HIVEMQ

The Blueprint for Agentic AI in Industrial Operations

Operationalizing Autonomous Intelligence at Scale

Category
Technical White Paper

Published
December 2025

Author
Kudzai Manditereza

Executive Summary

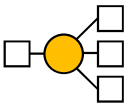
For decades, industrial companies have pursued the same fundamental goal: building production environments that balance resilience, efficiency, and responsiveness. While this challenge remains just as critical today as it was generations ago, the rules of the game have changed, dramatically.

The traditional approach of relying on steady, incremental improvements, simply cannot keep pace with the unprecedented scale and speed of change now required to operate in an increasingly volatile, disruptive and competitive environment. What's needed, instead, are production systems that don't just gradually get better, but those that are capable of dynamic adaptation and continuous learning.

For this reason, Agentic AI has become increasingly attractive for industrial companies. It naturally fits as the next evolutionary step for accelerating workflow-based improvements to reach the ultimate goal: building production environments capable of agile learning and dynamic adaptation to changing conditions.

However, the path from concept to operational reality remains unclear for most organizations.

This paper provides a systematic framework for operationalizing autonomous intelligence at scale across industrial enterprises. It addresses three fundamental challenges that prevent industrial companies from capturing the full value of Agentic AI:



First, the infrastructure gap. Traditional batch architectures can't support real-time responsiveness. Agents need continuous, contextualized data to act before problems escalate.

Second, the intelligence gap. Data access isn't enough. Agents need semantic understanding of equipment relationships, parameter impacts, and operational constraints for autonomous decisions.

Third, the trust gap. Autonomous systems affecting physical operations risk equipment, compliance, and safety. Governance frameworks enable deployment while maintaining control.

Table of Contents

Section 1: Establishing Real-Time Data Flow for Agentic AI Through Streaming and Unified Namespace

Why Agentic AI in Industrial Operations Needs Streaming Data

Fundamentally, agentic AI in industrial operations is about autonomy and agility. Detecting anomalies, making decisions, and adapting to changing conditions as they emerge. Like human operators who react to machine conditions as they happen rather than at fixed intervals, AI agents must respond instantly, drawing on domain knowledge while adapting to live

Streaming data from IoT devices, PLCs, and MES systems enables timely, autonomous decisions grounded in current operational knowledge. Unlike batch processing with delayed historian queries, streaming provides continuous, up-to-the-moment machine telemetry, process variables, and quality metrics, empowering agents to act in real time rather than on outdated insights.

While the benefits of streaming data for agentic AI are clear, many industrial organizations still face a critical challenge: how to actually implement this capability across complex, legacy environments. Unlocking real-time data for agentic AI requires deliberate, phased infrastructure transformation that respects operational technology constraints while building scalable, intelligent systems.

This section outlines a three-step roadmap: digitization, real-time flow, and shared semantic context, to progressively unlock capabilities required for agentic operations at scale.

Step 1: Digitize Operations to Build a Real-Time Foundation

Before streaming data, you must ensure it exists digitally. Many operations still rely on manual collection, paper records, or isolated systems. Begin by assessing your current state: identify which assets and processes are already instrumented versus those remaining analog or disconnected. Older equipment may require retrofitting with sensors and edge devices to capture operational data.

The goal isn't just connectivity, it's creating a complete digital representation of physical operations, including process variables, equipment states, quality measurements, and contextual metadata.

Critical mindset shift: Rather than collecting data for specific use cases, create a comprehensive, real-time digital model of your entire enterprise. When a quality issue emerges, AI agents need complete operational context to determine impacts on delivery, suppliers, schedules, and customers. This intelligent coordination requires access to real-time data across manufacturing, supply chain, quality, and planning systems.

Industrial Data Architecture Capabilities Essential for Scaling Agentic Operations

Seamless OT/IT Integration:	Quickly add new digital tools and onboard legacy systems without weeks of custom engineering. Heavy customization makes your transformation slow and unsustainable.
Unified Data Accessibility:	Standardize data, add context, and unify it across your organization through a common access layer, enabling AI agents to access a consistent, shared view of operations.
Scalable Data Architecture:	Seamlessly scale from one line to many, from single facility to global network. Plug in new assets or sites without rearchitecting your system.
High Data Availability:	Ensure resilient, fault-tolerant infrastructure. When data pipelines break, agentic AI systems become blind to operational reality.

Step 2: Adopt Event-Driven Architecture for Real-Time Adaptability

Once operations are digitized, shift from traditional point-to-point integrations to an event-driven architecture. Most organizations still use rigid, custom-engineered pipelines that are hard to scale and poorly suited for agentic AI's dynamic needs.

The MQTT Broker: Your Data Infrastructure's Central Nervous System

Instead of tight coupling through direct connections, systems interact through a common broker using a publish-subscribe model. This architectural decoupling delivers:

- **Seamless OT/IT Integration:** Systems simply publish or subscribe, no complex, bespoke integrations
- **Unified Data Accessibility:** All data flows through shared infrastructure, available in real time
- **Scalability:** Add new machines, applications, or AI agents without architectural changes
- **High Availability:** Clustering and failover ensure continuous data flow

Why Events Matter: The Time Value of Information

Event-driven architecture transmits data as discrete events reflecting what's happening now, sensor updates, machine state changes, quality measurements. This is critical because agentic AI depends on the time value of information: the sooner an agent observes and acts, the greater the impact.

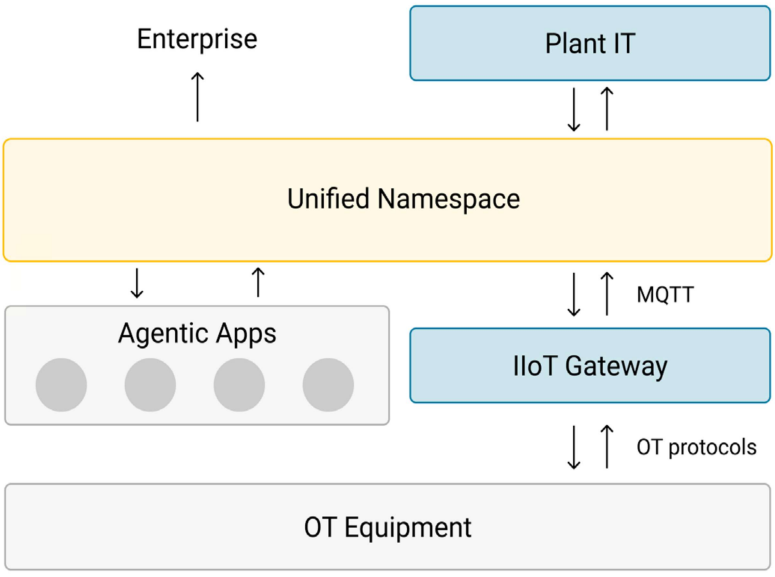
Events range from short-term (line breakdowns, anomalies) to long-term (requirement changes, process modifications). Each carries potential business impact with a narrow response window. Traditional architectures using polling or batch integration introduce delays that reduce AI effectiveness. Event-driven architecture lets AI agents see events as they happen, enabling real-time reasoning and intervention while still relevant.

Step 3: Establish a Unified Namespace for Semantic Consistency

Digitization creates data availability. Event-driven architecture enables data flow. But for agentic AI to thrive, you need semantic consistency—a common understanding of what data means across your enterprise. This is where the Unified Namespace (UNS) becomes transformative.

A UNS organizes all operational events in your MQTT infrastructure using a standardized structure aligned with your business context. Rather than each system maintaining its own naming conventions, the UNS provides a single, enterprise-wide semantic layer. This hierarchy typically follows ISA-95 but can be extended for domain-specific context.

For example, instead of cryptic tags like "PLC_007_AI_023," data is published to meaningful topics like "Enterprise/PharmaSite/Building2/TabletLine1/BlendingUnit/BlendUniformityPV." AI agents subscribe to hierarchical topics, receive data in predictable formats, and can dynamically discover available data streams and understand their context without manual integration.



Most critically, the UNS serves as the foundation for interoperability. Whether deploying AI agents, feeding historians, supporting MES applications, or enabling analytics, they all consume from the same semantic layer. This eliminates data silos and enables an open architecture where best-of-breed applications work together seamlessly.

