10.48047/jocaaa.2026.35.01.81

Modernizing Complex Enterprise Landscapes: A Cross-Industry Framework for Legacy System Transformation

Shiva Kumar Bhuram
Dorman Products, Inc., USA
Suresh Dameruppula
MYGO Consulting Inc., USA

Received: 31.01.2026 Revised: 02.02.2026

Abstract

Organizations across manufacturing, logistics, retail, and healthcare sectors continue operating on legacy enterprise systems that, while reliable, impose significant constraints on scalability, interoperability, and adoption of modern technologies, including cloud platforms, microservices, and artificial intelligence. These aging architectures accumulate technical debt through decades of customizations and tightly coupled integrations, creating barriers to digital transformation. A structured six-layer modernization framework addresses these challenges through baseline assessment, target architecture definition, pathway selection, clean-core extensibility, AI-driven optimization, and governance mechanisms. The framework enables incremental evolution while preserving operational continuity, reducing integration fragility, and establishing foundations for cloud migration. A manufacturing case demonstrates measurable improvements in inventory accuracy, integration reliability, and warehouse efficiency while significantly reducing technical debt. By integrating established architectural principles with generative AI capabilities, the framework provides enterprises with a systematic model for transforming legacy landscapes into adaptive, extensible digital platforms capable of supporting real-time analytics, event-driven processing, and intelligent automation without disrupting mission-critical operations.

Keywords: Legacy Modernization, Enterprise Architecture, Clean-Core Extensibility, AI-Driven Optimization, Digital Transformation

1. Introduction

Enterprise systems such as Enterprise Resource Planning (ERP), Warehouse Management Systems (WMS), Transportation Management Systems (TMS), Customer Relationship Management (CRM), Electronic Medical Records (EMR), and Manufacturing Execution Systems (MES) form the operational core of global organizations, supporting essential functions including financial reconciliation, manufacturing execution, order fulfillment, and regulatory reporting [6]. Many of these systems were deployed decades ago and have since accumulated extensive customizations, deeply embedding them within business processes and supply chain workflows.

However, the architectural foundations of these systems were designed for an era dominated by centralized computing, monolithic software design, and batch-driven integrations. These assumptions conflict with the demands of modern digital ecosystems that require real-time data processing, scalable integration, cloud-native agility, and AI-enabled decision support [1], [7]. As organizations pursue digital transformation, legacy systems increasingly struggle under the weight of technical debt, rigid integrations, limited observability, and outdated middleware.

10.48047/jocaaa.2026.35.01.81

Integration complexity remains a major obstacle. Traditional landscapes rely heavily on point-to-point interfaces, file-based exchanges, and proprietary protocols that limit modularity and create brittle dependencies [2], [8]. These limitations intensify as enterprises integrate cloud services, supply chain visibility platforms, IoT devices, and advanced analytics systems. Additionally, legacy environments often lack end-to-end visibility. Dependency maps, data lineage, and architectural documentation are frequently incomplete or outdated, increasing the risk of unintended impacts when making changes. Limited automation in testing, configuration, and deployment further slows modernization initiatives and raises operational risk [9].

Despite these challenges, legacy systems cannot simply be replaced. They embed domain knowledge, regulatory logic, and decades of operational refinement that are difficult to recreate without substantial risk [10]. Full replacement introduces significant cost, uncertainty, and transformation risk. Therefore, enterprises increasingly adopt structured modernization frameworks that combine incremental evolution with architectural safeguards.

The modernization framework presented in this paper synthesizes practical engineering patterns, industry modernization models, and AI-enhanced acceleration techniques. It provides a systematic, multi-layer methodology supported by diagrams, tables, and a comprehensive manufacturing case study. The goal is to help organizations modernize complex enterprise landscapes in a controlled, scalable, and academically grounded manner.

2. Research Motivation and Problem Context

Modern enterprises must operate in an increasingly dynamic environment characterized by real-time data flows, cloud-native ecosystems, automation, and AI-driven decision support. However, many organizations continue to rely on legacy systems that were architected for earlier computing paradigms. Modernization is therefore driven by a need to reduce technical debt, increase interoperability, and enable scalable digital capabilities [1], [6].

