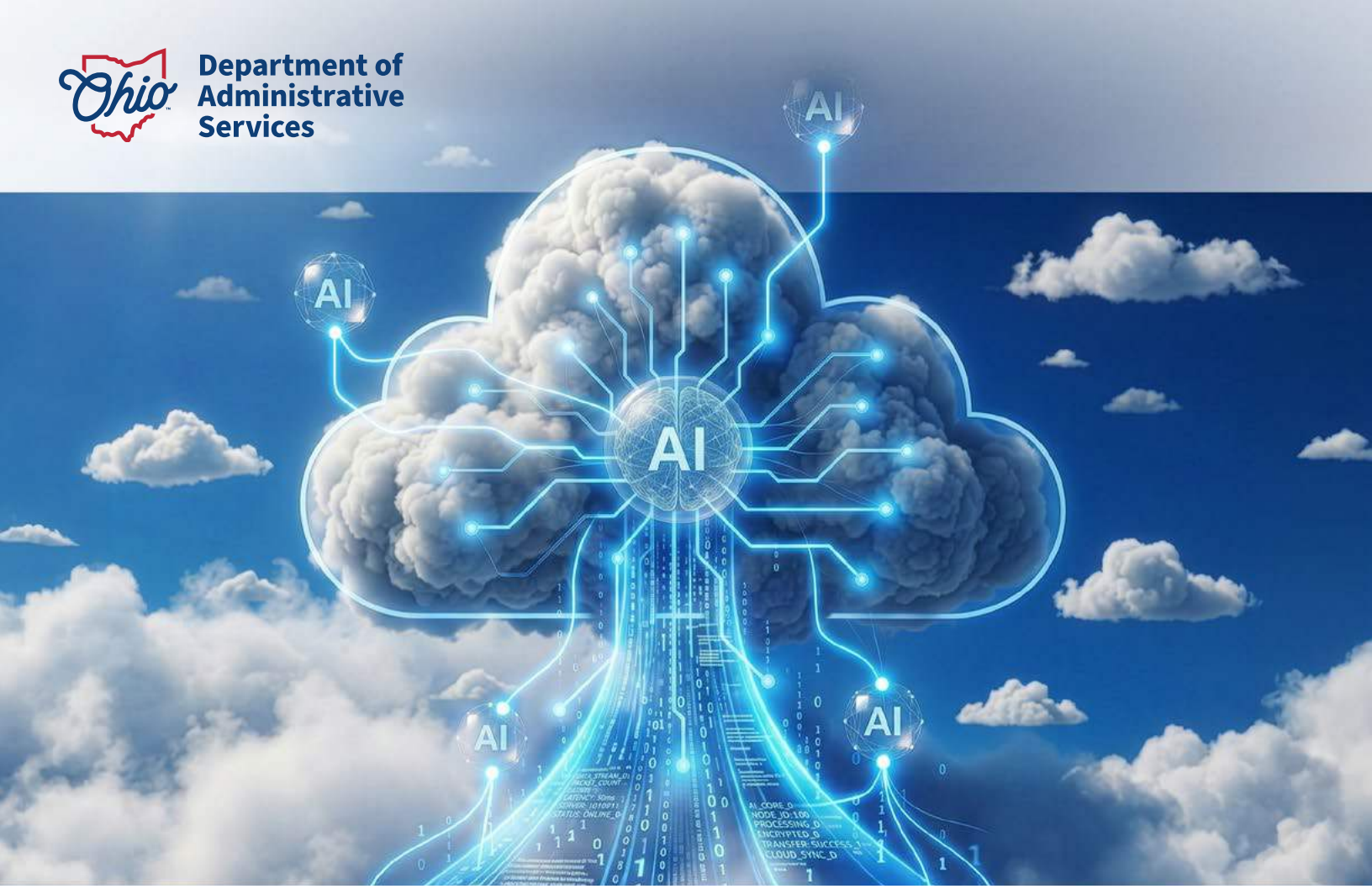
Department of Administrative Services



Driving Operational Excellence:

How AI Accelerated Statewide Cloud Modernization

Contact

Hoang Duong

Cloud Center of Excellence (CCoE) Lead
Ohio Department of Administrative Services

Hoang.Duong@DAS.Ohio.gov

Project Dates

May 2025 – April 2026

Agency

Ohio Department of Administrative Services

State

Ohio

Award Category

Operational Efficiency

EXECUTIVE SUMMARY

Legacy modernization has long been Ohio's most time-intensive government challenge — not because of what needs to be built, but because of how long it takes to understand what already exists. Agencies managing systems spanning decades, with millions of lines of undocumented code, faced a fundamental operational efficiency problem: modernization could not accelerate until system understanding did.

