

The Model Context Protocol in multifamily: from chatbots to governed agents

Your property management system vendor now ships an AI connector. Your leasing vendor has one too. Neither tells you what the agent is allowed to do once it is inside, who is accountable when it offers a concession it should not have, or how you would prove to a Fair Housing examiner that it treated two applicants the same way. This paper is about the layer that answers those questions.



Elizabeth Braman, CEO and Co-founder, Revolution RE

18-minute read · 49 sources · Architectural patterns, cross-industry lessons, and a market engagement framework

AT A GLANCE

Five things to know before the next AI demo

1

The Model Context Protocol has become the integration standard for AI agents.

Anthropic open-sourced it in late 2024; OpenAI adopted it in March 2025 and Google DeepMind in April 2025; in December 2025 it moved to the Linux Foundation's Agentic AI Foundation.^{2,6,9} By early 2026 the public ecosystem had passed 17,000 published servers.³

2

The protocol is a pipe, not a policy. The specification deliberately omits an authorization framework, so an unmediated server exposes internal systems, sprawls across the network, and leaves no central audit trail.^{1,26}

3

Regulated industries have already found the failure modes. Hospitality saw agents manipulated into unapproved rates; banking offloaded arithmetic to deterministic code and added confirmation gates; healthcare enforced "minimum necessary" access at the protocol layer.^{35,36,33}

4

Multifamily's exposure is statutory, not just operational. An agent with unconstrained discretion over leasing terms collides with the Fair Housing Act, with antitrust scrutiny of algorithmic pricing, and with fee-transparency rules.⁵

5

The answer is architecture. A gateway in front of every server, deterministic schemas on every high-consequence tool, programmatic confirmation gates on financial writes, and a four-phase rollout from read-only context to transactional write-back.^{22,27,5}

17,000+

published MCP
servers by early
2026³

30%

of enterprise
software vendors
expected to ship
native MCP
servers (Forrester
prediction)¹³

12

core threat
categories in the
CoSAI MCP
security
framework²²

4

rollout phases
from read-only
discovery to core
write-back

ABOUT THE RESEARCH

This paper is a desk study of 49 public sources: the protocol specification and its governance announcements, vendor documentation, analyst summaries from Gartner and Forrester as reported by secondary sources, the Coalition for Secure AI's security framework, academic preprints, and trade coverage from hospitality, banking, healthcare, and multifamily. Figures are as reported by the cited sources in October 2026 and have not been independently audited. No customer data was used.

CHAPTER 1

Enterprise AI is moving from answering questions to taking actions

Enterprise artificial intelligence is in the middle of a structural shift from conversational, retrieval-augmented generation ("RAG") toward autonomous, agentic task execution.¹ A large language model ("LLM") in this paradigm is no longer a passive engine that summarizes text. It acts as a reasoning controller that queries production databases, invokes specialized software tools, and executes transactions across distributed systems.³

The thing that held agentic architectures back was never the model. It was the integration layer.⁶ Bespoke point-to-point connections, vendor-specific software development kits, and fragile webhook chains meant that every model iteration needed its own costly engineering harness for every data silo it touched.⁷

Anthropic open-sourced the Model Context Protocol (“MCP”) in late 2024 as an open standard for how AI clients share context with, and invoke tools on, external systems.⁶ Within months it went from an emerging project to an industry standard: OpenAI added native support in March 2025, Google DeepMind followed in April 2025, and in December 2025 Anthropic, Amazon Web Services, Microsoft, Google, Cloudflare, Bloomberg, and Block contributed the protocol to the newly chartered Agentic AI Foundation (“AAIF”) under the Linux Foundation, giving it long-term institutional neutrality.^{2,7,9} Gartner has described MCP as a capability that transforms enterprise analytics and autonomous workflows, while warning engineering leaders to put rigorous controls around stability, governance, and security.^{10,12} Forrester has projected that thirty percent of enterprise software vendors will deploy native MCP servers and that half of enterprise resource planning suites will launch dedicated governance modules for agentic traffic.¹³

