Integration
Mobilization
Unification

From Legacy to Leading Edge

ADAPTIGENT

The End of COBOL Lock-In:

Lower Cost. Open Standards. Modern Integration.

White Paper

CONTENTS

INTRODUCTION

COBOL continues to power the backbone of global enterprise systems. Financial institutions, government agencies, logistics providers, and large enterprises rely on it to process transactions, manage records, and support operations at massive scale. In fact, **43% of banking systems are built on COBOL.** Its longevity reflects its stability, reliability, and deep integration into core business functions.

That same strength has created growing challenges. Many COBOL environments have become tightly coupled to specific compiler vendors, proprietary extensions, and legacy infrastructure. What once ensured consistency now limits flexibility, drives unpredictable costs, and complicates integration with modern platforms.

At the same time, expectations have shifted. Businesses demand real-time processing, API-driven ecosystems, and seamless cross-platform integration. Development teams expect modern tools, while customers and partners expect faster, more connected experiences.

Organizations are left in a difficult position. COBOL systems cannot simply be replaced because their business logic represents decades of investment and institutional knowledge. Rewriting them introduces significant risk, long timelines, and uncertain outcomes. As a result, forward-thinking enterprises are exploring creative solutions to modernize while balancing speed, risk, and cost.

A more practical approach is emerging. Instead of choosing between full rewrites and maintaining the status quo, organizations are modernizing COBOL in place. This approach reduces costs, eliminates vendor lock-in, and extends systems into modern architectures without discarding what already works.

This paper explores the hidden costs of traditional COBOL environments, outlines a pragmatic modernization strategy, and highlights real-world examples of organizations achieving greater flexibility and cost efficiency.

The Hidden Cost of "Status Quo COBOL"

Maintaining an existing COBOL environment often appears to be the lowest-risk option. Systems are stable, teams are familiar with the tooling, and core processes continue to function. Beneath that stability, however, cost structures continue to expand.

Licensing models tied to usage, distribution, or runtime environments introduce ongoing expenses that scale with business activity. Administrative overhead grows as organizations manage audits, compliance requirements, and vendor-defined licensing rules. These costs are rarely static and often increase over time without delivering proportional value.

Operational exposure also increases. Legacy environments were not designed for modern availability expectations, and when disruptions occur, the financial impact can be significant.

Industry estimates place the average cost of IT downtime at approximately $100,000 per hour. (ITIC, 2024)

For large enterprises in industries such as banking, government, and logistics, where transaction volumes and operational dependencies are especially high, the true cost of downtime can scale exponentially beyond these baseline estimates. What begins as a stable environment gradually becomes a cost center with limited flexibility and increasing risk.

Vendor Lock-In, Technical Debt, and Business Impact

Much of this cost and risk is tied to vendor lock-in. Many COBOL environments rely on proprietary language extensions and vendor-specific tooling introduced incrementally over years of development. This creates a dependency that affects both technology and strategy.

Portability becomes limited, making it difficult to move applications across environments without specialized effort. Integration with modern platforms often requires custom adapters or workarounds, adding layers of complexity that slow down change and reduce responsiveness to business needs.

At the same time, technical debt accumulates. Codebases become harder to maintain, and the cost of making changes increases. These constraints translate directly into business impact. Development cycles slow, releases take longer, and testing and deployment remain resource-intensive. Infrastructure costs stay high, particularly in centralized environments where scaling is tied to hardware or vendor-defined licensing models.

Organizations also face challenges connecting COBOL systems to modern digital initiatives. Whether enabling customer-facing applications, integrating with partners, or supporting internal analytics, legacy environments often require additional layers to participate in today's ecosystems.

The False Choice and a Better Path Forward

Faced with these challenges, organizations often believe they have only two options. One is to rewrite applications entirely. While this promises a clean slate, it introduces significant cost, long timelines, and substantial risk. Rewriting complex systems requires deep understanding of existing logic, and errors can disrupt critical business processes.

The alternative is to remain on existing platforms and accept the limitations. This preserves stability in the short term but allows cost and complexity to grow over time. Neither option provides a balanced path forward. A more practical approach is emerging. Organizations are modernizing COBOL in place, preserving proven business logic while removing the constraints that drive cost and limit flexibility.

