The Model Context Protocol (MCP): An Expert Analysis of its Role in LLM Integration, Agentic Systems, and Tool Use Standardization

1. Introduction: MCP as the Foundational Layer for Autonomous AI

The rapid expansion of Large Language Model (LLM) capabilities has transitioned academic curiosity into practical deployment. However, this transition has revealed a critical architectural gap: the lack of a standardized, reliable method for LLMs to interface dynamically with external resources and specialized tools. The **Model Context Protocol (MCP)** has emerged as the definitive response to this challenge, positioning itself as an essential open standard that defines a unified, bi-directional communication layer between AI models and external resources or tools.

The core purpose of MCP is to fundamentally enhance interoperability across heterogeneous AI ecosystems and significantly reduce the fragmentation that characterizes many early tool-augmented systems. MCP represents an architectural shift away from simple function calling—where LLMs rely solely on basic descriptions of available tools—to a protocol-driven method of context delivery. This method introduces a standardized, machine-actionable layer that explicitly links webservice endpoints with scientific concepts, operational schemas, and detailed metadata, thereby transforming passive tool descriptions into active context resources.

A review of high-impact scholarly works indicates a tight temporal clustering of foundational research regarding MCP—covering its definition, security analysis, performance benchmarks, and initial domain applications—all concentrated within the 2025 timeframe.[1] This synchronized emergence suggests that MCP is not a product of gradual academic evolution but rather a necessary and immediate industry response to critical architectural limitations. Early agentic systems quickly demonstrated scaling deficiencies when confronted with

complex, multi-step tasks, managing long contexts, and interacting with a large inventory of tools. The standardization offered by MCP directly addresses this necessity by providing the required architectural maturity.

2. Architectural Necessity: Deployment Models and Advanced System Integration

The effective integration of MCP necessitates a rigorous, defined approach to its server implementation and management. Unlike typical stateless APIs, MCP governs a unified, bi-directional communication protocol, requiring full-lifecycle governance. Research has defined the MCP server lifecycle as comprising four distinct phases: creation, deployment, operation, and maintenance. These phases are further delineated into 16 key activities, confirming that MCP extends beyond mere interface definition to become a structural requirement for system architecture definition.

FaaS-Hosted MCP Services for Scalability and Efficiency

A crucial technical innovation proposed for scalable MCP deployment is the utilization of cloud Functions as a Service (FaaS) to host MCP servers. The paper *AgentX: Towards Orchestrating Robust Agentic Workflow Patterns with FaaS-hosted MCP Services* empirically evaluated the success rate, latency, and cost associated with FaaS alternatives compared to local MCP server deployments. The architectural alignment between FaaS and MCP is evident: agentic tool use, particularly in production environments, tends to be sporadic and characterized by high-burst usage patterns. FaaS naturally accommodates this workload profile, offering significant benefits in terms of cost optimization, seamless scaling, and the elimination of persistent server infrastructure maintenance, which is indispensable for large-scale production agentic systems.

MCP in Mixture of Experts (MoE) Architectures

The Model Context Protocol plays a pivotal role in enabling sophisticated architectural patterns, such as the Mixture of Experts (MoE), particularly in highly specialized and critical

domains. In the context of asset discovery within Critical Infrastructure, the MoE architecture coordinates specialized, fine-tuned lightweight models for distinct operational phases. Here, MCP serves as the mechanism for injecting crucial **Model Context** by connecting the LLM-driven MoE to external, real-time threat intelligence sources.

Specifically, MCP facilitates the integration of threat intelligence feeds from platforms such as MITRE ATT&CK, which supports adversarial behavior mapping using an Industrial Control Systems (ICS) framework; MISP, for real-time indicator ingestion; and the CVE vulnerability database, which informs risk evaluation. By linking the LLM-based MoE to these constantly updated, external knowledge graphs, MCP facilitates semantic context awareness. This allows the system to dynamically interpret heterogeneous data sources and diverse protocols without relying on rigid, pre-determined taxonomies or a priori data classification, thereby refining expert routing decisions and enriching comprehensive asset risk assessments. MCP, therefore, functions as a dynamic routing and contextualization layer, proving essential for adapting AI systems to complex, real-time industrial environments.

