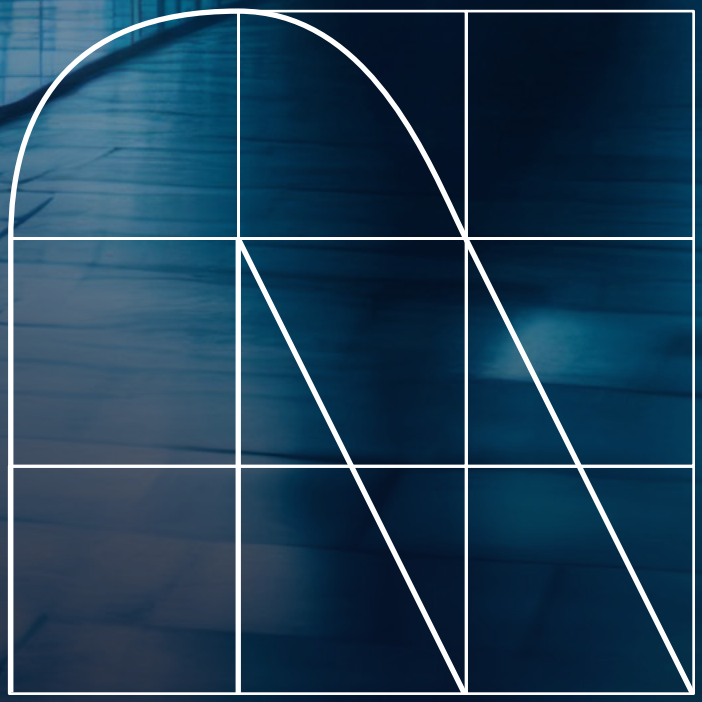
NTT DATA

The adaptive advantage: Powering enterprise applications with agentic AMS and autonomous testing



Content

03 The transformation: How agentic AI is reshaping application services

06 Agentic AMS: Autonomous application management for enterprise scale

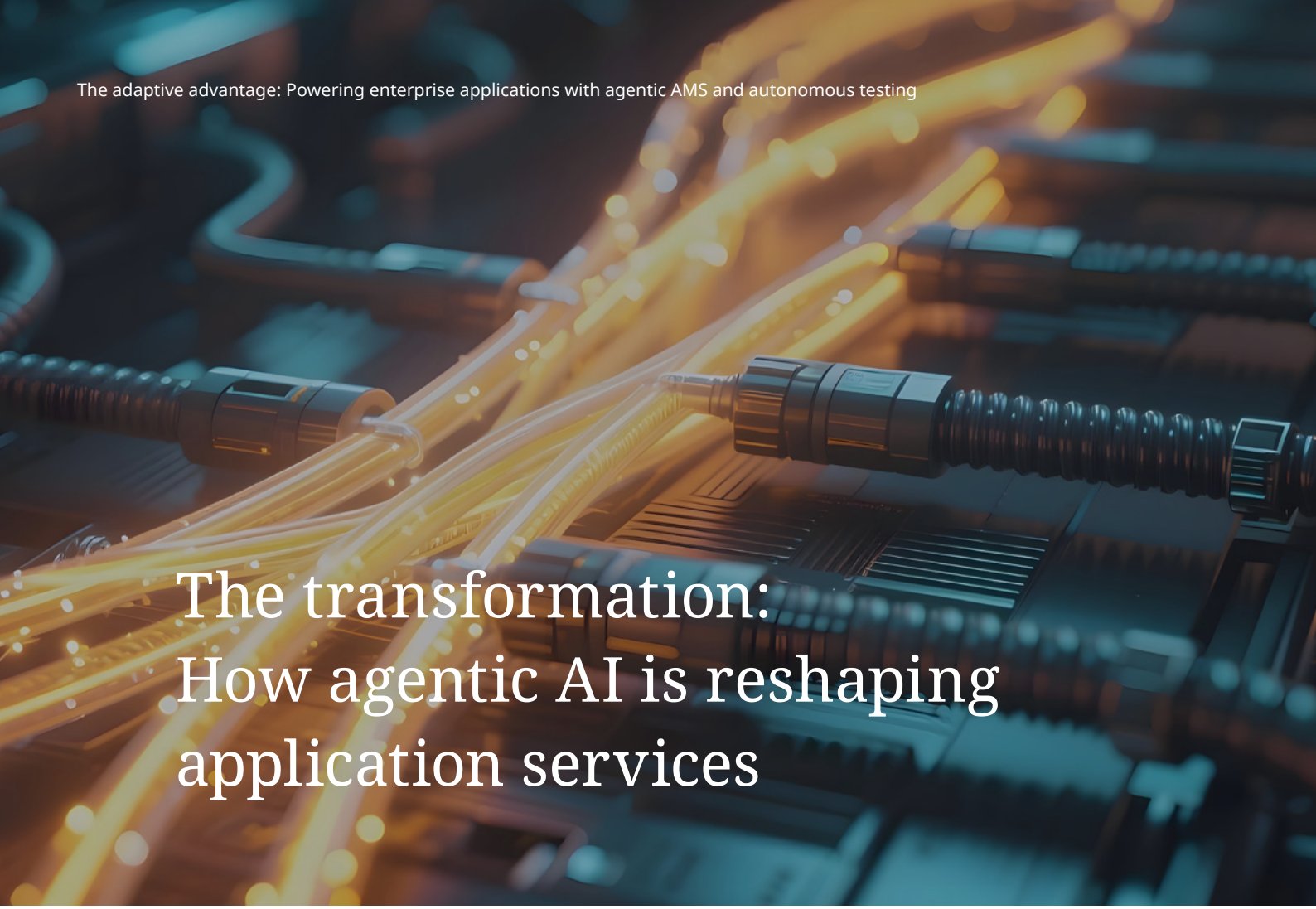
10 Autonomous testing: The intelligent quality engine for modern enterprises