The Ohio Department of Administrative Services (DAS) Cloud Center of Excellence (CCoE) solved this by embedding AI-assisted techniques directly into the earliest phases of its modernization lifecycle. By utilizing AI for system interpretation, functional summarization, dependency mapping, and specification generation, the program compresses what has traditionally been the most resource-intensive phase: the period before a single line of modern code is written. This establishes a clear definition of what each system must do, turning discovery, design, and execution into a single continuous flow. Crucially, this structure ensures AI accelerates delivery without ever displacing the human judgment, oversight, and agency ownership at the center of every engagement.

Where most modernization programs apply a single AI approach across all scenarios, Ohio built two purpose-fit methodologies calibrated to fundamentally different legacy environments — and established the governance structures to deploy each appropriately. For deep legacy reconstruction — environments where systems span decades and lack documentation — the program applies a context engineering approach. For structured enterprise upgrades — environments where the challenge is consistency and scale rather than interpretation — the program applies a deterministic, agent-based execution framework. These are not variations of the same approach — they are architecturally distinct methodologies designed for different risk profiles, applied within a unified governance model.

The results are significant and measurable. To date, the effort has resulted in a 50–60% reduction in documentation and estimation time, 40% faster system analysis, and 20–40% improvement in code generation across pilot engagements. One agency is independently continuing its own modernization after the program equipped it with the skills and frameworks to do so. Another has committed shared investment to expand the model further.

Ohio is not just modernizing faster — it is building the capability to sustain that pace independently. The program is positioned to scale, with an estimated four to five modernization-ready applications per engaged agency. With a proven model and a growing pipeline, the DAS CCoE is doing more than modernizing systems - it is driving operational excellence statewide.

Ultimately, the impact of this work extends beyond Ohio. The CCoE modernization effort tackles the same technical complexities currently prioritized by IT leaders across the country. Aligning closely with the NASCIO State CIO Top Ten Priorities — specifically AI (#1), modernization (#4), and cloud services (#10) — Ohio has successfully engineered an innovative path forward that serves as a highly adaptable model for state government IT.

IDEA

WHAT PROBLEM OR OPPORTUNITY DOES THE PROJECT ADDRESS?

Ohio's modernization challenge extended beyond legacy technology. Agencies across the state operate at varying levels of cloud maturity — some adopting modern practices, others still dependent on traditional hosting. As demand for cloud capabilities grew, agencies encountered four structural barriers that made modernization slow, inconsistent, and difficult to scale.

Capacity without consistency. Agencies lacked specialized resources to evaluate and execute modernization across complex application portfolios. A centralized team alone could not meet statewide demand, while a decentralized approach produced inconsistent results.

Systems without documentation. Many legacy applications — some developed over decades — lacked documentation, with business logic embedded deep within codebases. Teams were required to reverse engineer functionality, workflows, and dependencies before modernization could begin, making early phases extraordinarily time intensive.

Knowledge without transfer. Critical system understanding was held by a limited number of subject matter experts and rarely documented. This created bottlenecks in decision-making, introduced delivery risk, and made it difficult for new team members or partner agencies to contribute to modernization efforts.

Execution without a repeatable model. Modernization efforts required teams to piece together fragmented views of systems across modules, interfaces, and technologies — aligning backend logic, user interfaces, and data structures without a unified representation of system intent. This led to duplicated effort, rework, and prolonged delivery timelines.

WHY DOES IT MATTER?

These barriers had real consequences. Without the ability to move efficiently from system understanding to execution, agencies faced prolonged modernization timelines, escalating technical debt, and continued dependence on aging infrastructure that struggled to meet modern service expectations.

The stakes were not abstract. At the Ohio Department of Job and Family Services (JFS), the primary system managing Supplement Nutrition Assistance Program (SNAP) payment appeals for Ohio residents was running on an aging foundation — where every delay in modernization meant continued risk to a system that vulnerable Ohioans rely on to challenge benefits decisions. At the Ohio Facilities Construction Commission, a capital management system spanning 25 years had accumulated technical debt so significant that teams could not begin modernization without first spending weeks reconstructing what the system did. These were not edge cases — they were representative of the challenge facing agencies across the state.

Without a more efficient and consistent approach, Ohio risked a widening gap between the pace of modernization and the pace of public need.

WHAT MAKES THIS PROGRAM DIFFERENT?

Where most modernization programs apply a single AI approach across all scenarios, Ohio built two purpose-fit methodologies calibrated to fundamentally different legacy environments.