The synchronized publication of foundational papers in 2025 illustrates the immediate response of the research community to the architectural demands of scalable agents:

Table 1: Foundational MCP Research: Timeline and Focus (2025 Cluster)

Paper (Ref.)	**Publication Date (Approx.)**	**Primary Theme**	**Key Architectural Concept**
Model Context Protocol (MCP)...	March 2025	Foundational Definition & Security	Full Server Lifecycle, Threat Taxonomy
MCPmed: A Call for MCP-Enabled...	July 2025	Domain Specialization	Machine-Actionable Layer (FAIR Principles)
Help or Hurdle? Rethinking Model...	August 2025	Empirical Evaluation	MCPGAUGE (Proactivity, Overhead Analysis)
Integrating Generative AI and... [6]	August 2025	Agentic Integration	Context-Aware Autonomous Agents

AgentX: Towards Orchestrating...	September 2025	Workflow Orchestration	FaaS-Hosted MCP Services, AgentX Pattern

3. Trajectory I: Proactive Security Design and Threat Taxonomy

A defining characteristic of the Model Context Protocol research is the proactive and immediate definition of its security implications. The foundational paper, published concurrently with the protocol's definition, established a comprehensive threat taxonomy that maps security and privacy risks across the entire MCP server lifecycle. This architectural maturity demonstrates an understanding that enabling bi-directional communication between AI models and external execution environments inherently creates significant new attack surfaces.

The security taxonomy constructs a comprehensive categorization of risks across four major attacker types: **malicious developers, external attackers, malicious users, and security flaws**. This analysis encompasses 16 distinct threat scenarios and utilizes real-world case studies to validate potential attack surfaces and vulnerability manifestations within existing MCP implementations. Based on this extensive threat modeling, the research proposes a set of fine-grained, actionable security safeguards. These safeguards are explicitly tailored to address risks specific to each lifecycle phase (creation, deployment, operation, maintenance) and corresponding threat category, offering practical guidance for the secure adoption of MCP in enterprise environments.

The Contextual Security Enhancement vs. Risk Trade-Off

The adoption of MCP presents a critical trade-off between increased functionality and expanded threat exposure. While the protocol must be secured against the 16 identified threat scenarios, its architecture is simultaneously used to *strengthen* security in critical operational domains. For example, in Critical Infrastructure, MCP enables asset discovery systems to execute real-time, context-aware risk assessments by linking discovered assets to external threat intelligence. The protocol allows for a device profile to be immediately enriched with CVE data and MITRE ATT&CK mapping, which is impossible with traditional,

disconnected tools.

This duality highlights that while the protocol introduces a new, defined vector that can be exploited (e.g., through malicious tool creation or data poisoning), it also provides the necessary mechanism for creating contextually informed security agents. Organizations adopting MCP must recognize that the protocol defines a trust boundary that requires continuous monitoring and adherence to the proposed security safeguards detailed in the threat taxonomy.[1]

4. Trajectory II: Performance Validation and the "Hurdle" of Tool Use

The efficacy of the Model Context Protocol is not determined solely by its robust architectural definition but critically by the empirical performance of LLMs when leveraging its capabilities. The paper *Help or Hurdle? Rethinking Model Context Protocol-Augmented Large Language Models* introduced **MCPGAUGE**, the first comprehensive evaluation framework specifically designed to probe LLM–MCP interactions.

The MCPGAUGE Framework

MCPGAUGE is a benchmark suite built to move beyond theoretical benefits by rigorously testing practical deployment challenges. The framework includes a comprehensive 160-prompt suite and 25 diverse datasets spanning critical domains such as knowledge comprehension, general reasoning, and code generation. The scope of the evaluation was massive, involving around 20,000 API calls, testing six commercial LLMs, and utilizing 30 distinct MCP tool suites, representing a substantial computational investment (over USD 6,000).

The framework measures performance across four key dimensions critical for tool-augmented agent deployment:

1. **Proactivity:** Quantifying the LLM's capacity for self-initiated, appropriate tool use without explicit prompting.
2. **Compliance:** Measuring the LLM's adherence to the defined tool-use instructions and the MCP schema.
3. **Effectiveness:** Evaluating the overall task performance and accuracy achieved after

integrating the MCP tool.

4. **Overhead:** Assessing the computational cost and latency incurred due to the MCP interaction.

Challenging Prevailing Assumptions

The provocative title of the performance study—*Help or Hurdle?*—underscores a crucial analytical finding: the integration of MCP, while architecturally sound, does not *universally* guarantee performance enhancement. The comprehensive study revealed four key findings that directly challenge the prevailing assumptions about the inherent effectiveness of MCP integration.