14 The power of agentic AMS and autonomous testing together

16 Agentic operating model: The future state

18 Implications of agentic AMS and ATA in specific industries

20 Charting a course for the intelligent enterprise



The transformation: How agentic AI is reshaping application services

The winds of change: Why agentic AI's arrival is timely

Our digital landscape, a dynamic mix of legacy systems, cloud-native innovations and packaged solutions, has grown increasingly complex. It's an environment with familiar challenges: increasing portfolio diversity, a shortage of skilled workers and ever-present technical debt. NTT DATA's latest report on artificial intelligence, the **2026 Global AI Report: A Playbook for AI Leaders**, presents input from more than 2,500 executive and senior leaders in the context of three cohorts: AI leaders, AI laggards and all others.

The report highlights AI leaders' proactive approach to technical debt. A significant 29.5% of these leaders — a figure nearly double that of laggards at 17.2% and all others at 16.5% — identify technical debt (such as maintenance load and data fragmentation) as their top infrastructure blocker, diverting resources away from innovation. Leaders attack this blocker head-on. They view technical debt not only as a problem but as an opportunity. Agentic AI offers a strategic solution. It augments scarce expertise, reduces maintenance overhead and stabilizes complex application environments. And it represents a shift toward greater harmony in digital operations.

AI's evolution: From experimentation to essential capability

AI is no longer confined to experimental phases. The NTT DATA report confirms that AI has moved from a peripheral role to a central one. A significant 85.6% of AI leaders now integrate AI into their back- and mid-office workflows, considerably more than the 71.1% of AI laggards and 73.8% of all others. These implementations include application development and maintenance, rigorous testing and operational optimization — areas ripe for agentic capabilities. As AI integrates throughout the application lifecycle, organizations are shifting from isolated experiments to continuous, AI-powered service delivery.

> “Organizations that fully align their AI and business strategies are far more likely to realize tangible financial benefits from AI. 83.6% of ‘fully aligned’ organizations report a ≥5% profit increase from AI.”

The dynamic duo: When creation meets execution

A key development is the move beyond mere code suggestions. The NTT DATA report distinguishes generative AI (GenAI) as the 'creator' and agentic AI as the 'doer.' This distinction highlights a significant leap. Agentic AI enables autonomous, outcome-oriented systems that operate in closed loops. This means AI that not only generates content but actively executes, validates and optimizes workflows. It represents a transformation, shifting application services from reactive support models to adaptive, self-improving systems. This signifies a more dynamic future.

Why application management services (AMS) and testing offer immediate value for agentic AI

These domains are particularly well-suited for agentic applications and agentic intervention because of their structured workflows, repeatable tasks, and clear inputs and outputs. This aligns with the NTT DATA report, which shows AI leaders focusing on redesigning workflows in high-value domains to achieve productivity and cost benefits. Incident management, regression testing and release validation produce standardized artifacts and clear decision paths. This positions these tasks as strong candidates for agent-driven execution, with managed operational risk. They are a natural fit.

The symphony of agents: Orchestrating success

Multi-agent orchestration, such as what a CrewAI/OpenAI pipeline provides, has a suitable application here. The report notes that AI leaders view AI as an enterprise-wide capability, integrating it into their core delivery models. This is an expansive view. AMS and testing workflows naturally lend themselves to this approach. They compartmentalize into distinct roles suitable for monitoring, analysis, execution and validation. AMS, specifically, compartmentalizes workflows into distinct roles suitable for analysis and design, development, execution and monitoring. These roles map effectively to multi-agent orchestration, allowing agentic pipelines to not only mirror but improve existing service delivery structures. This can occur without disruptive reinvention.

Achieving measurable ROI

Measurable returns are a key benefit of adopting an AI leader approach to enterprise application advancement. Enterprises adopting this approach can realize benefits such as reduced maintenance cycles, quicker test creation and stable deployments. NTT DATA research supports these results. AI leaders are nearly 2.5 times more likely to achieve revenue growth above 10% and 3.6 times more likely to operate with margins of 15% or more compared to other organizations. Early, high-impact deployments in operational domains like AMS and testing can drive these gains. Here, agentic AI contributes to shorter cycle times and better quality while fostering reinvestment and scalable success. It's a robust strategy.

