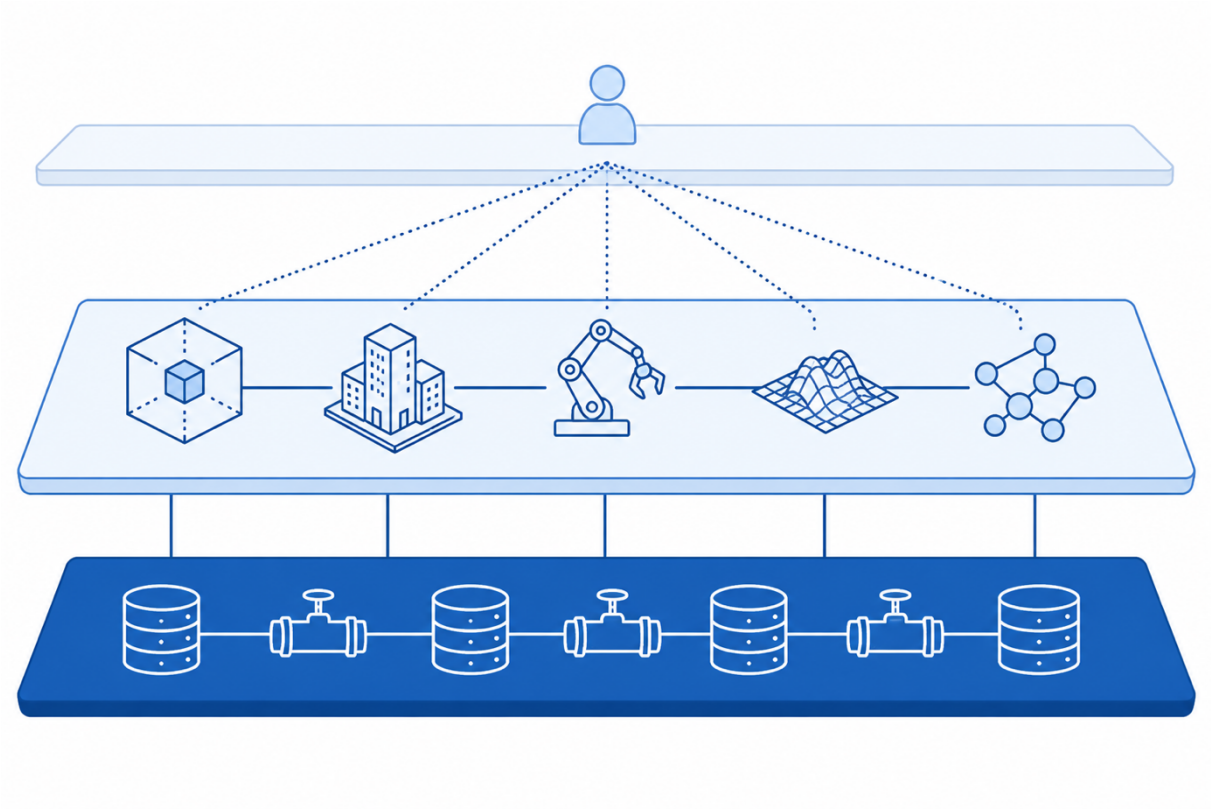
# Connecting Multiple Digital Twins: Why the Data Layer Has to Come Before the Agent Layer



Most utilities now run more than one digital twin. A water company's twins typically span the full source-to-sea cycle: catchment and raw water abstraction, treatment, distribution, wastewater collection, and final discharge, each one built by a different team, on a different platform, for a different purpose. An electricity distributor has the equivalent problem with substation, low-voltage network and flexibility-market twins. In both cases, a decision made in one twin can change what the next twin downstream has to deal with, and the question that follows is how to connect them. The instinctive answer in 2026 is to add an AI agent layer on top and let it talk to everything. That gets the sequence backwards.

Let’s start with what agent protocols actually do. The Model Context Protocol (MCP) standardises how an AI agent calls an external tool, collapsing what would otherwise be a custom integration for every agent-to-system pairing into one standard interface (an M x N issue). Agent2Agent (A2A), Google's complementary protocol, does the equivalent job for agent-to-agent coordination. Neither manages data, guarantees message delivery, enforces governance, or guarantees consistency across systems. Kai Waehner's April 2026 architecture analysis puts it plainly: “one protocol, one job.” MCP is an interface layer, not a data pipeline, and treating it as one is where this goes wrong.

The temptation to treat it as one is understandable given the pace of adoption. MCP passed 10,000 active public servers by December 2025, combined SDK downloads across the major languages exceed 97 million a month, and it now sits behind ChatGPT, Gemini, Microsoft Copilot, Cursor and VS Code. An Endor Labs analysis of 2,614 MCP implementations found 82% used file-system operations prone to path traversal and 67% used APIs related to code injection — implementation failures rather than flaws in the protocol itself, but a live exposure for any organisation giving an MCP-connected agent broad access without least-privilege controls. Scale that fast, and a thin, advisory role for the agent layer stops being caution and starts being basic risk management.

