



AI and Mainframe Modernization

What leaders need to know about COBOL-to-Java conversion

As organizations evaluate the future of long-running COBOL applications, interest in using AI, particularly large language models (LLMs), to accelerate modernization has grown significantly. AI can meaningfully support teams responsible for understanding and maintaining large, complex systems. At the same time, mission-critical mainframe and COBOL environments require more than code translation, and leaders need clarity on where AI adds real value. The goal is not to replace entire estates overnight, but to modernize in a way that protects reliability, preserves application fidelity, and positions the enterprise for long-term success.

Where AI is already delivering modernization value

GenAI is proving especially helpful in areas that improve understanding, productivity, and safety across COBOL systems:

- 01 Application understanding**
Automating the mapping of code paths, data flows, and dependencies, and rapidly documenting systems that have evolved over decades with limited historical context.
- 02 Developer productivity**
Providing natural-language Q&A, code explanations, and interactive learning support, while assisting with incremental refactoring, code cleanup, and navigation of unfamiliar modules.
- 03 Quality and safety**
Generating test scaffolding and regression tests to strengthen quality gates, and performing change-impact analysis for early identification of code-level risk.

Organizations combining AI assisted analysis with replatforming have seen developer productivity increase by 20–30% and testing cycles cut by more than 50%, demonstrating where AI reliably accelerates modernization.

These capabilities reduce onboarding time, improve confidence in changes, and enable targeted modernization work that provides a competitive edge, without altering core business logic or disrupting mission-critical operations.

Where AI falls short

Despite rapid advancements, full-scale, AI-driven conversion of COBOL-centric applications to Java remains challenging. Large enterprise environments typically include:

- Millions of lines of code across multiple languages and utilities.
- Integrated mainframe subsystems and middleware, including CICS®.
- Nonfunctional requirements such as performance, transactionality, stability, auditability, and security.
- Operational realities, including DR/HA, workload orchestration, SLAs, and compliance frameworks.
- Long-established business processes and data contracts that must be preserved precisely.

Mainframe and COBOL applications depend on more than their code. Their behavior is deeply tied to architecture, workflows, and decades of accumulated operational knowledge. Code translation and reverse engineering alone cannot reproduce these characteristics, making true end to end fidelity difficult to achieve at scale, especially given the scale of COBOL systems themselves. This is also where generic LLM based conversion introduces risk: because COBOL systems are underrepresented in AI training data and tightly coupled to business specific logic, even small hallucinations can distort critical behaviors and create costly downstream issues.

Many enterprises have already spent 5–8+ years attempting full rewrites before abandoning them due to complexity and risk, ultimately turning to replatforming as one of the most viable delivery paths.

A pragmatic approach: replatform safely and incrementally

Many organizations are adopting a phased modernization strategy with application replatforming rather than attempting full rewrites, let alone through AI. Replatforming moves applications from the mainframe to modern infrastructure—such as distributed or cloud environments—without changing the underlying business logic. This approach preserves reliability and functionality while delivering immediate benefits in cost, performance, agility, and integration flexibility. Organizations can also modernize how applications are developed, tested and operated on the mainframe itself, including with AI.

A replatforming-first approach allows teams to:

- Preserve decades of proven business logic.
- Reduce risk by avoiding semantic drift or logic breakage.
- Improve total cost-of-ownership (TCO) through infrastructure modernization.
- Introduce cloud, DevOps, and AI capabilities in a measured, low-risk way.
- Address skills challenges through AI-assisted development rather than wholesale technology shifts.
- Incrementally modernize applications to take advantage of new technologies.

This approach supports modernization speed while maintaining the stability expected of mission-critical workloads.

Organizations that have replatformed COBOL workloads in suitable scenarios routinely achieve 75–90% lower operating costs, including annual savings from \$1M to over \$25M, while preserving application fidelity.

Performance gains can be equally significant depending on workload characteristics and modernization goals: batch windows that once ran 12+ hours now complete in as little as 20 minutes, with some environments running up to 60x faster without rewriting COBOL.

Modernize on the mainframe with AI

For many organizations, modernization does not require moving applications off the mainframe. AI can also play a powerful role in extending and modernizing COBOL systems on IBM Z®, improving developer productivity, accelerating understanding, strengthening testing, and enabling API-driven integration—while preserving the platform’s unmatched reliability, security, and transaction integrity.

Whether organizations choose to modernize on the mainframe, replatform selectively, or pursue a hybrid strategy, the most successful approaches apply AI to reduce risk, preserve business logic, and deliver value incrementally.

How Rocket accelerates and de risks modernization

Rocket Software delivers modernization you can trust by combining deterministic, compiler grade analysis with responsible, explainable AI. Our technology precisely interprets every dependency, copybook, data flow, and control path—never relying on probabilistic guesses. This eliminates the operational risk common in AI only translation tools and creates a rock solid foundation for modernization that preserves the fidelity, reliability, and intent of mission critical COBOL workloads.

Building on that foundation, Rocket’s AI enhanced tooling accelerates your path forward with deep understanding, automated insights, and intelligently guided modernization options, all while keeping your business logic intact. **With Rocket Software, you get:**

- 01 Deep application and dependency analysis
- 02 AI-enabled modernization insights
- 03 Replatforming without rewriting
- 04 Cloud-ready, DevOps-enabled runtimes
- 05 Strategic modernization roadmaps
- 06 Expert partnership at every stage

One large enterprise successfully replatformed 13,000 applications with zero service disruption, demonstrating how preserving business logic provides reliability and scale that rewriting approaches cannot match.

AI is a powerful enabler when used thoughtfully. Successful modernization strategies apply AI to illuminate and accelerate the journey, without assuming entire estates can be safely converted in a single leap.

If you’d like a tailored modernization assessment, Rocket Software is here to help.

[Talk to an expert](#)



Modernization. **Without Disruption.**™

[Visit RocketSoftware.com](https://www.RocketSoftware.com)