2.1 Persistence of Legacy Systems

Legacy systems persist because they encode decades of domain-specific logic, regulatory compliance rules, and mission-critical workflows that are difficult to recreate without substantial risk [2]. Global surveys consistently show that core ERP and mainframe systems remain operational for more than 20–30 years in most enterprises due to stability, reliability, and the high cost of replacement [7]. Although robust, these systems face challenges:

- Outdated middleware and limited API support
- High integration fragility
- Complex, layered customizations
- Limited observability and documentation

These factors create rigidity that prevents organizations from adopting modern architecture patterns or integrating emerging technologies.

2.2 Cross-Industry Modernization Complexity

Although modernization challenges vary across industries, they share similar structural constraints grounded in legacy architecture, integration complexity, and data fragmentation. Table 1 summarizes representative challenges across major sectors.

Table 1 compares modernization challenges across industries, showing that common architectural choices, rather than industry-specific factors, drive long-term integration and modernization issues.

10.48047/jocaaa.2026.35.01.81

Industry	Primary System Landscape	Core Modernization Challenges	Business Impact
Manufacturing	ERP, MES, WMS, SCM	Production system integration constraints, supply chain visibility limitations, and real-time shop floor connectivity gaps	Delayed production cycles, inventory inaccuracies, supply chain disruptions
Healthcare	EMR, EHR, PACS, LIS	Clinical system interoperability barriers, patient data fragmentation, and regulatory documentation complexity	Patient safety risks, care coordination delays, compliance vulnerabilities
Retail	POS, E-commerce, CRM, OMS	Customer engagement platform disconnects, inventory management system fragmentation, and omnichannel integration gaps	Customer experience deterioration, stock-out incidents, and revenue leakage
Logistics	TMS, WMS, YMS, Route Planning	Real-time tracking capability limitations, carrier integration obstacles, and load optimization system constraints	Delivery delays, operational inefficiencies, and customer dissatisfaction

Table 1: Cross-Industry Modernization Challenges [3][4]

2.3 Gaps in Traditional Modernization Initiatives

Historical modernization strategies often emphasized platform migration or system replacement without addressing deeper architectural dependencies. Studies have shown that many modernization efforts underestimate integration complexity, undocumented workflows, and the volume of custom code embedded within legacy systems [9].

Key gaps include:

- Over-reliance on rip-and-replace approaches that disrupt operations
- Neglect of integration modernization, leaving brittle dependencies in place
- Limited analysis of data lineage and quality, leading to migration failures
- Underutilization of AI techniques for code analysis, workflow discovery, and test automation
- Weak governance models, resulting in inconsistent architecture decisions

Addressing these gaps requires a structured framework that integrates architecture-driven modernization, clean-core extensibility, and AI-supported acceleration. Such a framework ensures modernization activities are incremental, risk-controlled, and aligned with long-term enterprise strategy [10].

3. Methodology – A Multi-Layer Generative AI–Enhanced Modernization Framework

Modernizing complex enterprise landscapes requires a structured, incremental methodology that balances architectural evolution with operational continuity. The proposed framework consists of six

10.48047/jocaaa.2026.35.01.81

interconnected layers that collectively guide organizations through assessment, design, modernization, optimization, and governance. Each layer builds upon established enterprise architecture practices while incorporating advances in cloud-native design and AI-enabled engineering [3], [7].

3.1 Layer 1: Enterprise Baseline Assessment

Baseline assessment provides a comprehensive understanding of the current-state architecture, integrations, data flows, and technical debt. This step reveals modernization hotspots and dependency risks [5]. The assessment process encompasses systematic examination of existing system configurations, integration mechanisms, custom code repositories, and infrastructure dependencies. Organizations conduct comprehensive architectural audits that identify integration inventories, code complexity metrics, data lineage visualizations, and infrastructure dependency maps. This process creates foundational artifacts that illuminate system interdependencies, performance bottlenecks, and technical constraints embedded within operational environments. Thorough assessment enables organizations to prioritize modernization initiatives based on risk exposure, business criticality, and technical feasibility [5]. Without comprehensive baseline understanding, modernization efforts risk overlooking critical dependencies that could destabilize operational stability during transformation activities.

Table 2 outlines the baseline assessment across architecture, integration, code, processes, and infrastructure, ensuring a complete understanding of the current environment before modernization begins.