The data suggests that while the protocol successfully standardizes the communication layer, current LLMs often struggle with the complex *meta-reasoning* required to effectively leverage the context. Issues such as poor compliance with tool instructions and low proactivity indicate that models are not optimally tuned to understand *when* and *how* to invoke the specialized resources defined by MCP. Furthermore, the significant computational overhead incurred during MCP interaction can diminish the practical utility of the protocol, especially in latency-sensitive applications. This architectural limitation implies that future research must prioritize optimizing LLM training and fine-tuning specifically to improve MCP adherence, reduce interaction costs, and ensure the models reliably master complex context utilization.

The synthesis of these performance and security factors creates a balanced view of MCP adoption requirements:

Table 2: MCP Integration: Benefits, Risks, and Performance Dimensions

Category	**Observed Benefit**	**Identified Risk/Limitation**	**MCPGAUGE Dimension**
Architectural	Unified, dynamic tool discovery, FaaS scalability, MoE integration [1]	Requires full lifecycle governance (16 activities)	Effectiveness
Functional	Semantic context awareness, dynamic data	Poor LLM compliance and proactivity with tool	Proactivity, Compliance

	interpretation , Autonomy [6]	use	
Operational	Enhanced reproducibility, non-intrusive asset management [2]	Significant computational cost/overhead incurred	Overhead
Security	Integration of external threat intelligence (MITRE ATT&CK, CVE)	Exposure to 16 distinct threat scenarios (e.g., malicious developers)	N/A

5. Trajectory III: Advanced Agentic Workflow Orchestration

The utility of MCP is most acutely demonstrated in the design of robust, next-generation agentic workflows. Contemporary agentic AI systems often encounter bottlenecks related to navigating large sets of tools, executing complex, multi-step operations, and maintaining long-context history to mitigate hallucinations. While techniques like Chain-of-Thought (CoT) and ReAct provide partial solutions, a protocol-level solution like MCP is necessary for systemic reliability.

The AgentX Pattern and FaaS Integration

In response to these scaling challenges, researchers defined a novel agentic workflow pattern known as **AgentX**. This workflow is structurally composed of three specialized components: a stage designer, a planner, and an executor agent. The AgentX pattern integrates FaaS-hosted MCP services to manage tool interactions, enabling it to achieve performance that is competitive with or superior to existing state-of-the-art patterns, including ReAct and Magentic One. Empirical evaluations confirmed that utilizing FaaS-hosted MCP tools provides critical operational benefits related to success rate, latency, and cost for practical applications, thereby establishing a blueprint for orchestrating resilient agentic systems.

A Unified Framework for Context-Aware Autonomy

Beyond specific workflow patterns, MCP is identified as the central component in a proposed unified framework combining Generative AI, the Model Context Protocol, and Applied Machine Learning (ML).[6] This synergistic combination provides the means to power advanced agentic AI systems that are designed for autonomous decision-making and continuous self-improvement over time.[6]

In this unified vision, Generative AI facilitates the creation of new data and insights. However, the Model Context Protocol is the indispensable element that guarantees the AI models are **contextually aware** and can maintain functional integrity across diverse, dynamic environments and conditions.[6] Applied Machine Learning contributes the necessary component of real-world deployment and learning, enabling agents to adapt and make intelligent decisions instantaneously. This integrated framework, where MCP bridges generated intelligence with applied realism, is crucial for high-stakes domains such as healthcare, finance, and robotics, where adaptive decision-making is a primary operational requirement.[6]

6. Trajectory IV: Cross-Domain Specialization and Standardization

The versatility and architectural integrity of MCP are evident in its specialization across vastly different domains, from life sciences to critical infrastructure cybersecurity.

MCPmed: Operationalizing FAIR Principles in Biomedical Research

In bioinformatics, traditional web servers (e.g., GEO, STRING, UCSC Cell Browser) are primarily engineered for human users, making them difficult for LLMs and deep research agents to read and utilize autonomously. This human-centric design limits the potential for automated scientific discovery. To overcome this limitation, the community effort known as **MCPmed** proposes adapting MCP to bioinformatics web server backends.

