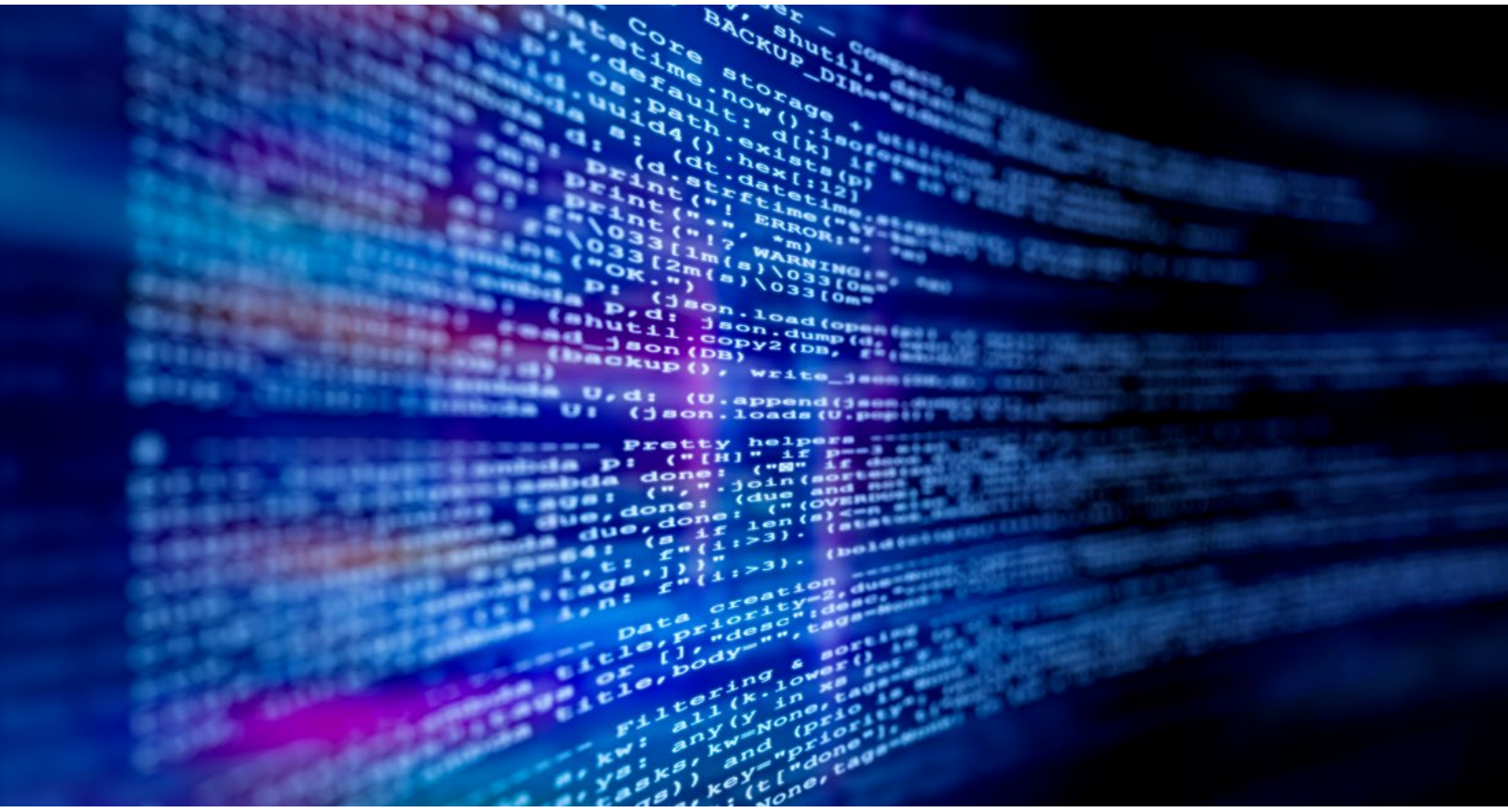
McKinsey & Company

Financial Services Practice

Can agentic AI (finally) modernize core technologies in insurance?

Insurers have long understood the need to transform their core technologies. So, why haven't they? Agentic AI may finally make the difference.

