



Agentic Data Fabric for AI-Native Industrial Systems

ENABLING PHYSICAL AI AND DISTRIBUTED AGENT ARCHITECTURE

Executive Summary

Industrial systems are undergoing a fundamental transformation as artificial intelligence moves from cloud-based analytics into real-time, safety-critical operations at the edge. This trend creates an urgent need for architectures that can bridge the gap between deterministic machine control and adaptive AI-driven decision-making. The emerging requirement in Industrial AI is for distributed, composable AI — systems where perception, planning, analytics, and actuation exist as independent capabilities. These capabilities can discover each other, form task chains dynamically, and coordinate across fleets of heterogeneous equipment operating in harsh, communications-constrained environments.

This paper presents a unified architecture for AI-native industrial systems using the Data Distribution Service (DDS®) standard as the framework to connect deterministic real-time communication, semantically structured operational data, and distributed AI agents into one system. The result is an architecture ready for AI-native industrial systems supporting end-to-end Physical AI models and distributed AI agents enabling next-generation systems to be built and deployed with lower cost and greater agility.

Introduction and Industry Context

Heavy Industry Systems inside established geofences with repeated workflows are strong candidates for end-to-end model deployment paired with deterministic safety layers. Open-weight and open-source model families means Heavy Industry systems can now be built by adapting and verticalizing existing foundations, rather than starting from scratch. That makes it economically viable to create site and task-specific systems while still preserving control over data, deployment boundaries, and life cycle management.

To enable the above scenario, a unified architecture is needed that connects traditionally siloed data using deterministic real-time communication, semantically structured operational data, and distributed AI agents into one system. This enables coordination of people, robots, vehicles, enterprise workflows, and simulation systems. The architecture must work across simulation, digital twin, edge, and cloud. It must support both Physical AI systems

acting on machines, as well as Digital AI systems acting on workflows. It must be modular, scalable, and secure by design. It should be designed to support a continuous closed-loop system from day one, rather than treating learning, simulation and deployment as disconnected phases.

Heavy Industry can adopt this approach directly by leveraging Automotive SDV hardware and cloud compute resources. The remaining gap is an architecture that can unify real-time communication, semantic context, AI agents, and deployment into one stack.

Edge Equipment Model Deployment Challenge

Deploying AI at the equipment level demands an end-to-end model architecture that ingests large context windows — on the order of one- to two-billion tokens — and produces actionable output of 2-6 tokens at control-loop frequencies of 30–50 Hz (roughly 300 tokens per second). This deployment can integrate physics-informed vision-language-action models with a structured agent pipeline of perception, decision, and action stages that operate under continuous safety-boundary enforcement.

Digital AI Agents and Semantic Systems

Industrial AI needs distributed, composable AI — both to form task chains dynamically and coordinate across fleets of heterogeneous equipment operating in communications-constrained environments.

Addressing this challenge requires three things;

- A real-time data fabric for deterministic communication across distributed machines and systems
- A semantic processing layer that transforms raw telemetry into contextualized knowledge
- A self-organizing agent framework that composes perception, reasoning, and actuation into adaptive task chains.

This forms an integrated architecture in which each layer depends on and amplifies the others, and the choice of data fabric fundamentally shapes what the upper layers can achieve.