Agentic AMS: Autonomous application management for enterprise scale

Agentic AMS represents a fundamental shift in how application management services are delivered. It transitions from human-led, ticket-driven execution to autonomous, outcome-oriented operations. Agentic AMS is built on a multi-agent architecture that uses a CrewAI/OpenAI pipeline. It analyzes, modifies, tests and deploys application changes with minimal human intervention. Agentic AMS functions as a highly capable, self-managing system for application services.

In this model, tickets from ITSM tools such as JIRA and ServiceNow evolve beyond static work requests. They become structured inputs that trigger coordinated agent workflows. Specialized agents interpret requirements, analyze the codebase, generate modifications, validate outcomes and prepare deployments. This creates a closed-loop system that continuously executes and learns.

A clear demonstration of this capability is the successful application of agentic AMS to a 4-million-line open-source customer relationship management (CRM) system. Autonomous agents generated and validated code changes across a massive and complex codebase. The agents operated consistently, accurately and with full end-to-end validation. Achieving this scale with traditional AMS models would be impractical, slow and cost-prohibitive.

This level of large-scale, autonomous code transformation aligns with the 2026 Global AI Report's definition of agentic AI as the 'doer.' These are systems that move beyond creation to execute, iterate and optimize in closed loops. It transforms static workflows into adaptive ecosystems. Achieving this scale and speed is not feasible with conventional AMS approaches.

Why agentic AMS matters

Accelerating modernization: The 2026 Global AI Report says that AI leaders prioritize core innovation — rebuilding applications with embedded AI instead of using simple additions. Agentic AMS supports this forward-thinking strategy by automating large-scale remediation, refactoring and modernizing activities across both legacy and hybrid portfolios. It provides a pathway to genuinely updating, not simply patching.

Transforming maintenance: Traditional AMS often feels reactive and ticket-centric. Agentic AMS, however, shifts this paradigm to autonomous, event-driven execution, where agents proactively analyze issues, recommend changes and implement fixes. This mirrors how AI leaders are redesigning workflows end-to-end to unlock significant productivity gains and cost benefits. It's a move toward anticipating, rather than simply reacting.

Improving quality: Every agent-generated code change is meticulously paired with autonomous validation, testing and verification. This aligns seamlessly with the report's focus on governed, enterprise-grade AI, moving organizations beyond merely asking, "Can we do this?" to confidently assert, "Can we prove we did it safely and reliably every time?" Quality becomes an inherent feature, not an afterthought.

Boosting reliability: AI leaders openly acknowledge that high maintenance needs and technical debt are notable barriers to innovation: 29.5% cite it as their primary infrastructure challenge. Agentic AMS reduces the need for niche legacy skills and instead embeds critical knowledge directly into agents that operate consistently across diverse and complex environments. It fortifies systems with consistent, built-in expertise.

Delivering consistency at scale: Agents execute standardized, governed workflows with full traceability and auditability. The NTT DATA report finds that 55.9% of leaders follow a centralized AI governance model and treat AI as an overarching enterprise capability — rather than an isolated IT experiment. Consistency is a goal and a built-in operational standard, even as operations expand.

As organizations strive to modernize aging portfolios while sustaining rapid delivery, agentic AMS offers a scalable, autonomous engine. It's designed to keep pace with business demand and navigate operational complexity, ensuring applications evolve with purpose and precision.

How agentic AMS works

Agentic AMS is thoughtfully constructed as an orchestrated, multi-agent pipeline. This design carefully mirrors and improves existing enterprise delivery models. It's a sophisticated system built for precision and adaptability.

Requirement interpretation: Intake agents adeptly parse ITSM tickets, change requests or event triggers, translating unstructured inputs into clearly actionable requirements and constraints. This ensures that every piece of work begins with a solid understanding.

Code understanding and impact analysis: Specialized analysis agents build a contextual understanding of the codebase, its dependencies and potential downstream impacts. This is a critical step for operating safely and effectively at scale.

Autonomous code modification: Execution agents then generate targeted code changes. These changes carefully align with specified requirements, architectural standards and governance policies. The goal is precise, policy-compliant manipulation.

Automated test generation: Testing agents dynamically create and update unit, integration and regression tests. They ensure these tests match each code change, providing immediate validation.

Validation: Validation agents verify correctness, security, performance and compliance. This robust process ensures that all changes meet established enterprise quality thresholds.

Deployment: Deployment agents prepare release artifacts and integrate smoothly with continuous integration/continuous delivery (CI/CD) pipelines. This enables controlled and confident promotion into production environments.

This architecture reflects the latest vision for AI-native systems. Reasoning, feedback and self-correction are embedded throughout the entire delivery lifecycle, as the 2026 Global AI Report underscores.