This article is a collaborative effort by Arun Gundurao, Krish Krishnakanthan, Rory Walsh, Sanjay Kaniyar, and Tanguy Catlin, representing views from McKinsey's Financial Services Practice.



April 2026

Modernizing the core technologies underpinning the insurance industry has long been viewed as essential. Yet excuses for failing to act persist, from complexity to cost to risk. No more. Agentic AI, drawing on experience from large-scale AI transformations, including work pioneered within QuantumBlack, McKinsey's AI arm, presents the opportunity to fundamentally transform how modernization is done end to end by capturing legacy knowledge at scale, compressing rework loops, and improving predictability across testing, reconciliation, and cutover.[1] It holds the promise of agents performing and coordinating discrete tasks with auditable outputs and human-in-the-loop controls—empowering leaders to rethink their modernization portfolio. And not only could it help insurers significantly improve financial viability by moving from one-off, bespoke programs toward a repeatable, scalable modernization factory,[2] but early adopters of agentic AI may gain a competitive edge.

The problem: Insurance core modernization's endless challenge

Insurers failing to modernize core systems often blame their inaction on specific structural costs and risks related to how technologies are built and operated.[3] And there's no denying these challenges are real: Executives know that legacy core modernization is hard because the "core" is a living socio-technical system that may comprise several decades' worth of sparsely documented embedded business rules, batch windows, custom interfaces, and data semantics.

In addition, in policy administration migrations, several common constraints may repeatedly create schedule slippage and re-baselining cycles,[4] from underdocumented product logic and actuarial settings to semantic gaps that often surface late and drive rework as well as cutover and runbook risks that force conservative sequencing. These dynamics have direct economic consequences that also reduce the appeal of modernizing core systems, including causing costly "double-bubble" periods in which insurers pay to run the legacy stack while funding the change program and prolonging the need to run systems in parallel, extending the timeline for decommissioning legacy platforms.[5]

A final but critical nuance undermining insurers' motivation is understanding where effort actually goes. In most insurance migrations, rewriting code or configuring the target platform is only a small portion of the work. A disproportionate share of time and investment often sits in understanding and configuring the rules, data conversion, quality control, reconciliation, operational readiness, and postmigration stabilization. That's why platform migrations often disappoint when teams attempt to recreate legacy complexity on a modern core. The discipline is knowing not only what to build but also what not to rebuild.

The potential solution: Agentic AI reshapes the cost curve

While the challenges of core system modernization have proved insurmountable for many insurers, agentic AI may provide a solution, drawing on experience from large-scale AI transformations, including work pioneered within QuantumBlack, McKinsey's AI arm. Autonomous or semiautonomous software agents can interpret legacy artifacts, produce structured documentation, generate and validate configuration or code, create and run tests, and coordinate workflows across the software delivery life cycle.[6]

This is different from a developer copilot: While copilots assist a user moment by moment, agents are designed to pursue a goal, break it into tasks, use tools and context, and iterate based on

1 "AI for IT modernization: Faster, cheaper, better," McKinsey, December 2, 2024; *Seizing the agentic AI advantage*, McKinsey, June 13, 2025.
2 For more, see "AI for IT modernization: Faster, cheaper, better," McKinsey, December 2, 2024; *Seizing the agentic AI advantage*, McKinsey, June 13, 2025; "The agentic organization: Contours of the next paradigm for the AI era," McKinsey, September 26, 2025.
3 For more, see Krish Krishnakanthan, Sanjay Kaniyar, Tanguy Catlin, and Sophie Ru, "How P&C insurers can successfully modernize core systems," McKinsey, May 12, 2025.
4 For more, see Krish Krishnakanthan, Sanjay Kaniyar, Tanguy Catlin, and Sophie Ru, "How P&C insurers can successfully modernize core systems," McKinsey, May 12, 2025.
5 For more, see Krish Krishnakanthan, Sanjay Kaniyar, Tanguy Catlin, and Sophie Ru, "How P&C insurers can successfully modernize core systems," McKinsey, May 12, 2025.
6 For more, see "AI for IT modernization: Faster, cheaper, better," McKinsey, December 2, 2024; *Seizing the agentic AI advantage*, McKinsey, June 13, 2025.

feedback and controls.[7] In a policy administration migration, this matters because the biggest bottlenecks are rarely typing code but rather the loops of discovery, mapping, testing, reconciliation, and cutover. Exhibit 1 outlines where agents can create efficiency within each domain step with, in our experience, typical productivity improvements ranging from 10 to 90 percent, depending on the step and the degree of automation.