So what does any of that have to do with an apartment portfolio? Quite a lot. Multifamily operations are fragmented across closed, proprietary property management systems (“PMS”) — predominantly Yardi Voyager, RealPage OneSite, Entrata, and AppFolio — alongside separate platforms for customer relationship management (“CRM”), smart-building access, maintenance ticketing, and revenue management.⁷ Conversational leasing assistants are widely deployed, but they have operated at an informational level: answering FAQs and collecting contact details without modifying core systems.⁵

MCP is what lets those assistants advance from informational chat to action: scheduling tours, validating applications, pricing units within approved bounds, and coordinating vendor work orders.⁵ And that is exactly where the risk lives. Multifamily housing is governed by federal statute, is under active scrutiny for algorithmic pricing, and is subject to state tenant-privacy law; an unconstrained autonomous action is a legal and financial liability, not just a product bug.⁵

The protocol decides whether an agent *can* reach your systems. **Governance decides what it is allowed to do when it gets there.** Treating MCP as a control plane rather than a data pipe is the whole argument of this paper.

CHAPTER 2

What the protocol does, and what it deliberately leaves out

Three participants, four primitives

MCP runs on a decoupled client-server architecture over JSON-RPC 2.0.¹² The specification confines itself to the standardized transport of context and tool invocations; it says nothing about how a model

reasons or manages its context window.⁴ There are three participants: the **MCP Host**, the user-facing AI runtime or orchestration engine; the **MCP Client**, the component inside the host that holds connections to servers and translates model intent; and the **MCP Server**, an independent program that exposes backend services, datasets, and executable business logic.¹⁷

The data layer defines four primitives.^{4,17}

✓ **Tools** — executable functions a model can discover, inspect, and invoke to cause side effects or run computations.

✓ **Resources** — read-only context, addressed by URI, that lets a model read records, templates, or logs without triggering a mutation.

✓ **Prompts** — parameterized, reusable workflow templates hosted on the server to guide a model through repeatable processes.

✓ **Elicitation and sampling** — the two-way extensions: a server can pause to request missing human input or explicit approval, and can request a model completion through the connected client.

Transports cover both local and networked environments. Standard input/output serves high-speed local process communication; distributed deployments use Streamable HTTP, which replaced the earlier Server-Sent Events implementation to improve security, session control, and firewall compatibility.^{12,21}

From $M \times N$ to $M + N$

Before MCP, connecting three agent applications to ten internal systems meant building, testing, and maintaining thirty custom connectors, and any schema change downstream or model change upstream broke them.^{7,8} With MCP each system exposes its capabilities once, through a server that advertises its schema in machine-readable metadata, and any compliant client can introspect and call it without custom client-side code.^{2,8} The practical consequence for an owner is an architectural boundary: the intelligence layer can be swapped without re-engineering the systems of record.¹

Dimension	Traditional REST APIs	Bespoke point-to-point AI integrations	Governed enterprise MCP gateway
Integration complexity	Linear: endpoints configured per system ⁴	Combinatorial sprawl of proprietary connectors ⁸	Decoupled abstraction, universal interfaces ⁸
Capability discovery	Manual inspection of OpenAPI schemas ⁴	Static parameter prompting that consumes the token window ⁴	Dynamic runtime introspection via <code>tools/list</code> and <code>resources/list</code> ⁴
Execution validation	Deterministic parameter checking ⁴	Inconsistent formatting, prone to hallucination ¹²	Strict JSON schema enforcement with rich error semantics ⁴
Identity and access	Static API keys or per-user OAuth 2.0 tokens ²²	Monolithic service credentials shared across workflows ¹	Federated OAuth 2.1, SPIFFE identities, RFC 8693 down-scoped token exchange ¹²
Governance and audit	HTTP access logs without agent intent ²²	Fragmented logs that cannot trace multi-step reasoning ²²	Centralized OpenTelemetry traces linking intent, reasoning, and outcome ²²