Example Use Case of Data Streaming for Agentic AI in Industrial Operations.

During a production run, tablet weight trends toward specification limits while energy consumption spikes. In traditional operations, this might go undetected until the next quality check, causing out-of-spec product and delays.

With the three-step infrastructure:

- **Real-Time Awareness:** Digitized sensors continuously stream blend uniformity, compression force, tablet weight, coating thickness, temperature, humidity, and energy consumption.
- **Event-Driven Intelligence:** A monitoring agent immediately detects the trending anomaly through MQTT architecture, capturing precise operational context: which product, batch phase, and how conditions compare to normal ranges.
- **Semantic Understanding:** When the AI agent subscribes to the UNS topic, it receives not just the current value but full context: specification limits, batch identity, recipe parameters, and relationships to upstream and downstream processes.
- **Collaborative Intelligence:** The monitoring agent shares anomaly context across specialized agents. The quality agent retrieves SOPs and batch records. The process optimization agent analyzes correlations between feeder speed, dwell time, and weight variability. The predictive maintenance agent identifies developing vibration signatures correlating with the energy spike.
- **Continuous Improvement:** Agents monitor outcomes and compare results against predictions. This feedback continuously refines decision models and expands operational intelligence.

These foundational capabilities **transform agentic AI** into **operational reality**, enabling autonomous, intelligent responses that preserve quality, optimize performance, and adapt continuously to changing production conditions.

Section 2: Enabling Contextual Intelligence for Agentic AI in Industrial Operations

The infrastructure outlined in Part 1, digitization, event-driven architecture, and the Unified Namespace, represents a crucial transformation in how industrial data flows through your enterprise. With these capabilities in place, AI agents and humans can now access real-time operational data through a semantically organized hierarchy. Data streams continuously from edge to enterprise, delivered with consistent naming conventions and hierarchical context.

However, this semantic hierarchy, while essential, is not sufficient to fully operationalize autonomous, agentic AI. The gap stems from the fact that having data is not the same as having intelligence. To reason, coordinate, and act safely across lines, sites, systems, and geographies, agents need linked meaning, not just well-named streams.

This section presents the next phase of our agentic operations blueprint: building the semantic foundation that transforms real-time data flow into **distributed data intelligence**. We'll explore why semantic graphs powered by domain-specific ontologies are essential, how they enable AI agents to reason with industrial knowledge, and the governance structures required to make this intelligence trustworthy, secure, and continuously improving.

The Limitations of Semantic Hierarchy in Industrial Data Intelligence

The Unified Namespace solves the problem of data accessibility and basic semantic organization. It tells you where data originates in your physical plant and provides consistent naming. But it doesn't answer the deeper questions that agentic operations demand:

- Why does this equipment failure affect downstream processes in specific ways?
- How do process parameters relate to quality outcomes across different product formulations?
- What are the regulatory implications of adjusting this process variable?
- Which safety protocols must be validated before an agent can autonomously modify setpoints?

These questions demand semantic richness, a deep, structured understanding of relationships, constraints, rules, and context that govern industrial operations. They require transforming data into actionable intelligence through an ontology-driven semantic graph that captures decades of engineering knowledge, regulatory requirements, and operational expertise.

To bridge this gap, industrial organizations must add a semantic graph layer that captures not just where data lives in your plant hierarchy, but what it means in the context of your operations, how it relates to other operational knowledge, and why it matters for decision-making.

Agentic AI Operations Use Cases Enabled by Graph Relationships

Unlike hierarchical models, semantic graphs capture complex, non-linear relationships reflecting the true intelligence of manufacturing processes: how materials flow, which processes depend on others, which parameters influence outcomes, and why operations halt. This makes AI agents context-aware, enabling them to reason about not just what is happening, but why.

Intelligent Material Flow Management

FeedsInto/ReceivesFrom shows physical flow sequence independent of hierarchy:

APIStorage → BlendingUnit → TabletPress_01 → CoatingPan_01 → Packaging

This enables AI agents to calculate lead times, identify bottlenecks, and predict cascade effects of upstream delays.

SuppliesTo/FeedsFrom shows direct supply relationships:

- APIStorage.SuppliesTo → BlendingUnit
- Silo_01, Silo_02.SuppliesTo → BlendingUnit
- BlendingUnit.FeedsInto → TabletPress_01

When API storage drops below minimum, monitoring agents instantly know the blending unit will run out in ~15 minutes, triggering alerts before interruption. When the tablet press reports feed interruption, root cause agents immediately query FeedsFrom to identify whether it's API depletion, excipient blockage, or blending malfunction.

Dynamic Production Capability Management

UsesFormulation links production line to product: TabletLine1.UsesFormulation → Formulation_Aspirin_325mg_v2.3

The line doesn't contain the formulation but is validated to produce it. The same formulation can run on multiple qualified lines. Planning agents query: "Which lines can manufacture this product?"

Autonomous Genealogy & Recall Response

Consumes/Produces tracks material genealogy for compliance and recalls:

Batch_20241113_001.Consumes → [MaterialBatch_API_Supplier_ACE_Lot_4427, MaterialBatch_Excipient_Starch_Lot_8821, ...]

Batch_20241113_001.Produces → FinishedGoodsLot_FG_20241113_A

If an API supplier recalls Lot ACE_4427, traceability agents execute automated queries:

- Backward: "Which batches consumed this lot?" → Batch_20241113_001
- Forward: "Which finished goods contain these batches?" → FG_20241113_A
- Distribution: "What's the distribution status?" → Queries ERP for shipments
- Documentation: Auto-generates recall impact assessment per 21 CFR Part 11

Full traceability in seconds rather than hours of manual investigation.

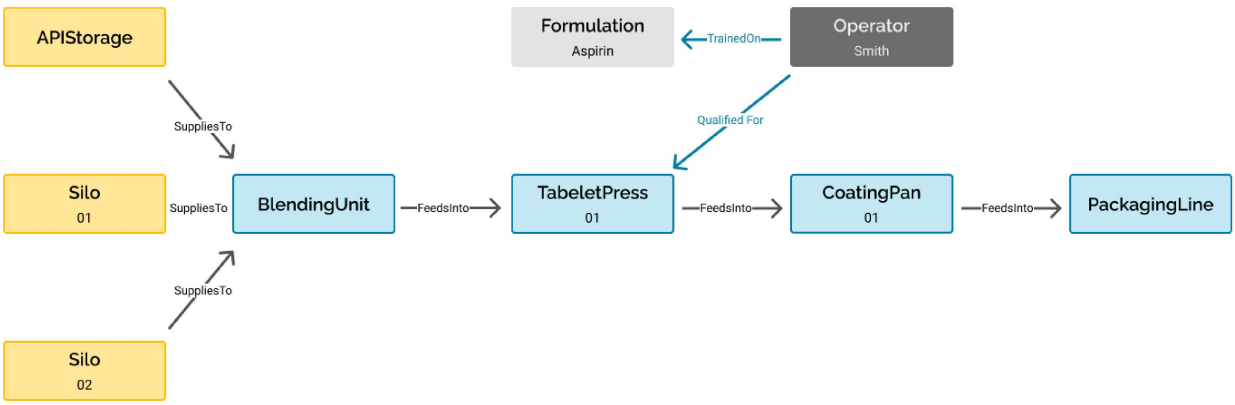
Intelligent Workforce & Asset Coordination

QualifiedFor tracks human resource qualifications:

Operator_J_Smith.QualifiedFor → [TabletLine1, TabletLine2, FormulationTraining_Aspirin, FormulationTraining_Ibuprofen]

Scheduling agents query: "Which operators can run Tablet Line 1 on night shift for Aspirin?" Returns only personnel with both equipment and formulation qualifications. When planning new product introductions, agents automatically identify which operators need additional training and estimate timeline before production can commence, preventing scheduling conflicts.

Below is a visualization of relationships semantic graphs enable you to capture.



Why Ontologies Matter for Agentic AI in Industrial Operations

For semantic graphs to be effective in industrial settings, they must be grounded in domain-specific ontologies. An ontology is a formal, structured representation of knowledge within a domain, defining what exists (entities), how things relate (relationships), and what each element means in context. It's a blueprint for shared understanding where every term, data point, and relationship has a clearly defined role.