By implementing MCPmed, web services gain a standardized, machine-actionable layer that explicitly links service endpoints with scientific concepts and their detailed metadata. This structured transition significantly enhances automation, improves reproducibility, and ensures interoperability. Fundamentally, MCP's adaptation in bioinformatics operationalizes the **FAIR (Findable, Accessible, Interoperable, Reusable) principles** for AI systems. LLM-driven research discovery hinges on automated access to structured data, and MCP provides the architectural mandate necessary to transition legacy infrastructure into accessible data resources for next-generation research agents.[2]

MCP in Critical Infrastructure Asset Discovery

The Model Context Protocol demonstrates equivalent value in the high-stakes domain of Critical Infrastructure (CI) cybersecurity. Asset discovery in Industrial Control Systems (ICS) requires deep contextual reasoning because traditional methodologies rely on deterministic tools that use fixed fingerprinting strategies and often fail to adapt to the heterogeneous, dynamic architectures characteristic of modern CI systems.[5]

An advanced architecture based on a Mixture of Experts (MoE) leverages LLMs to interpret and decode multi-protocol communications and correlate heterogeneous data sources. Within this architecture, MCP ensures the system's capacity for context awareness by facilitating the integration of external threat intelligence feeds (MITRE ATT&CK, MISP, CVE). This contextual linkage allows the LLM to generate explanations for discovery outcomes and produce adaptive discovery queries for novel devices, thereby supporting the development of non-intrusive asset management solutions and materially strengthening the cybersecurity posture of critical infrastructure systems.

Table 3: MCP's Role in Domain-Specific Challenges

Domain	**Pre-MCP Limitation**	**MCP Solution/Framework**	**Core MCP Function**
Bioinformatics/Research	Human-centric web servers limiting LLM readability	MCPmed community effort, lightweight breadcrumbs	**Interoperability:** Machine-actionable access to scientific data [7]
Critical	Deterministic tools	LLM-based MoE	**Context Injection:**

Infrastructure (ICS)	lacking contextual reasoning for asset ID	integrating threat intelligence via MCP	Linking real-time threat data (MISP, CVE) to operational assets

7. Google Scholar Abstract Collection: Model Context Protocol (MCP) in Markdown

The following collection presents high-impact academic abstracts related to the Model Context Protocol, addressing its role in LLM integration, agentic systems, and tool use, formatted strictly in Markdown as required.

Model Context Protocol (MCP): Landscape, Security Threats, and Future Research DirectionsAuthor(s): XINYI HOU, YANJIE ZHAO, SHENAO WANG, HAOYU WANG

Abstract:

> The Model Context Protocol (MCP) is an emerging open standard that defines a unified, bi-directional...source

---

AgentX: Towards Orchestrating Robust Agentic Workflow Patterns with FaaS-hosted MCP ServicesAuthor(s): Shiva Sai Krishna Anand Tokal, Vaibhav Jha, Anand Eswaran, Praveen Jayachandran, Yogesh Simmhan

---

Abstract:

> Generative Artificial Intelligence (GenAI) has rapidly transformed various

fields...source agentic workflows.

Help or Hurdle? Rethinking Model Context Protocol-Augmented Large Language ModelsAuthor(s): Wei Song, Haonan Zhong, Ziqi Ding, Jingling Xue, Yuekang Li

---

Abstract:

The Model Context Protocol (MCP) enables large language models (LLMs) to access...source tool-augmented LLMs.

MCPmed: A Call for MCP-Enabled Bioinformatics Web Services for LLM-Driven DiscoveryAuthor(s): Matthias Flotho, Ian Ferenc Diks, Philipp Flotho, Leidy-Alejandra G. Molano, Pascal Hirsch, Andreas Keller

---

Abstract:

Bioinformatics web servers are critical resources in modern biomedical research, facilitating interactive exploration of datasets through custom-built interfaces with rich visualization capabilities. However, this human-centric design limits machine readability for large language models (LLMs) and deep research agents. We address this gap by adapting the Model Context Protocol (MCP) to bioinformatics web server backends—a standardized, machine-actionable layer that explicitly associates webservice endpoints with scientific concepts and detailed metadata. Our implementations across widely-used databases (GEO, STRING, UCSC Cell Browser) demonstrate enhanced exploration capabilities through MCP-enabled LLMs. To accelerate adoption, we propose MCPmed, a community effort supplemented by lightweight breadcrumbs for services not yet fully MCP-enabled and templates for setting up new servers. This structured transition significantly enhances automation, reproducibility, and interoperability, preparing bioinformatics web services for next-generation research agents.

Integrating Generative AI and Model Context Protocol (MCP) with Applied Machine Learning for Advanced Agentic AI SystemsAuthor(s): Nilesh Bhandarwar

---

Abstract:

Generative AI, Model Context Protocol (MCP), and Applied Machine Learning...source is important.