The gateway pattern

Here is the part the demos skip. The open specification intentionally omits a native authorization framework, so a direct client-to-server connection in a production network produces endpoint sprawl, exposes internal targets, and has no central audit.^{1,26} Enterprise architectures resolve this with an **MCP Gateway**, a reverse proxy between AI hosts and backend servers, a pattern championed by cloud-native proxies such as Envoy AI Gateway, Kong AI Gateway, and the AAIF's AgentGateway project.^{27,29,30} The gateway is a Zero Trust control plane that does five jobs:²⁷

01

Centralized identity federation

Authenticates agent requests against the enterprise identity provider with OAuth 2.1 or SAML, then issues down-scoped, ephemeral tokens to each target server via RFC 8693 token exchange, closing the confused-deputy hole.¹²

02

Tool-level access control

Role- and attribute-based lists restrict which tools a given user, organization, or session can even see, so an agent cannot execute what it cannot discover.²⁵

03

Traffic inspection and data protection

Filters prompt-injection attempts, sanitizes responses, rate-limits, and scrubs resident information before a payload leaves the perimeter.²²

04

Routing and state

Routes tool calls across server instances, balances load, and keeps session affinity through multi-turn workflows.²⁷

05

Unified observability

Emits standardized OpenTelemetry logs so that every backend action can be traced back to the prompt that caused it.²²

CHAPTER 3

Three regulated industries have already paid the tuition

Hospitality, banking, and healthcare run legacy systems of record, hold sensitive data, and answer to regulators. Their early MCP deployments are the closest thing multifamily has to a pilot study.³²

EXHIBIT 2 CROSS-INDUSTRY COMPARISON OF MCP DEPLOYMENTS

Sector	Systems of record	Core challenges	Proven mitigations
Hospitality	Central reservation, property management, and revenue management systems ³⁴	Inventory depletion, unapproved discounts, rate scraping via exposed availability feeds ³⁵	Deterministic price floors in tool schemas, rate-limited inventory search, cross-departmental governance ³⁵
Financial services	Core banking, loan origination, portfolio analytics ³⁶	Calculation hallucinations, unapproved credit mutations, disclosure non-compliance ²²	Arithmetic offloaded to deterministic server-side code, inherited row-level security, mandatory confirmation gates ²¹
Healthcare	Electronic health records (Epic, Cerner), imaging, billing ³³	Protected health information exposure, legacy databases without APIs, context bleeding between sessions ³³	Auto-generated REST layers over legacy schemas, "minimum necessary" scoping, cryptographic session isolation ²²

Hospitality: the closest cousin

Hotels and apartments look alike from the inside: physical assets, a blend of transient and long-term occupancy, legacy PMS platforms, and revenue optimized by dynamic pricing.³⁴ Analyses presented around the Hospitality Industry Technology Exposition and Conference ("HITEC") described what happened when hoteliers connected experimental MCP servers directly to live availability, rates, and inventory engines: agents were manipulated through prompt injection into granting unapproved promotional rates, blocking inventory with bulk reservations, and exposing revenue strategy to scrapers.³⁵

The governance lesson was as important as the technical one. Implementations treated as pure IT projects broke down; the ones that worked split responsibility three ways — IT owned authentication and transport, revenue management set hard parameter boundaries no model could override, and operations governed guest communication — and put deterministic validation inside the server so pricing logic was isolated from generative manipulation.^{34,35}

Banking: deterministic math and confirmation gates

Finance tolerates neither stochastic arithmetic nor unverifiable actions.²³ Institutions deploying MCP — FactSet for portfolio analytics, nCino for mortgage origination — concluded that a generative model must never do raw math or execute a transaction unchecked.^{21,36,37} Their architectures separate qualitative reasoning from quantitative execution: the LLM assesses, while debt-service coverage ratios and amortization schedules are computed by unit-tested helper functions inside the server.²¹ nCino's mortgage architecture adds programmatic confirmation gates: an agent can retrieve tax filings and draft documents, but a final loan commitment halts until a qualified human signs off.³⁶ Coupled to the enterprise data fabric, agent connections inherit each user's row-level security, so a model cannot read what its operator could not.²³

