The enterprise guide to Agentic AI

Frameworks for strategic implementation and value creation

Abstract

This white paper provides a comprehensive framework for enterprise adoption of Agentic AI, addressing the gap between consumer-grade applications and effective enterprise implementation. It offers a strategic approach to decomposing complex business roles, orchestrating multi-agent systems, determining appropriate autonomy levels, and implementing solutions across industry verticals. Through detailed analysis and case studies, it demonstrates how organizations can move beyond rebranded automation to achieve genuine transformation with measurable business outcomes.

Table of contents

Introduction

The promise and reality of agentic AI in enterprise

Agentic AI promises to initiate a new S-curve of innovation, compelling enterprises to incorporate agentic solutions into their transformation agendas. While consumer-grade agentic use cases have demonstrated transformative success, enterprise implementations have shown fewer breakthrough results. Most enterprise applications have merely rebranded existing automation or AI solutions. A significant gap exists in understanding agentic AI and, more critically, in designing and executing effective agentic AI solutions.

Objectives of this white paper

This white paper provides a strategic framework for implementing agentic AI with a focus on practical execution. It explores how organizations can decompose complex job roles into agent-suitable tasks, orchestrate multiple agents within a cohesive system, anticipate and address common failure points and gradually evolve from human-assisted to fully autonomous operations. Through detailed case studies spanning banking and financial services (BFS), insurance, and finance and accounting (F&A), this white paper will attempt to demonstrate how agentic AI transforms operations, enhances decision-making and delivers measurable business value even as the underlying technologies continue to evolve.

Defining agentic AI

Agentic AI refers to AI systems that act as autonomous agents capable of understanding objectives, making decisions, taking actions and adapting their behavior to achieve specified goals. Unlike traditional automation or conventional AI systems, agentic AI possesses:

Goal-oriented reasoning: The ability to understand objectives and reason about the best approaches to achieve them

Autonomous decision-making: The capacity to make independent decisions based on available information and learned patterns

Adaptability: The capability to adjust strategies when confronted with changing circumstances or new information

Collaborative intelligence: The ability to work effectively with humans and other AI agents toward common goals

Self-improvement: The capacity to learn from experiences and outcomes to enhance future performance

Beyond automation: The agentic AI revolution

Evolution from RPA to agentic AI

Traditional robotic process automation (RPA) excels at executing predefined, rules-based tasks with high efficiency but lacks adaptability. In general, the success of RPA has been limited because it has lacked ability to reason and to quickly adapt to an ever-changing business and process landscape. AI-enhanced automation brings intelligence through machine learning but still operates within confined parameters. So, while AI solutions have excelled in predicting and prescribing outcomes and actions, it still had minimal to no ability to adapt, be autonomous, to reason and to interact with its ecosystem. On the other hand, agentic AI represents a transformative leap—autonomous entities that understand objectives, adapt to changing conditions and collaborate effectively with humans and other agents. While an AI agent doesn't need to necessarily use large language models (LLMs) or large reasoning models (LRMs), leveraging LLMs and LRMs do give the agents the ability to reason thereby driving more autonomy.

Consider transaction monitoring in banking: RPA might flag transactions that match predefined patterns, while AI automation might detect anomalies based on historical data. Agentic AI, however, would proactively investigate suspicious activities, gather relevant context, collaborate with other agents to establish a comprehensive risk profile and adaptively refine its approach based on outcomes.

Comparative framework

The differences between RPA automation, AI automation and agentic automation can be well understood in the following dimensions:

Dimension	Traditional RPA	AI automation	Agentic AI
Decision intelligence	Rules-based decisions	Pattern recognition and predictions	Goal-oriented reasoning and adaptive decision-making
Autonomy	Executes predefined processes	Learns from data but limited adaptability	Autonomous pursuit of goals, adapting to changing circumstances
Versatility	Task-specific	Domain-specific	Cross-domain capable
Human interaction	Requires human triggers and exception handling	Requires human oversight and intervention	Collaborates with humans as intelligent partners
Knowledge utilization	Limited to programmed logic	Utilizes training data patterns	Integrates domain knowledge, context, and experience

The business case for agentic AI

The shift toward agentic AI is strategic and not merely technological. Enterprises should seriously consider agentic AI to deliver the below benefits:

Enhanced adaptability: Agents can navigate complex, dynamic environments without constant reprogramming

Improved decision quality: By considering multifaceted contexts and collaborating with other agents

Reduced human cognitive load: Handling routine and complex tasks while escalating only when necessary

Accelerated innovation: Enabling rapid experimentation and implementation of new processes

Building reliable and scalable agentic AI solutions

Core implementation principles

The lack of widespread success of agentic AI solutions has less to do with the technology limitations but more to do with the enterprise approach to implementing its agentic AI program. It is important to focus on the following—rather obvious but often less thought through—aspects while building enterprise-grade agentic AI solutions:

1. Agent reliability: AI agents need to function consistently and deliver accurate results

2. Integrations: More often than not, the agents would be introduced in a complex ecosystem which includes multiple external tools and APIs. A key aspect of the success of an agent is therefore the investment in appropriate protocols that allow seamless integration and interaction with other tools, agents and APIs.

3. ROI-driven automation: Just because agentic AI is powerful and in vogue, we don't need to force fit agentic AI as the solution to every single automation opportunity. Simple ruleset-based automations can work seamlessly and provide better ROI for simple automations.

4. Avoid overengineering and avoid feature creep: Keep solutions simple and avoid adding unnecessary complexity. It is also important to resist the urge to add too many features, which can dilute focus and effectiveness.

5. Security measures: AI jailbreaks are as common and prevalent as are the new solutions. We might very easily get into a recursive problem where AI agents are trying to break other AI agents. This is a real threat and can not only shut down the AI program but can cause serious financial and reputational damage unless there is conscious investment in AI security protocols and tools.

6. Avoiding common pitfalls: Agentic AI programs face challenges similar to traditional automation when user-centric design is overlooked. Common issues include the lack of user-feedback loops and error-handling mechanisms, which are crucial for improving functionality and delivering a better user experience.

Understanding model context protocol and its potential role in the success of agentic AI solutions

Agentic AI, which extensively rely on LLMs interacting with external services, benefits from having a standardized protocol that governs these interactions. This led to the introduction of the “Model Context Protocol (MCP)”. MCP was first introduced by Anthropic as an open-source initiative in November 2024.

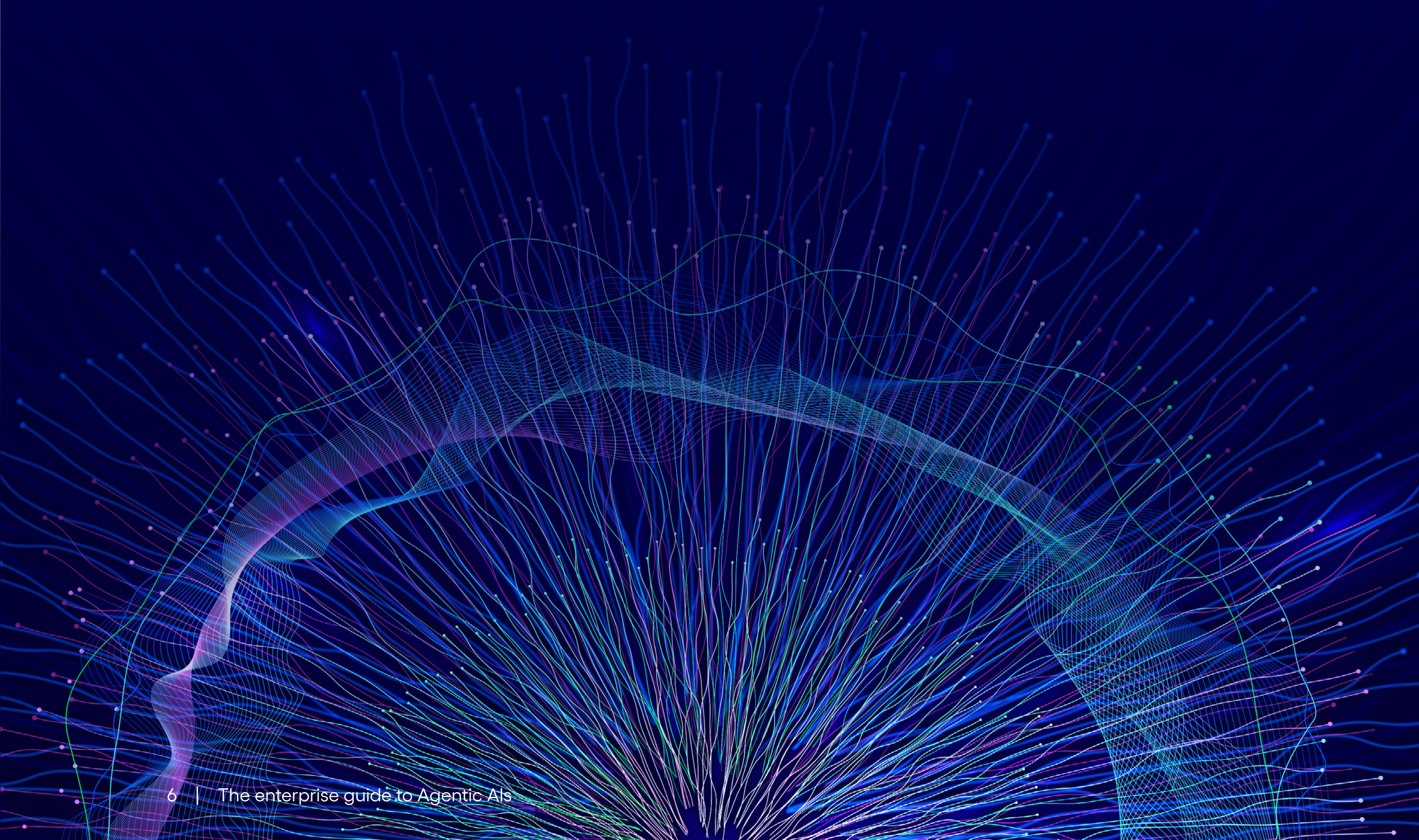
While basic LLMs could only predict text, enabling them to perform tasks required LLMs to be connected to external tools and APIs. In the context of agentic AI, enabling this connection is key to making the agent useful and scalable. A model context protocol introduces a standardized protocol that eliminates the complexity of connecting to multiple tools. It acts as a unified layer that translates between LLMs and external tools, simplifying their integration.