Value levers of agentic AMS

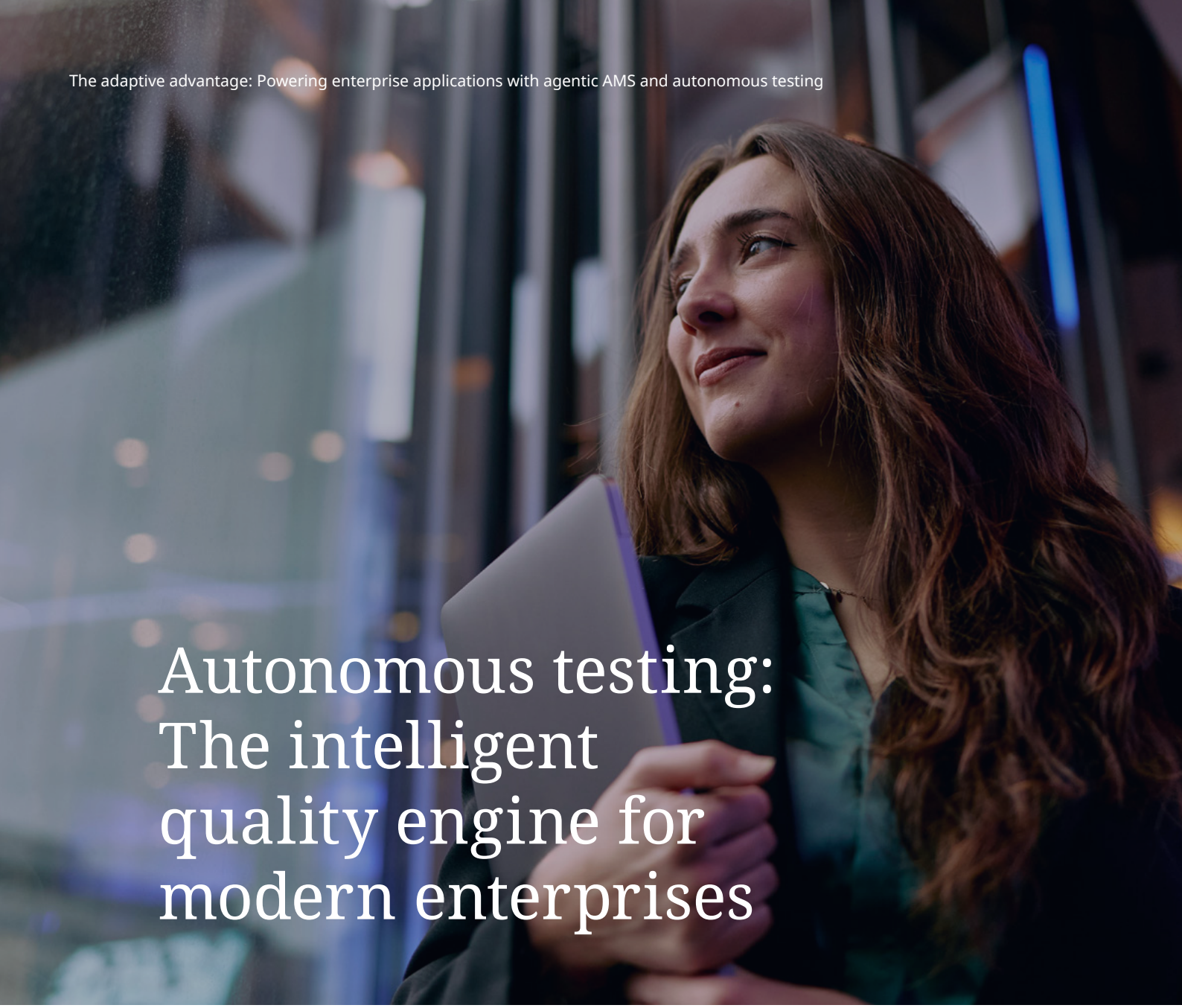
Here are the practical benefits that agentic AMS brings to the table:

- **Speed:** End-to-end cycle times see a marked reduction, often reducing weeks into a matter of hours. Agentic AMS systematically eliminates manual handoffs and common wait states.
- **Consistency:** Every change adheres to the same governed workflow. Built-in AI reasoning and automated testing at each step ensure a predictable and reliable process.
- **Scalability:** Agents operate around the clock, scaling elastically with demand. The 2026 Global AI Report finds that AI leaders wisely invest in scalable, secure platforms to accelerate growth.
- **Quality:** Automated code reviews, defect detection and regression testing improve reliability. This approach significantly reduces human error, a vital factor for sustaining high-margin operations at scale.