Two methodologies, one framework: For deep legacy reconstruction — environments where systems span decades and lack documentation — the program applies a context engineering approach. AI interprets undocumented codebases, generates functional summaries, produces API specifications, and maps system dependencies, transforming a prolonged manual discovery phase into a structured, accelerated knowledge-building process. Human review is mandatory at every stage — not as a checkpoint, but as the mechanism through which institutional knowledge and system intent are validated before execution begins.

For structured enterprise upgrades — environments where the challenge is consistency and scale rather than interpretation — the program applies a deterministic, agent-based execution framework. Purpose-built tools perform standardized framework and dependency upgrades across entire codebases with repeatable, auditable outputs. AI assists execution without introducing unpredictable behavior, and human review gates are maintained for complex or higher-risk transformations. No autonomous production changes occur at any stage.

These are not variations of the same approach — they are architecturally distinct methodologies designed for different risk profiles, applied within a unified governance model.

IMPLEMENTATION

WHAT WAS THE ROADMAP?

Intent before code: Underlying the CCoE project methodologies is a shared architectural philosophy: intent must be defined and validated before execution begins. Ohio's program embeds this principle through spec-driven development — an approach where cross-functional teams establish a clear, human-reviewed definition of what a system must do and how it must behave before AI-assisted execution begins. Think of it as the difference between handing a contractor a blueprint and asking them to figure out the building as they go. This eliminates the ambiguity that drives rework, reduces dependency on undocumented knowledge, and creates a traceable path from business requirement to delivered functionality — with AI accelerating execution against a defined target rather than guessing at one.

Responsible AI governance: Ohio's innovation is as much about governance as it is about technology. The program was designed from the outset to minimize AI-associated risks: deterministic execution models reduce hallucination risk, mandatory human review gates ensure accuracy and alignment, and no autonomous changes are made to production environments. The framework operates within existing state cloud and security controls and supports enterprise AI inventory and oversight requirements. This governance model is not a constraint on innovation — it is what makes the innovation sustainable and trustworthy in a public sector context where accountability to citizens is non-negotiable.

Built to transfer: Ohio's model was designed to be replicable. The core components are transferable to any state or jurisdiction managing legacy modernization at scale: a centralized delivery capability that supports multiple agencies simultaneously, a shared investment structure that distributes cost without fragmenting execution, a documented modernization playbook covering assessment through deployment, and governance mechanisms that coordinate priorities without removing agency ownership. Any organization can begin incrementally — validating the approach on one high-complexity application before expanding — without rebuilding the model from scratch.

The most meaningful evidence of transferability is not architectural — it is behavioral. When agencies experience the model firsthand and choose to expand their own investment, they demonstrate that the approach is not dependent on a specific team, vendor, or technology stack. It is a delivery model that organizations can adopt, adapt, and sustain independently — and that Ohio has already proven works across two distinct modernization environments.

WHO WAS INVOLVED?

Addressing these barriers required more than a technical solution — it required a governance foundation built on agency trust. The CCoE operating model was co-designed with seven agency IT teams to ensure it reflected real operational needs rather than centralized assumptions. Agencies contributed directly to the CCoE's vision, mission, roles, and governance structure — giving them a meaningful stake in the program's direction from the outset.

This collaborative foundation shaped how modernization work is delivered. Cross-functional teams — combining product owners, architects, engineers, and agency technologists — align business intent with technical execution on every engagement, ensuring modernization efforts remain grounded in agency missions while enabling consistent, repeatable delivery across the state. When the program expanded to embed AI-assisted techniques into the lifecycle, this structure became even more critical — ensuring AI accelerated delivery without displacing the human judgment and agency ownership at the center of every engagement.

HOW DID WE DO IT?

The primary cost driver in legacy modernization is rarely development — it is the effort required to understand what already exists before development can begin. In traditional approaches, teams spend the majority of early engagement time reconstructing business logic, mapping dependencies, and resolving ambiguities that should have been documented years prior. This front-loaded analysis cost is largely invisible in project budgets but consistently drives timeline overruns and resource strain across agencies.

Ohio's program directly targets this cost driver. By embedding AI-assisted techniques into the earliest phases of the modernization lifecycle — system interpretation, functional summarization, dependency mapping, and specification generation — the program compresses what has traditionally been the longest and most resource-intensive phase of any modernization engagement: the period before a single line of modern code is written. At a high level, the tool delivers the following functionality:

Context engineering: AI prompts analyze legacy modules to generate functional summaries and business level documentation. AI tools reference the existing screens and workflows for validation and review of front-end code.

