

Motivation

Professionals waste hours on manual report writing, though speech conveys richer detail. Automated report generation from speech uses agentic AI-autonomous agents that use external tools, reason over multi-step tasks, and manage the workflow. The goal is fully speech-driven reporting where the user speaks freely, the agent transcribes, structures content, asks follow up questions and delivers a polished report.

State of the art

Solutions like Nuance DAX demonstrate that voice-based documentation is already being deployed successfully. But these systems are typically proprietary, domain-specific, and difficult to adapt. Other new protocols extend this approach even further. The **Agent Communication Protocol (ACP)** supports multimodal interactions between agents, allowing them to communicate across formats like audio, images, and structured data for more complex workflows. The **Agent Network Protocol (ANP)** focuses on decentralization. Agents operate independently and interact through shared standards, much like microservices.

Protocol selection

To enable fully agent-driven report generation, two open protocols were selected: **Model Context Protocol (MCP)** and Agent-to-Agent (A2A). MCP provides standardized access to external tools like ASR or databases, while A2A handles communication between specialized agents. Both are lightweight, modular, and support scalable orchestration without tight coupling.

Protocols

Model Context Protocol

- One JSON-RPC rulebook lets the agent call every tool the same way.
- Any service that follows it becomes an auto-discovered MCP server. Core code stays untouched.
- Scopes per server guard sensitive data.
- Needed here: Whisper ASR, template storage, Postgres look-ups, vector memory, Slack alerts

Agent to Agent Protocol

- Each Agent advertises an AgentCard(name, skills, auth endpoint), enabling automatic discovery and capability matching.
- Oauth-based tokens ride along with every call. Scopes limit what a receiving agent may do with the data.
- Lets you add domain-specific helpers, without touching the orchestrator's code
- Uses plain JSON, so different agents can connect without special libraries

