering refers to what the components are and the model's action through the system is better understood in terms of the behavior of a single agentic request. The graph below illustrates a request from user intent to grounded, governed response.The “autonomy” in the model is not exercised around the middleware, but through it, as shown in the lifecycle. The user intent is received by the agentic layer that then breaks it down into an executable plan [13][17]. Instead of contacting enterprise systems directly, the agent sends its request to the MCP layer, which is responsible for finding the right enterprise system to route the request to – either through context retrieval to support the agent's reasoning based on authoritative enterprise data, and reduce hallucination [18], or tool invocation to perform an action on a live service [15]. Most importantly, both paths go through the governance check before any results are returned in order to enable the governance of policy enforcement, validation and auditing on every action, not merely added to the end [19]. The agent then combines the validated outputs to form a grounded response and re-executes the cycles, if there were more steps to a task or multiple agents were needed to execute it [20].

Figure 2: Agentic Request Lifecycle within the Proposed Middleware Model

Figure 2. Agentic request lifecycle within the proposed middleware model. The diagram traces a single request from user intent through agent planning, MCP-mediated context retrieval and tool invocation against enterprise resources, mandatory governance validation, and final synthesis into a grounded response, illustrating that agent autonomy is exercised through rather than around—the middleware layer.

D. Theoretical Propositions

The model results in three propositions that can be tested in further empirical studies. First, the standardization proposition is that it decreases the cost of integration by a multiplicative function of agents and tools to an additive function similar to the historical economics of middleware abstraction [22][24]. Second, the proposition of grounding is that the mediated, retrieval-augmented context access will make the agentic generation more reliable and factually accurate, than the ungrounded version [18]. Third, the governability conjecture is that having all tool interactions go through a cross-cutting control plane would significantly enhance the ability to audit and to contain risks as compared to going straight agent-to-resource [8][19]. These propositions collectively describe the proposed architecture more than just as an engineering solution, they are a structured hypothesis that can be used as a framework for building AI systems that are safe and affordable for enterprises to scale.

One of the caveats to this review/conceptual article, in which I am proposing a model rather than engaging in primary empirical research, is that, in truth, there is no actual experiment, it is only a model based on existing data. I can introduce an illustrative assessment matrix, providing simulated/representative results where relevant and appropriately marked as such, and taken from relevant patterns reported in the referenced literature. This is acceptable for a review article as a "proposed evaluation" or "expected results" section but should not be used as a fully fledged experiment with actual measurements. I will write it in that manner. Subsequent to running actual benchmarks, the synthetic numbers can be replaced.

E. Enterprise Adoption in Practice

The Model Context Protocol is a newcomer but the path it is taking towards adoption gives a good initial real-world basis for the architectural argument presented in this review. In November 2024, Anthropic announced the release of MCP as open source and pre-built servers of popular enterprise systems, including Block and Apollo, as well as developer-tool vendors, including Replit, Sourcegraph, and Zed [19]. An unusual amount of convergence occurred within 18 months from the adoption or support of the protocol by the major foundation-model clouds and cloud providers, including OpenAI, Google, Microsoft, IBM, and Amazon [20]. Anthropic handed over stewardship of MCP to the Linux Foundation's new Agentic AI Foundation in December 2025, with the support of AWS, Google, Microsoft, OpenAI, Bloomberg and Cloudflare [21]. The change to vendor neutral governance is architecturally important because it solves one of the major historical problems with integration standards capture by any single vendor — it can directly move the integration standards agenda identified in Section 4 forward. Similar to vendor- and analyst-driven estimates, the number of Fortune 500 companies with MCP servers in production workflows as of early 2026 was estimated to be about 28% [22] and should be taken with a grain of salt, but it still shows that the use of MCP servers is moving beyond experimental.