Most core technologies in the insurance industry use programming language, coding, and even systems created decades ago. In many cases, it's difficult to find people fluent in these languages and systems—or even workers able to understand them. That's why some of the most significant gains in enabling core modernization with agents come from decoding and translating outdated programming language, drafting rules, and mapping artifacts.

Exhibit 1

Agentic AI could deliver productivity improvements of up to 90 percent in the insurance-core-system modernization process.

Impact of agentic AI in the insurance-core-system modernization process

Domain step	Discovery, reverse engineering, and product or rule understanding	Target configuration acceleration for design and 3rd-party policy administration platforms	Data mapping, conversion, and quality	Testing, reconciliation, and defect cycle compression	Cutover, hypercare, and operations readiness	Program management and governance
How AI can help	Reduce SME[1] dependency and early-cycle uncertainty	Support informed customization decisions using approved patterns and constraints	Surface issues earlier and compress data readiness timelines	Shorten feedback loops and reduce SME effort	Enable role-based training and policy servicing with decision support and continuous learning	Provide continuous surfacing of execution and risk signals
	↓	↓	↓	↓	↓	↓
	Through rapidly reverse engineering legacy policy logic and batch behavior into decision tables, mappings, and explainable knowledge	By generating and validating configurations of 3rd-party policy administration platforms and tracing requirements end to end	Through automated mapping analysis, ETL[2] generation, profiling, anomaly detection, and test-data creation	By automatically generating, executing, diagnosing, and reconciling tests, thereby linking defects to root causes and financial outcomes	By orchestrating cutover runbooks and supporting hypercare operations	Through automating PMO[3] reporting, dependency tracking, and compliance artifacts
Typical productivity improvement, %	20–50	15–40	20–60	15–90	10–40	25–50

[1]Subject matter expert.
[2]Extract, transform, and load.
[3]Program management office.

McKinsey & Company

[7] For more, see "AI for IT modernization: Faster, cheaper, better," McKinsey, December 2, 2024; *Seizing the agentic AI advantage*, McKinsey, June 13, 2025; "The agentic organization: Contours of the next paradigm for the AI era," McKinsey, September 26, 2025.

Agents can read code written in archaic languages, reverse engineer the logic, and convert it into plain English. Likewise, agents can parse code and extract and document the business rules embedded within it. In many cases, an agent can accomplish within days what would take a trained subject matter expert months or even years to do.

Once these agent capabilities are established, the incremental cost of modernizing additional products and systems can fall quickly because the same agents, patterns, and context layers can be reused across waves and domains. In this way, agentic AI creates a portfolio option that insurance technology leaders have not had before. In addition to enabling the transition to a new core system (which is often a software-as-a-service platform), there is often a long tail of adjacent legacy applications, utilities, and interfaces that need to be migrated. Agentic approaches can make selective rewrites of this long tail economically viable, reducing maintenance cost and long cycles of migration and integration.[8]

Best practices to capture value

Reshaping the way insurers modernize their core systems with agentic AI requires more than simply applying the technology. It demands fundamental shifts in workflows, roles, governance, and sequencing. In our experience, three moves matter most:

Building modular agents to accelerate the whole process

The highest-performing technology migrations decompose work into reusable, composable agents across extraction, validation, transformation, orchestration, and generation. This approach improves control, makes outputs auditable, and enables reuse across discovery, data, testing, and cutover. In practice, it means treating agents as a library of atomic capabilities, each with clear inputs, acceptance criteria, and escalation paths to humans (Exhibit 2). It also makes it easier to update specific components as models and tooling evolve, without destabilizing the whole workflow.