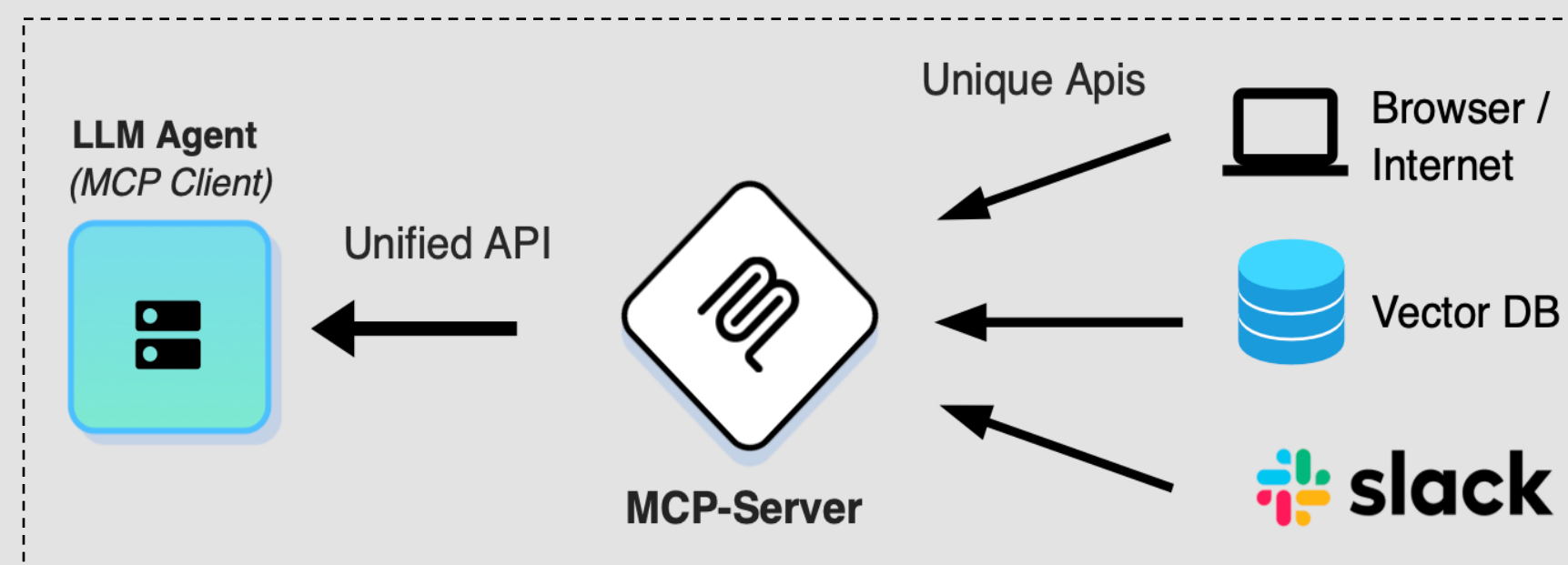


Figure 2. The architecture of the Model Context Protocol

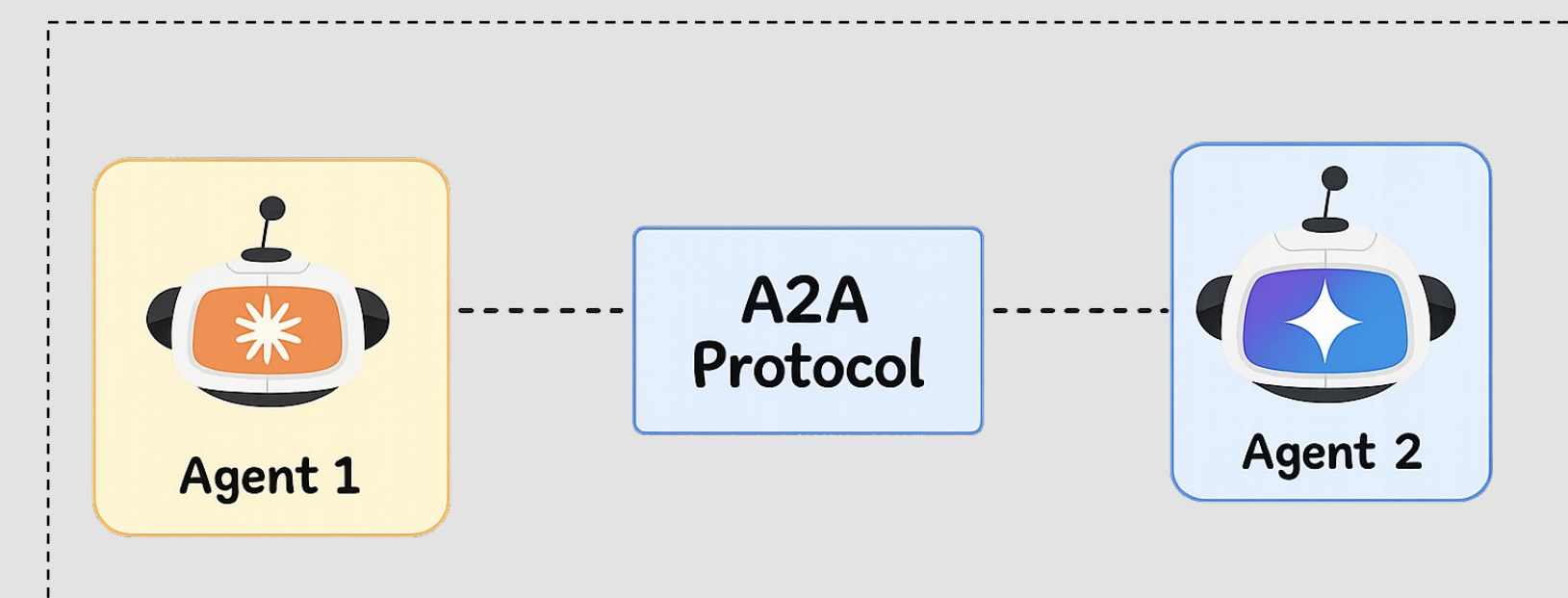


Figure 3. The architecture of the Agent to Agent Protocol

Combination of Agentic AI, MCP & A2A

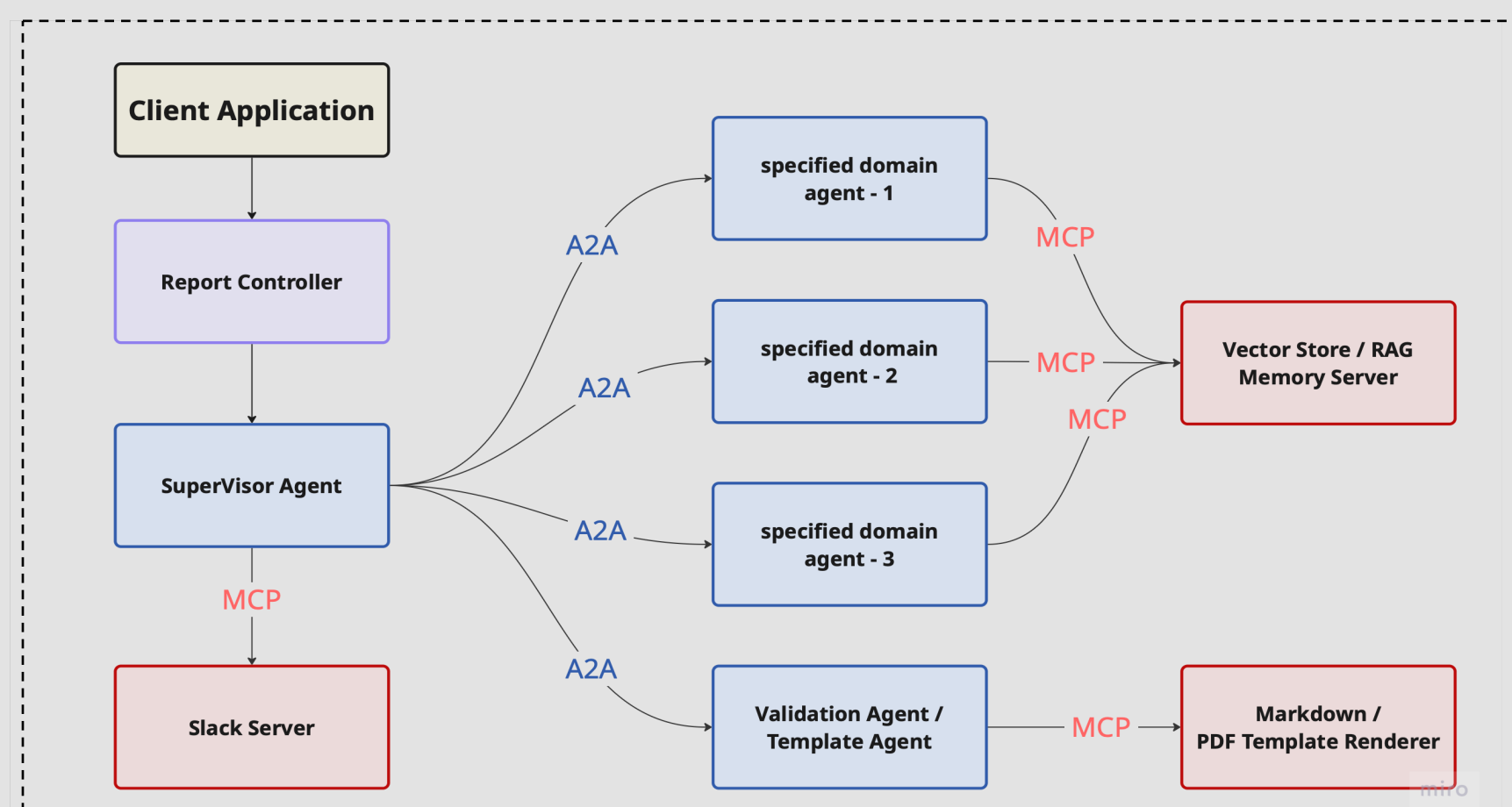


Figure 4. The architecture of how MCP & A2A could work together

Potential services via MCP

- **Whisper ASR** or **Siri** handles the initial voice transcription on-device.
- A **vector database** combined with a **RAG pipeline** helps agents retrieve relevant facts from prior reports or customers.
- **Filesystem access** provides PDF or Markdown templates for structured formatting depending on the domain.
- **Slack** can be used to send reports directly to teams or channels.

Evaluation

Traditional ASR tools output raw transcripts, which often lack context and require manual editing.

By combining Agentic AI with MCP and A2A, this architecture fills gaps in real time, routes content to the right agents, and delivers clean, structured output. It reduces engineering overhead, improves data integrity, and scales easily across domains.

Unlike proprietary systems, this design is fully open and modular, making it suitable for long-term use in diverse fields.

References

- [1] Ghosh, D. P., Ghosh, S., & Ghosh, B. (2025). *Beyond MCP & A2A: Core LLM limitations and the future of agentic AI in the construction industry*. ResearchGate.
- [2] Model Context Protocol. (n.d.). *Introduction*. ModelContextProtocol.io.
- [3] A2A Project. (n.d.). *Agent-to-Agent Protocol*. GitHub.
- [4] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Riedel, S. (2020). *Retrieval-augmented generation for knowledge-intensive NLP tasks*. arXiv.
- [5] Zhou, Y., Du, Y., Zheng, S., Tan, H., & Dai, W. (2024). *ReAct-style Agent Architectures: A Survey*. arXiv.



Figure 1. Example: Salesperson uses the app hands-free while driving