MCP’s role in enterprise agentic systems

MCP ecosystem typically includes:

- **MCP client:** User-facing apps
- **Protocol:** Standardized communication between clients and servers
- **MCP server:** Translates tool capabilities for the LLM
- **Service:** The actual external tool or database being accessed

It is important to note that the MCP standards haven’t reached a stage of maturity and is yet to see widespread adoption. There have also been concerns regarding performance and latency during interactions. However, just like the internet wouldn’t have scaled without a protocol like “HTTP,” agentic AI won’t scale without a standard like MCP. So, while a new standard might replace MCP, there is definitely a case for enterprises to start using the MCP standard now.



Strategic implementation framework

Decomposing roles into agent-suitable tasks

Traditional job roles typically encompass a complex (or in most cases "complicated") matrix of responsibilities, skills and knowledge. To effectively implement agent-based automation, it is essential to break down complex job roles into discrete, agent-suitable tasks. This requires a combination of methodological rigor and hierarchical decomposition, ensuring that agents can handle tasks effectively while preserving the synergy of the original job roles. Current approaches leverage several methodologies as below:

1. Agent-oriented methodologies (AOM):
   AOM extends object-oriented and knowledge engineering techniques by incorporating agent-specific attributes such as beliefs, desires, intentions and commitments. These methodologies enable the decomposition of tasks through:

   Object-oriented extensions: Use case analysis and CRC (class-responsibility-collaboration) cards identify agents and their roles, extending traditional models to include agent-specific mental states

   Knowledge engineering extensions: Model the cognitive and social dimensions of agents to capture nuances not covered by traditional approaches

2. Multi-agent systems (MAS):
   Multi-agent systems enable multiple agents to collaborate in completing complex workflows. This approach becomes particularly valuable when tasks require specialized expertise, coordination and dynamic adaptation.

   Task granularity: Tasks are divided among specialized agents, ensuring that complex processes are handled effectively

   Coordination frameworks: Orchestration platforms facilitate seamless task delegation and collaboration between agents

3. Role-based decomposition enhanced by decomposition spectrum:
   Role-based decomposition involves breaking down job roles into specific functions, skills and workflows. The decomposition spectrum adds a hierarchical framework to this methodology, refining the process as follows:

   Macro-level decomposition: This level breaks down entire job roles into major functional areas. It corresponds to functional analysis in role-based decomposition by mapping high-level responsibilities to agent capabilities. For example, in insurance underwriting, this step might involve segmenting the role into data collection, risk analysis and policy recommendations.

   Meso-level decomposition: At this level, functional areas are further divided into specific processes that define how tasks are executed. This aligns with workflow analysis, where the sequential and parallel workflows of a job role are mapped to agent tasks. For instance, risk analysis in underwriting could be divided into data validation, risk scoring and compliance checks.

   Microlevel decomposition: The most granular level identifies discrete tasks within processes and maps them to agent capabilities. This corresponds to skill mapping, where the necessary agent skills are aligned with task requirements. For example, data validation might be fully automated through AI agents capable of document parsing and anomaly detection.

Aligning decomposition with agent capabilities:

The optimal decomposition level depends on the complexity of the task, the maturity of available agent capabilities and the extent of required human oversight. For highly structured tasks, microlevel decomposition allows for full automation, while more nuanced processes may require meso-level decomposition with agent-human collaboration.

An integrated approach to task decomposition:

Among the above methodologies, the role-based decomposition combined with the multi-agent systems for complex workflows would be recommended. The hierarchical decomposition spectrum is easy to understand and visualize for domain and process SMEs and the technologists can enable the MAS framework to create a collaborative agent framework.

Agent orchestration and coordination strategies

Agent orchestration, i.e., the coordination of multiple agents toward common goals, is what transforms discrete intelligent entities into a cohesive, business-value-generating system. Effective orchestration requires:

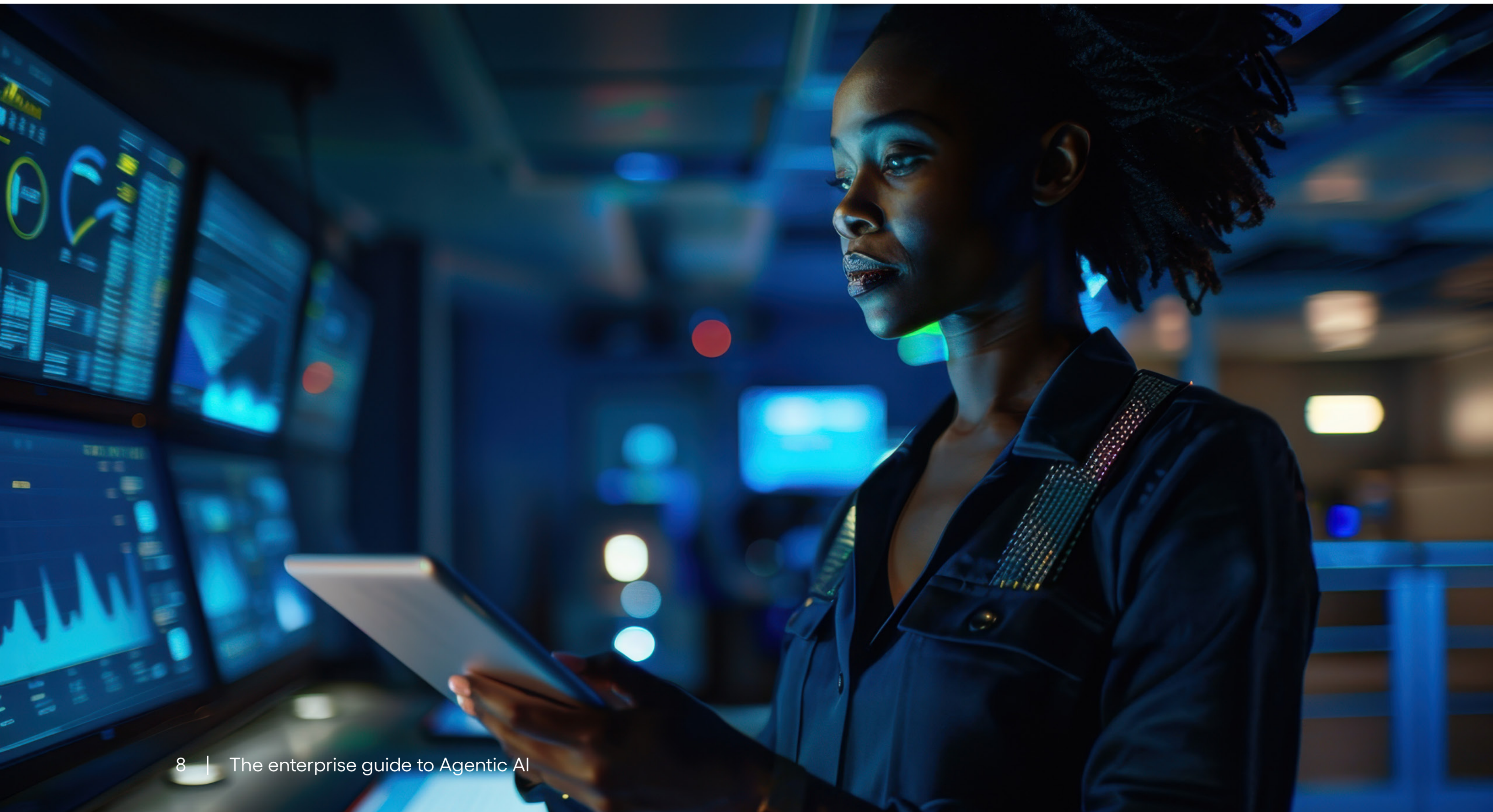
Clear role definition: each agent's responsibilities and boundaries