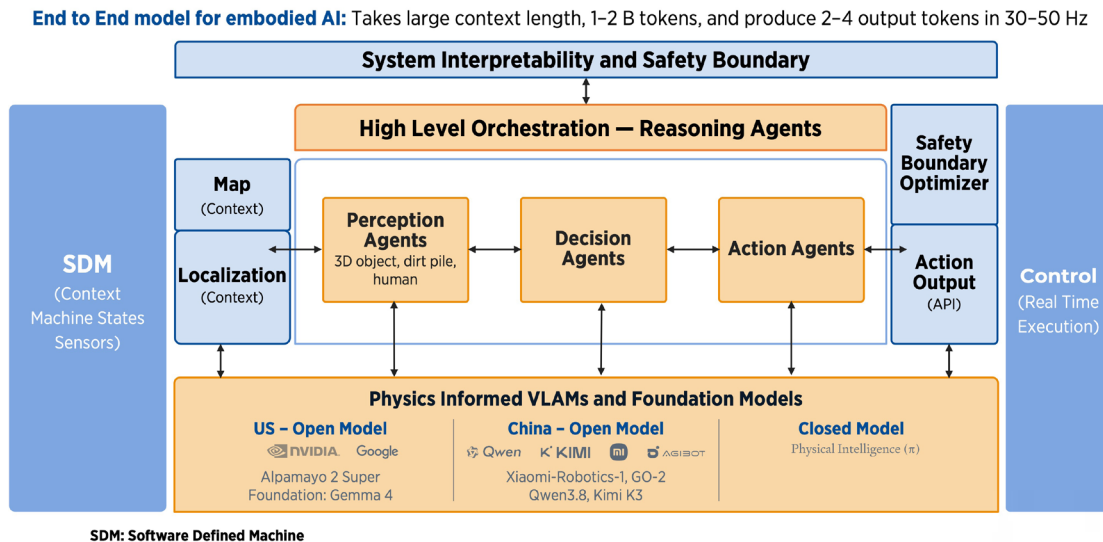


Figure 1: End-to-end model

Unified Architecture: DDS + Semantic Layer + Agents

Built on DDS, RTI Connex[®] provides the foundation. Data-centric architecture is what distinguishes this approach — specifically, applications publish and subscribe to named, typed data streams rather than sending messages to endpoints. An AI agent does not need to know which application produced an observation, or where it is — Connex delivers the data and enforces the delivery guarantees that the agent has specified.

When a new sensor or machine joins, its data becomes available to every agent whose subscriptions match. When a machine leaves, Connex detects the departure so that downstream agents can adapt, such as rerouting to alternative data sources or degrading gracefully, to avoid cascading failures. This is the property that makes the semantic and agent layers viable at industrial scale.

The semantic processing layer subscribes to raw data streams from Connex, normalizes and extracts features, and builds the indexed knowledge representations that provide operational meaning to the data.

The agent layer consumes semantically enriched knowledge, composing task chains at runtime, and feeding actuation commands and adaptive tasking back to the machines and sensors that execute them. The contextualized knowledge enables capabilities such as cross-domain correlation, anomaly detection, and adaptive response planning that would otherwise be impossible from the sensor data alone.

Instead of hard-coding pipelines or centralizing all logic in the cloud, services advertise their capabilities and constraints and the framework forms, monitors, and adapts chains on the fly under policy control. This approach preserves auditability and oversight, converting AI from monolithic applications into reusable, composable capabilities.

Connex's Quality-of-Service (QoS) features enable safety-critical control commands to be prioritized without application-level traffic management. An agent that requires the last-known-good position of every vehicle on site would use a specified QoS policy to receive current state information immediately upon joining, without querying a database or waiting for the next publication cycle.

Two feedback loops close the architecture. An adaptive tasking loop allows agents to re-task sensors and modify data collection based on evolving operational needs — for example, an agent that detects an anomaly can quickly publish new tasking parameters. In addition, a continuous learning loop feeds operational outcomes back into the semantic layer. These feedback paths transform the system from a static inference pipeline into an adaptive architecture that improves with operational experience.

System Properties: Scalability, Discoverability, Security

Ideally, the architecture should be able to shift from a single autonomous vehicle to a coordinated fleet of heterogeneous machines without re-architecting the communication or agent infrastructure. As machines join or leave a site, Connex adapts dynamically through a peer-to-peer architecture with no central broker, eliminating any central point of failure for robust system behavior.

This is enabled by automatic discovery of participants, topics, and data types at the middleware level — entities announce themselves and discover their peers without any central registry or directory service. The agent layer extends this with capability-based discovery: Agents advertise what they can do and what they need, and the framework matches capabilities to requirements at execution time. When a new machine arrives on site with its own embedded agents, those agents are automatically integrated into

the operational fabric without any pre-engineered integration. This enables multi-vendor, multi-OEM fleet operations.

Security is pervasive and granular and is built into the DDS standard itself, so it cannot be bypassed by application code. Authentication, encryption, and access control govern which agents can read data and which can issue commands. Security boundaries follow the data, so that agents from different OEMs can collaborate and share data without either entity exposing or allowing access to the other's model internals or operational parameters.

Simulation and Digital Twin Integration

Applying AI-driven autonomous behavior to physical equipment operating in unstructured, safety-critical environments means that rigorous simulation and testing is required.