Industry standards like ISA-95 (manufacturing systems) and Common Information Model (CIM) (energy systems) serve as foundational ontologies, providing consistent vocabulary and relationship structures for interoperability. When a data point is labeled "material lot," ISA-95 ensures unambiguous meaning for both humans and AI agents.

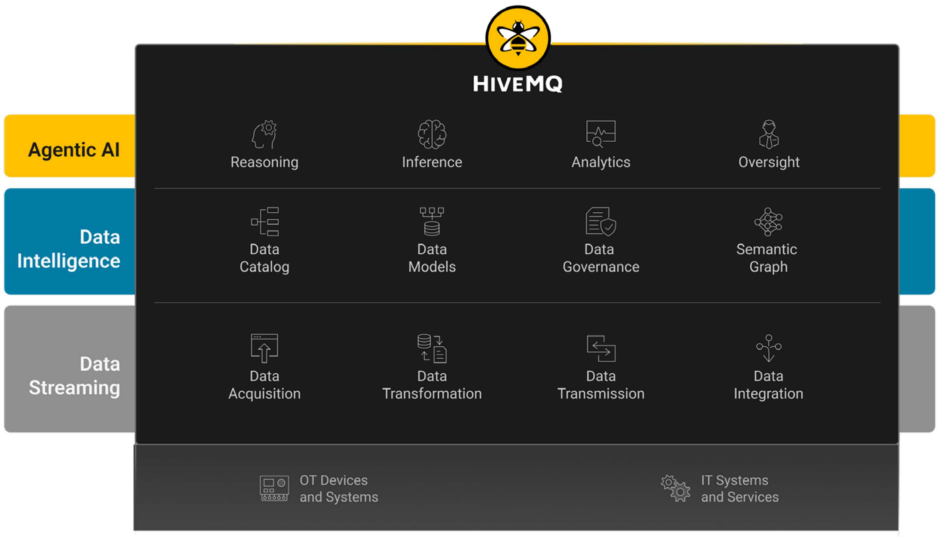
Why This Matters

This structured representation allows agentic AI to infer, reason, and act based on meaningful relationships. For example, if an ontology defines that a TabletPress must recordCompressionForce for each Batch, an AI agent can proactively identify when this parameter is missing, outside acceptable ranges, or inconsistent with the expected formulation.

Ontologies make data machine-interpretable, enabling AI agents to reason about relationships, dependencies, and intent rather than merely processing raw data. This not only enhances current data utility but future-proofs your ecosystem. As AI agents become more autonomous, their effectiveness will depend on access to well-defined domain semantics.

Building the Semantic Layer for Agentic AI Operations

The strategic value of ontology-driven semantic layers is clear: they transform raw data flow into actionable intelligence, enable AI agents to reason with industrial knowledge, and coordinate autonomous operations across complex manufacturing environments. However, understanding the "why" and "what" is only the beginning. The critical question remains: How do you actually implement this capability in your organization?



The answer lies in selecting or building a data platform with at least four of the essential capabilities below, that work together to create, maintain, and operationalize your semantic foundation.

1 Data Modelling: Making Data Agentic AI-Ready

Most plant data isn't inherently AI-ready. Even when labeled and mapped through the Unified Namespace, raw sensor telemetry cannot be directly related across plants, systems, or production contexts. The first step in building semantic intelligence is transforming raw operational data into standardized, AI-ready structures through consistent data models.

Data models must incorporate rich contextual metadata that goes far beyond simple measurements to capture what data means operationally. Critically, these models must automatically adapt as new fields, tags, or process changes are introduced, keeping your foundation always current without requiring manual re-engineering.

By converting all OT data into a small number of scalable, standardized schemas (typically a few core models covering equipment, processes, materials, and quality), manufacturers establish a consistent foundation for improvement across every line and plant. Because every site uses the same core models, agentic AI insights proven in one location can be applied enterprise-wide.

2 Semantic Graphing: Capturing Relationships and Intelligence

Semantic graphing capabilities enable you to map the relationships between your data models; turning the isolated data structures into a connected network of operational knowledge. This allows you to bring your ontology to life.

In short, data models specify the structure and context of individual entities, while the semantic graph defines how those entities relate, interact, and influence each other.

3 Data Catalog: Enabling Discovery and Transparency

As your semantic layer grows to encompass hundreds of data sources, thousands of entities, and complex relationship networks, discoverability becomes critical. AI agents need mechanisms to discover what data exists, what it means, and how to access it. Human data scientists, engineers, and operators need the same capabilities.

The data catalog serves as the navigational layer for your semantic foundation. It provides a searchable, browsable interface to your ontology, data models, and available data products.

4 Data Governance: Ensuring Trust, Quality, and Compliance

Agentic operations create a new challenge: autonomous systems making operational decisions based on data-driven intelligence. This autonomy demands trustworthy data, accurate, complete, timely, secure, and compliant. Without rigorous governance, AI agents can amplify data quality issues into operational problems.

Data governance for agentic operations must address three critical dimensions: Quality Rules and Validation, Lineage and Traceability, and Security and Access Control.

Section 3: Identifying Agentic AI Use Cases for Operational Efficiency in Industry

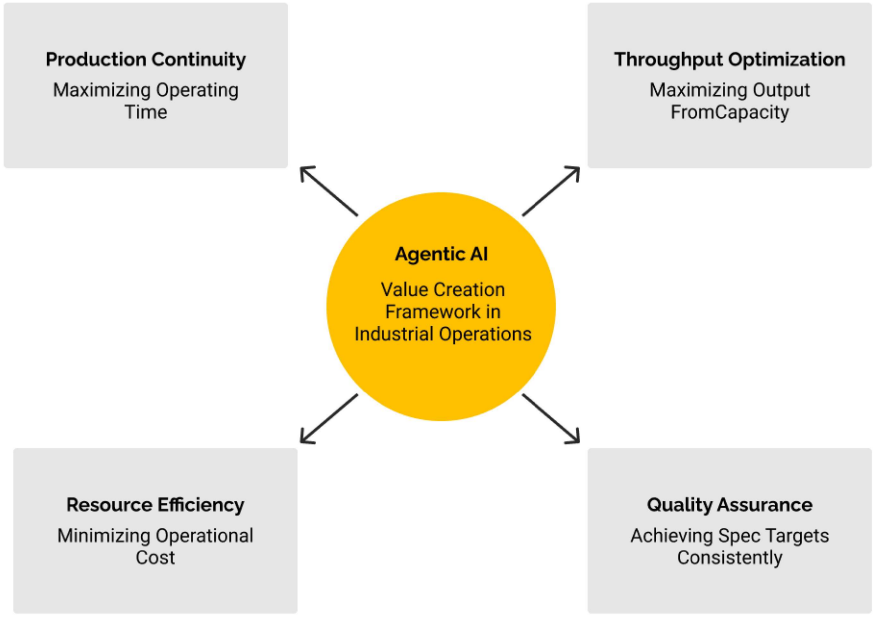
With real-time streaming infrastructure and semantic intelligence in place, your organization now possesses AI-ready operational data. The Unified Namespace delivers continuous data flow, and your semantic graph captures operational relationships.

This foundation represents a significant transformation in technical capability. Yet capability alone does not drive business outcomes. The critical question facing industrial companies is not whether they can deploy agentic AI, but where they should deploy it, at what level of autonomy, and with what expected return.

This section addresses the strategic challenge of translating your agentic AI infrastructure into operational impact. We present a systematic approach for identifying high-value opportunities, defining the appropriate level of agent autonomy for each use case, and building organizational readiness to scale from initial deployments to enterprise-wide autonomous operations.

Four Strategic Value Domains for Agentic Operations

Business outcomes from agentic AI cluster around four interconnected value domains. Each domain represents a distinct way that autonomous intelligence creates competitive advantage in manufacturing operations:



Production Continuity: Maximizing Operating Time

Equipment that isn't running produces nothing. Traditional reactive maintenance accepts unplanned downtime as inevitable. Agentic operations transform this through coordinated asset health management. Multi-agent teams continuously monitor equipment telemetry, identifying degradation patterns signaling impending failures. Rather than isolated alerts, agent networks reason about operational context: equipment dependencies, at-risk production schedules, available maintenance resources, and optimal intervention timing.

