Communication

Conceptualising RAG-Driven Agentic AI with Multi-Layer MCP for Seismic Structural Systems

Carlos Ávila * and Edgar Rivera

Posted Date: 5 December 2025

doi: 10.20944/preprints202512.0534.v1

Keywords: agentic AI in structural engineering; multi context protocol; agents; retrieval augmented generation

 doi:10.20944/preprints202512.0534.v1

Communication

Conceptualising RAG-Driven Agentic AI with Multi-Layer MCP for Seismic Structural Systems

Carlos Ávila * and Edgar Rivera

Mecánica Computacional e Inteligencia Artificial Aplicada (MCIAA), Ingeniería Civil, Facultad de Ciencias, Ingeniería y Construcción, Universidad UTE, Quito 170527, Ecuador

* Correspondence: carlos.avila@ute.edu.ec; Tel.: +593-999261838

Abstract

The integration of Generative AI into civil engineering is currently constrained by the susceptibility of Large Language Models (LLMs) to hallucination and their inherent lack of physics-based knowledge. To address these limitations, this paper presents a conceptual framework for the integration of Agentic Artificial Intelligence (AI) into the complete lifecycle of seismic-resistant structural engineering. The proposal employs a modular software architecture built on the Model Context Protocol (MCP), enabling distributed collaboration among specialised AI agents across six critical stages: (1) seismic hazard assessment, (2) structural modelling and analysis, (3) design and optimisation, (4) construction quality control, (5) structural health monitoring (SHM), and (6) ethical audit and explainability. In this architecture, agents operate as autonomous MCP Clients within a standardised context, orchestrating workflows by communicating directly with deterministic MCP Servers and the human user. This structure strictly manages tool execution through synchronous, verifiable MCP calls, ensuring that stochastic agentic reasoning remains decoupled from immutable numerical execution. By grounding generative outputs in physics-based engines and Retrieval-Augmented Generation (RAG), the framework ensures traceable reasoning, transparency, and professional accountability, offering a pathway for the ethical deployment of AI systems in civil and structural engineering.

Keywords: agentic AI in structural engineering; multi context protocol; agents; retrieval augmented generation

1. Introduction

The Architecture, Engineering, and Construction sector currently faces a dichotomy: the urgent need for integrated project delivery versus the reality of deep technological isolation [1]. This struggle is rooted in the widespread use of proprietary, specialised tools—such as CAD, FEM, and BIM —that operate on fixed procedural logic. While this fragmentation is problematic, it is sustained by the industry's historical imperative for deterministic methods [2], which provide the verifiable outcomes necessary for safety, regulatory compliance, and liability management [3]. These factors converge to establish a foundational definition of our current toolkit: deterministic engineering software is inherently characterised by its reliance on predefined, immutable logic.

Unfortunately, this rigid characterisation creates operational friction. Because these systems force information to flow in a linear, sequential manner [4], they lack the capacity for autonomous adaptation; consequently, any deviation or need for complex data interpretation necessitates significant, often manual, expert intervention [3,5]. Reliance on customised integration for specialised tools creates a brittle architecture, where complex logic demands high maintenance whenever systems change. Recognising this systemic fragility necessitates a fundamental evolution in traditional methodology. Therefore, the essential strategy requires orchestrating a cohesive, automated ecosystem that facilitates intelligent reasoning across core engineering platforms while strictly maintaining a 'Human-in-the-Loop' to guarantee process fidelity. This approach effectively

 doi:10.20944/preprints202512.0534.v1

eliminates the bottlenecks that currently delay civil engineering projects, establishing a robust foundation for advanced computational workflows.

To operationalise this cohesive link and transcend the limitations of linear workflows, the discipline must look beyond conventional tools. Consequently, the application of computational intelligence within Civil Engineering requires a rigorous distinction between conventional Machine Learning (ML) methodologies and the emerging paradigm of Agentic Artificial Intelligence (AI) [6,7]. This distinction characterises the autonomous agent not merely as a passive predictive model, but as a dynamic system interacting within its digital environment, capable of sensing conditions and executing goal-driven actions to influence specific outcomes. When this goal-oriented definition acts in concert with the advanced reasoning of foundational generative AI, it will enable these systems to operate with genuine autonomy, executing complex, multi-step tasks with proactive flexibility rather than merely responding to user prompts [6].