Here, the production agent chains can run in simulation without modification. Simulations can use the same QoS topics as their physical counterparts, so that AI agents aren't forced to distinguish between simulated or live data. This eliminates the re-integration and re-validation required when simulation and production environments use different communication stacks. Recorded scenarios, regression suites, and edge-case libraries all remain valid across the lab-to-field life cycle.

Digital twins enable continuous operational monitoring and subscribe to the same telemetry as physical assets. Operating against a predictive model rather than physical actuators makes it possible to evaluate proposed actions, detect drift between expected and observed behavior, and provide an always-current baseline against which anomalies become immediately visible. Connex QoS mechanisms ensure that data flows do not compete with safety-critical control traffic, and discovery allows instances to be spun up or down dynamically.

Deterministic test hooks enable continuous validation. This approach allows recorded data streams to be replayed through agent chains with exact reproducibility. In addition, automated regression suites can simulate edge cases that are dangerous or expensive to create physically, while validated chains can be promoted from simulation to live through a controlled, auditable process. This shortens the cycle from capability development to field deployment, while maintaining safety assurance and auditability.

Hardware Acceleration and AI Infrastructure

The models and agents in this architecture impose compute demands that conventional industrial controllers cannot meet. The architecture addresses this by decoupling compute placement.

Execution can occur at the edge, in the lab, or across a distributed fleet. Connex provides abstraction from physical hardware, spanning all three tiers without protocol translations or middleware boundaries, so that placement is an operational optimization with workloads placed where they make operational and economic sense. The framework integrates heterogeneous agents, foundation models, and local specialized models, while preserving the timing determinism required for safety cases. Furthermore, the framework exposes a vendor-neutral surface for integration instead of binding operators to a single cloud stack — this eliminates the integration boundaries that typically fragment industrial systems.

The edge tier encompasses compute resources that are co-located with the physical equipment they serve running safety critical workloads. If connectivity to the site network or cloud is degraded or lost, agents can continue operating with their local models and local data providing operational resiliency.

The site tier encompasses workloads that require data from multiple machines simultaneously, but cannot tolerate the latency or connectivity risk of a remote cloud for fleet traffic management and site-wide planning sequencing. Vehicles join and leave automatically with no operator intervention required.

Human operators employ visualization systems using the same Connex data fabric that the autonomous agents use. This shared data environment makes meaningful human oversight of operations possible.

Meanwhile, the cloud tier runs fleet-wide analytics, aggregating data across multiple sites to identify patterns invisible from any single vantage point. Model training and fine-tuning pipelines consume the operational data that flows up from edge and site tiers to improve the perception, planning, and decision models deployed back to the field. Large-scale simulation environments enable thousands of scenario variations to be exercised against the same agent chains that will run in production.

System Reference Architecture

The system reference architecture presented here is a **logical system architecture**. The same pattern can apply to a factory, mine, or other physical operating environment. The system connects to fixed or mobile assets, sensor and telemetry systems, control and enterprise systems, and human operators.

These systems connect through a **connector layer**. The connector layer includes protocol and vendor adapters, batch importers, canonical mappers and health monitoring. The purpose of this layer is to convert fragmented operational data into clean canonical events. Those events flow into the core platform.