Representative Use Cases: Coordinated predictive maintenance, automated spare parts optimization, equipment health monitoring with autonomous escalation, maintenance window scheduling based on production priority.

Throughput Optimization: Maximizing Output from Available Capacity

Most production lines operate below theoretical capacity due to bottlenecks, changeover losses, or suboptimal parameter settings. Each percentage point of improved utilization translates directly to additional output without capital investment.

Agentic AI excels here because the challenge is fundamentally about coordination across multiple simultaneous systems. Autonomous agents monitor real-time bottleneck migration, adjust process rates to balance flow, optimize changeover sequences, and coordinate material handling to prevent interruptions. Unlike traditional MES or scheduling systems, agent teams continuously learn from production outcomes, adapting to current conditions like ambient temperature, raw material characteristics, equipment condition, and operator skill levels.

Representative Use Cases: Dynamic bottleneck identification and mitigation, autonomous changeover optimization, real-time schedule adaptation, material flow coordination.

Quality Assurance: Achieving Specification Targets Consistently

Quality losses represent a double penalty: rejected material costs plus opportunity cost of capacity consumed producing out-of-spec product. In regulated industries, failures also trigger compliance investigations.

Agentic quality management shifts from detection to prevention through continuous process correlation analysis. AI agents learn relationships between process parameters and quality outcomes, detecting subtle drift patterns preceding specification violations. When quality agents identify emerging risks, they coordinate with process optimization agents to implement corrective adjustments before defects occur.

Representative Use Cases: Continuous quality prediction and process adjustment, automated root cause analysis, first-pass yield optimization, regulatory documentation generation.

Resource Efficiency: Minimizing Operational Costs

This domain addresses cost optimization across energy consumption, raw material utilization, labor efficiency, and working capital—ongoing expenses that compound across every production cycle.

Agentic approaches leverage the same capabilities deployed for throughput and quality but redirect them toward cost minimization. Energy management agents shift consumption to lower-rate periods without impacting production. Material optimization agents minimize waste and scrap. Inventory management agents balance carrying costs against stockout risks.

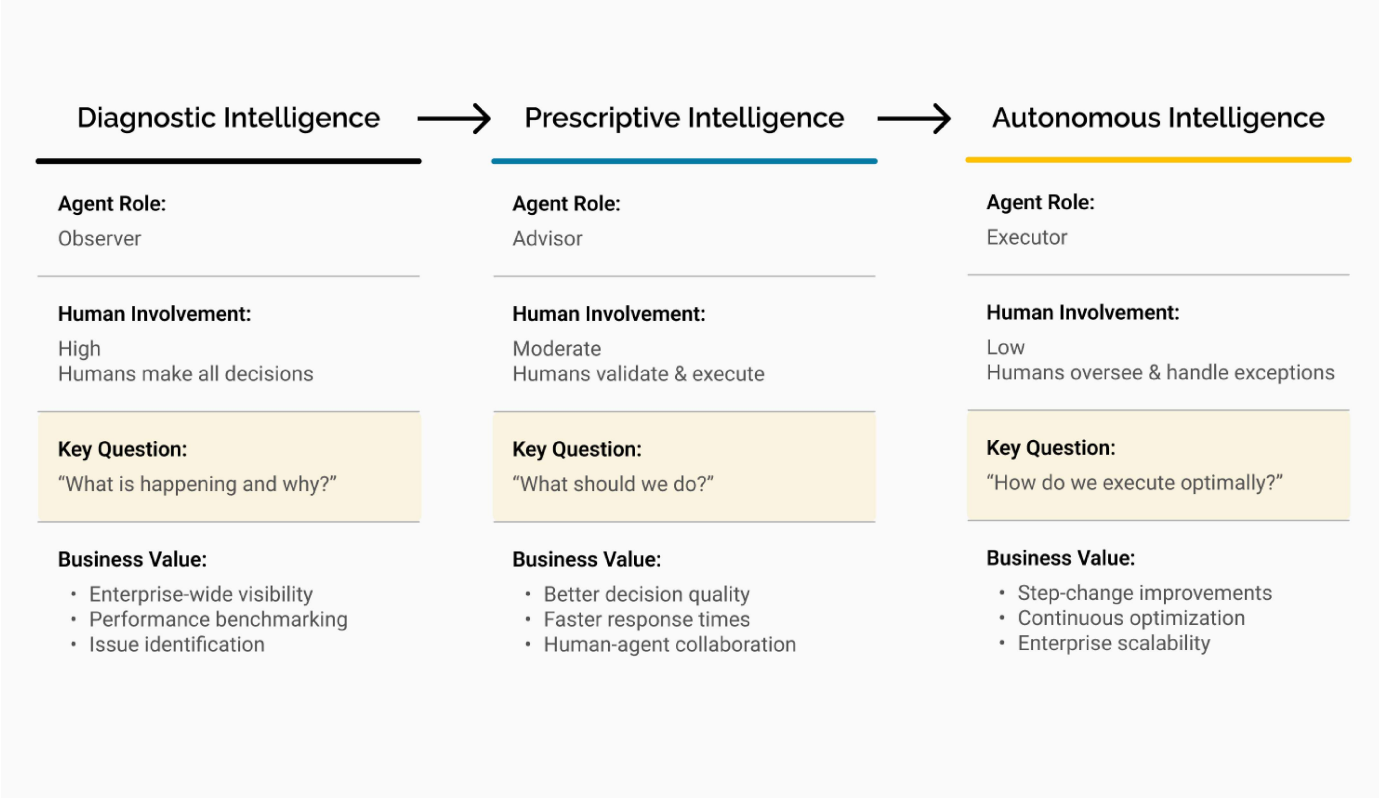
Critically, resource efficiency agents must coordinate with other value domains to avoid suboptimization.

Companies need governance frameworks balancing competing objectives through explicit cost functions.

Representative Use Cases: Autonomous energy optimization, predictive inventory management, waste and scrap minimization, labor scheduling based on skill requirements.

A Maturity Framework for Agentic Operations in Industry

Identifying which value domain to target represents only the first decision. The second critical choice is determining the appropriate level of agent autonomy for each use case. This decision must balance potential business impact against organizational readiness, technical complexity, and operational risk.



We propose a three-stage maturity framework that enables organizations to progressively expand agentic capabilities while building technical validation and organizational trust at each stage.

01 Stage 1: Diagnostic Intelligence - Establishing Visibility

AI agents serve as intelligent observers, continuously monitoring operational data streams, detecting anomalies, identifying correlations, and alerting humans to situations requiring attention. They answer: "What is happening and why?"

This foundation establishes baseline performance metrics and builds organizational familiarity with how AI agents reason about operations.

Agent Capabilities: Real-time monitoring against spec limits, automated event detection and correlation, pattern recognition, performance benchmarking across shifts/lines/facilities.

Business Value: Makes invisible operational dynamics transparent. Leaders gain enterprise-wide visibility into cost drivers and improvement opportunities.

Organizational Readiness: Minimal, agents don't make decisions, so focus is on training personnel to interpret insights.

02 Stage 2: Prescriptive Intelligence - Recommending Optimal Actions

Agents transition from observation to active guidance, generating specific recommendations for humans to execute. The question evolves to: "What should we do to optimize this?"

This stage delivers the highest ROI for most manufacturers by amplifying human expertise. Agents analyze thousands of variables simultaneously, identifying optimization opportunities beyond human cognitive bandwidth. Critically, it creates a feedback loop: when operators validate recommendations, agent models improve and operators develop intuition for when to trust agent reasoning.

Agent Capabilities: Multi-variable optimization, scenario analysis, root cause determination, proactive opportunity identification.

Business Value: Significant improvements through better decision quality and faster response. Human-agent collaboration enables optimization impossible for either alone.

Organizational Readiness: Moderate, requires training operators to interpret recommendations and establish feedback mechanisms capturing decisions and outcomes.

03 Stage 3: Autonomous Intelligence - Executing Decisions Independently

Multi-agent teams monitor, identify opportunities, decide, execute, and learn, all without human approval for routine operations. Humans maintain oversight and intervene only for exceptions or high-risk scenarios.