However, the deployment of the proposed Agentic AI necessitates a "Human-in-the-loop" paradigm to bridge computational autonomy with professional accountability. By synthesising the capabilities for self-direction and reflection inherent in Large Language Models (LLMs) [8], Agentic AI assumes the role of "Central Coordinator" within Multi-Agent Systems (MAS) frameworks [9] ; yet, this agentic coordination is most effective when anchored by human oversight to ensure technical validity and ethical alignment. This symbiotic structure enables autonomous agents to augment intricate workflows and mitigating risk [10], while preserving the engineer's role as the ultimate arbiter. While this operational independence introduces the acute challenge of establishing clear safety criteria for agent failures [11], the integration of Human-in-the-loop protocols confirms that Agentic AI constitutes the fundamental shift: moving the field from systems focused solely on prediction towards those capable of supervised, goal-directed management. (International Telecommunication Union 2025). While Large Language Models offer exceptional capabilities in natural language inference, their inherent reliance on static training data—characterised by fixed cut-off dates—often results in 'hallucinations' when processing fragmented or detailed queries. These constraints render unaugmented models insufficient for high-stakes engineering applications [12]. To address this fundamental limitation, the field has adopted Retrieval-Augmented Generation (RAG), defined as an integrative framework that couples the generative LLM with an efficient information retrieval system [12,13]. This architecture is specifically designed to augment the LLM's internal knowledge base with external, current, and domain-specific data, enabling effective performance where pre-trained data is typically incomplete [12]. By integrating authoritative documents from domain-specific repositories—such as seismic design codes [14–16], historical ground motion records [17], and experimental hysteresis data [18]—before generating a response, the system effectively mitigates the critical risk of hallucinations. This synthesis of data augmentation and error mitigation underpins the current architectural transition toward **Agentic RAG** systems; this shift reflects the reality that earthquake engineering workflows, such as performance-based assessments and non-linear analysis, require autonomous agents capable of complex planning and environmental interaction, extending far beyond simple text generation [8]. However, given the catastrophic consequences of structural failure in seismically active regions, this autonomy must be governed by a 'Human-in-the-loop' protocol. In this paradigm, the structural engineer serves as the definitive validation node, anchoring the agent's probabilistic reasoning to deterministic seismic provisions and ethical liability. Consequently, the most significant contribution of RAG in this context is its mechanism for factual grounding: by compelling the LLM to derive its outputs directly from specific regulatory clauses and validated spectral data, the framework drastically improves the accuracy and reliability of technical judgements compared to unaugmented generative models.

(Ahmad 2025)Despite the remarkable advances in AI, the core challenge of LLMs in engineering remains a lack of inherent physics-based knowledge, leading to what is termed "Recursive Hallucination" and "Structural Distortion" [19]. Essentially, LLMs can generate plausible-looking, yet structurally unsound, designs. To address this gap, a fundamental methodological reorganisation was proposed in late 2024 with the introduction of the Model Context Protocol (MCP), an open

 doi:10.20944/preprints202512.0534.v1

standard designed to harmonise the interface between LLMs and external, domain-specific data sources [20]. This protocol functions as a universal "socket"—analogous to a USB-C port for AI—enabling autonomous agents to dynamically discover and execute workflows, read resources, and utilise tools across disparate computational systems [21].

It is important to understand that the literature defines MCP not as a proprietary application, but as a communication protocol that standardises the interaction between an "AI Host" and the external world. This architecture is structurally underpinned by a client-host-server model, facilitating a standardised information exchange via JSON-RPC 2.0 messages [22]. This tripartite structure is fundamentally critical, as it effectively decouples the abstract reasoning capabilities of the AI from the domain-specific, immutable logic of engineering tools.

A recent study, supported by a dedicated dataset for validation, successfully demonstrated that MCP is a feasible and robust method of connection between an LLM and an external engineering API, specifically OPENSEESPY, enabling complex structural analysis via a structured CIDI prompt [23,24]. Therefore, the MCP represents a foundational structure and systemic definition of Civil Engineering applications, moving beyond mere generative capacity to verifiable, physics-informed execution.

While the individual utility of these technologies is recognised, the current state-of-the-art—as synthesised in recent reviews by Elsisi (2025) [25]—remains constrained by a 'linear pipeline' architecture. In this prevailing paradigm, AI functions primarily as a 'Copilot' or task-specific assistant, where the Model Context Protocol is utilised largely as a connectivity interface to facilitate conversation. Crucially, these existing frameworks rely heavily on post-hoc 'Human-in-the-Loop' validation to mitigate inevitable hallucinations. However, it is argued that relying on human oversight to police stochastic errors is insufficient for the strict liability requirements of seismic infrastructure.

To transcend these limitations, this paper proposes a unified orchestration framework that synthesises three critical technologies into a cohesive cognitive architecture. We argue that Agentic AI must operationalise the predictive and generative capabilities of LLMs, elevating them to a "Central Coordinator" role that provides intent and strategic planning. Simultaneously, Retrieval-Augmented Generation must be integrated to enforce strict regulatory grounding, mitigating hallucinations. Crucially, we introduce the Model Context Protocol as the foundational "USB-C for AI"—a standardised interface that decouples abstract reasoning from the immutable logic of engineering tools. By harmonising these elements, the framework moves the discipline from static, linear workflows to dynamic, physics-informed execution, effectively closing the validation gap required for autonomous civil infrastructure.