Assessment Dimension	Activities	Deliverables	Tools and Techniques
System Architecture	Platform inventory, version cataloging, technology stack mapping	Current-state architecture diagrams, technology inventory matrix	Architecture discovery tools, configuration management databases
Integration Landscape	Interface mapping, protocol analysis, data flow tracing	Integration catalog, dependency graphs, data lineage models	Integration monitoring tools, API discovery platforms
Custom Code Analysis	Complexity metrics, redundancy detection, and obsolete code identification	Code quality reports, refactoring recommendations, and technical debt inventory	Static code analyzers, AI-assisted code review tools
Process Mining	Transaction log analysis, workflow discovery, and exception pattern identification	Process flow diagrams, bottleneck analysis, optimization opportunities	Process mining software, workflow analytics platforms
Infrastructure Dependencies	Server mapping, network topology, storage allocation	Infrastructure dependency maps, capacity utilization reports	Infrastructure monitoring tools, cloud readiness assessments

Table 2: Layer-Specific Assessment Activities and Deliverables [5][9]

3.2 Layer 2: Target Architecture Blueprint

10.48047/jocaaa.2026.35.01.81

The target architecture defines a modular, scalable, and extensible future-state environment that supports multi-channel integrations, real-time analytics, and AI-enabled automation [5]. The blueprint establishes architectural principles that advance interoperability, observability, and long-term adaptability by separating capabilities into composable, service-oriented domains. Future-state architecture incorporates cloud-native design patterns, event-driven communication mechanisms, and API-led connectivity structures, enabling flexible integration across heterogeneous systems. The blueprint addresses multiple architectural dimensions encompassing application structure, data management, infrastructure allocation, security protocols, and integration frameworks. Organizations construct reference architectures that balance standardization with flexibility, enabling consistent implementation while accommodating domain-specific requirements. Target architecture blueprints deliver explicit guidance for transformation activities, ensuring alignment between tactical implementation decisions and strategic architectural objectives [5].

3.3 Layer 3: Modernization Pathway Selection

Modernization pathways help determine how each system component should evolve based on business criticality, technical constraints, and modernization readiness. Organizations categorize components using distinct pathways encompassing replatform for infrastructure migration, refactor for decomposing monolithic components, reengineer for process and architecture redesign, replacement for implementing new platforms or cloud-based solutions, and optimization for improving stable systems with limited modification. Pathway selection logic ensures modernization decisions align with risk tolerance and strategic priorities. Components undergo assessment across multiple dimensions encompassing technical condition, business value, replacement availability, integration complexity, and modernization urgency. This structured approach prevents arbitrary technology decisions and ensures that modernization investments address genuine organizational requirements rather than pursuing technology trends without strategic validation.

3.4 Layer 4: Clean-Core Extensibility and Integration Model

Clean-core extensibility minimizes customizations within legacy applications while enabling innovation through external services and composable extensions. This approach reduces upgrade friction, simplifies lifecycle management, and supports continuous innovation [5]. Organizations separate core system functionality from extension logic, maintaining standardized core systems while implementing business-specific requirements through side-by-side services, API extensions, and event-driven integrations. Clean-core approaches preserve vendor-supported functionality within core systems, reducing technical debt accumulation and maintaining upgrade compatibility. Extension architectures utilize modern integration technologies encompassing API gateways, service meshes, and event streaming platforms, creating flexible integration layers that connect core systems with innovative capabilities. This architectural separation enables organizations to implement new technologies and business capabilities without modifying core system configurations [5].

3.5 Layer 5: Generative AI–Driven Optimization

Generative AI enhances modernization efficiency by automating complex engineering activities encompassing code analysis and refactoring recommendations, documentation reconstruction, dependency visualization, test case generation, data quality pattern identification, and integration anomaly prediction [6]. Recent advancements in AI-assisted engineering demonstrate significant reductions in modernization cycle times and manual effort when applied to large codebases and integration environments. AI capabilities accelerate understanding of complex legacy systems by automatically extracting architectural patterns, identifying redundant code segments, and generating comprehensive

10.48047/jocaaa.2026.35.01.81

documentation from existing implementations. AI-driven tools support impact analysis by predicting downstream effects of proposed modifications across interconnected systems. Machine learning models trained on historical system behavior identify potential failure patterns and suggest preventive modifications. These capabilities particularly benefit organizations lacking comprehensive documentation or deep expertise in aging technologies [6].