This stage requires the most sophisticated infrastructure but delivers transformational capabilities: second-level response times, complex multi-system coordination, and continuous operation without fatigue. However, reliable autonomy depends on systematic exception handling, agents must recognize decision boundaries and escalate edge cases.

Agent Capabilities: Closed-loop process control, multi-agent coordination, self-learning, proactive exception escalation.

Business Value: Step-change improvements in productivity, quality, and efficiency. Once agents master optimization at one facility, intelligence deploys enterprise-wide within weeks versus years for human training.

Organizational Readiness: Extensive, requires robust change management, comprehensive testing in simulation, governance frameworks defining authority boundaries, and cultural transformation to autonomous system supervision.

A Taxonomy of Agentic Services

Understanding value domains and maturity stages provides strategic direction, but translating this into action requires a granular view of what agents actually do. This taxonomy describes specific agent capabilities organizations can deploy, organized around fundamental operational questions: What is our current state? Why are we experiencing this performance? What will happen next? What should we do? How do we execute optimally?

Monitoring Agents: Continuous State Assessment

Establish real-time awareness of operational conditions. Rather than passive data collection, these agents actively interpret streaming telemetry against specification limits, baselines, and predictive models to determine what deserves attention.

Core Capabilities: Multi-source data aggregation via UNS, automated anomaly detection, event correlation, intelligent alerting that filters noise and prioritizes critical conditions.

Diagnostic Agents: Root Cause Determination

When monitoring agents detect issues, diagnostic agents determine why by analyzing operational data to identify causal relationships. They traverse the semantic graph exploring equipment dependencies, process interactions, and material genealogy to isolate root causes.

Core Capabilities: Automated failure mode analysis, quality correlation analysis linking parameters to product attributes, material traceability, temporal pattern recognition.

Predictive Agents: Outcome Forecasting

Project future conditions based on current state and historical patterns, enabling proactive intervention before problems manifest and transforming reactive operations into anticipatory operations.

Core Capabilities: Equipment failure prediction and remaining useful life estimation, quality outcome forecasting, production schedule feasibility analysis, demand forecasting integrated with supply chain optimization.

Optimization Agents: Parameter Tuning and Resource Allocation

Identify opportunities to improve performance by adjusting process parameters, reallocating resources, or modifying sequences. At prescriptive maturity, they recommend actions; at autonomous maturity, they execute independently.

Core Capabilities: Multi-objective optimization balancing throughput, quality, and efficiency; constraint satisfaction considering equipment limits and safety boundaries; resource allocation; adaptive control.

Coordination Agents: Multi-System Orchestration

The most sophisticated capability—managing interdependencies, resolving conflicts between competing objectives, and orchestrating complex workflows spanning organizational boundaries.

Core Capabilities: Multi-agent collaboration protocols, workflow orchestration across planning/execution/quality/maintenance, conflict resolution, hierarchical decision-making with escalation protocols.

Learning Agents: Continuous Improvement

The meta-layer enabling all other agents to improve continuously. Analyze outcomes from diagnostics, predictions, optimizations, and coordination decisions to refine models and enhance future performance.

Core Capabilities: Reinforcement learning from operational outcomes and feedback, model updating incorporating new understanding, knowledge transfer enabling cross-context insights, performance monitoring tracking agent effectiveness over time.

Section 4: Establishing Governance Frameworks for Agentic AI in Industrial Operations

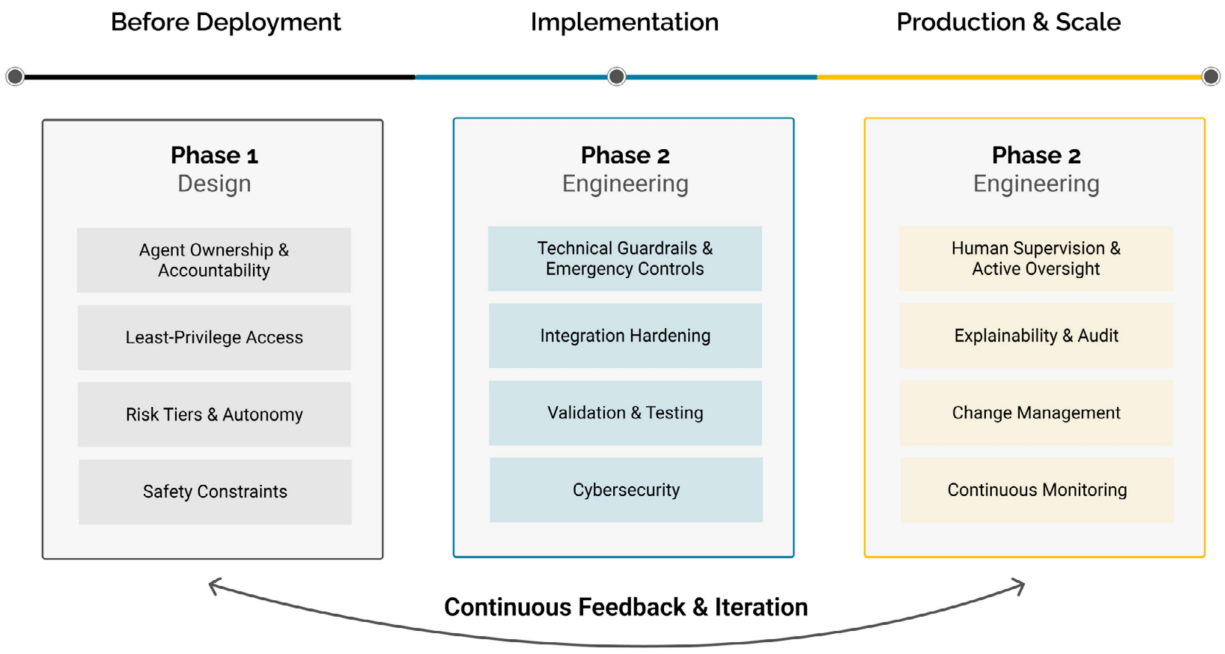
The potential of Agentic AI to dynamically optimize complex industrial operations, autonomously coordinate supply chains, and manage asset health promises unprecedented gains in efficiency, quality, and resilience.

However, deploying these powerful agents within high-stakes industrial environments presents a challenge fundamentally different from traditional IT applications. When an AI system can directly influence physical processes in industrial environments, the risks are no longer confined to data. They extend directly to equipment integrity, production continuity, regulatory compliance, and, most critically, human safety.

This high-stakes reality creates a critical tension for industry leaders: How can an organization harness the transformative power of agentic AI while maintaining absolute operational control?

The answer lies not in hesitant adoption, but in a deliberate, robust framework for governance and technical control built from the ground up. Success requires a new operational discipline that embeds safety, accountability, and oversight into the very architecture of these systems.

This section presents a systematic framework for implementing governance across three critical phases: designing agents with built-in safety constraints, engineering robust technical controls, and operating with continuous oversight and transparency. Each phase addresses distinct governance challenges while building toward a cohesive control environment that enables safe autonomy at scale.



Phase 1: Designing for Controlled Agentic Operations

Effective governance begins before system integration. Organizations must translate operational objectives into agent specifications that explicitly define authority boundaries, safety constraints, and accountability structures.

Establishing Clear Agent Ownership and Accountability

Every AI agent must have an assigned owner, a specific individual accountable for that agent's behavior, decisions, and outcomes. This mirrors how organizations assign responsibility for physical equipment and quality systems.

Effective agent ownership requires three elements:

Authority Documentation: Formal specification of what decisions the agent can make independently, which require human approval, and which are prohibited.

Performance Accountability: Defined metrics and thresholds triggering owner review.

Escalation Protocols: Clear procedures for when agents encounter situations beyond their design parameters, including how agents communicate uncertainty and who receives escalations.