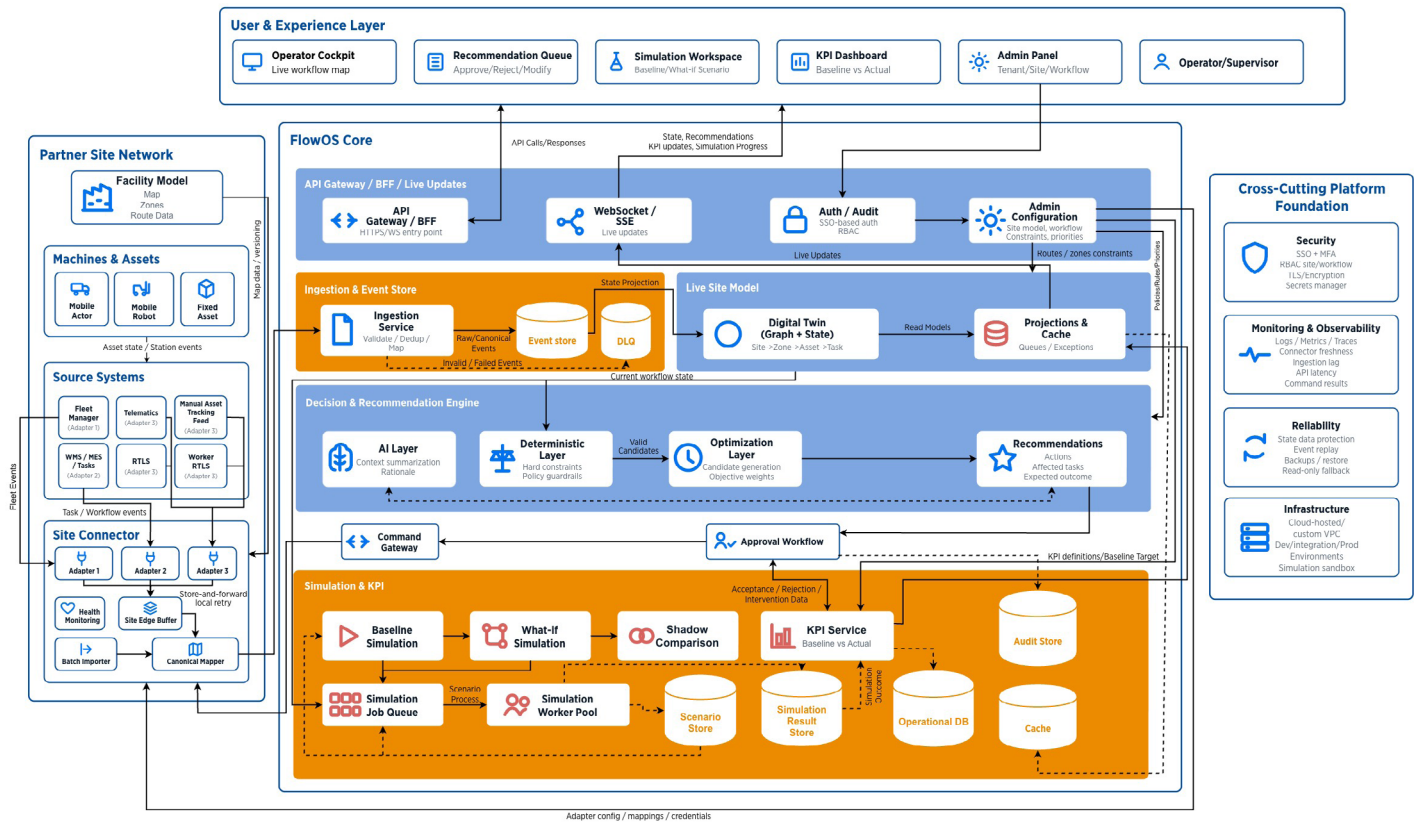


Figure 2: Logical system architecture

Conclusion

The convergence of real-time distributed systems, semantic data modeling, end-to-end model deployment, and distributed AI agents represents a step-change in how industrial systems will be designed, deployed, and operated.

The boundary between tiers is operationally fluid. A workload that normally executes at the site tier can be migrated to the cloud during a period of high site compute load or pushed to the edge during a connectivity outage, all without changing the agent, its interfaces, or its data contracts. New deployment patterns can be evaluated in simulation and promoted to production through the same validated pipeline used for any other agent chain update.

Heavy Industry is ready to adopt this shift. The reference architecture base layer described in this paper can enable this shift in capability.

RTI provides the real-time, data-centric communication fabric that makes large-scale distributed industrial systems practical, while **VerticalAI** provides the Heavy Industry AI system architecture that helps to bring intelligence into those systems. The architecture does not dictate where intelligence lives; instead, it provides the substrate to let intelligence live wherever it is most effective.

This whitepaper was produced jointly with Bibhrajit Halder. Halder is the Founder & CEO of VerticalAI. He started in autonomy with the DARPA Grand Challenge, and has developed autonomy solutions for Caterpillar, Ford, Faraday Future and Apple. Halder founded SafeAI in 2017 and VerticalAI in 2026.

ABOUT RTI

RTI is the real-time data streaming company for intelligent distributed systems. RTI Connex[®] software is the critical nervous system for over 2,000 designs across Aerospace and Defense, MedTech, Automotive, and Robotics.

RTI, Real-Time Innovations and the phrases "RTI Runs a Smarter World" and "Your systems. Working as one," are registered trademarks or trademarks of Real-Time Innovations, Inc. All other trademarks used in this document are the property of their respective owners. ©2026 RTI. All rights reserved. 50061 V1 1026