3.6 Layer 6: Governance and Value Realization

Governance ensures that modernization adheres to architectural standards, reduces risk, and remains aligned with enterprise strategy. Effective governance frameworks establish decision authority structures encompassing Architecture Review Boards ensuring alignment with architectural principles, Steering Committees overseeing modernization roadmaps, Security Councils ensuring compliance and risk controls, and Project Management Offices controlling execution discipline [5]. Governance mechanisms deliver consistent oversight across modernization initiatives, preventing fragmented decision-making and architectural inconsistency. Regular governance checkpoints validate that implementation activities conform to established architectural patterns, security requirements, and integration standards. Value realization frameworks track modernization outcomes against defined objectives, measuring improvements in system performance, integration reliability, operational efficiency, and business capability enablement. Governance structures ensure modernization efforts remain coordinated, controlled, and measurable across all transformation phases [5].

Table 3 outlines the comprehensive governance structure required to ensure successful modernization initiatives remain aligned with enterprise strategy and architectural standards. This multi-tiered governance structure creates clear decision authority hierarchies, prevents fragmented modernization efforts, and ensures consistent architectural decisions across distributed initiatives throughout the transformation journey.

Governance Body	Composition	Decision Authority	Meeting Frequency	Key Responsibilities
Architecture Review Board	Enterprise architects, solution architects, technical leads	Approve/reject architectural designs, standards compliance validation	Weekly during active modernization	Ensures alignment with architectural principles, validates integration patterns, and reviews technology selections
Steering Committee	Executive sponsors, business leaders, and IT leadership	Strategic direction, priority setting, resource allocation	Monthly	Oversees modernization roadmap, resolves business-IT conflicts, approves major investments

10.48047/jocaaa.2026.35.01.81

Security Council	Chief information security officer, security architects, compliance officers	Security approvals, risk acceptance, compliance validation	Bi-weekly	Evaluates API access controls, ensures data protection compliance, and manages security vulnerabilities
Project Management Office	Program managers, project managers, business analysts	Execution coordination, progress tracking, and risk management	Weekly	Controls project methodology adherence, monitors milestone achievement, and manages resource allocation
Change Advisory Board	Operations managers, release managers, business representatives	Change approvals, deployment scheduling, and rollback decisions	As needed	Validates deployment readiness, approves production changes, and coordinates go-live activities

Table 3: Governance Framework Structure [5][8]

4. Technical Innovations and Key Contributions

The framework introduces modular approaches for transforming legacy integration mechanisms through API-led connectivity, event-driven architectures, standardized data models, and lightweight orchestration capabilities. These integration patterns advance cross-system interoperability and enable seamless interaction between legacy systems, cloud-native applications, analytics platforms, and automation technologies. This represents a significant improvement over batch-oriented, file-transfer mechanisms prevalent in legacy environments, reducing failure rates and accelerating responsiveness.

4.1 Modern Integration Patterns

Modern integration patterns emphasize modularity, flexibility, and real-time communication capabilities that address constraints inherent in traditional integration approaches [7]. Organizations deploy API-led connectivity, establishing clear separation between system interfaces, process orchestration, and interaction layers. This architectural layering enables independent evolution of integration components without disrupting dependent systems. Event-driven architectures enable asynchronous communication patterns where systems publish state changes to event streams rather than executing synchronous point-to-point calls. This approach decouples system dependencies, improves resilience, and allows multiple consumers to react to the same events without additional integration effort. Standardized data models establish uniform data structures across heterogeneous systems, reducing transformation complexity and advancing data consistency throughout enterprise environments. Lightweight orchestration mechanisms manage complex workflows spanning multiple systems while maintaining loose coupling between participating components. These integration patterns collectively advance interoperability between legacy infrastructure and modern cloud-native applications, analytics platforms, and automation technologies [7]. Modern integration structures demonstrate substantial improvements compared to batch-oriented, file-

10.48047/jocaaa.2026.35.01.81

exchange mechanisms prevalent in legacy landscapes, reducing failure frequencies and accelerating response capabilities.