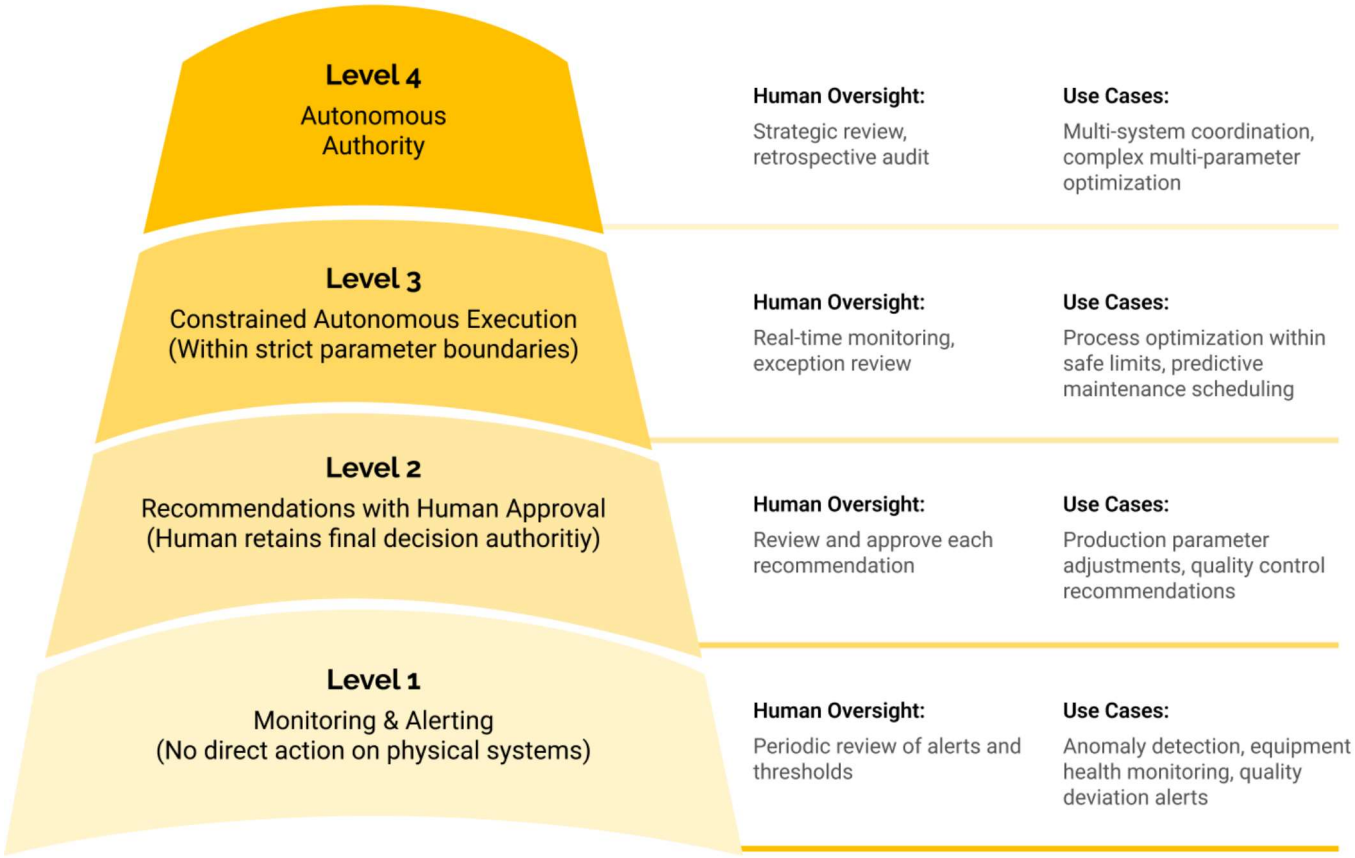
Implementing Least-Privilege Access Controls

Agentic AI eliminates historical OT/IT isolation, creating new attack surfaces. Least-privilege access controls grant agents only the specific data and system access required for their designated functions. These include **Data Access Boundaries**, **System Interaction Limits**, **Temporal Access Restrictions**, and **Network Segmentation**.

Defining Risk Tiers and Autonomy Thresholds

Risk tiering creates graduated autonomy levels matching agent authority to potential impact magnitude.

Level 1	Monitoring and Alerting: Agents observe data, detect anomalies, and notify operators without taking action. Lowest risk; all new deployments should begin here.
Level 2	Recommendations with Human Approval: Agents generate recommendations for qualified personnel to review and approve before execution.
Level 3	Constrained Autonomous Execution: Agents make and execute decisions independently within strictly defined parameter boundaries.
Level 4	Broad Autonomous Authority: Agents coordinate multiple systems and execute significant changes without real-time approval. Reserved for thoroughly validated use cases with comprehensive safety guardrails.



Embedding Safety and Ethical Constraints

Industrial agents must internalize safety rules and regulatory requirements as immutable constraints, behavioral hard limits that shape agent reasoning before decisions are considered. These include **Process Safety Boundaries**, **Regulatory Compliance Rules**, **Equipment Protection Limits**, and **Human Override Authority**.

These constraints require formal verification methods, embedded rule engines evaluating proposed actions against constraint libraries, and continuous monitoring flagging any attempts to operate outside established boundaries.

Phase 2: Engineering Robust Technical Controls for Agentic Operations

Design specifications establish what agents should do and the boundaries within which they must operate. The engineering phase transforms these specifications into implemented systems with technical safeguards that enforce desired behaviors and prevent unintended consequences.

Building Technical Guardrails and Emergency Controls

Every autonomous agent must include emergency intervention capabilities enabling rapid human override or complete system disengagement.

Emergency Stop Functionality: Like industrial equipment emergency stops, agent systems require equivalent capabilities. Operators must instantly suspend agent actions, halt in-progress operations, and prevent new decisions with a single command, accessible from workstations, mobile devices, and potentially physical control room buttons.

Automated Circuit Breakers: Agent systems should incorporate automated safeguards detecting anomalous behavior patterns and autonomously reducing agent authority pending human review. If an optimization agent makes parameter adjustments at substantially exceeding historical frequency, circuit breakers might restrict it to monitoring mode and alert supervisors.

Rollback Capabilities: When agents make suboptimal decisions, operators need the ability to quickly restore previous operational states.

Hardening System Integration Points

Agents interact with industrial systems through integration layers translating decisions into queries, workflows, and commands. Each integration point represents a vulnerability.

Strict Interface Definitions: Every agent-system connection must enforce rigorous input validation, type checking, and range verification. Interface hardening treats all agent outputs as potentially malformed until proven valid.

Transactional Integrity: When agents coordinate actions across multiple systems, implement transaction-like semantics ensuring consistency. If an agent adjusts both feed rate and downstream parameters, both must succeed together or fail together.

Establishing Rigorous Validation and Testing Protocols

Before assuming autonomous authority, agents must undergo extensive validation demonstrating reliable performance across representative operating conditions, edge cases, and failure scenarios.

Digital Twin Simulation: The most powerful validation environment is a high-fidelity digital twin accurately representing your physical operations. Subject agents to thousands of scenarios, normal production, process upsets, equipment degradations, quality excursions, observing reasoning and evaluating response safety and effectiveness.

Adversarial Testing: Red teams should attempt to manipulate agents into recommending unsafe actions, circumventing access controls, or violating design constraints.

Securing Agent Infrastructure Against Cyber Threats

Agentic AI creates new attack surfaces. Compromised agents could manipulate production operations, corrupt quality data, or trigger safety incidents.

Agent-Specific Threat Vectors: Beyond traditional IT security concerns, prompt injection attacks could manipulate agent reasoning by corrupting data or instructions, potentially causing harmful recommendations while appearing to operate normally.

Defense in Depth: Multiple overlapping control layers, network segmentation isolating agent systems, encrypted communications protecting commands and telemetry, behavioral monitoring detecting abnormal operation patterns, and immutable audit logging for forensic investigation.

Continuous Security Assessment: Unlike static industrial systems receiving security reviews during commissioning, agent systems require ongoing assessment as they learn, adapt, and expand into new operational contexts.

Phase 3: Continuous Oversight and Transparency for Agentic Operations

The final governance phase addresses how organizations maintain effective control as agents transition from validation environments to production operations, scale across multiple facilities and use cases, and evolve through continuous learning and capability enhancement.

Implementing Active Human Supervision

Industrial agents require continuous oversight by personnel empowered to intervene, equipped to understand agent actions, and accountable for performance outcomes.

Supervisory Roles: Different agent types demand different supervision models. Low-risk monitoring agents may require only periodic review by engineering staff validating alert thresholds. High-risk optimization agents executing autonomous adjustments need real-time supervision by qualified operators with process understanding and override authority. Multi-agent coordination systems spanning entire production lines demand oversight at the production leadership level.

