USING MCP TO ORCHESTRATE COMPOSABLE IDENTITY RESOLUTION AT SCALE

Morten Lileng
Global Head of Data & Identity Engineering | dentsu

dentsu | MERKLE

AGENDA

- Who we are, and what we ship
- The coordination problem: global scale, local rules, unique clients
- Why MCP, and what it gives us
- How we built it: architecture and trade-offs
- Live demo
- Results, and what to take away

dentsu | MERKLE

DENTSU.AUDIENCES SIMPLIFIES PERSON-BASED IDENTITY AND DATA THROUGH THE ONBOARDING AND ACTIVATION PROCESS

Create and grow person IDs across customers & prospects

Customer PII

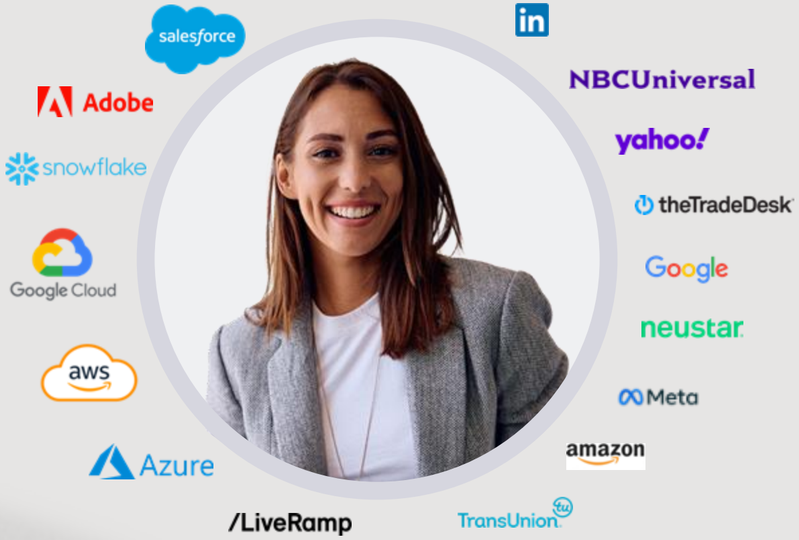
Partner PII

Brand/Partner Site & Visits

Dentsu Consumer & Professional IDs

Enhance person IDs with consumer and business data

Connect person IDs to experience, analytics, and data management



GLOBAL SCALE.
LOCAL RULES.
UNIQUE CLIENTS.

dentsu | MERKLE

LARGE GLOBAL CLIENTS: “HOW CAN WE EXPAND IDENTITY IN A PRIVACY-SAFE WAY IN INTERNATIONAL MARKETS?”

DIVERSE CLIENTS SPANNING VERTICALS & MARKETS…

- LEADING GLOBAL HOSPITALITY BRAND
- GLOBAL CLOUD-BASED SOFTWARE COMPANIES
- FRENCH GLOBAL LUXURY BRAND
- AMERICAN MULTINATIONAL FOOD COMPANY

…WANT TO USE OUR SNOWFLAKE NATIVE IDENTITY *IN INTERNATIONAL MARKETS*

DENTSU • IDENTITY

Matching isn’t the hard part. Coordinating is.

Five regions, five schemas, five sets of rules. No shared contract between them.

- Multiple regional data stores with different schemas, PII rules, and match logic
- No standard abstraction for routing, validation, or enrichment across systems
- Engineering teams spend cycles on orchestration plumbing, not matching logic
- Data sovereignty requirements make centralization impossible
- The problem is coordination. Not the match logic itself

dentsu | M

WHY MCP?

dentsu | MERKLE

What we used

Four pieces working together. MCP is the glue, not the whole system.

01

MCP orchestration layer

Open protocol for calling tools on remote systems. Carries instructions, never PII. The spine of the stack.

02

Regional identity engines

Five per-region matchers, each wrapped as an MCP server. Match logic and data stay local.

03

Enrichment & reference tools

Address, device, and consent services exposed as MCP tools. Pluggable without touching orchestration.

04

AI Planning agent

LLM that reads each record and decides which tools to call. Runtime routing, not a hard-coded pipeline.

dentsu

How a batch flows

Five identical stacks, one contract. Nothing crosses the region boundary.

STEP 01

Batch lands in Snowflake

Millions of records arrive, partitioned by region. Each region's data sits in its own Snowflake or Iceberg table.

STEP 02

Planner builds the plan

A planner runs inside each region. It reads local schema and picks which engines and tools to invoke.

STEP 03

Engines match in-region

Per-region MCP servers execute matching and enrichment inside the region's Snowflake account.