2. Materials and Methods

The system integrates seven layers organised according to the Model Context Protocol (Figure 1). At the top, the User & Interface Layer captures engineering intent and delivers results and explanations. The Agentic AI Layer interprets user goals (given in natural language, simplifying the user-machine interactions) and orchestrates multi-step workflows across domain agents. The Agents Layer, comprising specialised MCP clients (Hazard, Structural, Design, Quality Assurance, SHM, and Audit & Ethics Agents), translates high-level plans into structured tool calls and interprets computational responses. Communication with deterministic numerical engines is mediated by the MCP Gateway Layer, which ensures schema consistency, authentication, and routing, and by the Event Bus Layer, which broadcasts asynchronous events enabling reactive and concurrent operations. The MCP Server Layer contains the deterministic computational engines responsible for hazard analysis, structural simulation, code-compliant design, construction quality verification, structural health monitoring, and audit functions. At the bottom, the External Data Input Layer ingests as-built geometry, sensor streams, and traceability logs, supporting continuous life-cycle updates to models and decisions.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 December 2025 doi:10.20944/preprints202512.0534.v1

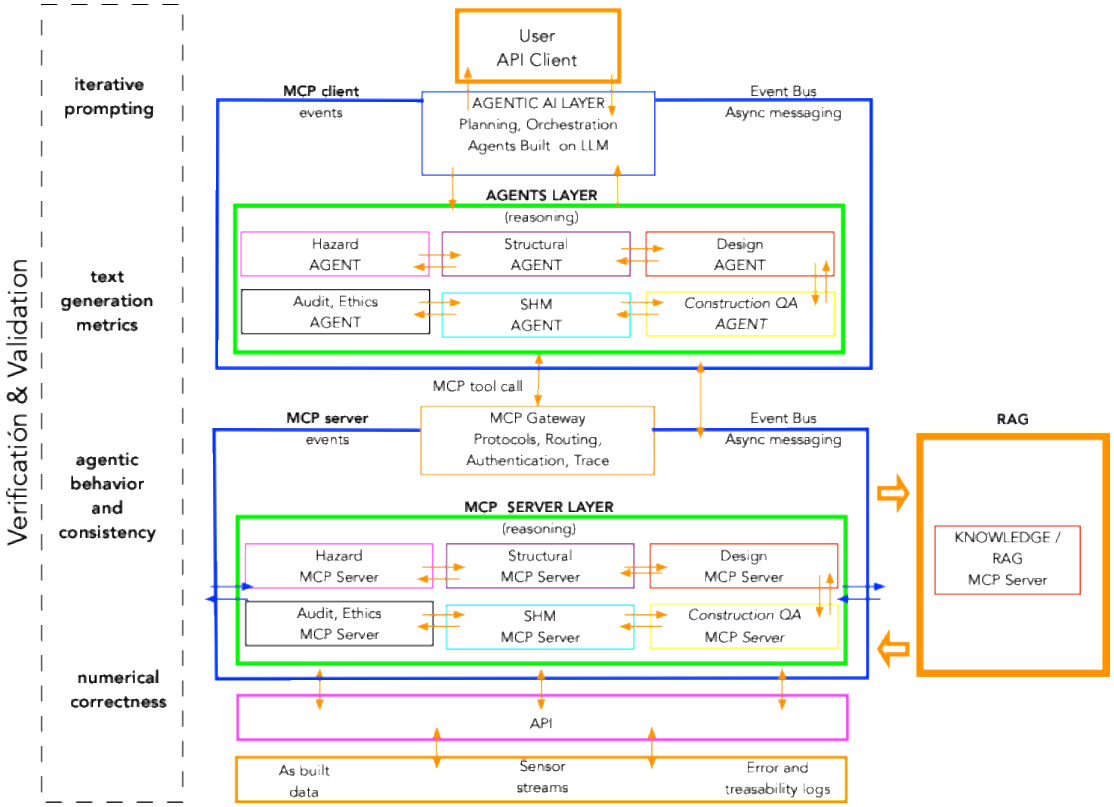

Figure 1. Layered Agentic AI architecture for end-to-end structural seismic engineering.

2.1. The Model Context Protocol Paradigm

The proposed framework adopts the Model Context Protocol to address the fundamental incompatibility between stochastic Large Language Model reasoning and the rigorous determinism required in seismic engineering. By enforcing a standardised client–server paradigm, the architecture strictly segregates cognitive planning from numerical execution, ensuring that the system functions as a reliable engineering tool rather than an uncontrolled generative model.

2.1.1. Client–Server Dichotomy

The architecture is defined by a strict functional separation between two layers:

- **The Client (Reasoning Layer):** The "Agents Layer" operates as the MCP client, comprising domain-specific agents (e.g., Hazard, Structural, Design) driven by LLM reasoning (Figure 1). These agents function as the system's orchestrators and are the sole initiators of requests. They utilise high-level cognition to decompose complex objectives and determine ***when*** to execute specific tasks—such as deciding to run a spectral analysis—without performing the calculations themselves.
- **The Server (Computational Layer):** The "MCP Server Layer" consists of stateless, deterministic engines that expose validated engineering tools (e.g., `run_response_spectrum`, `check_aci318`). These servers lack agency and reasoning capabilities; they strictly execute algorithms upon request and return structured, machine-interpretable outputs.

2.1.2. Safety and Determinism

This separation provides a robust mechanism for preventing "hallucinations" in safety-critical workflows. By routing all computations through the MCP Gateway to deterministic servers, the framework ensures that every structural demand, capacity check, or safety decision originates exclusively from validated physics-based algorithms. Consequently, the agents reason about the

 doi:10.20944/preprints202512.0534.v1

engineering process using textual logic, but the numerical ground truth is never generated by the LLM, thereby guaranteeing traceability and reproducibility.