Communication protocols: how agents share information and coordinate actions

Prioritization mechanisms: how tasks are prioritized across agents

Exception handling: how system failures and edge cases are managed

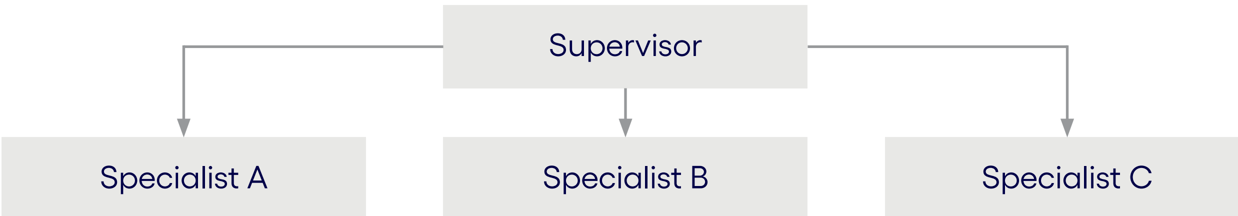
Performance monitoring: how agent effectiveness is measured and improved



Orchestration patterns:

Several orchestration patterns have emerged, each with distinct advantages:

1. **Supervisor-based orchestration:** In this pattern, a central supervisor agent coordinates the activities of multiple specialized agents. As an example, Amazon Bedrock's multi-agent collaboration framework uses a supervisor agent to manage specialized agents, improving task success rates and efficiency.



Advantages:

- Centralized control and monitoring
- Simplified task allocation and prioritization
- Clear accountability

Challenges:

- Potential bottlenecks at the supervisor level
- Single point of failure

This pattern works well for complex workflows requiring tight coordination, such as financial closing processes where multiple specialized agents must operate in sequence.

2. **Sequential pipelines:** This strategy involves organizing agents in a linear sequence where each agent performs a specific subtask and passes the result to the next agent. As an example, CrewAI's blog writing pipeline where planner, writer and editor agents work in sequence to produce a final article.

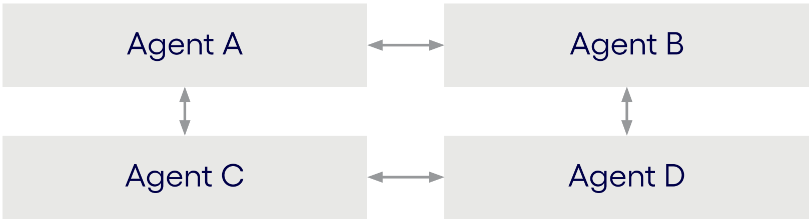
Advantages:

- Clear workflow
- Easy to understand and implement
- Suitable for tasks with linear progression

Challenges:

- Limited application
- Failure point on one agent can cause unexpected results in the subsequent agent(s)

3. **Peer-to-peer orchestration:** In this pattern, agents coordinate directly with each other. As an example, Fetch.ai's multi-agent economic platform enables autonomous economic agents to directly negotiate with each other in decentralized marketplaces without central coordination. These agents represent various stakeholders (consumers, providers, data owners) and conduct peer-to-peer transactions and information exchanges, improving resource allocation efficiency and reducing central bottlenecks.



Advantages:

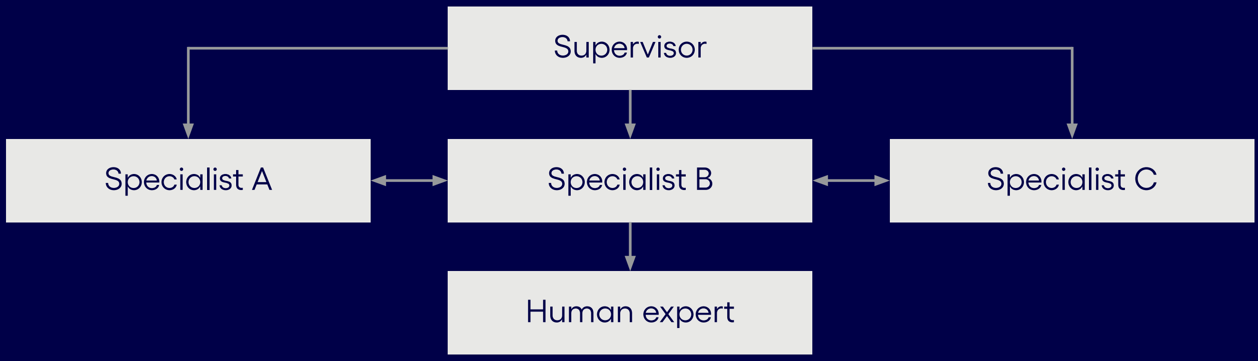
- No central bottleneck
- Greater resilience to individual agent failures
- More flexible adaptation to changing conditions

Challenges:

- More complex coordination logic
- Potential for conflicting actions

This pattern is effective for distributed systems where agents need to respond quickly to local conditions, such as fraud detection systems where multiple monitoring agents may need to collaborate rapidly.

4. **Hybrid orchestration:** Most mature agentic systems employ hybrid approaches, combining elements of both patterns. As an example, Microsoft's Project Bonsai combines both centralized and peer-to-peer approaches in industrial control systems. A high-level orchestrator agent determines overall manufacturing strategies while allowing specialized process control agents to communicate directly with each other during critical real-time operations. This hybrid approach maintains strategic oversight while enabling rapid local responses to changing conditions, resulting in a 25% increase in production efficiency in pilot implementations.



This approach allows for both centralized coordination and direct agent-to-agent communication, with strategic human involvement where needed.

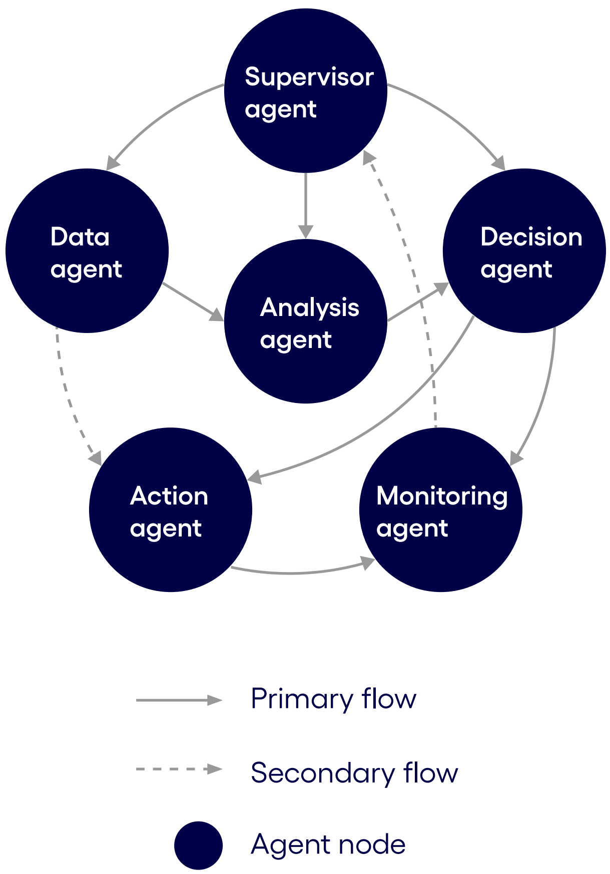
Advantages:

- Combines centralized oversight with local autonomy
- More resilient than purely centralized approaches
- More organized than purely peer-to-peer systems
- Adaptable to various task complexities
- Scalable for large agent ecosystems

Challenges:

- Higher implementation complexity
- Requires careful boundary definition between centralized and peer-to-peer components
- More complex debugging and monitoring
- Potential for communication overhead
- Risk of coordination conflicts between local and global decision-making

5. **Graph-based orchestration:** Graph-based orchestration represents agents and their interactions as a network of nodes (agents) and edges (communication pathways), enabling dynamic and non-linear workflows. For example, AWS uses a graph-based model in its agent interaction framework to support complex coordination patterns and enhance scalability across distributed systems.