Healthcare: legacy mediation and minimum necessary

Healthcare carries HIPAA and HITRUST obligations in environments where, by one vendor's account, over 80 percent of legacy imaging and database infrastructure lacks modern REST APIs.³³ Providers bridged the gap with data virtualization layers that introspect legacy SQL Server, Oracle, and DB2 schemas and generate compliant endpoints, which are then wrapped by containerized MCP servers — modern discovery for agents without a multi-year database migration.³³ The privacy rule is enforced at the protocol layer: a documentation agent asking for patient history receives scoped queries, recent vitals or lab trends, never the whole chart, and frameworks such as Innovaccer's HMCP build tenant segregation and session boundaries into the transport so one interaction's data cannot bleed into the next.^{22,33}

Every one of these industries learned the same thing: **open protocol access is not enterprise readiness.** The readiness is in what you put around it.

CHAPTER 4

The threat model is documented, and it is not hypothetical

MCP in production introduces risks that bypass conventional perimeter defenses, because a language model is a non-deterministic translator between human language and system commands. A malicious instruction can arrive disguised as ordinary data and turn a connected agent into a vector for privilege escalation or exfiltration.²²

The Coalition for Secure AI ("CoSAI") has published an enterprise framework of twelve core threat categories (MCP-T1 through MCP-T12) and nearly forty attack vectors, aligned with the academic threat model of Narajala and Habler.^{18,22} Five matter most for a property operator.

T1

Authentication spoofing and confused-deputy attacks

Centralized servers often hold elevated keys. If identity verification fails somewhere in a multi-agent delegation chain, an attacker rides the agent's broad credentials into backend systems.²²

T2

Missing object-level access control and context bleeding

Without authorization enforced at the tool boundary, an agent reads records across organizational lines and carries confidential data into shared context windows.²²

T3/4

Instruction-boundary collapse and injection

Models process control instructions and raw data in the same token stream. An adversarial prompt embedded in a maintenance request, a vendor invoice, or a prospect inquiry can hijack the reasoning loop and trigger unauthorized tool calls.²² In multifamily, every one of those data fields is user-writable.

T6

Tool poisoning and rug-pull attacks

Agents select tools by their natural-language descriptions. An attacker who can alter a tool description, schema, or parameter mapping can deceive an agent into sending private data to an external endpoint.^{20,22}

T11

Supply chain risk and shadow MCP infrastructure

Open-source server repositories are a gift to developers and a liability to security teams when an engineer connects an ungoverned community server to internal infrastructure, opening unmonitored egress paths.¹

Defense in depth, by layer

✓	Micro-segmentation and sandboxing.	Servers that run business logic or database queries live in isolated containers (gVisor, Kata Containers) with immutable root filesystems and strict outbound policy.	22
✓	Cryptographic tool integrity.	Tool descriptions and schemas are treated as executable code: a private, certified registry requires signatures before registration, which defeats runtime tampering.	19,22
✓	Context-aware ephemeral access.	No persistent API credentials; the gateway issues short-lived, down-scoped tokens mapped to the active task and revokes them on completion.	18
✓	Structured human confirmation.	High-risk calls do not rely on a passive modal that induces consent fatigue; critical mutations require authenticated, parameter-inspected confirmation through a structured interface.	4,22