Adoption challenges and risk mitigation

Agentic AMS, like any significant technological advancement, introduces new considerations. The 2026 Global AI Report thoughtfully addresses many of these, including:

- **Code quality and vulnerabilities in AI-generated code:** The layered validation agents, secure platform integrations and centralized governance approaches AI leaders favor mitigate these concerns. A multi-pronged strategy builds confidence.
- **Role disruption and developer resistance:** The NTT DATA report highlights that AI leaders adopt an "expert-first" approach. They use AI to augment the capabilities of experienced employees rather than aiming for direct replacement. Positioning agents as collaborative partners is key to successful adoption.
- **Foundation model dependency:** Leaders address this risk through multi-model and hybrid AI deployment strategies. This ensures flexibility and avoids vendor lock-in while maintaining broad use.
- **Market saturation and competitive noise:** As AI adoption proliferates, differentiation often stems from effective execution at scale. AI leaders focus on end-to-end redesign, strong governance and measurable outcomes, moving beyond isolated tools or pilot programs. It's about impact, not presence.



Autonomous testing: The intelligent quality engine for modern enterprises

If agentic AMS accelerates changes, then autonomous testing ensures rapid and intelligent validation of those changes. It represents a fundamental evolution beyond traditional test automation. Instead of using manually scripted test cases and static regression suites, autonomous testing uses a set of intelligent agents that collectively manages the entire quality assurance (QA) lifecycle. These agents work with minimal configuration, continuously learning from application changes, usage patterns and historical outcomes to optimize quality validation in real time.

At the core of this approach is NTT DATA's Autonomous Test Agent (ATA) framework, which applies multi-agent orchestration to testing activities traditionally performed by QA analysts. The result is an adaptive quality engine that transforms unstructured inputs, such as raw requirements, user stories, design artifacts or application URLs, into executable, validated test assets without manual intervention.

This shift enables testing to move at the same pace as modern delivery models. As agentic AMS fast-tracks code change and modernization, autonomous testing ensures that every change is validated intelligently, continuously and at enterprise scale.

What these agents enable

Faster, higher-quality test creation: AI agents can adeptly analyze natural language artifacts, user stories, design documents or even application URLs to derive test cases. This capability removes the need for testers to write each scenario manually, significantly reducing cycle time compared to traditional test development.

Intelligent test planning: Instead of relying on static test suites, autonomous agents prioritize tests based on recent code deltas, telemetry, usage patterns and historical defect data. This aligns risk-focused quality checks with what matters most to the business.

Automated generation of synthetic and masked test data: Creating realistic, compliant test data can be a major bottleneck in QA. Autonomous agents can generate and mask data that meets specific business and privacy constraints without laborious manual effort.

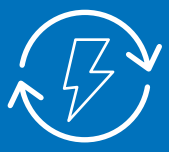
Autonomous script generation: Agents can produce executable test scripts directly from analysis for modern testing frameworks, including Selenium, Cypress, Playwright and Appium. This eliminates much of the manual scripting traditionally required for test automation.

Parallelized execution and intelligent failure analysis: Tests run concurrently across multiple environments. Agents perform root-cause analysis, adaptively retry failures and provide smart diagnostics. This feedback goes directly into continuous test optimization, all conducted autonomously.

What autonomous testing is

Autonomous testing means that purpose-built AI agents adeptly manage the entire QA lifecycle. Rather than relying on manual scripting and execution, a suite of intelligent agents performs tasks that mirror traditional QA work, but with automation and continuous learning embedded at the core. Agents:

Interpret requirements and generate test cases



Plan and prioritize based on impact and risk

Produce and manage test data

Run and analyze tests across various environments

Provide insights and feedback, delivering continuous improvement

This model effectively mirrors how human QA analysts work, but it scales autonomously. It reduces manual effort while simultaneously increasing coverage and consistency across releases.

NTT DATA's agent marketplace framework – our five agents

The ATA framework is built around five specialized, autonomous testing agents, each of which is responsible for a distinct phase of the QA lifecycle. Together, they form an end-to-end, self-coordinating pipeline that mirrors and enriches human QA workflows.

01

Requirement analysis and user story generation agent

This agent ingests raw, unstructured inputs such as emails, chat transcripts, partial specifications, screenshots or multilingual documents. From these artifacts, it generates refined user stories and expanded acceptance criteria, even when requirements are incomplete or ambiguous. This capability ensures testing can begin earlier and with greater clarity than traditional approaches.

02

Test planning agent

Rather than executing static test suites, this agent prioritizes tests dynamically. It looks at new code changes, old problems, data from sensors and how people use the application to focus on the areas that need the most attention and deliver the most value.

03

Test case design agent

This agent automatically generates detailed test cases, including objectives, preconditions, step-by-step execution flows and expected results. These test cases are often more comprehensive and consistent than their manually authored equivalents, with no manual scripting effort.

04

Test data generation agent

To address one of the most persistent QA bottlenecks, this agent autonomously generates realistic, domain-aware test data. It supports synthetic data creation and masking techniques that comply with business rules, privacy requirements and regulatory constraints.

05

Test script generation and execution agent

The final agent converts test designs directly into executable scripts for modern frameworks such as Selenium, Cypress, Playwright and Appium. Tests run in parallel across environments. The agent analyzes problems, tries different things and fixes them automatically. This information helps the system improve itself.