Advantages:

- Supports complex workflows with non-linear interactions
- Allows for dynamic adaptation
- Enhances scalability

Challenges:

- Increased complexity in creating, maintaining and debugging the interaction graph as the system scales
- Difficulty in dynamically modifying the graph structure during runtime to adapt to changing environments
- Computational overhead and potential performance bottlenecks when traversing complex graphs with many nodes and edges

Communication protocols:

Effective interagent communication is crucial for coordination. Key protocols include:

- Agent communication protocols, originally formalized through Agent Communication Languages (ACLs) such as KQML and FIPA-ACL, provided structured semantics and intent-driven messaging between agents. While these foundational models introduced key concepts in agent interaction, modern multi-agent systems, especially those built on LLMs, rely on more scalable and lightweight methods such as RESTful APIs, event buses, WebSockets, and message queues (e.g. Kafka, RabbitMQ) to enable asynchronous, tool-integrated and dynamic agent communication
- Publish/subscribe paradigm that decouples publishers (agents that generate messages) from subscribers (agents that receive messages), supporting asynchronous communication

Visualization and monitoring:

Effective orchestration requires visibility into agent activities and system performance. Modern agent orchestration platforms offer:

Process visualization: Real-time views of agent workflows and activities

Performance dashboards: Metrics on agent effectiveness, efficiency and outcomes

Exception queues: Interfaces for addressing cases requiring human intervention

Audit trails: Comprehensive records of agent actions and decisions

These capabilities enable organizations to monitor, troubleshoot and continuously improve their agentic systems.

Determining appropriate autonomy levels for AI agents

Not all tasks are suitable for fully autonomous agentic automation. Determining the appropriate level of autonomy and human oversight is crucial for balancing efficiency with reliability, handling exceptions and maintaining compliance. Here are key guidelines for determining the level of automation using agentic AI:

Data quality and integrity: Human oversight is essential for ensuring the accuracy and completeness of data inputs. Regular audits and data quality checks should be performed by human experts to maintain data integrity throughout automated processes.

Exception handling: Humans should be involved in managing cases that fall outside the parameters of automated systems. Organizations must establish clear escalation paths for complex or unusual cases that require human judgment and intervention.

Regulatory compliance: Human experts must ensure that automated processes comply with industry regulations and standards. Regular compliance audits and updates to \automated systems should be overseen by human specialists to prevent violations and maintain adherence to evolving requirements.

System monitoring: Organizations should implement continuous monitoring systems for automated processes. Human experts should review system performance metrics and address any anomalies or issues promptly to prevent cascading failures.

Decision validation: For critical decisions, organizations should implement a human-in-the-loop approach where AI recommendations are validated by human experts before execution to ensure appropriate outcomes.

Environment predictability: The task environment must be reasonably predictable for AI agents to function effectively without constant human intervention. Tasks with high variability or uncertainty may require greater human oversight.

Risk evaluation and consequence severity: Organizations should implement a human-in-the-loop approach for tasks where errors could have moderate impacts. Strong human oversight should be maintained for tasks where errors could lead to significant financial, legal or reputational damage.

Audit trails: Organizations should ensure that automated systems can provide comprehensive audit trails and clear explanations for decisions made to support regulatory compliance and process transparency.

Data availability and quality: Organizations need to assess the availability and quality of data required for agent training and automation.
Higher levels of autonomy are suitable when high-quality, comprehensive data is available to train and operate AI systems effectively.

Human value-add: Organizations should consider whether human judgment adds significant value to the task. Tasks requiring creativity, empathy, ethical judgment or complex contextual understanding may require more substantial human involvement.

This assessment framework helps prioritize tasks for appropriate levels of human involvement. For example, routine data reconciliation scores high on definability and predictability with low consequence severity, making it ideal for full autonomy of agentic automation. In contrast, complex fraud investigations might require significant human collaboration given their unpredictability and high consequence severity.

Implementation roadmap across the autonomy spectrum

Agentic AI implementation exists on a spectrum from human-led to fully autonomous:

Copilot (human-led):
Agents provide suggestions and support, but humans make decisions and take actions

Collaboration (shared control):
Agents handle routine tasks autonomously but escalate complex cases to humans

Supervision (agent-led):
Agents operate autonomously with human oversight and intervention capabilities

Autonomy (agent-driven):
Agents operate independently with minimal human involvement

For complex use cases, organizations should progress deliberately along this spectrum, building trust and capabilities at each stage:

Phase 1: Foundation building

- Identify high-value use cases
- Conduct task suitability assessments
- Develop initial agent prototypes
- Establish governance frameworks
- Implement change management programs

Phase 2: Copilot deployment

- Deploy initial agents in copilot mode
- Establish feedback mechanisms
- Collect performance data
- Refine agent capabilities
- Build user confidence

Phase 3: Collaborative autonomy

- Transition suitable tasks to collaborative mode
- Implement interagent communication
- Develop orchestration capabilities
- Refine exception handling
- Enhance monitoring and analytics

Phase 4: Supervised autonomy

- Expand agent autonomy with human supervision
- Implement advanced orchestration patterns
- Develop predictive capabilities
- Enhance self-healing mechanisms
- Optimize system performance

Phase 5: Intelligent enterprise

- Deploy fully autonomous agents where appropriate
- Implement advanced learning and adaptation
- Develop cross-domain capabilities
- Optimize human-agent collaboration
- Continuously evolve the system

Real-world case studies: Industry implementation case studies

Having established the conceptual foundations of agentic AI—from its distinctive capabilities beyond traditional automation to frameworks for decomposing roles, determining appropriate autonomy levels and orchestrating multiple agents—we now turn to practical implementation. The following industry-specific case studies demonstrate how these theoretical principles translate into tangible business outcomes.

Each use case illustrates the complete journey of agentic AI implementation: from problem identification and task decomposition to agent orchestration strategies and measurable results. These examples provide not just conceptual validation but actionable blueprints that organizations can adapt to their specific contexts. By examining these implementations in detail, we bridge the gap between theoretical potential and practical execution, showing precisely how agentic AI delivers transformative value in complex enterprise environments.

Real-world use case 1: Finance and accounting—Implementing agentic AI for group and local closing for an insurance enterprise

Traditional job roles typically encompass a complex (or in most cases "complicated") matrix of responsibilities, skills and knowledge. To effectively implement agent-based automation, it is essential to break down complex job roles into discrete, agent-suitable tasks. This requires a combination of methodological rigor and hierarchical decomposition, ensuring that agents can handle tasks effectively while preserving the synergy of the original job roles. Current approaches leverage several methodologies as below:

The closing challenge:

Financial closing processes in multinational insurance companies typically consume significant resources while facing strict regulatory deadlines. These processes involve complex workflows across multiple systems, reconciliations between diverse data sources and consolidation of reports from numerous local entities. The repetitive yet nuanced nature of these tasks makes them ideal candidates for agentic AI implementation.

Strategic role decomposition for financial closing:

Applying our hierarchical decomposition framework to the closing process reveals natural divisions that align with agent capabilities. The financial closing function can be segmented into six core functional areas (macro-level decomposition) and each functional area breaks down into defined processes with clear inputs, outputs and workflows (meso-level decomposition) as below:

S.no	Macro-level decomposition	Meso-level decomposition
1	Data collection and validation	• System data extraction • Data standardization and normalization • Completeness validation • Cross-system data reconciliation
2	Account reconciliation	• Balance matching across systems • Discrepancy identification and classification • Resolution tracking • Documentation management
3	Adjustment processing	• Adjustment identification and categorization • Journal entry creation • Approval workflow management • Posting execution
4	Financial consolidation	• Currency translation • Intercompany elimination • Minority interest calculation • Group-level adjustments
5	Financial reporting	• Statement generation • Regulatory compliance verification • Disclosure preparation • Report distribution
6	Performance analysis	• Variance analysis • Trend identification • Anomaly detection and explanation • Commentary generation

At the most granular level, we identify specific tasks and their suitability for agent automation (microlevel decomposition). For example, within data collection and validation we have the below microlevel decomposition:

- Connecting to source systems and extracting data (highly structured, ideal for full automation)
- Standardizing data formats across systems (rule-based, suitable for automation)
- Applying validation rules to identify data gaps (well-defined, suitable for automation)
- Resolving complex data quality issues (variable complexity, requires human collaboration)

Autonomy assessment framework in action:

Applying the autonomy assessment framework to closing tasks reveals varied autonomy levels:

Task category	Predictability	Consequence severity	Data quality	Human value-add	Recommended autonomy
Data extraction	High	Low	High	Low	Full autonomy
Standard reconciliations	High	Medium	High	Low	Supervised autonomy
Complex reconciliations	Medium	High	Medium	High	Collaborative
Recurring journal entries	High	Medium	High	Low	Supervised autonomy
Non-standard adjustments	Low	High	Medium	High	Human-led
Intercompany eliminations	High	Medium	High	Low	Supervised autonomy
Financial statement preparation	High	High	High	Medium	Supervised autonomy
Management commentary	Low	High	Medium	High	Human-led

Orchestrating the closing ecosystem:

For financial closing, a hybrid orchestration approach delivers the optimal balance of control and efficiency:

Financial close orchestrator agent: Serves as the primary coordinator, maintaining the close calendar, tracking dependencies and ensuring timely completion of all tasks

Specialized agents:

- **Data integration agent:** Extracts and standardizes data from source systems
- **Reconciliation agent:** Performs account reconciliations and tracks unresolved items
- **Journal entry agent:** Creates, routes and posts standard adjustments
- **Consolidation agent:** Performs currency translation and intercompany eliminations
- **Reporting agent:** Generates financial statements and regulatory reports
- **Analysis agent:** Identifies variances and generates

Human integration points:

- Review and approval of non-standard adjustments
- Resolution of complex reconciliation discrepancies
- Final review of financial statements
- Development of strategic commentary

Communication flow:

The system implements:

Event-driven communication: Agents respond to triggers like data availability or task completion

Status tracking: Comprehensive monitoring of task progress and bottlenecks

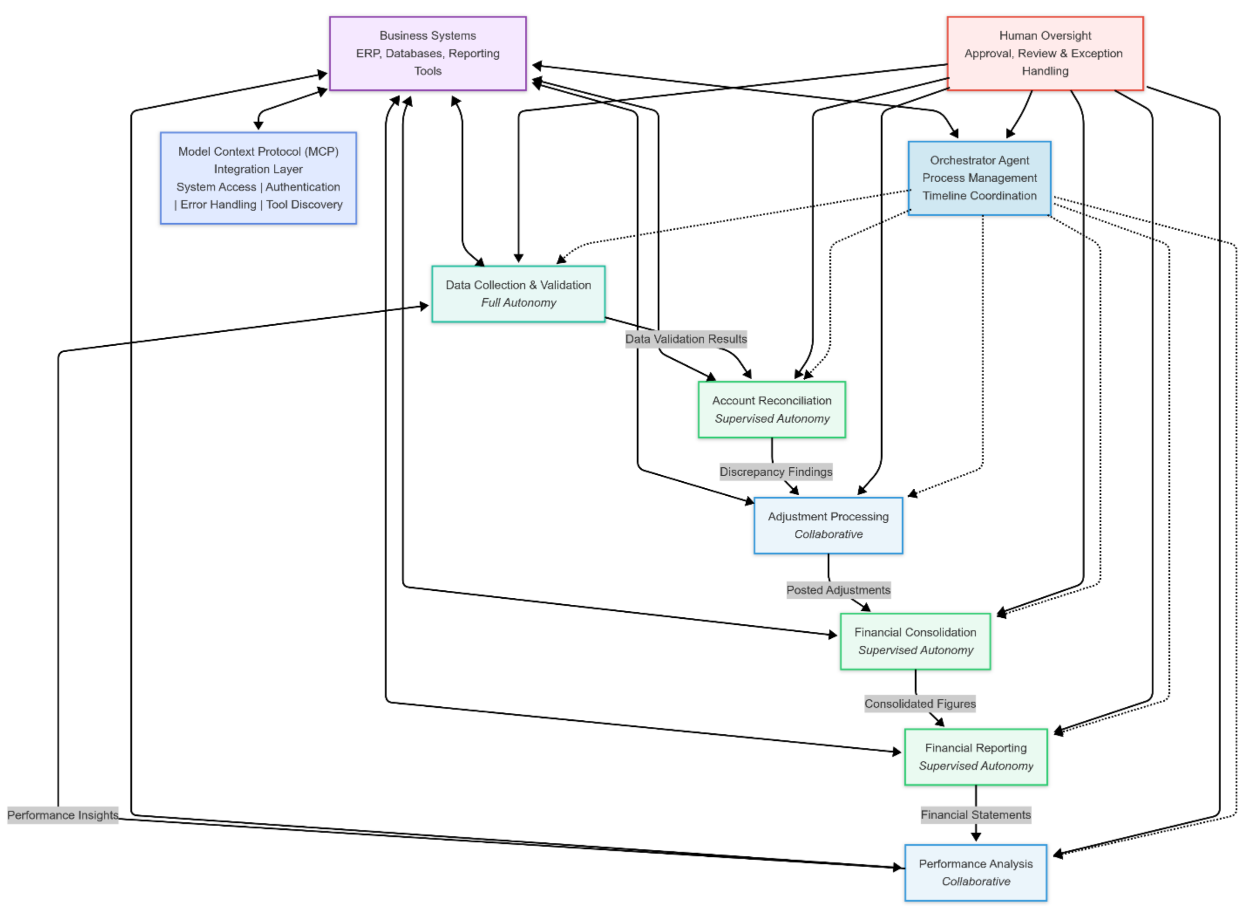
Exception routing: Automated escalation of issues requiring human judgment

Feedback loops: Performance metrics capture for continuous system improvement

MCP integration for financial systems—The model context protocol provides significant advantages for financial closing by standardizing agent interactions with:

- ERP systems and financial databases
- Legacy accounting applications
- Consolidation tools
- Regulatory reporting platforms
- Document management systems

MCP implementation reduces integration complexity and maintenance overhead while providing consistent error handling and authentication across the financial ecosystem.



Visualization and monitoring—A comprehensive visualization dashboard should be implemented featuring:

Process monitoring:

- Real-time closing calendar with completed, in-progress and pending tasks
- Status indicators for each account and entity
- Critical path analysis showing bottlenecks

Performance metrics:

- Time-to-close tracking compared to historical benchmarks
- Reconciliation completion rates
- Exception volumes and resolution times
- System resource utilization

Exception management:

- Centralized queue of items requiring human review
- Categorization of exceptions by type, entity and priority
- Historical context of similar past exceptions

Audit trail:

- Comprehensive logging of all agent actions
- Documentation of human decisions and overrides
- Time-stamped record of all system interactions

Implementation pathway—A phased implementation approach delivers incremental value while managing risk:

Foundation phase: Deploy data integration and reconciliation agents with high human oversight

Expansion phase: Add journal entry and consolidation agents with supervised autonomy

Optimization phase: Implement reporting and analysis agents with collaborative autonomy

Maturity phase: Reduce human oversight for well-performing processes and enhance agent sophistication

This strategic implementation of agentic AI in financial closing can reduce cycle time by 30%–50%, decrease resource requirements by 40%–60% and significantly improve accuracy while enhancing analytical capabilities.

Real-world use case 2: Enhanced banking transaction monitoring system using agentic AI

The transaction monitoring challenge:

Most global banks struggle with their transaction monitoring system, which generates thousands of false positives daily, overwhelming investigators and allowing genuine fraud to slip through. Transaction monitoring represents an ideal application for agentic AI due to its complex blend of structured data processing, pattern recognition and nuanced investigation requirements. The evolving nature of financial crimes necessitates systems that can adapt autonomously while maintaining regulatory compliance. A comprehensive agentic AI approach involves strategic role decomposition, appropriate autonomy allocation and sophisticated orchestration.

Strategic role decomposition for transaction monitoring:

Applying our hierarchical decomposition framework to the transaction monitoring process reveals natural divisions that align with agent capabilities. The transaction monitoring function can be segmented into six core functional areas (macro-level decomposition) and each functional area breaks down into defined processes with clear inputs, outputs and workflows (meso-level decomposition) as below:

S.no	Macro-level decomposition	Meso-level decomposition
1	Data collection and normalization	• Source connectivity and management • Data extraction and normalization • Cross-system reconciliation • Quality validation
2	Pattern analysis and risk assessment	• Pattern recognition and anomaly detection • Historical comparison and trend analysis • Behavioral analysis • Risk scoring and threshold management
3	Alert management	• Alert triage and categorization • Priority assignment based on risk factors • Resource allocation and workload balancing • Alert aging and escalation management • False positive identification and reduction
4	Investigation and contextual analysis	• Context gathering • Document analysis • Customer profile enrichment • Transaction relationship mapping • Documentary evidence gathering • Entity network analysis
5	Regulatory reporting	• Case documentation compilation • Evidence assembly and preservation • Narrative generation for suspicious activities • Regulatory filing preparation and submission • Audit trail maintenance and documentation
6	Process orchestration and oversight	• Workflow sequencing and management • Exception handling and escalation • Performance monitoring and optimization • Cross-agent communication facilitation

At the most granular level (microlevel decomposition), tasks are classified based on agent suitability:

Highly structured tasks (data extraction, normalization): Full agent autonomy on tasks like data extraction and normalization, initial pattern matching against known typologies, basic alert prioritization, routine documentation gathering, etc.

Semi-structured tasks (initial risk scoring): Agent-led with oversight with tasks like risk scoring for complex scenarios, alert disposition for moderate-risk cases, contextual information synthesis, draft narrative generation, etc.

Complex judgment tasks (final SAR determination): Human-led with agent assistance with tasks like final suspicious activity determinations, complex investigation strategy, regulatory filing approval, model tuning and threshold adjustment, etc.