---

8. Synthesis of Research Findings and Future Standardization Challenges

The analysis confirms that the Model Context Protocol represents a crucial architectural shift required for the reliable, scaled deployment of tool-augmented AI systems. The rapid, synchronous emergence of foundational papers in 2025 demonstrates that MCP is an immediate, practical necessity for overcoming the limits of first-generation agentic systems, particularly their struggles with complex multi-step tasks and maintaining long-context history.

MCP defines the standard that formalizes LLM-tool interaction, moving past simple API calls to a structured, bi-directional protocol that is essential for robust, deployable systems like the AgentX workflow pattern and MoE frameworks.[3] Furthermore, the architectural design of MCP, demonstrated by its immediate consideration of FaaS deployment , indicates that the protocol was developed with enterprise scalability, cost optimization, and microservices integration as core requirements.

The long-term viability of MCP hinges on successfully navigating two dominant, interconnected challenges:

The Dual Imperatives: Security and Efficiency

1. **Security Risk Management:** The foundational documentation of MCP proactively

established a comprehensive threat taxonomy. This analysis categorized security and privacy risks across 16 distinct scenarios and four attacker types, mandating that the adoption of MCP requires rigorous implementation of security safeguards tailored to each stage of the server lifecycle. The protocol introduces an expanded attack surface, and its secure deployment requires organizations to rigorously manage the trust boundaries inherent in external execution environments.

2. **Efficiency and Model Compliance:** The empirical data provided by MCPGAUGE shows that the protocol's theoretical benefits are often undercut by practical performance limitations. High computational overhead and deficiencies in LLM proactivity and compliance reveal that the primary barrier to effective tool-augmented AI is not solely the communication standard itself, but the difficulty in tuning LLMs to utilize the external context correctly and efficiently. This implies that developers must focus heavily on optimizing model training specific to MCP adherence to mitigate computational cost and ensure accurate tool invocation.

Path to Sustainable Growth

The success of domain-specific implementations, such as MCPmed in bioinformatics and the MoE integration in Critical Infrastructure cybersecurity , confirms the protocol's adaptability. The application in bioinformatics is particularly significant as MCP effectively acts as the technical standard enabling LLM discovery within legacy, human-centric data infrastructure, thereby operationalizing the FAIR principles for AI systems.[2]

Future research and development efforts must continue to strengthen MCP's standardization across diverse applications, refine trust boundaries within execution environments, and focus heavily on methods for optimizing LLM performance to address the compliance and overhead issues identified by MCPGAUGE. The sustainable growth of tool-augmented AI is directly correlated with the maturity and robustness of the Model Context Protocol.

Alıntılanan çalışmalar

1. Model Context Protocol (MCP): Landscape, Security Threats ... - arXiv, erişim tarihi Ekim 13, 2025, https://arxiv.org/abs/2503.23278
2. MCPmed: A Call for MCP-Enabled Bioinformatics Web ... - arXiv, erişim tarihi Ekim 13, 2025, https://arxiv.org/abs/2507.08055
3. AgentX: Towards Orchestrating Robust Agentic Workflow ... - arXiv, erişim tarihi Ekim 13, 2025, https://arxiv.org/abs/2509.07595
4. Help or Hurdle? Rethinking Model Context Protocol-Augmented ..., erişim tarihi Ekim 13, 2025, https://arxiv.org/abs/2508.12566
5. Asset Discovery in Critical Infrastructures: An LLM-Based Approach - MDPI, erişim tarihi Ekim 13, 2025, https://www.mdpi.com/2079-9292/14/16/3267

6. Integrating Generative AI and Model Context ... - ResearchGate, erişim tarihi Ekim 13, 2025, https://www.researchgate.net/profile/Nilesh-Bhandarwar/publication/395238999_Integrating_Generative_AI_and_Model_Context_Protocol_MCP_with_Applied_Machine_Learning_for_Advanced_Agentic_AI_Systems/links/68b895e1d9261f6f51b124fa/Integrating-Generative-AI-and-Model-Context-Protocol-MCP-with-Applied-Machine-Learning-for-Advanced-Agentic-AI-Systems.pdf
7. MCPmed: A Call for MCP-Enabled Bioinformatics Web Services for LLM-Driven Discovery, erişim tarihi Ekim 13, 2025, https://arxiv.org/html/2507.08055v1
8. Dr. Luigi Coppolino | Author | Parthenope University of Naples, 80133 Naples, Italy - SciProfiles, erişim tarihi Ekim 13, 2025, https://sciprofiles.com/profile/1411308?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name