Defining Modern COBOL

Open & Portable

Modern COBOL environments adhere to ANSI standards, reducing reliance on proprietary extensions and improving portability across platforms.

Integrated & Extensible

COBOL systems integrate with modern languages, frameworks, and application ecosystems, enabling broader interoperability and modernization.

Cloud & Infrastructure Ready

Support for distributed and cloud deployment models enables scalability and alignment with modern infrastructure strategies.

Where NetCOBOL Delivers Immediate Value

A key factor in modernization is the choice of compiler and runtime environment, since the right platform can remove long-standing constraints while preserving existing functionality. NetCOBOL addresses several of the most common challenges associated with legacy COBOL environments, particularly in cost, simplicity, and flexibility.

From a cost perspective, eliminating runtime fees has a direct and measurable impact. Organizations no longer need to pay for application distribution, which simplifies cost models and reduces long-term expenses. This is reinforced by a more straightforward licensing approach that reduces reliance on complex structures, lowering administrative overhead and minimizing the need for ongoing audits and compliance management.

From a technical standpoint, ANSI compliance allows organizations to move away from proprietary syntax, resulting in a cleaner and more portable codebase that is easier to maintain and extend. At the same time, NetCOBOL supports modernization without requiring a full rewrite. Existing applications can be extended into modern environments through languages such as C#, C++, and VB within the .NET framework, enabling teams to introduce new capabilities while continuing to leverage proven business logic.

Migration is also designed to minimize disruption. Automated tools support conversion while maintaining functional equivalence, which significantly reduces the risk typically associated with large-scale replatforming efforts.

FAST MIGRATION STORIES: REAL COMPANIES, REAL RESULTS

Why Real-World Proof Matters

Modernization strategies are ultimately judged by outcomes, not intent. Organizations need to understand how quickly value can be realized, how risk is managed, and how performance changes after implementation. Real-world examples provide insight into what is achievable and how similar challenges have been addressed. They also help ground decision-making in evidence rather than assumptions, making it easier to evaluate trade-offs and set realistic expectations. In environments where stability is critical, this kind of proof is often the difference between hesitation and action.

Cross-Case Insights

Across these examples, several consistent outcomes emerge. Organizations are able to preserve their existing COBOL investments while reducing costs and accelerating time to value, since systems do not need to be rebuilt from scratch. At the same time, integration improves, enabling participation in modern application ecosystems, while scalability increases as systems move to more flexible infrastructure models.

BANKING: SKANDINAVISK DATA CENTER

Skandinavisk Data Center supports banking operations for more than one hundred financial institutions and needed to scale its infrastructure while maintaining reliability and controlling costs. To achieve this, the organization transitioned from an IBM mainframe environment to a Microsoft-based platform using NetCOBOL, allowing it to retain its existing COBOL applications while modernizing the underlying infrastructure.

The results were substantial. **Annual operating costs were reduced by $16 million, while system availability reached 99.8%** to support high transaction volumes across its client base. **More than 11.2 million lines of COBOL code were preserved and successfully migrated** to a modern platform, demonstrating that even large-scale environments can be modernized without sacrificing stability or requiring a full rewrite.

DISTRIBUTION: SIMON & SCHUSTER

Operating in a high-demand distribution environment, Simon and Schuster required improvements in processing speed, system responsiveness, and overall agility. The organization also needed to integrate its systems more effectively without disrupting existing applications. By implementing NetCOBOL alongside Adaptigent's platform, it extended its COBOL systems into a more flexible and connected architecture.

This approach delivered immediate impact. **Business agility improved by 300%, IT trouble tickets were reduced by 75%, and batch processing times were significantly shortened.** As a result, order fulfillment became faster and more responsive, highlighting how modernization can drive both operational efficiency and customer-facing performance.

GOVERNMENT: WASHINGTON STATE LICENSING

The Washington State Department of Licensing manages critical public services supported by a large COBOL codebase and required a modernization approach that minimized risk while avoiding the disruption of a full system replacement. By adopting NetCOBOL, the agency was able to transition to a modern platform while preserving its existing applications and maintaining continuity of service.