Autonomy assessment framework in action—The effectiveness of an agentic AI system depends on appropriately calibrated autonomy levels. Human oversight concentrates where:

- Regulatory consequences are significant
- Judgment requiring domain expertise is needed
- Novel patterns require interpretation
- System adaptation decisions are necessary

Applying the autonomy assessment framework to closing tasks reveals varied autonomy levels:

Agent type	Autonomy level	Human oversight	Rationale
Data aggregation	High	System-level monitoring	Highly structured task with defined parameters
Risk scoring	Medium-high	Regular model review	Statistical nature with established patterns
Alert management	Medium	Threshold adjustment, complex case review	Balance between efficiency and accuracy
Investigation	Medium-low	Guidance on complex cases, verification of findings	Contextual judgment requirements
Reporting	Low	Comprehensive review	High regulatory consequences
Orchestrator	Medium	System-level monitoring	Process coordination with dynamic adaptation

Orchestration and coordination strategy—Transaction monitoring benefits from a hybrid orchestration approach that combines centralized control with flexible agent interaction. The system employs a layered orchestration model:

Strategic layer—The orchestrator agent establishes overall case priorities, allocates resources and monitors system performance

Tactical layer—Specialized agents communicate directly for adjacent processes:

- The risk scoring agent feeds directly to the alert management agent
- The investigation agent coordinates with the reporting agent to ensure findings are properly documented

Operational layer—Individual agents operate autonomously within their domains while reporting status to the orchestrator

Agent communication patterns—The system implements:

Vertical communication—Status updates and strategic directives flow between the orchestrator and specialized agents

Horizontal communication—Adjacent agents exchange information directly to minimize latency

Human integration points—Designed interfaces where human expertise integrates with agent processing

Exception handling—The orchestration framework includes sophisticated exception management:

- Automatic escalation of anomalous patterns
- Dynamic reallocation of resources for urgent cases
- Explicit human decision points for regulatory-sensitive determinations

MCP integration for financial systems—Model context protocol would significantly enhance transaction monitoring capabilities by providing:

Unified data access layer—Standardizing connections to core banking systems, watchlists and external data sources

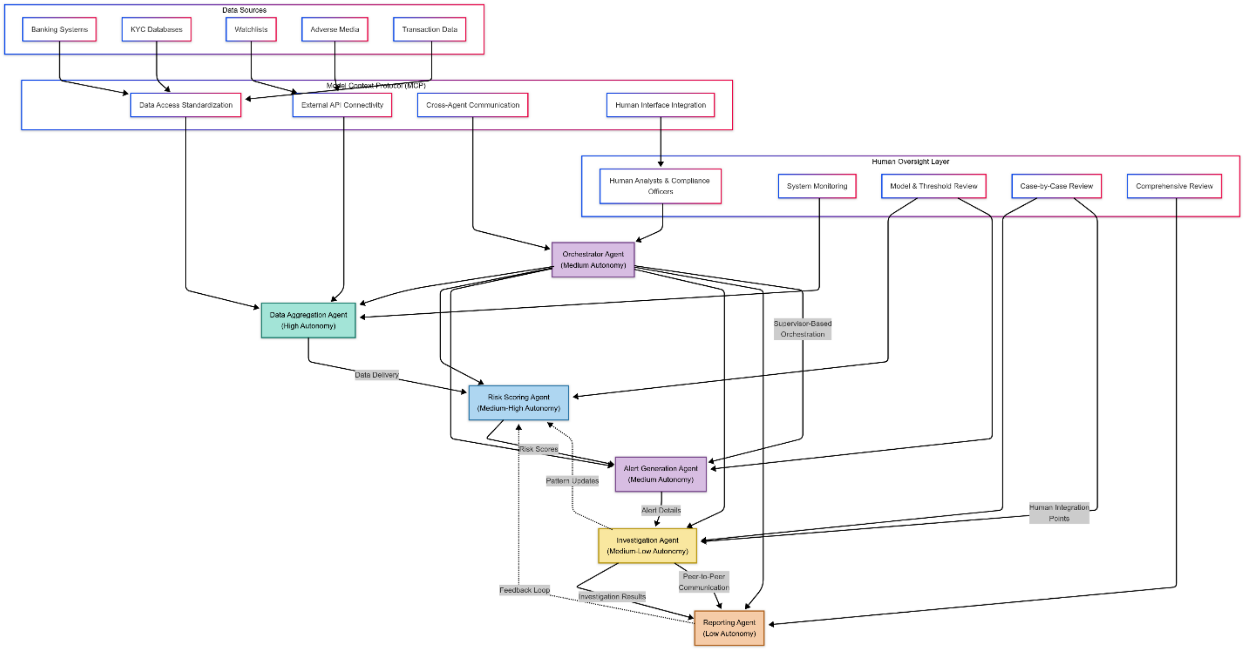
Tool integration framework—Enabling seamless incorporation of specialized tools:

- Network analysis visualization
- Document extraction utilities
- Regulatory filing interfaces

Agent communication standardization—Establishing consistent interaction patterns between agents regardless of their underlying technology

Human-agent collaboration interface—Standardizing how agents present information to human experts and incorporate their feedback

While MCP continues to mature, early adoption provides significant architectural advantages and positions the system for future enhancements as the protocol evolves.



Visualization and monitoring—A multilayered visualization strategy ensures comprehensive visibility:

Operational dashboard:

- Real-time transaction flow monitoring
- Agent activity tracking and performance metrics
- Exception queues with priority indicators
- Resource utilization visualization

Investigation interface:

- Case details with agent-gathered evidence
- Entity relationship mapping
- Risk factor visualization and explanation
- Transaction timeline analysis

Management analytics:

- Alert volume and disposition trends
- Efficiency metrics and bottleneck identification
- False positive/negative rate analysis
- Regulatory compliance tracking

System intelligence monitoring:

- Agent learning curve visualization
- Pattern adaptation effectiveness
- Human-agent collaboration metrics
- Model performance and drift indicators

Implementation pathway—A phased implementation approach delivers incremental value while managing risk:

Phase 1: Foundation building (months 1–3)

- Process mapping and task classification
- Data source inventory and integration planning
- Agent prototype development and testing
- Performance baseline establishment
- Governance framework development

Phase 2: Augmentation mode (months 4–6)

- Data aggregation agent deployment as an assistant tool
- Risk scoring agent implementation with human validation
- Alert management agent for prioritization recommendations
- Investigator feedback collection for continuous improvement
- Performance metrics establishment and monitoring

Phase 3: Collaborative execution (months 7–12)

- Data aggregation transition to supervised autonomy
- Risk scoring expansion to handle routine case disposition
- Investigation agent deployment for contextual enrichment
- Reporting agent implementation for draft generation
- Orchestrator agent introduction for basic workflow management
- Exception handling protocol development

Phase 4: Intelligent adaptation (months 13–18)

- Enhanced risk scoring with advanced pattern recognition
- Investigation agent with reasoning capabilities expansion
- Reporting agent evolution to generate complete filing drafts
- Dynamic orchestration with adaptive workflow management
- Predictive capabilities for emerging fraud patterns
- Advanced performance analytics implementation

Phase 5: System maturity (months 19–24)

- Autonomous processing for routine transactions
- Cross-system learning for pattern adaptation
- Advanced contextual analysis implementation
- Optimized human-agent collaboration framework
- Continuous evolution mechanisms based on outcomes

Expected business impact: A comprehensive agentic AI approach to transaction monitoring delivers transformative results:

Detection effectiveness: 30%–40% increase in true positive identification through sophisticated pattern recognition and contextual analysis

Operational efficiency: 70%–85% reduction in false positives, allowing investigators to focus on high-value cases

Investigation speed: 60%–75% reduction in case resolution time through automated context gathering and analysis

Cost optimization: Potential for $10–$20M annual savings for large institutions through reduced manual processing

Regulatory compliance: Enhanced standing with regulators through more comprehensive, consistent and well-documented monitoring

Adaptive capability: Continuous system evolution to address emerging financial crime techniques

Real-world use case 3: Enhanced P&C underwriting with agentic AI

The property and casualty (P&C) insurance underwriting process presents an ideal candidate for agentic AI transformation due to its complex workflow, reliance on multifaceted data sources and need for consistent risk evaluation. Implementing agentic AI in this domain can dramatically improve operational efficiency, risk assessment accuracy and customer experience while maintaining strict regulatory compliance.