2.2. Multi-Layer Architectural Topology

The proposed framework (Figure 1) establishes a hierarchical topology comprising seven distinct strata, rigorously organised under the Model Context Protocol to decouple stochastic LLM reasoning from deterministic engineering verification.

At the summit, the **User & Interface Layer (Layer 1: Human-in-the-Loop Oversight)** defines the regulatory boundary, capturing high-level engineering intent whilst serving as the definitive approval gate for design alternatives and explainable outputs. Directly beneath, the **Agentic AI Layer (Layer 2: Executive Planning & Orchestration)** acts as the system's cognitive core. This layer decomposes complex, multi-stage workflows—spanning hazard assessment to structural health monitoring—managing cross-domain dependencies, based on AI reasoning, without executing direct computations.

Operational logic is delegated to the **Agents Layer (Layer 3: Domain-Specific MCP Clients)**. Comprising specialised agents for Hazard, Structural Analysis, Design, Quality Assurance (QA), SHM, and Audit, this layer translates the orchestrator's natural language plans into structured MCP tool schemas. To ensure protocol integrity, the **MCP Gateway (Layer 4: Secure Middleware & Routing)** mediates all traffic between clients and servers. It enforces strict authentication and schema validation, guaranteeing that stochastic agent requests adhere to the rigid input requirements of the numerical engines.

Simultaneously, the **Event Bus (Layer 5: Asynchronous Reactive Backbone)** employs a publish-subscribe model to broadcast system-wide triggers—such as hazard.cms.ready or qa.deviation.alert—enabling real-time cross-domain reactivity. The computational foundation resides in the **MCP Server Layer (Layer 6: Deterministic Computational Engines)**. This suite executes validated physics-based algorithms (e.g., PSHA, FEM, Code Checks) and strictly excludes hallucination. Crucially, this layer integrates a **Knowledge/RAG Server**, which retrieves contextual regulatory data (e.g., ACI 318-25 clauses) to support decision-making without altering numerical results. Finally, the **External Data Input Layer (Layer 7: Lifecycle Data Ingestion)** anchors the digital twin in physical reality by feeding as-built BIM models, sensor streams, and construction logs into the upstream servers.

2.3. Integration of Controlled Retrieval-Augmented Generation

The framework integrates RAG technology by encapsulating it within a dedicated Knowledge MCP Server, strictly adhering to the client–server architecture defined by the Model Context Protocol. Rather than functioning as an unchecked generative layer, this server is implemented as a deterministic endpoint that exposes specific tools—such as search_codes, search_qa_guidelines, and search_projects—which agents invoke via the MCP Gateway.

Technically, this integration relies on the MCP Gateway to enforce schema validation on all retrieval requests, ensuring that queries for regulatory clauses (e.g., ACI 318, ASCE 7) or historical project data are structured and secure. Upon invocation, the Knowledge Server queries a vectorised knowledge corpus and returns structured JSON containing ranked passages and source metadata, explicitly avoiding free-form conversational outputs.

This architecture establishes a rigorous Contextual vs. Computational Separation. The RAG-driven Knowledge Server is solely responsible for providing textual justification and interpretive context, while physics-based computations—such as Finite Element Method (FEM) and Probabilistic Seismic Hazard Analysis (PSHA)—are executed by isolated, stateless MCP servers. This separation prevents LLM hallucinations from corrupting numerical workflows while ensuring that every engineering decision is traceable to a specific, immutable document within the vector store.

 doi:10.20944/preprints202512.0534.v1

2.4. System Dynamics and Lifecycle Integration

The proposed framework (Figure 1) orchestrates system dynamics through a hybrid execution model that harmonises deterministic control with reactive agility, ensuring robust performance across the engineering lifecycle.

2.4.1. Synchronous Execution

The core operational mechanism relies on a standard synchronous request-response cycle managed by the MCP Gateway. In this phase, domain agents (MCP clients) initiate tool calls—such as struct-server.run_response_spectrum or design-server.check_aci318—which the Gateway validates for schema consistency before routing to the appropriate deterministic MCP server. This strict mediation ensures that all safety-critical calculations remain reproducible, authorised, and strictly isolated from potential reasoning errors inherent in the LLM layer.

2.4.2. Asynchronous Reactivity

To accommodate dynamic inputs, the architecture employs an Event Bus Layer that establishes event-driven loops distinct from the linear request cycle. This mechanism broadcasts asynchronous state changes, such as qa.deviation.alert during construction or shm.alert.damage during operation, allowing the Agentic AI to react concurrently to emerging hazards without blocking ongoing computations.

2.4.3. Lifecycle Continuity

The convergence of deterministic tool execution and reactive, event-driven monitoring unifies the project lifecycle through a shared digital thread. By ingesting as-built geometry, sensor streams, and traceability logs via the External Data Input Layer, the system links initial design assumptions with real-world Quality Assurance verification and long-term Structural Health Monitoring. This integration transforms static engineering models into a continuous, closed-loop ecosystem capable of dynamically updating assessments in response to physical degradation or seismic events.