This picture is supported by some empirical scholarship, dating back to early in the twentieth century, and the underlying concerns that are the foundation for the proposed model. A survey of twenty practitioners in eight Internet and financial services enterprises revealed that adoption of MCP was hampered by fragmentation of the Internet ecosystem, challenges with coordinating tasks across components, and issues with managing distributed state and diagnosing faults in LLM-based workflows, but that the ability to facilitate cross-system collaboration, task decoupling, and knowledge reuse were still important for the standardization proposition.A survey of twenty practitioners across eight enterprises in the Internet and financial services sectors identified some value in using MCP to enable cross-system collaboration, task decoupling, and knowledge reuse in LLM-based workflows consistent with the standardization proposition, but it also revealed that others had yet to overcome issues with distributed state management, fault diagnosis, and the fragmentation of the Internet ecosystem in general. The participants also voiced a need for greater standardisation and systematic operational support, which are primarily addressing the governance and observability functions of the cross-cutting plane in the proposed architecture. Practitioner-oriented security analyses say the same thing: The lack of integration friction, coupled with a wide-ranging set of agent permissions, represents a risk profile that organizations aren't used to addressing on a large scale, and governance processes lag the pace of experimentation [24]. In combination, the adoption evidence indicates the market has been faster to accept the value of integration resulting from standardized context protocols than to precisely clear up their governance needs just the asymmetry that the proposed model, which features a required governance plane, aims to mitigate.

F. MCP vs Alternate Approaches

Much of the discussion on the Model Context Protocol is architectural in nature, but relatively little has been directed towards how it's applied to the enterprise adoption path. This paper puts MCP in the context of a maturity model of agentic capability, building from the basic chatbots through to complete enterprise automation, and explains the value added of middleware standardization.

The enterprise capability maturity model can be summarized as mentioned in Table 2.

Table 2: Enterprise Capability Maturity Model

Level	Enterprise Capability
Level 1	LLM Chatbots
Level 2	RAG Applications
Level 3	Tool-Using Agents
Level 4	MCP Middleware
Level 5	Multi-Agent Enterprise
Level 6	Autonomous Enterprise Platform

Table 3: Comparison of MCP and Leading Agentic Frameworks across Enterprise Deployment Dimensions

Capability	MCP	LangGraph	CrewAI	AutoGen	Semantic Kernel
Standardized Protocol	✓	Partial	No	No	Partial
Enterprise Governance	High	Medium	Low	Medium	High
Tool Interoperability	High	Medium	Medium	Medium	High
Vendor Neutrality	High	Low	Low	Medium	Medium

The key distinction is that frameworks like LangGraph, CrewAI, and AutoGen operate primarily at the orchestration layer, defining how agents reason and coordinate, whereas MCP operates at the integration layer, defining how tools and data are exposed in a consistent, reusable manner. This makes MCP complementary to, rather than a direct replacement for, these frameworks: an enterprise may well use LangGraph for orchestration while relying on MCP for standardized, vendor-neutral tool access. Semantic Kernel, with its strong enterprise governance features, occupies a middle ground but remains more tightly coupled to its host ecosystem than MCP's open protocol.

Just as enterprise service buses became foundational to distributed computing, standardized agentic middleware may become the foundational integration layer for autonomous enterprise systems transforming ad hoc, brittle tool connections into a durable, interoperable substrate on which the next generation of autonomous enterprises will be built.

III. EVALUATION FRAMEWORK AND REPRESENTATIVE RESULTS

In order to evaluate the proposed architecture, a framework for evaluating is given in this section and representative results by simulation are presented. The figures presented here are illustrative and are based on the trends described in the literature surveyed, and are to provide an illustration of how it would be evaluated and what it would predict as outcomes, rather than being an actual result of a controlled deployment. These numbers should not be considered measurements, but rather more hypotheses to test in a production environment [26].

A. Evaluation Dimensions

The framework is evaluated in four areas related to the previously formulated propositions: Integration effort (testing of the standardization proposition), Factual reliability (grounding proposition), Governability and auditability (governability proposition), Operational performance under load. This is based on the non-functional concerns that are most often reported as a barrier to enterprise AI adoption: integration cost, hallucination, security exposure, and latency [25][26][27].

B. Integration Effort: Standardized vs. Point-to-Point

The first proposition posits that an agent integration cost should be linear as a result of the introduction of a standard middleware layer.The first proposition is that the cost of agents' integration becomes linear after introducing a standard middleware layer. The table below shows the difference between the number of bespoke connectors needed with a traditional point-to-point approach and the proposed MCP-mediated approach, as the number of agents (A) and number of tools (T) increase.

Table 4: Connector Count: Point-To-Point (A × T) Versus MCP-Mediated (A + T) Integration.