CHAPTER 5

Why multifamily is the hard case

The legacy PMS bottleneck

Multifamily technology sits on platforms engineered decades ago for general ledger accounting, property operations, and lease record-keeping.⁷ Yardi Voyager and RealPage OneSite are the source of truth for millions of apartment units, and neither was designed for low-latency, dynamic interaction with autonomous agents.^{2,14} Third-party vendors connecting to those cores face three structural barriers:

gatekept access with certification fees, per-unit surcharges, and long approval cycles;⁷ fragile synchronization that runs on hourly or nightly batches, so an AI leasing agent can quote yesterday's price on a unit leased this morning;⁵ and the engineering overhead that forces even well-funded PropTech firms such as EliseAI to spend substantial resources building and maintaining custom connectors across PMS, CRM, and other platforms.^{14,15}

Level 1 and Level 2

Funnel Leasing draws a useful line between two stages of multifamily AI.⁵ Level 1, informational AI, reads data and answers questions about amenities, policies, and floor plans; it is low risk because it modifies nothing. Level 2, action-oriented AI, creates guest cards, schedules self-guided tours through smart-access hardware, modifies maintenance tickets, and posts lease concessions. Moving to Level 2 is where an invalid write or a misquoted term stops being a bad chat and becomes a corrupted record in the portfolio's reporting.⁵ The same analysis makes the distinction this paper is built on: a traditional integration is a passive pipe that moves data, whereas MCP, properly governed, is the control plane that defines what an agent may do with that data once it has it, demands human intervention when a parameter is breached, and leaves an auditable trace.⁵

Regulatory realities

01

The Fair Housing Act

Housing providers may not discriminate on the basis of race, color, national origin, religion, sex, familial status, or disability. An agent exercising unconstrained natural-language discretion in a leasing negotiation can produce inconsistent qualification criteria, disparate concessions, or steering through nothing more sinister than stochastic variation in its output — and the owner carries the liability.⁵

02

Antitrust scrutiny of algorithmic pricing

With state and federal investigations into algorithmic rent-setting, operators need demonstrable autonomy over pricing and concessions. Agents acting across a portfolio must run on transparent, auditable business logic that shows rates were set independently, without collusive cross-portfolio sharing.⁵

03

Fee transparency

Federal and state regulators increasingly require complete disclosure of mandatory fees — technology, trash valet, administrative surcharges. A leasing agent must be prevented from making inaccurate or incomplete pricing representations, which means every quoted figure is validated against the certified PMS fee schedule through a deterministic tool schema.⁵

CHAPTER 6

A blueprint for the governed multifamily gateway, and a four-phase roadmap

Four layers

The architecture isolates the systems of record, enforces the regulatory rules, and keeps everything observable.⁵ At the top sit the AI clients: conversational leasing agents, resident communication portals, automated maintenance dispatchers. They talk only to an enterprise MCP gateway deployed as a reverse proxy inside the management company's secure cloud.²⁷ The gateway is the enforcement layer: it validates workload identity, enforces OAuth 2.1 tokens, sanitizes prompts, masks resident personally identifiable information, and applies access lists so each agent discovers only the tools its role permits.²²

Behind the gateway, domain-specific servers each wrap one operational application.

✓ **Core PMS server** — interfaces with Yardi Voyager or RealPage OneSite, exposing read-only financial data plus strictly validated transactional tools for lease execution and payments. 15

✓ **CRM and leasing server** — manages guest cards, schedules tours, and validates concessions against marketing guidelines. 5

✓ **Smart-building server** — issues time-limited guest credentials through access-control platforms such as ButterflyMX for verified tour appointments. 14,48

The servers enforce deterministic guardrails.⁵ A tool that calculates a concession or evaluates creditworthiness rejects unstructured natural-language input and mandates rigid parameters — certified income values, verified identity identifiers, approved concession tiers — before running an internal, deterministic computation.²² If an agent tries to offer an unapproved discount or alter qualification guidelines, the server returns a structured error and the model is forced back inside its authorized parameters.⁴

The roadmap

Borrowing the implementation frameworks that banking and healthcare arrived at, a management company should roll MCP out in four phases that expand capability only as operational maturity is demonstrated.⁵ Start small, build incrementally — a data strategy, like a diet, fails when the first step is the whole programme.