2.5. Systemic Mitigation of Stochastic Variance in Autonomous Engineering Agents

The integration of a dedicated Validation and Verification layer is not merely an architectural enhancement but a fundamental imperative for the safe deployment of Agentic AI in structural engineering. While the proposed architecture leverages the reasoning power of LLMs, it is required to recognised that these components are inherently probabilistic. Unlike the deterministic solvers in the MCP Server Layer—which provide mathematically guaranteed results for hazard curves and spectral response—agentic outputs are susceptible to "hallucinations," omissions, and misalignment with physics-based ground truths.

Therefore, the Validation and Verification layer is essential as an independent "meta-evaluator" to bridge this reliability gap. By systematically benchmarking agentic explanations against deterministic data using established NLP metrics and "LLM-as-a-Judge" paradigms, it can be ensured textual accuracy. Furthermore, this layer is critical for validating agentic behaviour itself, verifying that autonomous agents strictly adhere to engineering workflows and safety constraints. Consequently, to transform stochastic AI reasoning into a robust professional tool, the Validation layer provides the necessary continuous monitoring and regression testing required to mitigate risk and ensure that the rigour of structural safety standards is never compromised by the fluidity of generative models.

 doi:10.20944/preprints202512.0534.v1

3. Results and Discussion

3.1. The Epistemological Shift: From Linear Prediction to Autonomous Orchestration

Historically, the Civil Engineering sector's reliance on proprietary tools has established a "linear, sequential" workflow that fundamentally lacks the capacity for autonomous adaptation [26]. Recent comprehensive reviews, such as Elsisi (2025), confirm that current AI integrations remain constrained by this paradigm, typically functioning as linear pipelines or "Copilots" that rely heavily on post-hoc human validation to mitigate stochastic errors. While this approach reduces risk, it fails to fundamentally resolve the reliability gap required for autonomous seismic infrastructure. Consequently, current integration efforts frequently result in a "brittle architecture" where point-to-point connections require significant manual maintenance. This operational friction demonstrates that the solution to industry fragmentation is not merely better software interoperability, but rather "cognitive orchestration" [27] —a move from reactive error correction to architectural prevention.

Achieving this orchestration requires a fundamental distinction between conventional Machine Learning and Agentic AI [28]. Unlike predictive ML, Agentic AI is defined as systems contextually aware and autonomously goal-directed [29]. By elevating the LLM to the role of a "Central Coordinator," this framework provides the "self-direction and reflection" necessary to decompose complex tasks without human micro-management. The convergence of this cognitive layer with traditional deterministic logic confirms that the discipline is transitioning from systems focused solely on prediction toward those capable of autonomous, goal-directed management.

3.2. The Client–Server Dichotomy as a Defense Against Recursive Hallucination

The most significant technical barrier to deploying Generative AI in structural engineering is Recursive Hallucination [30], a phenomenon where LLMs generate plausible but physically unsound designs due to a distinct deficiency in arithmetic reasoning and contextual stability [31]. Without deterministic grounding, these models function merely as statistical guessers, rendering them unsuitable for the complexity of socio-technical engineering networks. Consequently, the resolution to this "Black Box" dilemma requires rigorous contextualisation, effectively achieved through RAG frameworks that anchor the model in verified data. To operationalise this, this study proposes the MCP as the critical communication interface between the AI application and authoritative external knowledge bases—specifically design codes, standards, and technical manuals—thereby mitigating hallucinations through enforceable engineering constraints.

The primary barrier to Generative AI in structural engineering is Recursive Hallucination [30], often resulting in physically unsound designs. To resolve this, rigorous contextualisation is essential. This study employs Retrieval-Augmented Generation to mitigate hallucinations, operationalised via the MCP. MCP functions as the critical interface connecting the AI to external, authoritative sources—such as building codes and technical manuals. This architecture ensures that model outputs are not merely statistical probabilities but are grounded in verifiable engineering standards.

By adopting a strict Client–Server Dichotomy [32], the framework effectively decouples abstract reasoning from domain-specific immutable logic. The discussion validates the definition of MCP not merely as a technical standard, but as a safety mechanism. The Agents (Clients) operate in the domain of textual logic, reasoning and planning and The Engines (Servers) operate in the domain of deterministic physics.

This separation ensures that the AI is physically incapable of "guessing" a structural response; it must invoke verified tools (e.g., run_response_spectrum) to obtain a result, aligning with the integration strategies [33]. As demonstrated in recent validation studies using OPENSEESPY [23,24], this architecture ensures that whilst the *process* is orchestrated by AI, the *product* remains mathematically pure and reproducible. Furthermore, this architecture addresses the cognitive attention thresholds defined by Nielsen [34], ensuring cycle times (6–12s) that allow the engineer to maintain focus on the dialogue without context switching. Consequently, the rigorous application of MCP transforms Generative AI into the "foundational structure" for verifiable,

 doi:10.20944/preprints202512.0534.v1

physics-informed execution. While the MCP secures numerical determinism, the integration of RAG is strictly necessary to confine the Artificial Intelligence within the "ranges" of professional acceptability. Unaugmented Large Language Models [12,13] suffer from a "knowledge cut-off" and a propensity for hallucination that renders them non-compliant for high-stakes engineering.