Strategic role decomposition for P&C underwriting: Successful implementation of agentic AI in P&C underwriting begins with methodical role decomposition following the hierarchical approach outlined earlier. The P&C underwriting function can be segmented into four core functional areas (macro-level decomposition) and each functional area breaks down into defined processes with clear inputs, outputs and workflows (meso-level decomposition) as below:

S.no	Macro-level decomposition	Meso-level decomposition
1	Information collection and validation	• Document intake and classification (applications, inspections, claims history) • Structured data extraction and normalization • Third-party data integration (property records, satellite imagery, weather data) • Discrepancy identification and resolution
2	Risk evaluation	• Historical loss pattern analysis by property type and geography • Catastrophe model integration and interpretation • Market condition assessment and competitive positioning • Predictive modeling for loss propensity and severity
3	Policy administration	• Coverage configuration and limit determination • Rating factor application and premium calculations • Quote generation and proposal preparation • Policy document creation and delivery • Renewal assessment and retention strategy
4	Compliance management	• Jurisdictional rule checking and validation • Form and endorsement selection based on regulatory requirements • Documentation verification and certification • Regulatory reporting and filing management

At the most granular level (microlevel decomposition), tasks are classified based on agent suitability. For example:

Document intake: Classification of incoming documents, extraction of key fields, validation against expected formats

Risk scoring: Application of specific risk models, comparison against industry benchmarks, identification of risk factors

This structured decomposition creates the foundation for designing a multi-agent system that can effectively handle the complexity of P&C underwriting while maintaining appropriate human oversight where required.

Autonomy assessment framework in action—P&C underwriting requires careful calibration of agent autonomy levels based on risk complexity, regulatory requirements and potential financial impact:

Full autonomy (minimal oversight)

- Standard data gathering from established sources
- Validation of policy information against third-party databases
- Basic policy document generation for standard coverages
- Routine compliance checks against well-defined regulatory requirements

High autonomy (exception-based oversight)

- Risk assessment for standard residential properties
- Pricing for well-established risk profiles
- Renewal processing for policies without significant changes
- Catastrophe exposure calculations

Collaborative autonomy

Complex commercial property risk evaluation

- Non-standard property assessment (high-value, unique construction)
- Coverage customization for specialized needs
- Pricing for policies with multiple exceptions or unique features

Human-led

- Novel or emerging risk scenarios without established underwriting precedent
- Complex regulatory situations or jurisdictional edge cases
- High-value or strategic client negotiations
- Situations with limited or ambiguous data availability

This graduated approach ensures appropriate human involvement based on risk complexity and potential impact.

Orchestration and coordination strategy—P&C underwriting benefits from a hybrid orchestration approach combining:

Supervisor-based orchestration: An underwriting orchestrator agent provides centralized workflow management, ensuring cases progress appropriately through the underwriting lifecycle and maintaining visibility across the entire process

Peer-to-peer collaboration—Specialized agents communicating directly when efficient, in ways such as:

- Risk assessment and pricing agents collaborating on factor-specific premium adjustments
- Compliance and documentation agents confirming regulatory requirements for policy forms
- Data collection and risk assessment agents requesting additional information when anomalies are detected

The core agent ensemble typically includes:

Data acquisition agent: Collects, validates, and normalizes information from applications, third-party sources and historical records

Risk assessment agent: Evaluates property characteristics, location factors and applicant history to determine risk level

Pricing agent: Applies appropriate rating factors and calculates premiums based on risk assessment

Compliance agent: Ensures all regulatory requirements are met for the specific jurisdiction and policy type

Documentation agent: Generates and manages all required policy documentation

Client communication agent: Handles routine communications with clients and brokers

Underwriting orchestrator agent: Coordinates the overall workflow and manages exceptions

Agent communication patterns—The system implements:

Vertical communication: Status updates and strategic directives flow between the orchestrator and specialized agents

Horizontal communication: Adjacent agents exchange information directly to minimize latency

Human integration points: Designed interfaces where human expertise integrates with agent processing

Exception handling: The orchestration framework includes sophisticated exception management:

- Automatic escalation of anomalous patterns
- Dynamic reallocation of resources for urgent cases
- Explicit human decision points for regulatory-sensitive determinations

MCP implementation for enhanced integration—Implementing model context protocol provides significant advantages for P&C underwriting, particularly for:

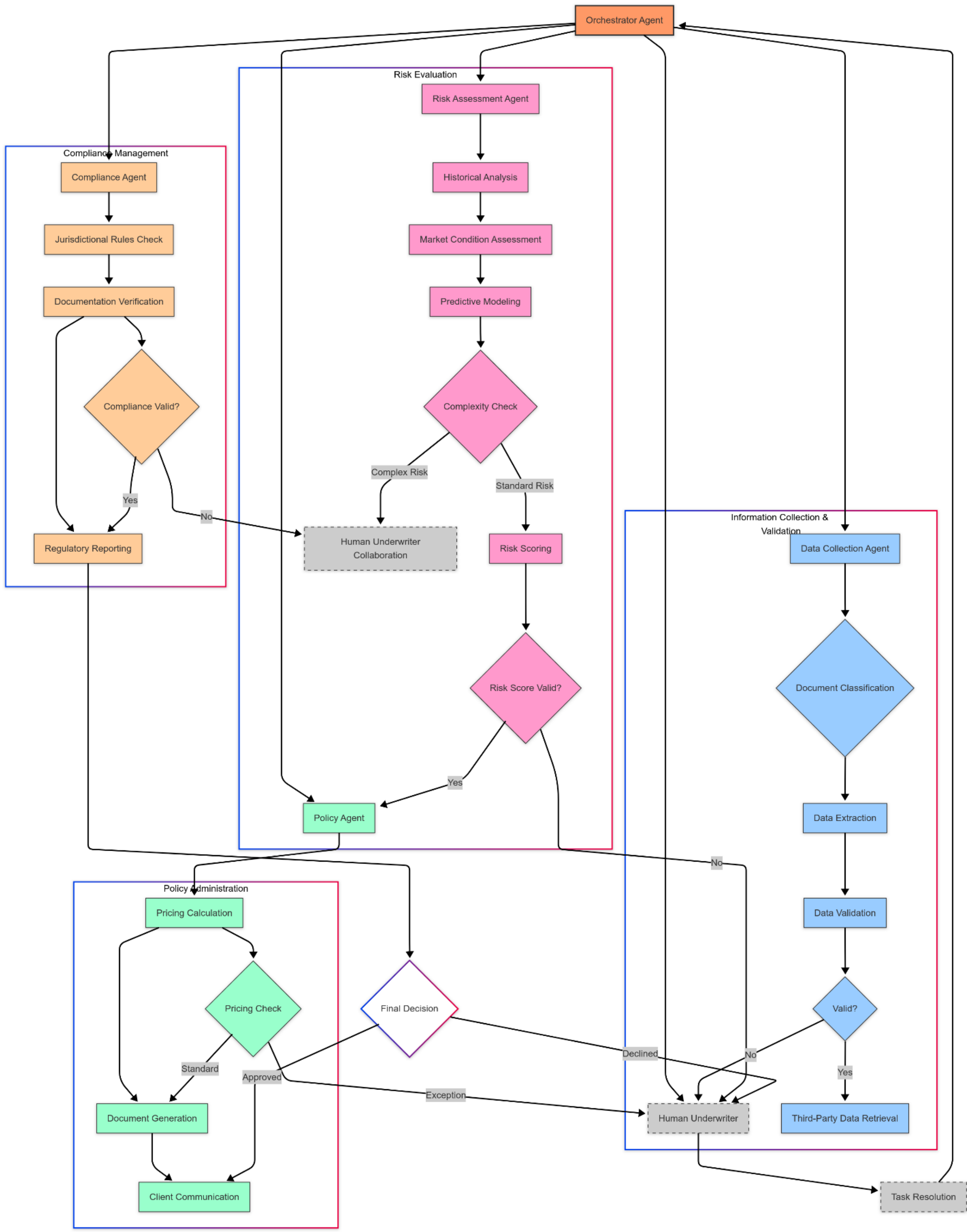
External data integration: Standardizing connections to property databases, catastrophe models and aerial imagery services

Tool orchestration: Providing agents with consistent access to rating engines, document generation systems and compliance databases

Interagent communication: Facilitating standardized information exchange for complex risk evaluations

Implementation would involve:

- Developing MCP clients for each agent
- Creating a central MCP server to translate between agents and external services
- Implementing standardized protocols for information exchange
- Establishing security measures for sensitive data handling through MCP


Orchestrator Agent
Risk Evaluation
Risk Assessment Agent
Historical Analysis
Market Condition Assessment
Predictive Modeling
Complexity Check
Complex Risk
Standard Risk
Human Underwriter Collaboration
Risk Scoring
Risk Score Valid?
Yes
No
Policy Agent
Compliance Management
Compliance Agent
Jurisdictional Rules Check
Documentation Verification
Compliance Valid?
Yes
No
Regulatory Reporting
Information Collection & Validation
Data Collection Agent
Document Classification
Data Extraction
Data Validation
Valid?
No
Yes
Human Underwriter
Third-Party Data Retrieval
Task Resolution
Policy Administration
Pricing Calculation
Pricing Check
Standard
Approved
Exception
Document Generation
Client Communication
Final Decision
Declined