4.2 AI-Enhanced System Modernization

A critical innovation involves incorporating generative AI accelerators throughout modernization operations. Recent developments demonstrate that artificial intelligence automates code interpretation, remediation, documentation generation, dependency visualization, and test construction across extensive legacy landscapes [8]. This substantially reduces modernization timelines and minimizes human error. AI platforms examine existing code repositories, identifying redundant logic, obsolete patterns, and improvement opportunities that would require extensive manual effort to uncover. Natural language processing capabilities extract business logic from code artifacts, generating human-comprehensible documentation that clarifies system behavior and decision logic. Machine learning algorithms trained on architectural patterns recognize structural similarities across different implementations, proposing consistent refactoring approaches. Automated test generation tools examine existing system behavior, constructing comprehensive test suites that confirm functionality before and after modernization operations. These capabilities demonstrate particular value when organizations lack detailed documentation or expertise in obsolete technologies. By embedding artificial intelligence across multiple framework layers, modernization accelerates, achieves greater precision, and reduces dependence on scarce legacy-system expertise [8]. This advances modernization maturity and supports sustainable transformation at scale.

4.3 Governance-Integrated Modernization

The framework emphasizes governance as a fundamental component ensuring modernization initiatives remain aligned with long-term enterprise objectives [8]. Governance mechanisms encompassing architecture review boards, modernization steering committees, and standardized decision checkpoints prevent fragmented modernization and reduce architectural drift risk. Architecture Review Boards confirm that proposed solutions conform to established architectural principles, security requirements, and integration standards before implementation authorization. Steering Committees provide strategic direction, prioritize competing initiatives, and resolve conflicts between business unit requirements and enterprise-wide standards. Security Councils assess risk exposure introduced through modernization operations, ensuring compliance with regulatory requirements and organizational security protocols. Project Management Offices establish execution discipline through standardized project methodologies, progress monitoring mechanisms, and resource allocation frameworks. These governance structures create decision authority hierarchies, clarifying responsibility boundaries and escalation paths throughout modernization programs. Regular governance checkpoints occurring at defined modernization milestones confirm continued alignment between implementation operations and strategic objectives [8]. Effective governance ensures modernization investments deliver anticipated value while preserving operational stability and architectural consistency across distributed initiatives.

4.4 Summary of Contributions

The fundamental contributions of the framework encompass a structured, multi-layer modernization methodology providing systematic guidance through complex enterprise transformation operations [7]. Clean-core extensibility blueprints establish architectural patterns that separate core system functionality from innovation logic, enabling continuous capability advancement without destabilizing foundational systems. Modern integration patterns appropriate for hybrid and cloud landscapes enable seamless communication between legacy infrastructure and modern digital services. AI-augmented engineering mechanisms accelerate modernization operations by automating labor-intensive tasks encompassing code

10.48047/jocaaa.2026.35.01.81

examination, documentation reconstruction, and test generation [8]. Governance-driven execution discipline ensures modernization initiatives remain coordinated across organizational boundaries, preventing duplicative investments and architectural fragmentation. Collectively, these contributions constitute scalable, repeatable frameworks for transforming complex enterprise ecosystems while preserving operational continuity and strategic alignment throughout extended transformation journeys.

5. Implementation and Case Studies

Note: The case study presented in this section is an illustrative composite developed for the purposes of demonstrating the application of the proposed modernization framework. It synthesizes common patterns observed in ERP and WMS modernization programs and reflects outcomes frequently reported in modernization research and industry transformation studies. The scenario does not describe a specific organization, nor does it rely on proprietary or confidential data.

5.1 Background

A global manufacturing enterprise operated a legacy ERP system tightly integrated with a heavily customized Warehouse Management System (WMS). Over two decades, the systems accumulated nearly a thousand custom objects, more than seventy point-to-point integrations, and numerous undocumented workflows [9]. These constraints caused recurring operational issues, high integration maintenance costs, and complexity that hindered digital transformation efforts. The organization needed to modernize without disrupting warehouse operations, which supported continuous production and multi-site distribution networks. The manufacturing environment comprised distributed facilities requiring uniform operational processes while accommodating regional variations in regulatory requirements and business practices. Existing technology foundations incorporated successive generations of hardware and software systems deployed progressively across extended periods, creating heterogeneous landscapes with inconsistent architectural patterns. Historical customization decisions executed to address immediate operational needs generated entangled dependencies complicating subsequent modification attempts [9]. The enterprise recognized that continued reliance on aging foundations would ultimately restrict competitive capabilities and obstruct implementation of emerging technologies vital for future operational distinction.