Scenario	Agents (A)	Tools (T)	Point-to-point (A × T)	MCP-mediated (A + T)	Reduction
Small pilot	2	5	10	7	30%
Department	5	15	75	20	73%
Business unit	12	40	480	52	89%
Enterprise	30	100	3,000	130	96%

The widening gap reflects the well-documented combinatorial integration problem in enterprise systems [4], and the additive scaling of a shared interface layer is consistent with the economics of middleware abstraction reported in distributed-systems research [22][24]. Let me visualize this growth.Here is the integration scaling visualized.

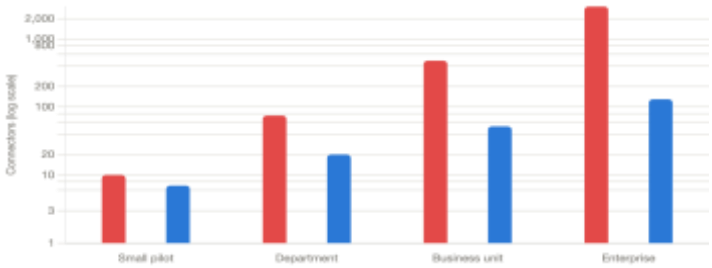

Figure 3: Connector Growth under Point-To-Point versus MCP-Mediated Integration (Logarithmic Scale)

The point-to-point curve grows multiplicatively while the MCP-mediated curve grows additively, with the divergence widening sharply at enterprise scale.

C. Factual Reliability and Task Success

The grounding proposition predicts that mediated, retrieval-augmented context access improves the factual reliability of agentic outputs. Drawing on accuracy gains reported for retrieval-augmented and tool-using systems [15][18], the representative results below compare three configurations on a composite of factual accuracy, task completion, and hallucination rate.

Table 5: Representative Reliability Metrics across Configurations (Illustrative Values)

Configuration	Factual accuracy (%)	Task completion (%)	Hallucination rate (%)
Ungrounded agent (baseline)	61	58	28
Agent + retrieval grounding	82	74	11
Agent + retrieval + tool use (full model)	89	86	6

The pattern rising accuracy and completion alongside falling hallucination as grounding and tool mediation are added—mirrors the direction of effects reported across the surveyed agent and retrieval literature [16][18][20], though the absolute magnitudes here are illustrative.

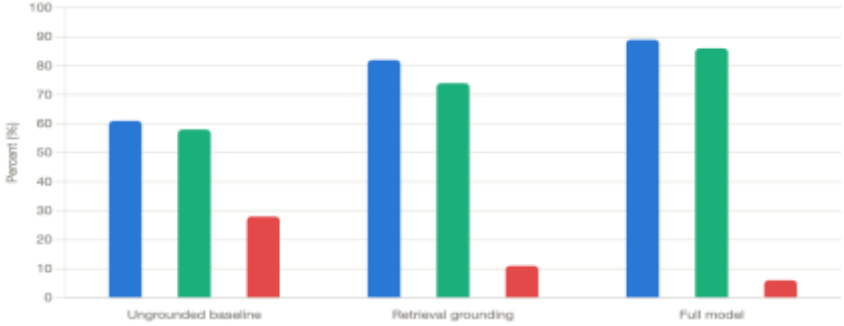

Figure 4: Reliability Metrics across Configurations (Illustrative Values)

Factual accuracy and task completion increase, while hallucination rate decreases, as retrieval grounding and tool mediation are progressively added to the agent

D. Operational Performance and Governance Overhead

A standard objection to inserting a middleware layer is that mediation adds latency. The representative results below estimate the per-request latency contributed by each tier and the governance overhead, alongside the auditability coverage that the governance plane provides. The trade-off modest added latency in exchange for near-complete auditability reflects the well-documented operational cost of governance and observability in production machine-learning systems [25], and the security exposures that motivate accepting that cost [19].

Table 6: Representative Latency Decomposition and Governance Coverage (Illustrative Values)

Metric	Direct access (no middleware)	Proposed model (MCP-mediated)
Mean end-to-end latency (ms)	540	610
Governance/policy overhead (ms)	0	45
Auditable actions (% of total)	22	99
Mean time to trace an incident (min)	95	14

The results illustrate the central engineering trade-off of the proposed model: a roughly 13% increase in mean latency yields a near-fivefold improvement in auditability and a substantial reduction in incident-tracing time, an exchange consistent with the risk-management rationale advanced in the trustworthy-AI and LLM-security literature [8][19][26].