That distinction matters more for digital twins than for most integration problems, because connecting two twins is rarely a conversational exchange. It is a control-loop coupling: an upstream decision in one twin's model can degrade the downstream performance of another. NIST IR 8356 names the specific failure mode this creates if the wrong layer carries the connection — “linked-DT corruption.” Twin definitions are routinely built on top of each other, so a corrupted or inconsistent definition in one twin propagates errors into every twin that depends on it. An open-ended, negotiated agent-to-agent call is exactly the wrong mechanism for an exchange that needs to be deterministic, auditable and replayable.

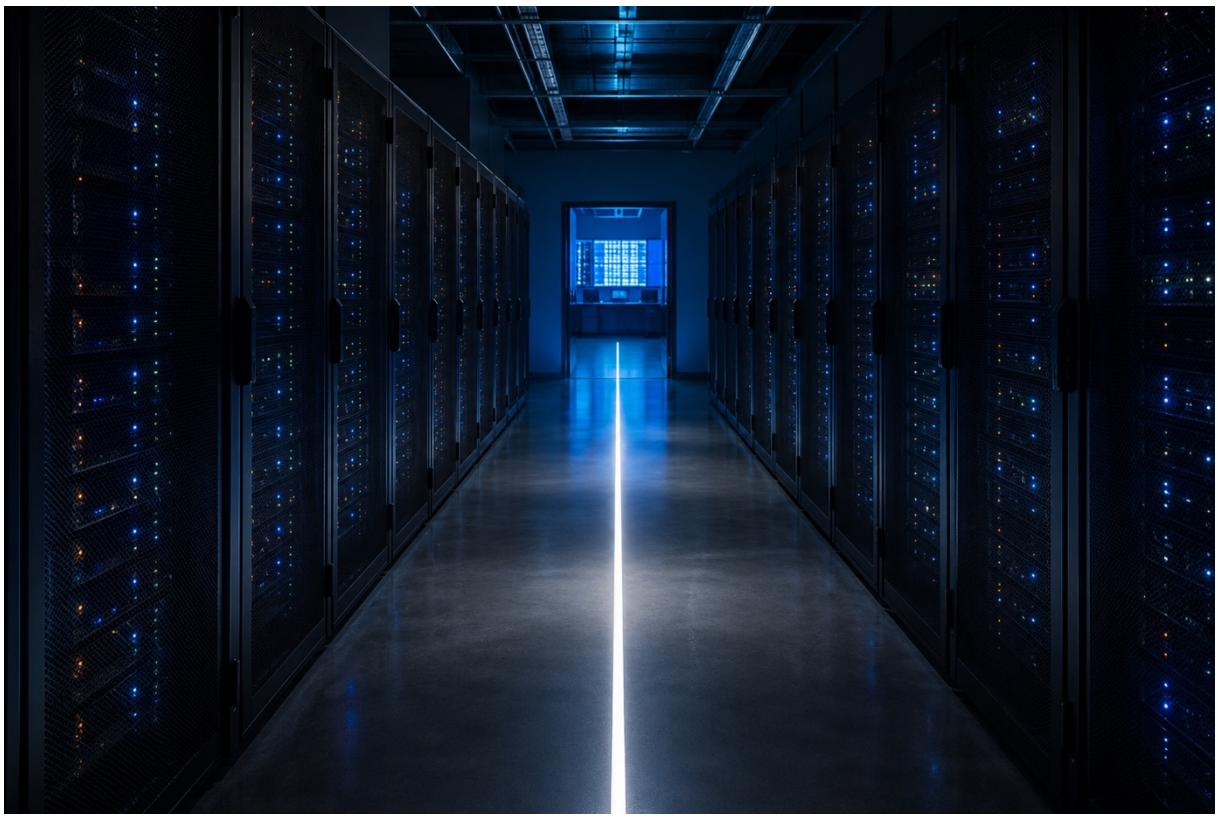
The right sequence is layered rather than binary. A governed data backbone — Microsoft Fabric or Apache Kafka — should be the substrate of record for cross-twin exchange, with explicit, versioned data contracts between twin owners, particularly for anything high-frequency, safety-related or compliance-relevant. Agent protocols belong above that layer, reserved for genuinely exploratory or advisory interactions, with a human in the loop and no write-back access into a control system. Waehner calls this the “real-time context engine” pattern: Kafka and Flink govern data freshness and

consistency, and MCP supplies the interface an agent uses to consume it. The consistency guarantee comes from the streaming layer. It never comes from MCP itself.