Shifting from a single-program mindset to a modernization portfolio

Agentic capabilities fundamentally change the unit economics of modernization because once core agents are built and governed, their marginal cost of reuse sharply declines. That means modernization can be framed not as a single large-scale migration but as a coordinated portfolio of opportunities across the full technology estate.

McKinsey research on AI-enabled software engineering finds that the greatest impact comes when AI is embedded across workflows and scaled systematically, rather than confined to isolated use cases.[9] In an insurance context, this means looking beyond migrating the administration of single policies to the broader landscape: multiple product lines, billing and claims integrations, satellite systems, and high-maintenance legacy utilities. This does not mean that value cannot be realized from tackling modernization in phases, such as end-to-end prospecting—design choices can be tailored to achieve specific outcomes according to a particular cadence or timeline. But the ideal end state for optimal impact is full modernization of the legacy core.

With a reusable agent stack in place, the incremental effort to modernize additional products or adjacent applications has the potential to fall materially. That allows leaders to evaluate platform migrations and decommissioning decisions, as well as selectively rewrite long-tail legacy applications, as part of an integrated road map. And this portfolio lens enables compounding value creation while maintaining governance and platform discipline.

Redesigning roles, governance, and risk management for agentic execution

Insurance modernization is regulated and operationally sensitive. Agentic workflows need controls by design such as human-in-the-loop approvals at stage gates, traceability from requirements to configuration to test evidence, and clear model-validation practices. Agentic AI fundamentally reshapes roles and creates more full-stack “product definers” and “product builders,”

[8] For more, see “AI for IT modernization: Faster, cheaper, better,” McKinsey, December 2, 2024; *Seizing the agentic AI advantage*, McKinsey, June 13, 2025.

[9] Charlotte Relyea and Martin Harrysson, “The AI revolution in software development,” McKinsey, April 1, 2026.

Exhibit 2

AI agents can be deployed across the migration workflow.

Deployment of AI agents, by category

McKinsey AI agent categories

Interaction | Reasoning | Extraction | Validation | Transformation | Orchestration | Generation | Learning

1 Discovery, reverse engineering, and product or rule understanding

- **Document extraction** (eg, extract relevant fields, pull policy docs and control memos)
- **Table extraction** (eg, convert tabular content, turn decision tables)
- **Entity extraction** (eg, identify entities, products, systems, owners)
- **Summarization** (eg, condense long text, produce "rule intent and edge cases and open questions")
- **Classification** (eg, cluster rules by product, transaction type, complexity, or exception patterns)
- **Root-cause analysis** (eg, diagnose patterns or dependencies, surface dependency chains, and anomalies)
- **Interview** (eg, conduct structured Q&A, guide SMEs[1] through prompts to elicit rule intents and exceptions)
- **Report generation** (eg, produce structured reports to create an "explainable rule pack" per policy domain)

2 Target design and configuration acceleration

- **Planning** (eg, sequence multistep plans, draft configuration build plan per domain)
- **Decision tree** (eg, apply constraints or patterns to drive if-then configuration decisions and exception handling)
- **Process validation** (eg, check configurations against defined FAST[2] design standards and governance guardrails)
- **Quality assurance** (eg, test and validate outputs against acceptance criteria before promoting to the next stage)
- **Code generation** (eg, generate code snippets and templates, draft repeatable templates and scripts)
- **Content generation** (eg, create configuration notes and requirement-to-configuration trace statements)
- **Approval** (eg, run stage-gate reviews for configuration packages, log decisions)
- **Routing** (eg, route configuration questions or approvals to the right product SMEs, architects, and delivery owners)

3 Data mapping, conversion, and quality

- **Table extraction** (eg, digitalize mapping specs and transformation rules)
- **Format conversion** (eg, transform data formats, standardize mapping artifacts, extracts, and test data sets)
- **Classification** (eg, group fields by domain, criticality, transformation type, and expected quality controls)
- **Data quality** (eg, assess and flag issues, detect missing or invalid values and inconsistent domain codes)
- **Fact checking** (eg, verify claims vs trusted sources)
- **Scenario generation** (eg, create realistic test cases, generate representative test data sets)
- **Monitoring** (eg, track signals and alerts, flag anomalies)