3.3. RAG as a "Constitutional" Governance Layer

Rather than functioning merely as a conversational enhancement, the Knowledge MCP Server is positioned as a "Regulatory Governor." This reclassification is critical; Civil Engineering mandates rigorous factual accuracy and cannot tolerate the epistemic risks associated with unsupported assertions or probabilistic errors.

This architecture encapsulates RAG within a deterministic Model Context Protocol (MCP) environment, enforcing a strict logical dichotomy between Contextual Justification—sourced directly from immutable regulatory standards such as ACI 318 and NEC15—and Computational Truth, derived from physics-based Finite Element Method servers. By compelling the agent to anchor all decisions in these external, verifiable contexts, the approach effectively mitigates the risk of hallucination inherent in unaugmented models. Consequently, the framework guarantees "traceable reasoning," wherein every engineering decision—from preliminary design assumptions to final compliance checks—is inextricably linked to a verifiable document. This ensures that the system provides not just answers, but valid arguments where the premises logically support the conclusion, thereby satisfying the industry's stringent requirements for liability management and professional accountability.

3.4. Achieving Lifecycle Continuity: A Closed-Loop Event-Driven Ecosystem

The persistence of latency in feedback loops between physical assets and digital models fundamentally constrains traditional civil engineering. Current practices often render models static and unresponsive to real-time degradation, acting merely as archival snapshots rather than dynamic operational tools. Recent empirical studies confirm that traditional "information integration" fails to address the "surge of criticalities" inherent in complex infrastructure, necessitating a shift toward intelligent, low-latency processing [35]. To realise true Lifecycle Continuity, we must transition to an Event-Driven Engineering Ecosystem. This argument rests on the necessity of implementing an asynchronous Event Bus Layer. Unlike synchronous systems that idle awaiting manual input, this architecture operates autonomously using distributed, stateless microservices, effectively solving the latency problem through streamlined, non-blocking workflow execution [36].

In critical scenarios such as Structural Health Monitoring, the system detects specific telemetry events—exemplified by shm.alert.damage—and independently triggers computational re-analysis. This capacity for self-directed assessment constitutes decisive evidence that the architecture bridges the validation gap required for autonomous civil infrastructure, moving beyond passive sensing to "AIoT-enabled decentralised fault diagnosis" [37]. Therefore, because the framework resolves the latency problem through asynchronous processing, and because it enables immediate, automated reaction to seismic or degradation events, it successfully transcends traditional static limitations. It is inferred that the proposed framework transforms the digital model into a dynamic, "closed-loop ecosystem" capable of "perception-fusion-prediction" cycles. This is robust, as the capacity for real-time adaptation is a necessary condition for modern structural resilience, a condition which this event-driven architecture demonstrably satisfies.

3.5. Ethical Implications: The Engineer as "Executive Reviewer"

The integration of autonomous orchestration into structural engineering often precipitates concerns regarding human disempowerment; however, valid reasoning dictates that Agentic AI must be conceptualised as advanced instruments facilitating digital interaction rather than as professional replacements [38]. Our framework explicitly operationalises this premise by positioning

 doi:10.20944/preprints202512.0534.v1

the User & Interface Layer as the definitive approval gate. By utilising the MCP to automate the drudgery of data transfer and employing RAG to verify regulatory compliance, the system facilitates a critical professional transition. The engineer is thereby liberated from the role of a "Manual Calculator," burdened by interoperability issues, to assume the station of an "Executive Reviewer" focused on high-level design intent and ethical responsibility. Consequently, this "Human-in-the-Loop" paradigm ensures that the operational velocity required by modern autonomy does not compromise the expert judgment and strict domain expectations indispensable for safety-critical infrastructure.

3.6. Future Research Venues

To transition the proposed framework from a conceptual proposal to a verifiable industrial reality, future research must rigorously stress-test the autonomy and verifiability of its constituent layers through a series of targeted pilot cases. Specifically, the Agentic AI's capability to autonomously decompose vague retrofit goals for non-standard existing structures must be validated to ensure operation without human micro-management. Concurrently, the system's "Event-Driven" capacity necessitates testing via Real-Time QA, utilising as-built LiDAR point clouds to verify the autonomous detection of geometric deviations. Furthermore, establishing "Lifecycle Continuity" requires proving the system's ability to autonomously initiate nonlinear analysis upon detecting damage during simulated seismic events. Finally, the "Contextual Grounding" of the Knowledge Server must be confirmed by subjecting the RAG architecture to conflicting international codes, thereby testing its ability to provide traceable reasoning across distinct regulatory environments. Executing these specific validation scenarios constitutes the essential condition for establishing the framework's industrial viability

4. Conclusion

The integration of the Model Context Protocol constitutes the critical enabler for the proposed agentic AI architecture in structural seismic engineering. Given that seismic design mandates absolute determinism, traceability, and strict regulatory compliance—standards that traditional LLMs cannot independently guarantee—MCP provides the essential secure, standardised client–server foundation. By isolating domain-specific computations, such as probabilistic seismic hazard analysis and nonlinear dynamic assessments, within validated MCP servers, the framework strictly segregates reasoning from calculation. This architectural separation effectively precludes LLM hallucination from compromising numerical workflows, ensuring that all safety decisions remain grounded in deterministic algorithms. Simultaneously, MCP empowers the agentic layer to operate safely within an event-driven, closed-loop ecosystem, where the MCP Gateway and Event Bus manage asynchronous updates ranging from hazard readiness to structural health monitoring. Consequently, MCP transcends the role of an auxiliary component to function as the foundational mechanism that transforms agentic AI from a mere conversational interface into a trustworthy, auditable, and lifecycle-aware computational system indispensable for safety-critical infrastructure.

