Downloaded from buildwithruud.com · Kami Document System

Data 360 Zero-Copy & Amazon Connect MCP: Separating

2026.11.16 · /blog/data-360-zero-copy-snowflake-amazon-connect-mcp

1. Why Zero-Copy Read Federation Is Not a Write Strategy

Salesforce Data 360 (formerly Data Cloud) Zero-Copy federation with Snowflake, Redshift, and Databricks eliminates bulk ETL duplication for analytical reads and AI grounding. Similarly, connecting Amazon Connect and Salesforce Service Cloud via the Model Context Protocol (MCP) allows conversational agents to reason over live customer context.

However, both architectures fail in production when teams confuse **read federation** with **write ownership** or **telemetry streaming**:

1. **The Telemetry Callout Trap**: Polling or writing live Amazon Connect queue depth, wait time, and service-level metrics into custom Salesforce objects burns daily API quotas and turns your CRM into a lagging, conflicting copy of the contact center.
2. **The Server MCP vs. WebMCP Boundary**: Using a backend Server MCP integration user when a human is actively working inside a browser session discards session context; conversely, making a browser agent scrape DOM selectors breaks on every UI deploy.

2. Clean-Room Reference Architecture

Contact Center Telemetry: Treat Amazon Connect as the system of engagement and telemetry source of truth; stream aggregated real-time state to the Service Cloud UI layer while persisting only structured case intake, reason codes, and resolution outcomes in Salesforce.

Zero-Copy Reads + Outbox Writes: Use Data 360 Zero-Copy to hydrate Customer 360 and Agentforce context from Snowflake, but route all state-mutating commands through a Single-Writer service backed by an atomic Transactional Outbox.

WebMCP for Session-Bound UIs: On public and portal surfaces (like `dev.buildwithruud.com`), expose schema-validated `use_webmcp_tool` descriptors (`describe_site`, `list_actions`, `submit_inquire`) while hard-blocking `/admin` routes.

3. Portable Engineering Rule

Never use your transactional CRM database as a real-time telemetry buffer, and always match the agent protocol (Server MCP vs. WebMCP) to where the user session actually lives.