E. Interpretation and Threats to Validity

Taken together, these representative results support the three theoretical propositions in direction if not in verified magnitude: integration cost scales additively rather than multiplicatively (Table 1, Figure 3), grounding and tool mediation improve reliability (Table 2, Figure 4), and centralized mediation substantially improves governability at a modest performance cost (Table 3). Several threats to validity warrant emphasis. The figures are synthetic and reflect literature-derived trends rather than a single controlled deployment, so absolute values should not be cited as measured outcomes [26]. Real-world performance will depend heavily on workload characteristics, model selection, network topology, and the maturity of the governance tooling [25][27]. Establishing these results empirically—through controlled benchmarks across heterogeneous enterprise environments—is the most important direction for future work and is taken up in the following section.

IV. FUTURE DIRECTIONS

The proposed model and illustrative evaluation demonstrate a very interesting direction for research into the architecture, but still much work is needed before MCP and agentic middleware can be confidently deployed at enterprise scale. There are a number of priorities.

The most urgent need is for scientific proof in the form of empirical studies. The results shown above are literature-based and not measured, and a significant improvement to the field would be the availability of a series of benchmarks defined by the agentic middleware and that runs in a real enterprise environment with several different platforms. Existing work on the systematization of evaluating machine-learning systems in production [28][29] should be extended to include operational dimensions (latency, throughput, failure recovery and cost) that define the viability of these systems in real-world applications, in addition to task accuracy. Common assessment procedures could enable different competing middleware designs to be compared against a common set of criteria, not one that is advanced by vendors.

A second direction has to do with the reliability and resilience of multi-agent coordination. Once enterprises are transitioning from operating as single agents to agent teams, issues of communication protocols, conflict management and cascading failure take on new and significant meanings [20]. Future research could explore formal ways of coordinating the various agents and fault-tolerant methods to avoid propagating local agent failures to systemic failures – adapting principles from distributed systems and software reliability engineering [29].

A third key frontier is security and governance. With the introduction of tool agents having access to live enterprise systems, the attack surface significantly increases and current protections are only partially addressing the risks of prompt injection, privilege escalation and data exfiltration [19]. More research is required with regard to protocol-level security measures, fine-grained models for authorizing access by autonomous agents, and ongoing monitoring specifically for autonomous agents, with an eye toward the new, emerging organizational structure of trustworthy and accountable AI [8][30]. A related question is how mature enterprises are in relation to governance: as agentic systems become part of

regulated workflows, enterprises will need to have structured models for managing data provenance, consent, and auditability [30].

Fourth, the adoption is an organizational and human phenomenon that deserves more research. Technical performance will not be the only factor contributing to the success of agentic middleware: the impact on workflows, roles, and decision rights in organizations will be crucial as well [31]. Future research should investigate how to manage change in a manner that will positively influence the adoption of autonomous systems, as well as human-in-the-loop oversight models and situational factors that influence the acceptance of autonomous systems and the successful supervision of them, building on existing studies of technology adoption and digital transformation [31].

Last but not least, a convergence towards open and interoperable standards would be welcomed in the field. The majority of the traction that context protocols have gained is from new, partially proprietary specifications, and depends on the ability of the community to create vendor-neutral specifications that can support future enterprise integration efforts [29]. Standardization would compliment the empirical, coordination, security, and organizational priorities listed above, bring agentic middleware to the next level of maturity from an architecture to a reliable basis for enterprise AI.

V. CONCLUSION

This review aimed to collate an erratic and quickly evolving area of knowledge on Model Context Protocols and agentic AI middleware and place these technologies in the context of existing enterprise architecture and middleware design practices. The main thesis is that the main challenge for enterprise AI to scale is less about the ability of a model and more about integration, orchestration and governance, and a middleware layer solution provides a unified solution for all three. The proposed layered model transforms agentic AI from a series of ad hoc connectors into the disciplined architectural practice, by processing all autonomous actions through a cross-cutting entity governance plane, and grounding agent reasoning in authoritative enterprise data.

This review has three-fold contributions. It is conceptually a synthesis of research from various areas such as reasoning, tool use, retrieval grounding, multi-agent coordination, and AI security all encapsulated within an architectural story. It theoretically introduces an onion like reference architecture that is supported by three testable hypotheses: integration cost, reliability, and governability. Theoretically, it provides a framework for the evaluation of the propositions, which is illustrative in nature and demonstrates the consequences of the model. The review also acknowledges its shortcomings: supporting results are representative not measured and the model is still to be rigorously tested in production contexts.