This approach resulted in more than $1 million in annual savings, along with improved system integration and the elimination of a large-scale code conversion effort. The case demonstrates that even highly regulated environments can modernize effectively without introducing unnecessary risk or complexity.

OPEN, MODERN COBOL: APIS, INTEGRATION, AND FLEXIBILITY

What Open COBOL Means in Practice

Open COBOL environments are built on standards that enable interoperability across platforms and tools, reducing dependency on specific vendors and allowing organizations to evolve their architecture over time. This standardization not only improves portability but also simplifies maintenance and makes it easier to onboard developers who are familiar with modern development practices.

API-Driven Enablement

APIs have become the primary mechanism for connecting systems and enabling digital services, with **more than 83% of web traffic is now API-driven.**

By exposing COBOL logic as APIs, organizations can extend the value of existing systems into new channels, supporting customer applications, partner integrations, and internal services without requiring fundamental changes to core logic.

Integration with Modern Development Environments

Modern development teams expect tools that support collaboration, debugging, and rapid iteration, and COBOL environments must align with these expectations to remain effective.

NetCOBOL integrates with widely used platforms such as Visual Studio and Eclipse, while also supporting languages such as C#, C++, and VB. This allows cross-functional teams to work more seamlessly and reduces friction between legacy systems and modern development workflows.

Flexible Deployment Options

Modern COBOL applications are no longer limited to centralized mainframe environments. They can be deployed across Windows and Linux systems, as well as cloud platforms such as Azure, enabling distributed architectures that scale based on demand.

This flexibility also supports hybrid environments, where legacy and modern systems coexist within a unified and more adaptable architecture.

Performance, Reliability, and Future Readiness

Modern COBOL platforms deliver reliable performance and stability while reducing operational complexity.

Support for graphical interfaces, web access, and databases like SQL Server and Oracle enables modernization without disrupting core operations.

CONCLUSION: A NEW MODEL FOR COBOL MODERNIZATION

A Shift in Approach

COBOL remains deeply embedded in the systems that run modern enterprises, and for most organizations, replacing it is neither practical nor necessary. The challenge is not the language itself, but the constraints introduced by the environments in which it operates. Vendor lock-in, rising licensing costs, and limited integration capabilities have made it increasingly difficult to adapt these systems to current business demands.

As expectations shift toward real-time processing, API-driven architectures, and greater operational agility, organizations are reevaluating how their COBOL environments fit into a modern technology strategy. The shift underway is not about abandoning COBOL, but about removing the limitations that prevent it from evolving alongside the rest of the enterprise.

Modernization Without Disruption

The experiences outlined in this white paper demonstrate that modernization does not require a full rewrite or disruptive transformation. Organizations can reduce cost, improve performance, and increase flexibility by modernizing COBOL in place. This approach preserves decades of business logic while addressing the structural issues that drive complexity and risk.

By adopting standards-based platforms, organizations can reduce dependency on proprietary extensions and improve portability across environments. At the same time, enabling API access and integration with modern development tools allows COBOL systems to participate fully in broader application ecosystems. These changes make it possible to extend the value of existing systems rather than replace them.

Technologies such as NetCOBOL support this model by simplifying licensing, eliminating runtime fees, and providing compatibility with modern infrastructure and development frameworks. As seen in the case studies, this approach can deliver measurable results, including significant cost savings, improved system performance, and faster time to value.

References

[1] Reuters. COBOL BLUES. 2024. Accessed 2026.

[2] ITIC. ITIC 2024 Hourly Cost of Downtime Part 2. 2024. Accessed 2026.

[3] Akamai. State of the Internet Security Report. 2019. Accessed 2026.

Adaptigent is a software technology company offering solutions to help businesses harness the power of APIs for innovation and growth. A global distributor of the Fujitsu NetCOBOL compiler, Adaptigent seamlessly integrates core systems. More than 2,500 organizations globally trust Adaptigent solutions. Visit www.adaptigent.com.

The information contained in this document represents the current view of Adaptigent on the issues discussed as of the date of publication. Because Adaptigent must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Adaptigent, and Adaptigent cannot guarantee the accuracy of any information presented after the date of publication. This white paper is for informational purposes only. Adaptigent makes no warranties, express or implied in the document.