This is where an agent protocol genuinely earns its place: not carrying the operational coupling itself, but answering a hypothetical about it. Picture an agent attached to the treatment twin asking the distribution twin's agent a “what if” — if treatment changes its chlorination dosing for the next six hours, what does that do to disinfection residuals by the time water reaches the far end of the network? That is an exploratory, advisory question, well suited to MCP or A2A, and it should come back to a person before anyone changes a dosing setpoint. The moment the answer would trigger an automatic write-back into a control system rather than a recommendation for a person to review, it has left the agent layer's job and entered the data backbone's.

This is not a theoretical sequencing argument. Microsoft's Foundry IQ is built to unify Work IQ, Fabric IQ, Azure SQL, File Search and MCP sources behind one retrieval layer. At Build 2026, Microsoft took the knowledge-base layer and its MCP server to general availability, with full SLA coverage; the individual source connectors remain in public preview. A hyperscaler is putting the governed-data-unification layer underneath the agent-facing protocol layer in production, rather than waiting for every connector to mature first. Any utility weighing how to let AI agents query multiple digital twins should read this as a working example of the sequencing argument, not a finished feature to copy outright.

One UK water utility I have advised on is already building it in this order. Its group-wide Microsoft Fabric programme is structured as a Lakehouse plus Medallion architecture — Bronze, Silver, Gold — with Zero Trust security enforced through row- and column-level controls, and OT data such as SCADA telemetry streamed into Event Streams before any AI layer touches it. The agent-facing layer is not even in scope yet. The data backbone is being engineered first, on purpose, which is the order this argument requires.



The same principle holds across organisational boundaries, not only within one company. Connected Places Catapult's Climate Resilience Demonstrator (CReDo) links energy, water and telecoms infrastructure models to flag cascading risk from extreme weather, and does it by sharing derived insight between partners rather than the underlying asset data itself. CReDo+ Beta has secured close to £10 million, with UK Power Networks among the partners, backed by Ofgem and Innovate UK. CReDo is not built on an agent protocol — it is a shared-data model — but it draws the access-control line in exactly the place this argument would put it: insight crosses the organisational boundary, raw data does not.

None of this is primarily a technology decision, and the evidence for that predates AI agents entirely. SWAN's 2024 Digital Twin Values Guide is built around when to invest rather than how to build, and its Vitens case study found operators adopted daily digital twin use quickly once they were genuinely engaged — readiness was organisational, not technical. A utility that has not yet settled who owns which twin's data is not ready to decide which protocol connects them, regardless of how mature that protocol is.

Ofgem's November 2025 open letter on architectural coordination for the energy sector states that “no existing body or group can provide overall guidance and governance in architectural coordination in an unbiased, logical, and altruistic manner,” and names AI explicitly as a complexity multiplier rather than a solution. Protocol choice sits inside whatever data-governance model an organisation adopts. It cannot substitute for deciding who owns each twin's data and who is accountable when one twin's output degrades another's.

This maps onto a tension I track using my Land-Labour-Capital-Data-Energy-Algorithms framework. Data and Algorithms are both candidates to govern this relationship, and only one of them can: either the data layer sets the terms the algorithm operates within, or the algorithm is left improvising rules the data layer should have enforced. For a board connecting its first two digital twins, that is the decision to make before anything else. The protocol the agents eventually speak comes after, and by comparison, it is the easy part.

# Sources


1. “MCP vs. REST/HTTP API vs. Kafka: The Architect’s Guide to Agentic AI Integration” — Kai Waehner, 10 April 2026. https://www.kai-waehner.de/blog/2026/04/10/mcp-vs-rest-http-api-vs-kafka-the-architects-guide-to-agentic-ai-integration/
2. “Donating the Model Context Protocol and Establishing the Agentic AI Foundation” — Anthropic, 9 December 2025. https://www.anthropic.com/news/donating-the-model-context-protocol-and-establishing-of-the-agentic-ai-foundation
3. “Security and Trust Considerations for Digital Twin Technology,” NIST IR 8356 — Voas, Mell, Laplante, Piroumian, NIST, February 2025. https://nvlpubs.nist.gov/nistpubs/ir/2025/NIST.IR.8356.pdf
4. “Foundry IQ: Build Smarter Agents Faster with Unified Knowledge and Serverless Retrieval” — Pablo Castro, Microsoft Foundry Blog, 2 June 2026 (Build 2026). https://devblogs.microsoft.com/foundry/build-smarter-agents-faster-with-foundry-iq/
5. “Climate Resilience Demonstrator (CReDo)” — Connected Places Catapult. https://cp.catapult.org.uk/project/climate-resilience-demonstrator-credo/
6. “Digital Twin Values Guide” — SWAN Forum (Smart Water Networks Forum), 2024. https://swan-forum.com/wp-content/uploads/2024/06/SWAN-Forum_Digital-Twin-Values-Guide.pdf
7. “Energy Digitalisation Governance: Architectural Coordination” — Ofgem open letter, Marzia Zafar, 4 November 2025. https://www.ofgem.gov.uk/policy/energy-digitalisation-governance-architectural-coordination