01

Read-only discovery and context retrieval

Expose only read-only resources: floor plan specifications, certified pricing matrices, community rules, verified amenity details. Validate low-latency retrieval, eliminate hallucination in the conversational assistant, and establish baseline OpenTelemetry logging so every data access is traceable.^{21,26}

02

Low-risk operational writes

Introduce scoped write tools for low-liability workflows: creating guest cards, scheduling tours, logging maintenance inquiries. Enforce tool-level role-based access and watch execution patterns for unexpected errors.^{5,26}

03

Governed process execution

Expand to multi-step tasks: initial applicant screening, work-order triage and vendor dispatch, renewal processing. Schemas enforce Fair Housing and fee-disclosure rules; deterministic helpers evaluate the financial data.^{5,14}

04

Core transactional write-back

Allow agents to apply approved concession structures, post resident payments, and generate binding leases in the primary PMS, with programmatic confirmation gates that halt execution until a designated property manager reviews and cryptographically approves the parameters before anything reaches the general ledger.⁵

Phase 1 is where most operators should spend longer than they want to. **An agent that reads clean, standardized data is already a better leasing assistant than one that writes to messy data.**

CHAPTER 7

The buying conversation: pipelines versus governed contracts

Engaging property management leadership, institutional owners, and PropTech vendors means positioning MCP around business value, risk, and scalability, never around the protocol itself.⁵

EXHIBIT 3

STAKEHOLDER VALUE MATRIX

Persona	Strategic driver	The objection you will hear	The honest answer
CTO / VP of IT	Stability, cyber hygiene, less integration maintenance, no vendor lock-in ¹	"We already have REST APIs and an integration platform. Another protocol is complexity." ⁵	APIs transport data; MCP governs execution. REST lacks dynamic discovery and semantic validation, so every new agent needs brittle hardcoded glue. A gateway takes the problem from $M \times N$ to $M + N$ and shields systems of record from model churn. ^{4,8}
Head of Operations	Efficiency, staff retention, leasing velocity, portfolio scale ⁵	"Autonomous assistants make unpredictable decisions and add oversight burden for the on-site team." ⁵	Reliable execution inside defined guardrails. Level 2 agents create guest cards, book appointments, and route maintenance directly in the PMS, with parameter boundaries that keep them inside operating policy and take the administrative load off site staff. ⁵

Persona	Strategic driver	The objection you will hear	The honest answer
General Counsel / Compliance	Fair Housing liability, fee rules, tenant privacy, audit defence ⁵	"Giving an AI write permission in our leasing database is a housing-discrimination lawsuit waiting to happen." ⁵	Governed autonomy through deterministic enforcement. Reasoning is separated from execution; schemas enforce Fair Housing and disclosure rules; full OpenTelemetry trails give the audit its evidence. ^{1,4,22}
PropTech executive	Engineering velocity, shorter implementations, wider integration footprint ³²	"Custom integrations to closed PMS platforms eat our roadmap." ^{7,15}	Build once, integrate everywhere. A compliant server makes the product reachable from Claude, Copilot, ChatGPT, and custom frameworks, shortening deployments from months to days. ²

The conversation stalls in the same place every time: "Our stack is already connected through modern APIs and webhooks."⁵ The answer is not to argue about protocols; it is to draw the distinction between data transit and governed execution.

	An API is a passive pipe.	It moves data, carries no machine-readable semantics, offers no runtime discovery, and leaves parameter parsing and error handling to each application.	4,5		MCP is an active contract.	It dictates what an agent can discover, read, and execute; validates parameters before execution; enforces constraints; and halts for human sign-off when policy says so.	4,5
---	----------------------------------	---	-----	---	-----------------------------------	---	-----



The durable advantage is infrastructure, not models.

Foundation models ⁵ are commoditizing quickly; the value is in owning a secure, governed, observable protocol plane that connects your data to whichever model performs best this quarter.