10.48047/jocaaa.2026.35.01.81

Figure 1 links technical modernization metrics to business outcomes, showing how improvements in reliability, observability, and debt reduction drive agility, resilience, and long-term strategic value.

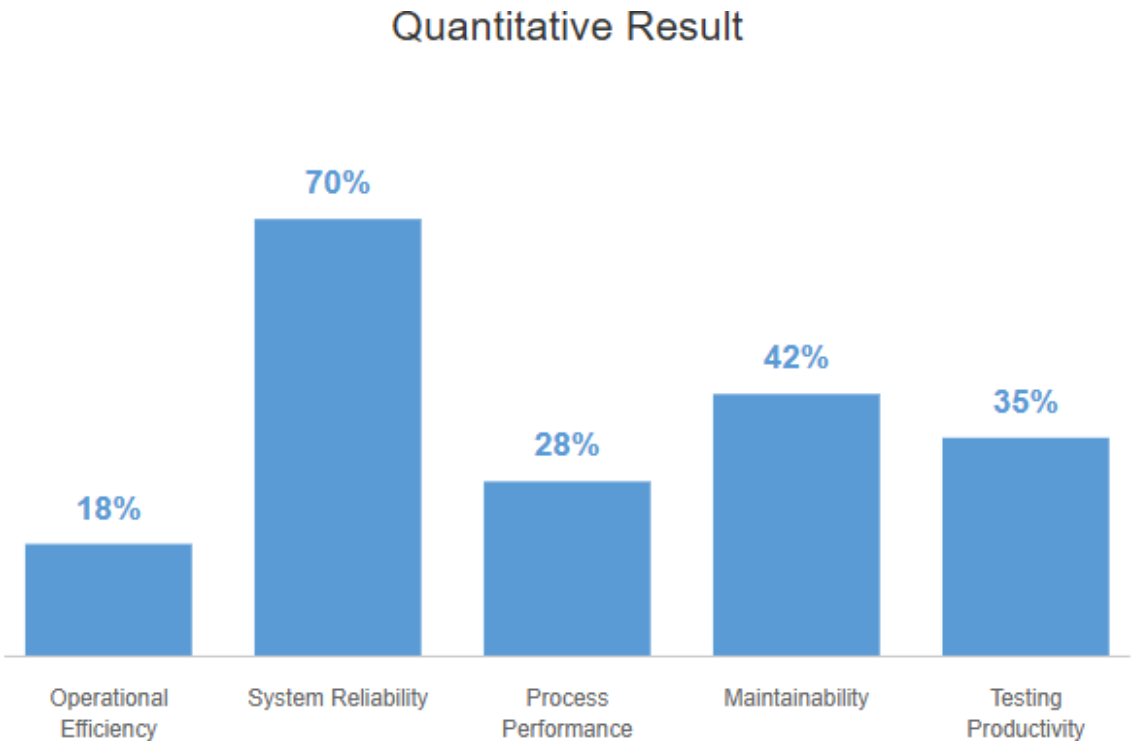

Fig. 1: Strategic Impact Metrics Summary [9][10]

5.2 Challenges Identified

The baseline assessment identified systemic challenges consistent with those documented in enterprise modernization research. Organizations confront inconsistent inventory accuracy precipitating frequent reconciliation operations, high warehouse task execution time affecting labor productivity, unreliable batch integrations between ERP and WMS infrastructure, difficulty incorporating automation equipment encompassing conveyor systems and handheld scanning devices, and customizations blocking ERP upgrades generating extended maintenance cycles [10]. These issues made incremental modernization difficult without a structured, layer-based approach. Data quality issues manifested through discrepancies between physical inventory and system records, necessitating manual verification processes consuming substantial operational resources. Integration failures materialized unpredictably, disrupting warehouse operations and requiring emergency interventions to restore operational continuity. Performance degradation during peak operational periods indicated scalability constraints within existing architectural configurations. Limited observability prevented rapid root cause identification when system disruptions materialized, prolonging resolution times and magnifying business consequences. Documentation deficiencies meant subject matter expertise resided primarily with individual personnel rather than being systematically captured, creating knowledge concentration vulnerabilities [10]. Absence of automated testing structures elevated modification risk and prolonged deployment cycles for minor enhancements.