Visualization and monitoring—The system includes a comprehensive visualization layer:

Process flow dashboard: Real-time visualization of applications moving through the underwriting pipeline, showing current status and agent handling each case

Agent performance metrics:

- Accuracy rates for each agent
- Processing times by task type
- Exception rates and types
- Human intervention frequency and reasons

Risk visualization tools:

- Heat maps of risk factors across the portfolio
- Comparative visualizations of risk assessments
- Anomaly detection and highlighting

Workload management Interface:

- Queue visualization for human underwriters
- Priority-based case assignment
- Capacity monitoring and load balancing

Audit and compliance dashboard:

- Complete audit trails of all agent decisions
- Regulatory compliance status by jurisdiction
- Documentation completeness metrics

Implementation pathway—A phased implementation approach balances rapid value delivery with appropriate risk management:

Phase 1: Foundation building (months 1–3)

- Conduct detailed task decomposition using the multilevel framework
- Develop and train initial agents focused on data acquisition and standard risk assessment
- Establish governance framework and performance metrics
- Create underwriter training program for the new human-agent collaboration model

Phase 2: Augmented underwriting (months 4–6)

Deploy initial agents as underwriter assistants, providing recommendations but not making independent decisions

- Implement comprehensive feedback mechanisms to improve agent accuracy
- Develop performance dashboards tracking agent recommendation quality
- Begin development of pricing and compliance agents

Phase 3: Selective autonomy (months 7–12)

- Graduate data acquisition agent to autonomous operation for standard inputs
- Deploy pricing and compliance agents in assistant mode
- Implement initial orchestration capabilities between agents
- Establish exception handling workflows with clear escalation paths
- Create visualization tools for process monitoring and bottleneck identification

Phase 4: Orchestrated operations (months 13–18)

- Deploy underwriting orchestrator agent to manage end-to-end process flow
- Transition standard, low-complexity risks to fully agent-led processing with human supervision
- Implement advanced agent communication protocols
- Optimize system performance based on accumulated operational data
- Enhance exception handling and self-correction mechanisms

Phase 5: Adaptive enterprise (months 19–24)

- Expand autonomous processing to medium-complexity risks
- Implement advanced learning capabilities based on human feedback patterns
- Develop cross-functional integration with claims and customer service
- Refine human-agent collaboration for complex underwriting scenarios
- Deploy continuous improvement mechanisms for ongoing optimization

Throughout implementation, maintaining appropriate human involvement remains essential, particularly for complex risks, regulatory edge cases and strategic client relationships.

Expected business impact—Properly implemented agentic AI in P&C underwriting delivers substantial measurable benefits:

- 60%–70% reduction in routine underwriting processing time
- 30%–40% improvement in underwriter productivity for complex risks
- 25%–35% decrease in policy issuance errors
- 15%–20% improvement in risk selection accuracy
- Enhanced customer experience through faster quote turnaround
- More consistent application of underwriting guidelines across the portfolio
- Improved capture and utilization of institutional knowledge
- Greater scalability during peak submission periods

The transformed underwriting operation achieves a powerful balance between automation efficiency and human expertise, enabling underwriters to focus on complex risk evaluation and strategic client relationships while routine processes proceed autonomously.

Conclusion and forward outlook

Key success factors

As demonstrated through our industry case studies, agentic AI represents a transformative approach to enterprise operations—not merely an incremental improvement over existing automation. The strategic framework presented in this white paper—decomposing complex roles, orchestrating multi-agent systems, determining appropriate autonomy levels and implementing through measured phases—provides organizations with a roadmap for capturing this value.

Future evolution of enterprise agentic AI

Organizations that succeed with agentic AI implementation share common characteristics: They start with clear business objectives rather than technology capabilities, they progress deliberately along the autonomy spectrum-building confidence and capabilities at each stage and they invest in robust governance and human-agent collaboration models.

Getting started: Next steps

As agentic AI technologies continue to evolve, early adopters following these principles are positioned to create sustainable competitive advantages. The gap between consumer and enterprise applications is narrowing, and forward-thinking organizations are already establishing the foundations for truly intelligent enterprises. The question is no longer whether agentic AI will transform industries, but rather which organizations will lead this transformation, and which will be left behind.

For executives and transformation leaders, the time to act is now—not with sweeping replacements of existing systems, but with strategic implementations that demonstrate value while building toward a more comprehensive vision. By applying the frameworks and lessons presented here, organizations can navigate the complexity of agentic AI implementation and realize its transformative potential.

Engineering AI for Impact in BFSI

Cognizant engineers AI for impact. We help clients embrace AI with confidence and modernize their business foundations, achieve hyper-productivity and drive growth and innovation. We've built a "last mile" for AI implementation – platforms, services and IP that accelerate scaled adoption while also enhancing the quality of AI outputs, enabling governance and orchestration, and optimizing cost. And we're partnering with enterprise leaders to architect multi-agent, dynamic operations for unparalleled agility and efficiency.

With our expertise in engineering, cloud, data and AI, we speed our clients' transformation journeys and help them stay relevant in a fast-changing world. We're translating AI ambitions into market-leading capabilities, delivering tangible business impact and sustained competitive advantage."

References:

1. Andy Nguyen (2025). "Model Computer Programming (MCP): From Simple Text Prediction to Powerful Tool Users." Medium. https://medium.com/@huyhoan0705/model-computer-programming-mcp-from-simple-text-prediction-to-powerful-tool-users-292bc054259c

2. Reddy, M. (2025) . "The Evolution of LLMs: From Text Prediction to MCP—The New Standard in AI Integration." Medium. https://medium.com/@mukeshreddy662369/the-evolution-of-llms-from-text-prediction-to-mcp-the-new-standard-in-ai-integration-cc0392ebfcb0

3. Markovate. (2025) . "Agentic AI Architecture: A Deep Dive". Markovate Blog. https://markovate.com/blog/agentic-ai-architecture/

4. AWS. (2025). "Automate tasks in your application using AI agents" Amazon Web Services Documentation. https://docs.aws.amazon.com/bedrock/latest/userguide/agents.html

5. Fetch.ai. (2020). "Autonomous Economic Agents: an update." Medium. https://medium.com/fetch-ai/autonomous-economic-agents-an-update-9385dfc1abd1

6. Fetch.ai. (2024). "Communicating with other agents." Fetch.ai Documentation. https://fetch.ai/docs/guides/agents/intermediate/communicating-with-other-agents

7. Snowflake. (2024). "Your Enterprise Data Needs an Agent." Snowflake Blog. https://www.snowflake.com/en/blog/ai-data-agents-snowflake-cortex/

8. Floch, J. et al. (2024). "A survey of flexible agent interaction approaches." ACM Digital Library. https://dl.acm.org/doi/abs/10.3233/MGS-130208

9. Anthropic. (2024). "Introducing the Model Context Protocol." Anthropic News. https://www.anthropic.com/news/model-context-protocol

10. SmythOS. (2024). "Agent Communication Protocols: An Overview." SmythOS Documentation. https://smythos.com/ai-agents/ai-agent-development/agent-communication-protocols/

Author

Anoop Nair
Senior Vice President, Global Head of FSI - IOA
anoop.nair@cognizant.com

Follow in

Anoop Nair is the Senior Vice President and IOA FSI Global Vertical Leader at Cognizant. In this role, he is responsible for driving strategy and market share, while ensuring customer success and strengthening delivery of modern business operations for the Financial Services and Insurance (FSI) sector. He's passionate about harnessing AI and Gen AI to drive real business impact—unlocking insights, accelerating outcomes, and reimagining operations to create lasting value for clients.

cognizant®

Cognizant helps engineer modern businesses by helping to modernize technology, reimagine processes and transform experiences so they can stay ahead in our fast-changing world. To see how Cognizant is improving everyday life, visit them at www.cognizant.com or across their socials @cognizant.

World Headquarters
300 Frank W. Burr Blvd.
Suite 36, 6th Floor
Teaneck, NJ 07666 USA
Phone: +1 201 801 0233
Fax: +1 201 801 0243
Toll Free: +1 888 937 3277

European Headquarters
280 Bishopsgate
London
EC2M 4RB
England
Tel: +44 (01) 020 7297 7600

India Operations Headquarters
5/535, Okkiam Thoraipakkam,
Old Mahabalipuram Road,
Chennai 600 096
Tel: 1-800-208-6999
Fax: +91 (01) 44 4209 6060

APAC Headquarters
1 Fusionopolis Link, Level 5
NEXUS@One-North, North Tower
Singapore 138542
Tel: +65 6812 4000

© Copyright 2025–2027, Cognizant. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the express written permission of Cognizant. The information contained herein is subject to change without notice. All other trademarks mentioned here in are the property of their respective owners.

WF3440600