Together, these five agents create a fully autonomous, continuously improving testing ecosystem capable of scaling across enterprise portfolios.

Capabilities that transform quality

Each agent carefully accelerates and elevates QA activities, leading to transformative results. By orchestrating these five agents, autonomous testing delivers a set of capabilities that fundamentally reshapes enterprise quality engineering.

Faster, better-quality test creation: Tests are generated directly from natural language artifacts and application context. This eliminates manual scenario writing and significantly reduces test development cycles.

Intelligent, risk-based test planning: Autonomous prioritization ensures testing effort aligns with business risk, recent changes and real-world usage patterns rather than static regression lists.

Automated synthetic and masked test data generation: High-quality, compliant test data is produced on demand. This removes a major source of delay and manual effort in traditional QA workflows.

Autonomous script generation across frameworks: Executable test scripts are generated automatically for web, mobile and cross-platform environments. This removes the need for manual automation scripting.

Parallelized execution and intelligent failure analysis: Tests execute concurrently across environments. They have built-in root-cause analysis, smart retries and actionable diagnostics that continuously improve test effectiveness over time.

Demonstrated efficiency

Industry observations consistently indicate that autonomous testing can deliver significant advantages.

Reduced total cost of ownership (TCO): It lowers costs across planning, test data creation, script generation and execution. This strategically shifts costs from manual labor toward automated throughput, optimizing resource allocation.

Accelerated release cycles: With built-in guardrails and early defect detection, autonomous testing helps significantly speed up release cycles. This enables quicker innovation and faster time-to-market.

External studies further support this trend: Autonomous QA systems can generate thousands of test cases in minutes. They can execute complex suites significantly faster than human teams and reduce analysis effort by large margins. The result is quicker delivery combined with high confidence in the quality of the software.

Why testing is the perfect on-ramp for agentic AI

Autonomous testing sits at a crucial intersection of predictability, velocity and risk management, making it an ideal entry point for agentic AI.

High predictability, high-volume tasks: These tasks lend themselves exceptionally well to AI-driven automation. Such systems thrive on structured, repeatable processes, making testing a natural fit for AI's strengths.

Clear quality outcomes: With defined metrics like pass/fail rates, coverage and defect density, testing provides measurable ROI and robust governance. This clarity helps demonstrate tangible value quickly.

Immediate reduction in manual effort: QA professionals can focus on strategy and sophisticated risk modeling. It redirects their expertise to higher-value activities.

Essential validation of autonomous code changes: This capability is crucial for changes produced by agentic AMS. It ensures that accelerated change does not compromise trust, stability or the customer experience. Validation is the bedrock of rapid, reliable innovation.

The power of agentic AMS and autonomous testing together

Individually, agentic AMS and autonomous testing deliver meaningful gains in speed, efficiency and quality. When brought together, they form a future-state application operating model. In this model, change and validation are continuously coordinated, intelligently governed and executed at enterprise scale.

This combined model moves organizations beyond linear delivery pipelines. It points toward a closed-loop, AI-native system where applications can evolve safely, rapidly and sustainably. It is a more integrated and dynamic approach to application lifecycle management.

How the combined model works

Agentic AMS generates validated code changes: Multi-agent pipelines autonomously interpret requirements, analyze impact and implement code changes across complex application landscapes. These changes are produced consistently and align with architectural standards, security policies and governance controls. This dramatically reduces manual effort and cycle time.

Autonomous testing ensures every change is safe, compliant and high-quality: As code changes, autonomous testing agents quickly create, run and analyze tests that are tailored to those changes. This ensures functional correctness, performance stability and compliance before changes progress in the pipeline. The result is confidence in speed without sacrificing control.

Closed-loop intelligence amplifies speed, quality and governance: Feedback from testing agents continuously informs AMS agents. This refines code generation, improves test selection and helps reduce repeat defects. This closed-loop learning model enables both systems to improve over time, transforming static workflows into adaptive, self-optimizing delivery engines.

Reduces regression risk and accelerates modernization initiatives: Large-scale modernization efforts often stall due to concerns about regressions in legacy systems. By pairing autonomous code change with continuous, intelligent validation, organizations can modernize incrementally and with greater confidence. This enables updates to large codebases while maintaining stability and business continuity.

Humans provide oversight, governance and strategic direction: While agents operate at scale, human involvement remains central. Architects, engineers and QA leaders define guardrails, approve promotion thresholds, manage exceptions and focus on higher-value decisions. This “expert-first” model ensures AI augments human capability rather than replacing it.

Why this matters

The integration of agentic AMS and autonomous testing fundamentally reshapes application services:

Modernization accelerates without increasing undue risk.

Resilience improves through continuous validation and learning.

Operational costs decrease as autonomous execution replaces manual effort.

Delivery models shift from reactive to adaptive patterns.

Together, these capabilities create a scalable, autonomous application management engine. This engine keeps pace with business demand, absorbs complexity and enables organizations to continuously evolve their applications with confidence, speed and robust governance.