Supervision Interfaces: Effective oversight requires purpose-built interfaces communicating what agents observe, how they reason, and what actions they take. Dashboards must display current parameters alongside agent-recommended targets, expected outcome improvements, confidence scores, and reasoning factors. They must support rapid drill-down investigation into the data analyzed, patterns detected, constraints respected, and alternatives considered.

Escalation and Exception Handling: Agents must recognize three situations demanding escalation: novel scenarios outside validated operating experience, conflicting objectives requiring human judgment on trade-offs, and high-consequence decisions exceeding autonomous authority thresholds. Effective escalation requires clear communication of uncertainty, explaining what makes the situation unusual, what information is lacking, and trade-offs of available options.

Ensuring Explainability and Comprehensive Audit Trails

In regulated environments, autonomous agent actions must meet the same documentation standards as human operator decisions.

Decision Transparency: Every autonomous action must include a clear record explaining the reasoning. Explainability must be actionable for domain experts who aren't AI specialists.

Complete Data Lineage: Audit trails must maintain complete lineage showing what information agents accessed, when, and what transformations or analyses they performed.

Tamper-Resistant Record Keeping: Audit trails must be secured against modification or deletion.

Maintaining Disciplined Change Management and Version Control

Agent systems improve through learning and adapt to changing production requirements, but uncontrolled changes pose significant risks. Change management must govern all modifications with the same rigor applied to control system programming.

Formal Change Control: Every modification, updated training data, revised decision logic, new integrations, expanded authority boundaries, must proceed through a structured change control process.

Version Control: Organizations must maintain rigorous version control tracking exactly what model versions, training datasets, decision rules, and integration configurations are deployed in each production environment.

Even with robust governance, organizations should resist immediately deploying agents at the highest autonomy levels. Successful organizations adopt graduated autonomy pathways that systematically expand agent authority as both technical performance and organizational trust mature. Effective governance requires human adaptation, operators learning to interpret recommendations, engineers developing intuition for when agent reasoning proves reliable, and leadership building confidence through demonstrated results.

Section 5: Establishing Multi-Agent Frameworks for Coordinated Industrial Intelligence

---

The governance frameworks established in Section 4 create the control environment necessary for deploying autonomous agents safely in production operations. With clear ownership, bounded authority, and comprehensive oversight in place, organizations can confidently grant agents the autonomy required to drive operational outcomes.

However, a critical realization emerges as you begin deploying these governed agents across your manufacturing operations: the most valuable industrial challenges cannot be solved by individual agents operating in isolation. True operational impact comes from their coordination: preventing quality defects, optimizing maintenance timing to minimize production disruption, and adjusting downstream processes to accommodate upstream constraints.

This coordination challenge reflects a fundamental characteristic of industrial operations: they are inherently multi-system, multi-objective problems. Production throughput depends on equipment availability, which depends on maintenance execution, which depends on parts inventory, which depends on supply chain coordination.

Traditional industrial automation addresses this complexity through hierarchical control: MES systems coordinate production scheduling, SCADA systems manage process control, EAM systems orchestrate maintenance. Each layer operates within its domain, with coordination happening through rigid integration points and human decision-making at domain boundaries.

Agentic operations demand a fundamentally different coordination model. When autonomous agents make real-time decisions based on streaming operational data, rigid hierarchies and pre-programmed integration workflows become bottlenecks. The pace of operational change outstrips the speed at which centralized control systems can process information, evaluate options, and coordinate responses.

The alternative is distributed intelligence: networks of specialized agents that coordinate dynamically through shared operational context, negotiate trade-offs using common objective functions, and adapt their collaboration patterns to changing production conditions. This multi-agent approach mirrors how skilled manufacturing teams operate, with domain experts communicating continuously, sharing situational awareness, and collectively solving problems that no single specialist could address alone.

This section presents systematic frameworks for designing multi-agent systems that deliver coordinated operational intelligence at scale.

Principles of Multi-Agent Operating Models in Industrial Operations

Principle 1: Shared Intent > Local KPIs

In traditional industrial operations, each system optimizes for its own key performance indicator, machine uptime, batch yield, maintenance intervals, without visibility into broader trade-offs. Multi-agent operations replace this fragmented logic with a shared objective function; a formalized statement of operational goals and constraints (for example, maximizing OEE with hard limits on energy use, product quality, and safety compliance).

Agents reading from the same Intent Contract can now negotiate trade-offs using a common value system. They evaluate which decision contributes more to the shared objective function given current constraints. This ensures that local actions reinforce, rather than undermine, enterprise outcomes.

Principle 2: Common Context is Non-Negotiable

Coordination is impossible if agents interpret the world differently. Every participant in a multi-agent ecosystem, whether an equipment agent, scheduling agent, or process optimization agent, must see the same operational reality. This is achieved through a common contextual substrate, grounded in the Unified Namespace for event flow and the ontology for meaning, relationships, and constraints.

Private data models, proprietary tag dictionaries, and inconsistent naming conventions are incompatible with coordinated intelligence. A shared context transforms agent communication from translation to collaboration; every event becomes immediately intelligible and actionable across the entire ecosystem.

Principle 3: Bounded Autonomy by Risk Tier

Treating all agent decisions as equally consequential leads to either over-restriction (human approval for every trivial decision, negating automation value) or under-restriction (agents making high-consequence decisions without adequate safeguards, creating operational and regulatory risk).

Agents operate within clearly defined autonomy tiers, described i n section 4, that determine what they may observe, recommend, execute, or coordinate.

These tiers are enforced dynamically at runtime through policy engines that evaluate decision context, process state, and safety rules before action execution.

Principle 4: Human Co-Agency, Not Fallback

In a multi-agent operation, humans are not exception handlers; they are collaborative agents in the loop. Human-agent collaboration must be designed deliberately with clear roles, information sharing, and coordination protocols. Every agent interaction includes clearly defined:

- **Escalation paths** (who to alert and under what conditions),
- **Handoff protocols** (what data, confidence levels, and context to transfer), and
- **Latency targets** (how quickly a human must respond before fallback procedures initiate).

This transforms operators from reactive overseers into informed collaborators, supervising autonomous systems, validating reasoning, and intervening only where human judgment adds value. The result is a symbiotic workforce where human expertise scales through digital augmentation.

Principle 5: Safety and Compliance as Hard Constraints, Not Guidelines

Traditional approaches treat safety rules and regulatory requirements as procedures that agents should follow; guidelines embedded in prompts, documentation that agents reference, or checks that agents perform. This creates vulnerability: prompt injection could bypass safety rules, reasoning errors could misinterpret regulations, and model updates could inadvertently disable compliance checks.

Safety and compliance rules must be compiled into decision paths and enforced at the infrastructure level, making unsafe or non-compliant actions unrepresentable, not just unlikely or warned against, but impossible to execute regardless of agent reasoning.

Multi-Agent Orchestration Patterns In Industrial Operations

With design principles established, organizations face a critical architectural decision: Should we implement entity-centric digital twin agents or service-based capability agents?

This decision represents fundamentally different philosophies about how intelligent systems should coordinate: through hierarchical control or through emergent collaboration.

The Digital Twin Architecture: Entity-Centric Intelligence

Digital twin architecture deploys one agent per significant operational entity. Each physical machine, each production order, each material batch receives its dedicated agent that maintains complete operational awareness of that entity's state, history, relationships, and objectives.

Each agent maintains its entity's complete digital state in the semantic layer, receives all telemetry relevant to its entity through UNS subscriptions, and coordinates with related agents through communication protocols.

Advantages in Industrial Operations:

Excels when success depends on deep contextual understanding of individual entities. Press_01_Agent accumulates months of operational history specific to this machine, its bearing wear patterns, humidity sensitivity, performance with different formulations, and response to maintenance interventions. This deep context enables detection of subtle anomalies that centralized service agents analyzing aggregate data would miss.

Beyond contextual depth, this delivers emergent adaptability through distributed intelligence. When TabletPress_01_Agent detects bearing failure, it communicates with its agent network. Blender_04_Agent adjusts batch sizes for reduced downstream capacity, CoatingPan_02_Agent recalibrates for intermittent feed, and ProductionOrder_Agents reroute to alternate equipment. No central controller orchestrated this

response, each agent, understanding its role and relationships, contributes specialized intelligence to collectively manage the disruption.