CONCLUSIONS

Treat the protocol as governance, or pay for it later

The standardization of MCP under a neutral foundation gives multifamily a durable basis for AI integration.² Replacing bespoke connectors with a universal abstraction resolves the integration bottleneck that has constrained this industry for a generation, and it lets management companies move from informational chatbots to agents that streamline workflows, lower operating costs, and improve the resident experience.^{5,6} But the experience of hospitality, banking, and healthcare is unambiguous: protocol access alone is not readiness. Unmediated agents against core property databases invite prompt injection, data poisoning, privacy breaches, and statutory liability.^{3,5}

Four commitments follow.

1

Mandate a gateway. Every agentic request passes through an enterprise MCP gateway that provides identity federation, tool-level access control, injection filtering, and OpenTelemetry logging.^{22,27}

2

Constrain high-consequence tools with deterministic schemas. No open-ended natural language reaches a tool that evaluates an application, quotes a rate, or applies a concession.^{5,22}

3

Gate financial and contractual writes. High-impact mutations halt until a designated manager reviews and authorizes them, as banking and healthcare already do.⁵

4

Follow the four phases. Read-only discovery, then low-risk writes, then governed processes, then core transactions — building confidence and validating controls at each step.⁵

The case against, taken seriously

Is all of this overkill for a regional operator with a few thousand units and no platform team? Two objections deserve an answer. The first is cost: a gateway, a tool registry, and signed schemas sound like an enterprise programme. They scale down better than they sound. Phase 1 needs little more than read-only servers and logging, and the PMS vendors' own connectors increasingly ship with governance of their own. The second is that the vendor connector already solves it. It does, for that vendor's system. It does not govern the agent that spans three operators on two platforms, and it does not standardize what "occupancy" means across them. A gateway governs actions; a common data model governs meaning. An operator needs both, and the second is the one most of the industry is still skipping. Our earlier piece on the Monday morning problem AI will not solve on its own takes up that half of the argument.

Organizations that treat MCP as another API wrapper will meet fragility, security holes, and compliance exposure. Those that recognize it as an architectural governance layer will build agentic environments that protect net operating income, hold up under regulatory scrutiny, and remain flexible enough to adopt whatever the next generation of real estate technology turns out to be.^{1,5} The protocol is settled. The governance is the work.

ABOUT THE AUTHOR

EB

Elizabeth Braman

CEO and Co-founder, Revolution RE

Elizabeth is a serial entrepreneur, licensed attorney, and CCIM who founded Revolution RE to solve data standardization in multifamily real estate. She holds a JD, an MBA in finance, and an LLM in tax. About Revolution RE.