STEP 04

Resolved IDs written back

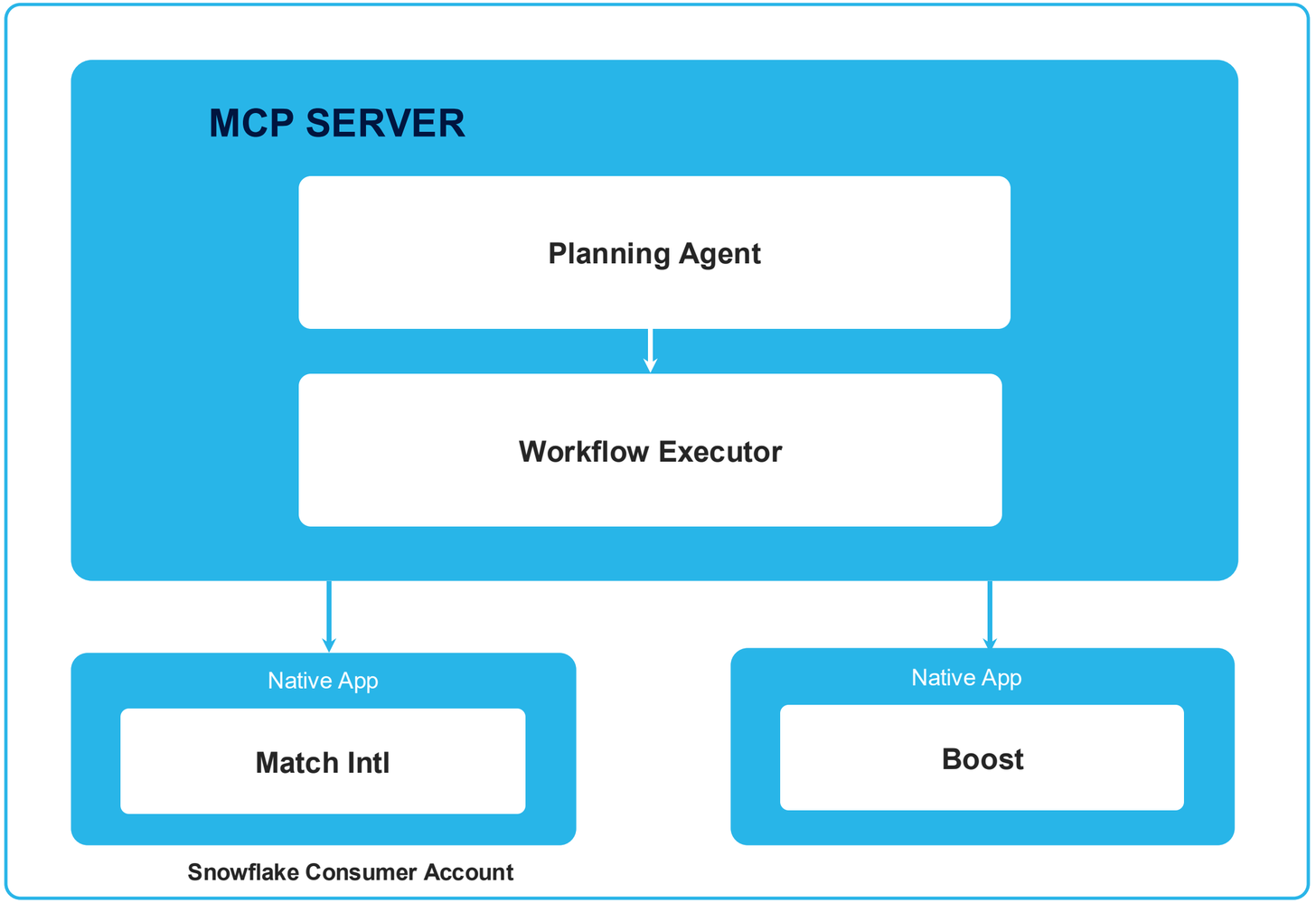
Each region writes its resolved-ID table locally. Data, compute, and control all stay in-region.

dentsu | M

HOW WE BUILT IT

Cross-App Agent Orchestration

How a Cortex Agent coordinates across independently governed Native Apps



Why Architected This Way

- **Agent tools are database-bound.** Workflow Executor lives in the same Native App as the agent. Inter-app Communication (IAC) bridges the rest.
- **No direct cross-app tool calls.** IAC is the only supported path for a Cortex Agent to reach procedures in another Native App.
- **Provider isolation by design.** Match and Boost deploy and evolve independently. Neither knows about the other.
- **No data moves.** Coordination passes identifiers, not PII.
- **Consumer account controls all connections.** IAC approvals are explicit and consumer-governed.