Entity-centric architecture also enables simplified coordination because agents know exactly which other agents they depend on, keeping reasoning tractable as facility complexity grows.

Challenges Requiring Mitigation:

The primary drawback is agent proliferation overhead. A facility with two hundred pieces of equipment, three hundred concurrent production orders, and five hundred active material batches requires one thousand agents operating simultaneously. Organizations must invest in edge computing infrastructure and distributed platforms that support this scale.

The Service-Based Architecture: Capability-Centric Intelligence

Service-based architecture deploys a smaller number of sophisticated agents, each providing intelligence services consumed across many operational instances. One EquipmentHealthService_Agent monitors all facility equipment. One QualityPredictionService_Agent forecasts outcomes for all product formulations

Agents consume operational data from the UNS and maintain state in the semantic layer, but they manage state for multiple entities simultaneously rather than dedicating cognitive capability to single instances.

Advantages in Industrial Operations:

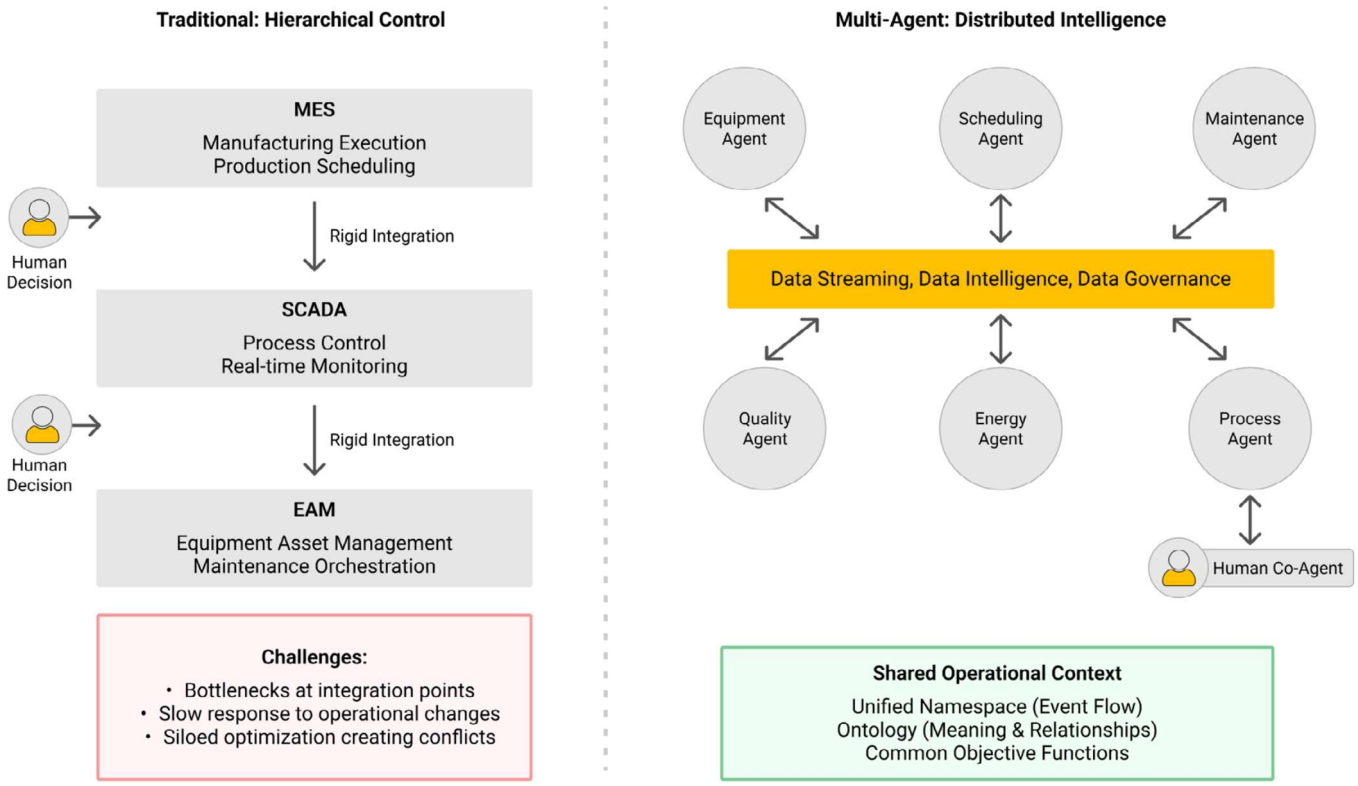
Operational efficiency through resource consolidation. Sophisticated agent models, deep learning networks, multi-objective optimization algorithms, complex quality correlation models, are expensive to develop and computationally intensive. Deploying them once and amortizing costs across hundreds of instances makes economic sense, particularly for smaller facilities.

Rapid learning through data aggregation. EquipmentHealthService_Agent analyzing patterns across fifty presses identifies failure mode correlations invisible to agents observing single machines.

Simplified version management, when improving quality prediction algorithms, update one QualityPredictionService_Agent rather than propagating changes to hundreds of agents. New capabilities deploy enterprise-wide instantly.

Challenges Requiring Mitigation:

The fundamental limitation is reduced contextual adaptation. Service-based architecture inherently operates through hierarchical control: a single agent manages multiple operational instances from a centralized decision point. While the agent may execute faster than human managers, it remains constrained by the same top-down structure that limits traditional automation.



Hybrid Architectures: Combining Approaches Strategically

The comparison above might suggest organizations must choose one approach exclusively. In practice, most industrial organizations will ultimately deploy hybrid architectures that combine entity-centric agents for operational contexts where adaptability and fault tolerance matter most, with service-based agents for capabilities that genuinely benefit from centralized data aggregation and shared infrastructure.

Multi-Agent Communication Framework in Industrial Operations

Successful multi-agent systems, whether centralized or distributed, require an agent-to-agent communication framework through which they expose skills, share tasks, communicate intent, and return results. However, the choice of transport protocol for this framework is the difference between a resilient, scalable ecosystem and a brittle, unmanageable web of dependencies.

Traditional point-to-point communication protocols impose fundamental limitations that conflict with the distributed, autonomous nature of industrial agent systems. When agents communicate through direct connections, the number of required integrations grows quadratically with system scale. More critically, this tight coupling prevents the dynamic, self-organizing behavior that distinguishes truly agentic systems from conventional automation.

Agent-to-Agent Communication Through Event-Driven Protocols

In agentic industrial systems, agent-to-agent communication should follow the same event-driven principles that govern real-time operational data flow. Instead of relying on direct, point-to-point messaging between agents, interactions occur through shared coordination topics within the UNS.

For example, when the Press_01_Agent detects abnormal vibration patterns indicating a likely bearing failure, it publishes an event to a relevant topic in the UNS. Other agents, subscribed to that namespace, react accordingly as part of a coordinated response.

This publish-subscribe pattern for agent coordination delivers three critical benefits:

Decoupling: Agents don't need to know which other agents exist or where they're deployed. They publish coordination events to semantic topics, and any agent with relevant responsibility can subscribe and respond.

Auditability: All agent-to-agent communication flows through the UNS, creating complete audit trails of coordination decisions. When investigating why a production schedule changed, engineers can replay the agent coordination events that led to that decision.

Extensibility: Adding new agent capabilities requires no changes to existing agents. If you deploy a new EnergyOptimization_Agent that needs to participate in maintenance scheduling decisions, it simply subscribes to the relevant coordination topics and begins contributing to those workflows.

You can read more about this approach on this whitepaper,
An MQTT Architecture for Scalable Agentic AI Collaboration

About HiveMQ

HiveMQ is the Industrial AI Platform helping enterprises move from connected devices to intelligent operations. Built on the MQTT standard and a distributed edge-to-cloud architecture, HiveMQ connects and governs industrial data in real time, enabling organizations to act with intelligence. With proven reliability, scalability, and interoperability, HiveMQ provides the foundation industrial companies need to operationalize AI, powering the next generation of intelligent industry. Global leaders including Audi, BMW, Eli Lilly, Liberty Global, Mercedes-Benz, and Siemens trust HiveMQ to run their most mission-critical operations.

Learn more at hivemq.com