4 Testing, reconciliation, and defect cycle compression

- **Scenario generation** (eg, create test cases and regulatory or control cases)
- **Report generation** (eg, create reconciliation reports, defect packs, and readiness summaries)
- **Quality assurance** (eg, validate against benchmarks, verify outputs vs expected results, prevent regressions)
- **Summarization** (eg, condense evidence, ie, what failed, where, likely driver, and next action)
- **Prioritization** (eg, rank by criteria, sort defects by business impact, operational risk, and relevance)
- **Root-cause analysis** (eg, diagnose underlying drivers, connect defects to different drivers)
- **Monitoring** (eg, watch defect aging, recon variance trends and regression spikes)

[1]Subject matter experts.
[2]Flexible, appropriate, structured, and transparent.
[3]Program management office.
Source: McKinsey Agents@Scale; McKinsey analysis

McKinsey & Company

Exhibit 2 continued

AI agents can be deployed across the migration workflow.

Deployment of AI agents, by category

McKinsey AI agent categories

Interaction | Reasoning | Extraction | Validation | Transformation | Orchestration | Generation | Learning

5 Cutover, hypercare, and operations readiness

- **Scheduling** (eg, orchestrate cutover runbook timing and checkpoints)
- **Routing** (eg, direct work to owners, route incidents to the correct groups to resolve)
- **Approval** (eg, log go/no-go decisions, manage checkpoint approvals)
- **Escalation** (eg, route complex cases upward, trigger human escalation for high-severity or ambiguous cases)
- **Conversational** (eg, guided "ops assistant" interaction for hypercare triage and runbook execution)
- **Form filling** (eg, capture incident details, decisions, actions taken, and evidence in consistent fields)
- **Rewriting** (eg, tailor updates for execs vs operators)
- **Pattern learning** (eg, learn recurring issue patterns, improve playbooks and decision support over time)

6 Program management and governance

- **Report generation** (eg, structured reporting, automate PMO[3] packs: milestones, risks, dependency tracking, KPIs)
- **Summarization** (eg, condense inputs, synthesize updates into exec-ready narratives and governance updates)
- **Fact checking** (eg, verify claims, validate status assertions and metrics against source-of-record artifacts)
- **Monitoring** (eg, continuously flag schedule-slip risk, defect leakage, variance)
- **Approval** (eg, manage stage gates, maintain auditable decision trail)
- **Prioritization** (eg, rank focus areas, highlight critical path items and top risks requiring leadership intervention)

[1]Subject matter experts.
[2]Flexible, appropriate, structured, and transparent.
[3]Program management office.
Source: McKinsey Agents@Scale; McKinsey analysis

McKinsey & Company

making talent upskilling critical. For technology leaders, this implies treating agents as a new production system that requires defined privileged access, monitoring of behavior and outcomes, and the use of auditable artifacts for regulatory and internal assurance, with humans involved throughout.

What to do next

Core modernization has for good reason long been viewed as one of the largest and most risk-laden technology investments insurers undertake. Yet agentic AI directly addresses the bottlenecks and expertise gaps that prevail today. The strategic question becomes not whether to experiment with agents but how to deploy them in ways that materially improve certainty on cost, risk, and timeline. That requires three shifts: embedding agents across the full modernization life cycle rather than in isolated tasks, sequencing modernization as a portfolio to capture reuse and declining marginal cost, and redesigning roles, governance, and risk management for agentic execution.

Insurers that combine reusable agent capabilities with strong governance and platform rigor can accelerate legacy decommissioning while preserving control. The advantage is not faster coding alone but a structurally different and value-compounding modernization model. And it may finally deliver the modern technology backbone insurers have long known is essential.

Arun Gundurao is a partner in McKinsey's New York office, **Krish Krishnakanthan** is a senior partner in the Connecticut office, **Rory Walsh** is a partner in the Dublin office, and **Sanjay Kaniyar** is a partner in the Boston office, where **Tanguy Catlin** is a senior partner.

The authors wish to thank Ari Libarikian, Nick Milinkovich, Shalaka Kulkarni, and Sid Kamath for their contributions to this article.