Agentic operating model: The future state

Agentic AMS and autonomous testing offer more than mere task automation. They fundamentally reshape how application teams operate. The outcome is an agentic operating model where humans establish direction and guardrails while intelligent agents execute at scale, with speed, consistency and continuous learning.

This model reflects a significant shift from effort-based delivery to outcome-driven, AI-native operations.

How workflows change

From manual execution to supervised autonomy

Developers shift to "approvers" and "integrators": Developers no longer spend most of their time on routine fixes, refactoring or repetitive changes. Instead, agents generate code modifications, so developers can:

- Review and approve changes
- Handle complex integrations
- Define architectural patterns and constraints

Their focus effectively moves up the value chain, from writing every line of code to shaping how systems evolve.

QA shifts to "quality supervisors": QA teams transition from manually creating and executing tests to supervising autonomous quality systems. Their role becomes one of:

- Defining quality thresholds and risk policies
- Reviewing exceptions and failure patterns
- Continuously improving test strategies

This elevates QA from execution to sophisticated governance and quality intelligence.

Product teams gain continuous engineering capability: With agents handling much of the execution, product teams can:

- Validate ideas more rapidly
- Ship smaller, safer changes with greater frequency
- Respond continuously to client feedback

Engineering transforms into an always-on capability, rather than a series of gated releases.

New key performance indicators (KPIs)

Measuring outcomes, not effort

Traditional metrics, such as "tickets closed" or "test cases executed," become less relevant in an agentic model. New KPIs pivot toward speed, quality and learning:

- **Autonomous test coverage** measures the percentage of application behavior continuously validated by autonomous testing agents.
- **Agent success rate** tracks the ratio of agent-generated changes that successfully pass validation and are promoted without requiring human rework.
- **Time-to-change** assesses the elapsed time from a request or event trigger to a validated, deployment-ready change.
- **Mean time to resolution (MTTR)** measures the speed at which agents diagnose, remediate and validate production issues.
- **Technical debt elimination velocity** quantifies the rate at which legacy issues, deprecated components and code smells are autonomously remediated over time.

These KPIs provide clear visibility into progress, enabling leaders to manage AI-driven delivery with the same rigor applied to financial or operational performance.

Human and agent collaboration model

"Human in the loop" by design

The agentic operating model is not a hands-off approach. It is human-governed by design, echoing the human-in-the-loop patterns outlined in NTT DATA's ATA framework.

Humans define intent and guardrails

Architects, developers and QA leaders establish:

- Coding standards
- Test coverage expectations
- Security and compliance policies
- Promotion and rollback rules

Agents execute within boundaries

Agents operate autonomously within these predefined constraints, generating code, tests and insights at scale while respecting enterprise controls.

Humans intervene by exception

Human attention is strategically reserved for:

- High-risk changes
- Novel scenarios
- Policy violations or ambiguous outcomes

This ensures humans focus their judgment where it is most critical.

Continuous learning loop

Feedback derived from human reviews is integrated back into agent behavior. This steadily improves accuracy and relevance. It also builds trust over time.

Implications of agentic AMS and ATA in specific industries

Manufacturing

Context and industry pressures

The manufacturing industry faces a complex landscape: increasing complexity within global supply chains, significant dependence on legacy MES/SCADA integrations, and regular, rapid maintenance of enterprise resource planning (ERP) and custom applications across geographically distributed plants. The industry is also seeing growing adoption of smart factories and Industry 4.0 architectures. These factors collectively create a demanding operational environment.

Implications of agentic AMS

- **Automates** the process of updating custom manufacturing execution applications.
- **Reduces** operational downtime due to slow patching processes or manual code changes.
- **Enables** quicker modernization of older shopfloor integrations, bringing them into contemporary operational efficiency.
- **Supports** predictive maintenance strategies through more rapid implementation of necessary system changes.

Implications of autonomous testing

- **Validates** integrations across OT and IT boundaries.
- **Ensures** quality and compliance for multisite deployments, a complex challenge in manufacturing.
- **Speeds** up version testing in environments characterized by complex hardware dependencies.

Automotive

Context and industry pressures

The automotive industry is quickly moving toward software-defined vehicles (SDVs), necessitating rapid iteration for over-the-air (OTA) updates. This shift is compounded by the development of sophisticated digital platforms that cover retail, supply chain and mobility services. These pressures dictate highly agile and reliable development processes.

Implications of agentic AMS

- **Automates** large volumes of code changes across critical applications, such as product lifecycle management (PLM), logistics, retail and service platforms.
- **Improves** code governance, which is especially vital for the safety-critical or compliance-heavy workflows prevalent in the automotive industry.
- **Supports** the continuous engineering cycles essential for integrating SDV ecosystems.

Implications of autonomous testing

- **Validates** high-volume releases across potentially hundreds of application endpoints.
- **Delivers** consistent quality assurance across global engineering teams and their suppliers, ensuring unified standards.
- **Verifies** multiplatform user experiences, encompassing dealer, consumer, vehicle and mobility applications.