Author Contributions: Conceptualization, Carlos Avila; Formal analysis, Carlos Avila and David Rivera; Investigation, Carlos Avila and David Rivera; Methodology, Carlos Avila; Supervision, Carlos Avila; Writing – original draft, David Rivera; Writing – review & editing, Carlos Avila and David Rivera.

Funding: Please add: "This research received no external funding".

Data Availability Statement: "No new data were created or analyzed in this study. Data sharing is not applicable to this article.".

Conflicts of Interest: "The authors declare no conflicts of interest.".

 doi:10.20944/preprints202512.0534.v1

Abbreviations

ACI	American Concrete Institute
AI	Artificial Intelligence
AIoT	Artificial Intelligence of Things
API	Application Programming Interface
ASCE	American Society of Civil Engineers
BIM	Building Information Modeling
CAD	Computer-Aided Design
FEM	Finite Element Method
JSON	JavaScript Object Notation
LiDAR	Light Detection and Ranging
LLM	Large Language Model
MAS	Multi-Agent Systems
MCP	Model Context Protocol
ML	Machine Learning
NLP	Natural Language Processing
PSHA	Probabilistic Seismic Hazard Analysis
QA	Quality Assurance
RAG	Retrieval-Augmented Generation
SHM	Structural Health Monitoring

References

1. Ikudayisi AE, Chan APC, Darko A, Adedeji YMD. Integrated practices in the Architecture, Engineering, and Construction industry: Current scope and pathway towards Industry 5.0. Journal of Building Engineering 2023;73:106788. https://doi.org/10.1016/J.JOBE.2023.106788.
2. Onatayo D, Onososen A, Oyediran AO, Oyediran H, Arowoiya V, Onatayo E. Generative AI Applications in Architecture, Engineering, and Construction: Trends, Implications for Practice, Education & Imperatives for Upskilling—A Review. Architecture 2024, Vol 4, Pages 877-902 2024;4:877–902. https://doi.org/10.3390/ARCHITECTURE4040046.
3. Khodabakhshian A, Puolitaival T, Kestle L. Deterministic and Probabilistic Risk Management Approaches in Construction Projects: A Systematic Literature Review and Comparative Analysis. Buildings 2023;13. https://doi.org/10.3390/buildings13051312.
4. Gomes AM, Azevedo G, Sampaio AZ, Lite AS. BIM in Structural Project: Interoperability Analyses and Data Management. Applied Sciences (Switzerland) 2022;12. https://doi.org/10.3390/app12178814.
5. The deepset Team. AI Agents and Deterministic Workflows: A Spectrum, Not a Binary Choice | deepset Blog 2025. https://www.deepset.ai/blog/ai-agents-and-deterministic-workflows-a-spectrum (accessed November 20, 2025).
6. Ren Y, Liu Y, Ji T, Xu X. AI Agents and Agentic AI-Navigating a Plethora of Concepts for Future Manufacturing 2025.
7. Acharya DB, Kuppan K, Divya B. Agentic AI: Autonomous Intelligence for Complex Goals - A Comprehensive Survey. IEEE Access 2025;13:18912–36. https://doi.org/10.1109/ACCESS.2025.3532853.
8. Ahmad HM. Multi-Agent Retrieval-Augmented System for Domain-Specific Knowledge in Struc-tural Engineering. Aalto University, 2025.
9. Rawat A, Witt E, Lill I. A Conceptual Framework For Llm-Based Multi-Agent Systems In Construction Management. CIB W78 Conference on IT in Construction, Porto: 2025.
10. Das K, Khursheed S, Paul VK. The impact of BIM on project time and cost: insights from case studies. Discov Mater 2025;5. https://doi.org/10.1007/S43939-025-00200-2.
11. International Telecommunication Union. AI Standards for Global Impact: From Governance to Action. Geneva: 2025.
12. Jaakko Junttu. Enhancing information accessibility in infrastructure construction. Tampere University, 2025.