dentsu

Composable Identity Resolution - MCP Architecture

CLIENT

Data Engineer

Claude Code

MCP Client

OAuth 2.0

MCP

SNOWFLAKE ACCOUNT

NATIVE APP | MATCH INTL

MCP SERVER

- search-individuals
- resolve-identity
- apply-hygiene
- query-results

RBAC enforced · OAuth governed · executes in consumer account

IDENTITY RESOLUTION PIPELINE

Validate → Hygiene → Match → Append → Output

Country-routed via component registry · UK · Mexico · Other

KNOWLEDGE BASE

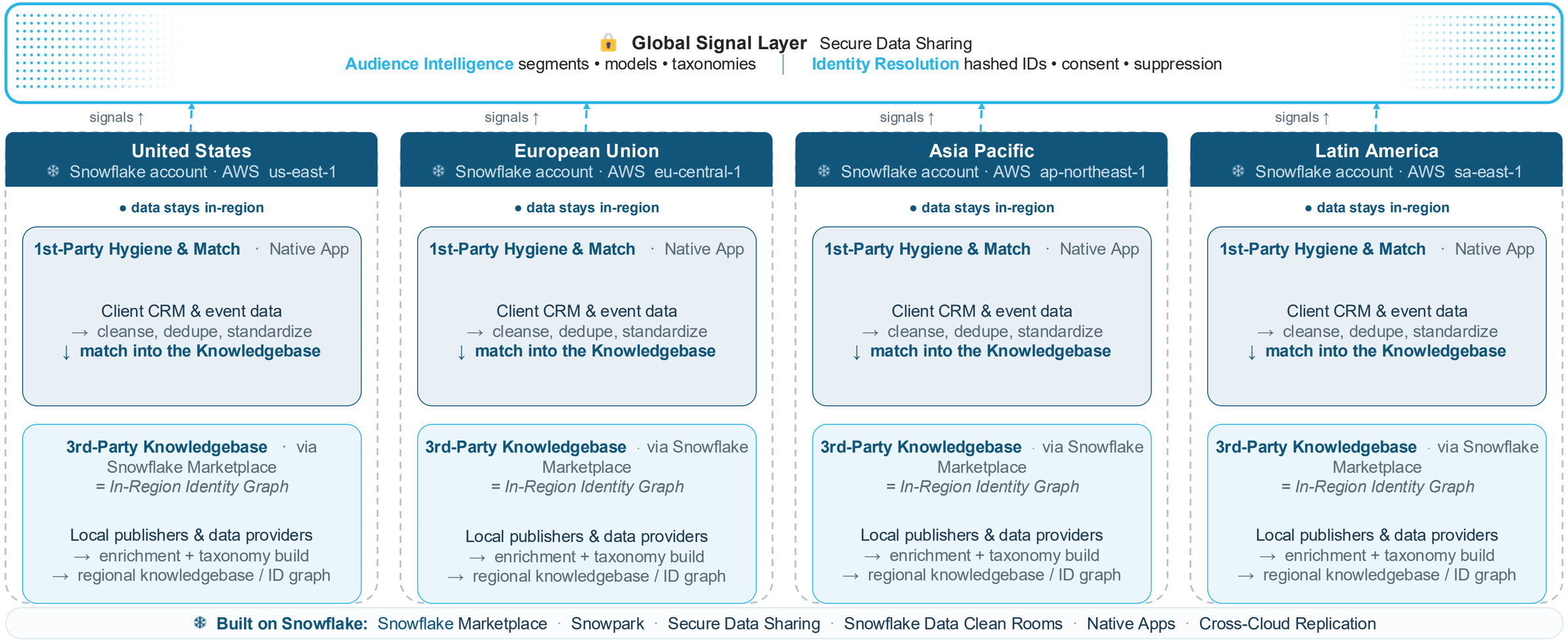
INDIVIDUALS	ADDRESSES	HOUSEHOLDS	TOUCHPOINTS
Names	Standardized addresses	Family units	Email · Phone· Device· IP· Other

SNOWFLAKE TECH

- Snowflake MCP Server
- Snowpark
- *Cortex Search (TBC)*
- Cortex Analyst
- Semantic Views

Data Sovereignty by Design

PII never crosses regional boundaries. Encrypted IDs carry the signal, not the data



DEMO

dentsu | MERKLE

RESULTS

dentsu | MERKLE

Production Outcomes

What shipped, what it's delivering, and what we held the line on

Markets

3

Live on a single composable orchestration layer

Match Rate

70%

Post-migration household resolution rate (pre-migration baseline: 60%)

Cross-Border PII

0

Cross-border PII transfers. Keys held in region

Time to Market

4 wks

Per-market rollout; minimum downstream rewrites

Source: client production data, Q2'26. Figures illustrative pending final validation.

dentsu | M

Lessons Learned: Cortex Search in Native Apps

- Cortex Search Service cannot be created inside a Snowflake Native App (confirmed product gap as of April 2026)
- Cross-account access patterns (secure share, Cortex Knowledge Extensions, REST API) are policy-incompatible when client PII must stay within the app
- "Available in Snowflake" does not mean "available inside a Native App." Validate object support in the app package early
- Engage Snowflake product early on architectural dependencies. Workarounds can ship a demo, but production needs a committed roadmap answer

What You Can Take From This

If you're building on Snowflake for regulated, multi-market data

- Snowflake is raw material, not the product. Your value is how you assemble it. Keep the layer on top thin so you can swap parts as the platform evolves.
- Make data residency physical, not a policy promise. Encrypt identifiers locally and keep each market's data in its own box so "data stays home" is enforced by the architecture, not a rule.
- Verify every primitive works inside a Native App before you commit. "Available in Snowflake" isn't the same as "available in an app package." Get Snowflake's written yes on every object in your design.

dentsu | M

THANK YOU

Morten Lileng

Global Head of Data & Identity Engineering | dentsu

dentsu | MERKLE