System Architecture & Data Flow Security Overview

Mindshine Technologies Inc. · Document 14 · Version 1.1 · August 12, 2026 · Status: **Current**

Architecture

TIVA Cortex presents data pipelines as governed configuration: data sources, typed data elements, source-field mappings, processing rules, and targets — every mutation a **versioned commit** in a per-project, branchable configuration store (full history preserved). **TIVA Flow** is the execution engine that runs that configuration: typed transformation graphs dispatched per stage, with per-stage record counts and rejects surfaced, so processing integrity is measurable per run. Business workflows, project agents, published apps, and trained models sit on the same governed base.

Data flow — with its control points

1. **Ingress:** customer data enters through authenticated connectors (files, PostgreSQL, MongoDB, APIs, feeds; S3/warehouse integrations by deployment). Connection credentials are held per source; schemas are captured at connect.
2. **Processing:** transformations run in the EKS runtime under machine- readable contracts — pinned schemas, typed elements, enforced compatibility. Unmapped columns never land; rejects are visible, not dropped.
3. **Landing:** governed targets in per-project stores; element-level bindings define exactly what lands where. PII columns must carry access policies (audited per project, closed-loop).
4. **Consumption:** dashboards, apps, agents, and analytics read the governed targets under the same central authorization policy.

LLM integration

LLM-agnostic: Anthropic, OpenAI, Google, Amazon Bedrock, or customer-hosted models per deployment; customer API keys supported. Prompts carry recipe metadata and — only where the customer has not flagged regulated data — bounded row samples. **Customer data is never used to train foundation models.**

Deployment & isolation

Managed deployment runs on **Amazon EKS (us-east-1)** with per-environment namespaces and an **isolated namespace per deployed customer application**; tenant isolation in the product is enforced by organization-scoped queries behind the central authorization policy. Workloads are hardened (non-root, read-only root FS, seccomp, admission-policy least privilege). Private-cloud, AWS/Azure, and on-premises deployment patterns are supported; in customer-controlled deployments, isolation boundaries follow the customer's architecture with the same application-layer controls.

Owner: CTO. **Review:** on material architecture change. **Contact:** security@mindshine.com.