13. Barnett S, Kurniawan S, Thudumu S, Brannelly Z, Abdelrazek M. Seven Failure Points When Engineering a Retrieval Augmented Generation System. Proceedings of 3rd International Conference on AI Engineering Â•fl Software Engineering for AI (CAIN 2024) 2024;1.
14. ACI Committee 318. Building Requirements for Structural Concrete and Commentary. Vol. 1 (20. Farmington Hills: American Concrete Institute; 2019.
15. American Society of Civil Engineers. Minimum Design Loads and Associated Criteria for Buildings and Other Structures, ASCE/SEI 7-22. Reston: 2022.
16. CAMICON, MIDUVI. Norma ecuatoriana de la construcción - NEC: NEC-SE-MP - Mamposteria estructural. Quito: 2014.
17. Rios D, Altamirano M, Ilbay D, Tlapanco J, Rivera-Tapia D, Avila C. Beyond Prescriptive Codes: A Validated Linear-Static Methodology for Seismic Design of Soft-Storey RC Structures 2025. https://doi.org/10.20944/preprints202511.0145.v1.
18. Orakcal K, Massone LM, Ulugtekin D. A Hysteretic Constitutive Model for Reinforced Concrete Panel Elements. International Journal of Concrete Structures and Materials 2019 13:1 2019;13:51-. https://doi.org/10.1186/S40069-019-0365-9.
19. Xu B. Hallucination is Inevitable for LLMs with the Open World Assumption 2025.
20. Anthropic. Introducing the Model Context Protocol 2024. https://www.anthropic.com/news/model-context-protocol (accessed June 17, 2025).
21. Model Context Protocol. What is the Model Context Protocol (MCP)? 2025. https://modelcontextprotocol.io/docs/getting-started/intro (accessed November 24, 2025).
22. Google Cloud. What is Model Context Protocol (MCP)? 2025. https://cloud.google.com/discover/what-is-model-context-protocol (accessed November 24, 2025).
23. Avila C, Ilbay D, Rivera D. Human–AI Teaming in Structural Analysis: A Model Context Protocol Approach for Explainable and Accurate Generative AI. Buildings 2025;15. https://doi.org/10.3390/buildings15173190.
24. Avila C, Ilbay D, Tapia P, Rivera D. Toward Responsible AI in High-Stakes Domains: A Dataset for Building Static Analysis with LLMs in Structural Engineering. Data (Basel) 2025;10:169. https://doi.org/10.3390/data10110169.
25. Elsisi A. Large Language Models Application in Civil and Structural Engineering: Review. SSRN 2025. https://doi.org/10.2139/ssrn.5803822.
26. Sacks R, Eastman C, Lee G, Teicholz P. BIM Handbook. Wiley; 2018. https://doi.org/10.1002/9781119287568.
27. Wang L, Ma C, Feng X, Zhang Z, Yang H, Zhang J, et al. A survey on large language model based autonomous agents. Front Comput Sci 2024;18:186345. https://doi.org/10.1007/s11704-024-40231-1.
28. Xi Z, Chen W, Guo X, He W, Ding Y, Hong B, et al. The rise and potential of large language model based agents: a survey. Science China Information Sciences 2025;68:121101. https://doi.org/10.1007/s11432-024-4222-0.
29. Franklin S, Graesser A. Is It an agent, or just a program?: A taxonomy for autonomous agents, 1997, p. 21–35. https://doi.org/10.1007/BFb0013570.
30. Ismayilzada M, Paul D, Montariol S, Geva M, Bosselut A. CRoW: Benchmarking Commonsense Reasoning in Real-World Tasks. The 2023 Conference on Empirical Methods in Natural Language Processing, 2023. https://doi.org/10.48550/arXiv.2310.15239.
31. Yang X, Chen B, Tam Y-C. Arithmetic Reasoning with LLM: Prolog Generation & Permutation. 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Mexico: 2024.
32. Ray PP, Pratim PR. A Survey on Model Context Protocol: Architecture, State-of-the-art, Challenges and Future Directions 2025. https://doi.org/10.36227/techrxiv.174495492.22752319/v1.
33. Liang H, Kalaleh MT, Mei Q. Integrating Large Language Models for Automated Structural Analysis 2025. https://doi.org/https://doi.org/10.48550/arXiv.2504.09754.
34. Nielsen J. Usability Engineering | Enhanced Reader. 1st ed. San Francisco: Morgan Kaufmann Publishers Inc.; 1994.

 doi:10.20944/preprints202512.0534.v1

35. Wang K, Zhou X, Guan J. The construction of an integrated cloud network digital intelligence platform for rail transit based on artificial intelligence. Sci Rep 2025. https://doi.org/10.1038/s41598-025-29732-6.
36. Bertozzi N, Geraci A, Bergamasco L, Ferrera E, Pristeri E, Pastrone C. A Distributed Event-Orchestrated Digital Twin Architecture for Optimizing Energy-Intensive Industries. Proceedings of the 10th International Conference on Internet of Things, Big Data and Security, SCITEPRESS - Science and Technology Publications; 2025, p. 337–44. https://doi.org/10.5220/0013364400003944.
37. Chillon Geck C, Al-Zuriqat T, Elmoursi M, Dragos K, Smarsly K. AIoT-enabled decentralized sensor fault diagnosis for structural health monitoring. 11th European Workshop on Structural Health Monitoring, EWSHM 2024, NDT.net; 2024. https://doi.org/10.58286/29577.
38. Shneiderman B. Human-Centered AI: A New Synthesis. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 2021;12932 LNCS:3–8. https://doi.org/10.1007/978-3-030-85623-6_1.