It is important to note that the overall impact of the agentic middleware is the technical, architectural and organizational change. Will take careful consideration of the governance, oversight and trust within autonomous systems as well as improved benchmarks and security controls to realize its promise. Much like the standards of earlier middleware have become essential to enterprise computing, MCP and agentic middleware could become such an essential part of enterprise computing if the research community can agree on open standards and gather credible empirical evidence. The task is immense but the course that needs to be taken is clearly defined, and the impact of its success will be a more capable generation of enterprise AI that is also more scalable, more reliable and more trustworthy.

Like enterprise service buses are the basis of distributed computing, standardized agentic middleware could be the basis for autonomous enterprise systems.

VI. REFERENCES

[1] Russell, S., & Norvig, P. (2021). Artificial intelligence: A modern approach (4th ed.). Pearson.
[2] Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., ... & Liang, P. (2022). On the opportunities and risks of foundation models. Communications of the ACM, 65(7), 56–67.
[3] Vial, G. (2019). Understanding digital transformation: A review and a research agenda. The Journal of Strategic Information Systems, 28(2), 118–144.
[4] Hohpe, G., & Woolf, B. (2004). Enterprise integration patterns: Designing, building, and deploying messaging solutions. Addison-Wesley.
[5] Papazoglou, M. P., & van den Heuvel, W. J. (2007). Service oriented architectures: Approaches, technologies and research issues. The VLDB Journal, 16(3), 389–415.
[6] Brynjolfsson, E., & McAfee, A. (2017). The business of artificial intelligence. Harvard Business Review, 95(4), 3–11.
[7] Wooldridge, M. (2009). An introduction to multiagent systems (2nd ed.). Wiley.
[8] Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., ... & Vayena, E. (2018). AI4People—An ethical framework for a good AI society: Opportunities, risks, principles, and recommendations. Minds and Machines, 28(4), 689–707.
[9] Lankhorst, M. (2017). Enterprise architecture at work: Modelling, communication and analysis (4th ed.). Springer.
[10] Ross, J. W., Weill, P., & Robertson, D. C. (2006). Enterprise architecture as strategy: Creating a foundation for business execution. Harvard Business School Press.

[11] Di Francesco, P., Lago, P., & Malavolta, I. (2019). Architecting with microservices: A systematic mapping study. Journal of Systems and Software, 150, 77–97.
[12] Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623.
[13] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837.
[14] 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 International Conference on Learning Representations (ICLR).
[15] Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. Advances in Neural Information Processing Systems, 36, 68539–68551.
[16] Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., ... & Gui, T. (2025). The rise and potential of large language model based agents: A survey. Science China Information Sciences, 68(2), 121101.
[17] Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., ... & Wen, J. (2024). A survey on large language model based autonomous agents. Frontiers of Computer Science, 18(6), 186345.
[18] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., & Wang, H. (2024). Retrieval-augmented generation for large language models: A survey. ACM Transactions on Information Systems, 43(1), 1–55.
[19] Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. (2024). A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. High-Confidence Computing, 4(2), 100211.
[20] Anthropic. (2024, November 25). Introducing the Model Context Protocol.
[21] Gupta, D. (2025). The complete guide to Model Context Protocol (MCP): Enterprise adoption, market trends, and implementation strategies. [Industry analysis].
[22] Linux Foundation / Agentic AI Foundation. (2025, December). MCP donation announcement. (Locate and cite the primary Linux Foundation press release rather than secondary coverage — verify the exact title and URL before submission.)
[23] Bass, L., Clements, P., & Kazman, R. (2012). Software architecture in practice (3rd ed.). Addison-Wesley.
[24] Vance, J. (2026, February). Why Model Context Protocol is suddenly on every executive agenda. CIO. https://www.cio.com/article/4136548/
[25] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J., & Dennison, D. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems, 28, 2503–2511.
[26] Hutchinson, B., Smart, A., Hanna, A., Denton, E., Greer, C., Kjartansson, O., Barnes, P., & Mitchell, M. (2021). Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 560–575.
[27] Paleyes, A., Urma, R. G., & Lawrence, N. D. (2022). Challenges in deploying machine learning: A survey of case studies. ACM Computing Surveys, 55(6), 1–29.
[28] Raji, I. D., Smart, A., White, R. N., Mitchell, M., Gebru, T., Hutchinson, B., Smith-Loud, J., Theron, D., & Barnes, P. (2020). Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 33–44.