Automated code generation: Specialized GenAI prompts accelerate backend and front- end development by producing repository interfaces, services and controller code, UI components, integration tests, and more.

API specification generation and validation: OpenAPI specifications are generated for seamless mapping between backend APIs and UI components, ensuring traceability and validation of end-to-end integration.

Deployment and debugging automation: GenAI facilitates defect resolution including bug fixes, gap analysis, and documentation, in combination with manual reviews to ensure quality and completeness.

The program's funding progression tells its own efficiency story. Initial modernization efforts at OFCC were seeded by DAS investment to validate the approach against one of the state's most complex legacy environments. When the model proved effective, OFCC chose to continue and expand using their own agency funding.

At JFS, the program operates under a shared funding model with DAS. This structure reflects a deliberate shift from isolated, project-based spending toward shared infrastructure investment — recognizing that modernization capability, like cloud infrastructure itself, delivers greater return when treated as a shared enterprise asset rather than an agency-specific cost.

Together, these funding models demonstrate that the program generates value at two levels: it reduces the cost of individual modernization engagements by compressing analysis timelines, and it builds reusable capability that agencies are willing to fund independently.

IMPACT

WHAT DID THE PROJECT MAKE BETTER?

Ohio's AI-assisted modernization program demonstrated that the most significant efficiency gains are not achieved during development—they are achieved before it. In traditional engagements, documentation, effort estimation, and backlog creation alone can consume six to ten weeks of senior architect time. Ohio's program directly compresses this phase, redirecting effort from low-value system reconstruction to high-value design and execution. The result is a fundamental improvement in ROI that rapidly accelerates the delivery of critical citizen services. A prime example is the appeals process for SNAP payments at JFS, where delivering system reliability and timeliness directly impacts vulnerable residents challenging benefit decisions.

Two pilot engagements validated this across distinct modernization scenarios. At OFCC, the program was applied to a capital management system spanning 25 years and more than 400,000 lines of undocumented Perl code — one of the state's most complex legacy environments. What began as a single pilot module expanded to eight additional modules as the model proved effective, with OFCC now independently modernizing its remaining modules using the skills, frameworks, and approaches developed through the program. At JFS, the program was applied to HATSX — the primary system managing SNAP payment appeals for Ohio residents. Using a deterministic, agent-based execution framework, the program performed large-scale framework and dependency upgrades with repeatable, auditable outputs, bringing a critical system for Ohioans closer to a modern, maintainable foundation.

These two engagements represent opposite ends of the modernization spectrum — deep legacy reconstruction at OFCC and structured enterprise upgrade at JFS. Together, they demonstrate that Ohio's model is not environment specific. It is repeatable across fundamentally different legacy scenarios.

HOW DO YOU KNOW?

Across both engagements, the program produced measurable efficiency gains at every stage of the modernization lifecycle.

The largest improvements occurred in the early-stage activities that traditionally consume the most time and senior resource effort:

- 50–60% reduction in time required for documentation, effort estimation, and backlog creation
- 40% reduction in time required for API specification and system analysis
- 20–40% improvement in code generation and implementation throughput
- 50% reduction in time required for code reviews and debugging

These results were not incidental. They reflect a deliberate shift in how modernization work is structured — compressing the analysis phase that has historically been the largest cost and time driver in legacy modernization programs.

At OFCC, where the legacy system lacked comprehensive documentation, the program enabled teams to progress from analysis into active execution within the same engagement — a transition that in traditional approaches would have required a separate, prolonged discovery phase. At JFS, the deterministic agent-based framework performed large-scale framework and dependency upgrades consistently and repeatably across the application portfolio, reducing manual effort and upgrade risk simultaneously.

The most compelling validation of these results is not the metrics themselves — it is what agencies did next. At OFCC, what began as a single pilot module grew to nine modernized modules, with the agency now independently continuing its remaining modernization work — applying the skills, frameworks, and delivery practices developed through the program without centralized support. At JFS, the shared investment model established for HATSX reflects a deliberate organizational commitment to the approach, not a one-time engagement. These are not anecdotal signals — they are documented decisions by agency leaders who experienced the model first-hand and chose to build on it.

WHAT NOW?

The program is now positioned to scale across Ohio's agency portfolio. With an estimated four to five modernization-ready applications per engaged agency, the pipeline represents a substantial and sustained opportunity to apply the same AI-assisted model — and the same efficiency gains — across the state's legacy application inventory. The foundation is in place. The model is proven. And the agencies that have experienced it are already building on it independently.