Financial services

Context and industry pressures	Implications of agentic AMS	Implications of autonomous testing
Financial services operates under intense regulatory scrutiny, demanding mandatory auditability and the need to modernize legacy core systems swiftly. There's also considerable pressure to deliver digital-first experiences and accelerate product cycles, all while managing rising cyber and operational risks. It's a sector where precision and speed are paramount.	• **Acts** as a central, intelligent layer to manage complex regulatory requirements, such as Office of Foreign Assets Control (OFAC) sanctions, anti-money laundering (AML) responsibilities, data privacy and industry-specific regulations. • **Automates** the remediation of legacy codebases, reducing the operational risk inherent in older systems. • **Improves** system integrity by eliminating manual coding errors, a key to financial reliability. • **Speeds up** modernization efforts for regulatory reporting and crucial compliance updates. • **Supports** rapid product innovation by significantly shortening change cycles.	• **Guarantees** comprehensive regression coverage for business-critical financial workflows, providing essential stability. • **Improves** test traceability and compliance reporting, meeting stringent regulatory requirements. • **Supports** continuous testing in systems that handle real-time transactions. • **Reduces** dependency on fragmented manual testing teams, which consolidates efforts and improves efficiency.

Insurance

Context and industry pressures	Implications of agentic AMS	Implications of autonomous testing
The insurance industry is characterized by highly customized systems for policy, claims, underwriting and billing. Product complexity inflates testing and maintenance costs, and there's a growing customer expectation for digital-first experiences. The need to deploy new insurance products and pricing logic rapidly adds to these pressures.	• **Automates** the modification of rating engines, policy administration workflows and rule sets, all common elements in insurance systems. • **Speeds** the modernization of legacy COBOL, Java and .NET insurance systems. • **Ensures** consistency across various policy variations and state-specific rules, a complex aspect of insurance.	• **Automates** the end-to-end testing of underwriting, claims and billing journeys. • **Ensures** accuracy across thousands of potential coverage combinations, a critical aspect of policy management. • **Reduces** the risk of regulatory compliance issues that can arise from manual test gaps. • **Improves** release predictability for both carriers and third-party administrators (TPAs).

Charting a course for the intelligent enterprise

The journey through the capabilities of agentic application management services and autonomous testing reveals a compelling truth: We are on the cusp of a profound transformation in how enterprises manage and evolve their digital core. What was once a domain characterized by reactive processes and manual interventions is now swiftly moving toward a future of proactive, outcome-driven autonomy.

We've seen that agentic AI is far more than a technological upgrade; it's a strategic imperative. It solves the problems of different applications, handles skills gaps well and tackles technical debt quickly by improving knowledge and making operations easier. The distinction between generative AI as the 'creator' and agentic AI as the 'doer' underscores a shift toward systems that not only innovate but also execute, validate and optimize with remarkable precision.

The combined force of agentic AMS and autonomous testing creates a powerful, symbiotic relationship. This model constantly controls change and testing, delivering applications that are faster to market and have better quality, consistency and resilience. Across diverse industries, from the intricate supply chains of manufacturing to the tightly regulated financial services sector and the rapid innovation cycles of the automotive and insurance industries, the benefits are clear: reduced operational risk, accelerated modernization and a robust assurance of quality.

Crucially, this is not a vision of technology replacing human ingenuity. Instead, it carefully redefines roles. Developers become system architects. QA professionals supervise quality intelligence. And

product teams become agents of continuous innovation. The principle of human-in-the-loop control ensures that strategic oversight, critical judgment and ethical governance remain strong, guiding agents to operate within clear limits.

In an era where adapting quickly is not simply an advantage but a necessity, the agentic operating model provides a scalable, adaptive engine for continuous application evolution. It empowers organizations to respond to demand, absorb complexity and innovate with unwavering confidence and speed. The path ahead invites us to embrace a future where our applications are not only managed but intelligently and autonomously optimized, perpetually ready for what comes next.

Ready to explore the next steps for your organization?

Here are a few ways to continue your journey:

01 **Direct engagement:**
If the vision of a more intelligent, adaptive and autonomous enterprise resonates with your strategic objectives, we invite you to connect with NTT DATA today. Let's explore how agentic AMS and autonomous testing can be tailored to transform your application services and drive tangible value for your organization.

02 **Focus on exploration and partnership:**
The future of application management is here. It offers unprecedented opportunities for efficiency and innovation. We encourage you to reach out to our NTT DATA specialists to discuss your unique challenges and discover how a partnership can unlock the full potential of agentic AMS and autonomous testing within your enterprise.

03 **Emphasizing next steps:**
Understanding the power of agentic AMS and autonomous testing is only the beginning. We welcome the chance to learn more about your needs, assess your readiness and work together to create a practical plan to implement these new solutions. Contact NTT DATA for a personalized discussion.

For a deeper dive into practical implementation, be sure to consult our companion guide on getting started with agentic AMS and autonomous testing.

Visit nttdata.com to learn more.