1. Mirantis, "Securing Model Context Protocol for Mass Enterprise Adoption," mirantis.com.
2. Homesage.ai, "Top MCP Servers Transforming Real Estate in 2026," homesage.ai.
3. Airia, "What is Model Context Protocol? The Enterprise Guide to MCP," airia.com.
4. Infosys, "Integration Fabric: The Future of APIs with Model Context Protocol," infosys.com.
5. Funnel Leasing, "Why Model Context Protocol is essential for multifamily AI deployment," funnelleasing.com.
6. Anthropic, "Introducing the Model Context Protocol," anthropic.com.
7. Adventures in CRE, "What Is MCP and Why Should CRE Professionals Pay Attention?" adventuresincre.com.
8. Shakudo, "What is MCP (Model Context Protocol) & How Does it Work?" shakudo.io.
9. Linux Foundation, "Linux Foundation Announces the Formation of the Agentic AI Foundation," linuxfoundation.org.
10. Gartner, "Innovation Insight: Model Context Protocol," gartner.com.
11. Gartner, "Model Context Protocol: The New Must-Have Capability to Drive AI," gartner.com.
12. K2view, "MCP Gartner insights for 2025," k2view.com.
13. Forrester, "Predictions 2026: AI Agents And New Business Models," forrester.com.
14. EliseAI, "Integrations," eliseai.com.
15. EliseAI, "Supported Integrations Per Product," support.meetelise.com.
16. Zenodo, "Why MCP Becomes the Control Plane for Enterprise AI," zenodo.org.
17. Model Context Protocol, "Architecture overview," modelcontextprotocol.io.
18. Narajala and Habler, "Enterprise-Grade Security for the Model Context Protocol (MCP)," arXiv:2504.08623, arxiv.org.
19. Narajala and Habler, "Enterprise-Grade Security for the Model Context Protocol (MCP)," HTML version, arxiv.org.
20. MDPI, "Model Context Protocol Threat Modeling and Analysis," mdpi.com.
21. agentic-ops, "Real Estate MCP Server: Property Intelligence & Workflow," github.com.
22. Coalition for Secure AI, "Securing the AI Agent Revolution: A Practical Guide to MCP Security," coalitionforsecureai.org.
23. Strategy Software, "Enterprise AI: Closing the Context Gap," software.strategy.com.
24. SAP, "What Is Model Context Protocol (MCP)?" sap.com.
25. FactSet Insight, "Enterprise MCP Part 3: Security and Governance," insight.factset.com.
26. Tray.ai, "MCP governance in the enterprise," tray.ai.
27. Kong, "What is an MCP Gateway? Key to Secure Enterprise AI at Scale," konghq.com.
28. Infosecurity Magazine, "MCP Is Creating Major Governance Gaps, Researchers Warn," infosecurity-magazine.com.
29. Agentic AI Foundation, "Projects," aaif.io.
30. Envoy AI Gateway, "Model Context Protocol (MCP) Gateway," aigateway.envoyproxy.io.
31. alphaXiv, "Enterprise-Grade Security for the Model Context Protocol (MCP)," alphaxiv.org.
32. Mint, "Zoho bets on AI to rewrite the enterprise software playbook," livemint.com.

33. DreamFactory, "MCP Security for Healthcare," dreamfactory.com.
34. Cendyn, "2026 is the year of MCP: what hoteliers need to know," cendyn.com.
35. HITEC, "Model Context Protocol (MCP): The Hidden Risks Hoteliers Must Uncover," hitec.org.
36. The Fintech Times, "nCino Adds MCP Layer to Connect AI Agents to Mortgage Suite," thefintechtimes.com.
37. FactSet, "Model Context Protocol (MCP)," factset.com.
38. LoanPro, "What is Model Context Protocol and what does it mean for lending?" loanpro.io.
39. Nitor Infotech, "Model Context Protocol (MCP) in Healthcare," nitorinfotech.com.
40. Keragon, "Medical MCP Servers: How Model Context Protocol...", keragon.com.
41. Guru, "What Is Epic MCP? Exploring the Model Context Protocol and AI," getguru.com.
42. Vouched, "Unlocking the Future of Healthcare with the Model Context Protocol," vouched.id.
43. Shiji Insights, "Model Context Protocol in Hospitality: An Overview," insights.shijigroup.com.
44. FactSet, "FactSet Expands Model Context Protocol Suite to Portfolio Analytics," factset.com.
45. MarkTechPost, "Critical Security Vulnerabilities in the Model Context Protocol (MCP)," marktechpost.com.
46. EliseAI, "Why Speed Matters: How EliseAI Gets Practices Live in Weeks, Not Months," eliseai.com.
47. Baseten, "How EliseAI trains specialized models for healthcare and housing," baseten.co.
48. ButterflyMX, "EliseAI Integrations," butterflymx.com.
49. Wolters Kluwer, "How Model Context Protocol supports the future of agentic healthcare," wolterskluwer.com.

© 2026 Revolution RE · www.revolutionre.com · The interactive edition of this paper, with live sources, is at www.revolutionre.com/post/mcp-multifamily-real-estate-white-paper