10.48047/jocaaa.2026.35.01.81

5.3 Application of the Modernization Framework

The assessment phase produced detailed architectural documentation encompassing mapped integrations including file-based exchanges, custom RFC calls, and middleware workflows, created integration dependency graphs highlighting fragile synchronous flows, conducted custom code analysis revealing redundancy in picking, put-away, and replenishment logic, applied process mining techniques identifying high-frequency exception loops in warehouse operations, and constructed data lineage models for materials, batches, and warehouse documents [9]. This aligned with baseline assessment techniques recommended in modernization literature.

The target architecture prioritized modularity, scalability, and real-time connectivity through designed API-led integration between ERP and WMS, introduced event-driven backbone triggering warehouse tasks, decomposed WMS logic into modular microservices encompassing cartonization, wave planning, replenishment, and exception handling, and deployed observability components encompassing event logs and traceability dashboards. This blueprint reflected modern architectural principles recommended for distributed enterprise systems.

Components underwent classification using defined pathways where ERP experienced retain and refactor due to stable processes and regulatory dependencies making full replacement risky. WMS custom modules underwent reengineering due to high complexity and performance issues requiring modular redesign, and legacy middleware confronted replacement through modern API gateway and lightweight orchestration services [10].

A clean-core approach minimized ERP customizations through relocating warehouse business logic external to ERP, constructing side-by-side extensions for cartonization, wave planning, and real-time validation, reducing ERP custom code substantially, and ensuring ERP remained a stable system of record while innovation occurred externally.

AI tools accelerated the modernization initiative through AI-assisted code review identifying unused or redundant programs, AI-generated test scripts reducing testing effort, predictive analytics identifying integration failure patterns preventing warehouse disruptions, and AI-based dependency extraction accelerating documentation reconstruction for long-untouched custom modules.

Governance mechanisms ensured controlled execution through weekly Architecture Review Boards validating design patterns and integration models, Security Council evaluating API access, data protection, and compliance, dedicated PMO coordinating phased rollouts avoiding warehouse disruptions, and Steering Committee providing decision authority for prioritization and risk mitigation [9].

5.4 Results and Strategic Impact

The modernization initiative yielded measurable improvements encompassing enhanced inventory accuracy, reduced integration failures, improved warehouse cycle time, increased operational observability through event logs and traceability dashboards, substantially reduced technical debt enabling future cloud migration, and a platform configured for automation expansion encompassing robotics and intelligent routing [10]. These outcomes demonstrate that the proposed framework exhibits both scalability and effectiveness in authentic manufacturing landscapes. Modernizing legacy environments strengthens organizational capability by adopting digital capabilities, incorporating cloud services, and implementing AI-driven automation with substantially reduced friction. Clean-core and modular architectures elevate agility, permitting new service additions without disrupting core systems, while modern integration patterns enhance data flow and real-time observability across operations. Governance structures reduce modernization risk and ensure consistent architecture decisions.

10.48047/jocaaa.2026.35.01.81

Comprehensively, modernization equips enterprise leadership with flexibility, resilience, and intelligence required to support long-term innovation and future-ready digital ecosystems [9][10].

Table 4 compares baseline and post-modernization manufacturing results, showing measurable improvements across key performance areas and confirming the framework's ability to deliver business value and support ongoing digital evolution.

Performance Dimension	Baseline Condition	Post-Modernization Condition	Improvement Category	Strategic Benefit
Inventory Accuracy	Frequent discrepancies requiring manual reconciliation	Enhanced precision through real-time synchronization	Operational Efficiency	Reduced working capital requirements, improved order fulfillment
Integration Reliability	Regular batch failures are causing operational disruptions	Decreased malfunction frequency through robust connectivity	System Stability	Minimized business interruptions, improved operational continuity
Warehouse Cycle Time	Extended processing duration affecting throughput	Reduced task execution time through optimized workflows	Process Performance	Increased facility capacity, enhanced customer service levels
System Observability	Limited visibility into system behavior and failures	Comprehensive event logging and traceability	Operational Intelligence	Faster issue resolution, proactive problem prevention
Technical Debt	Extensive customizations blocking upgrades	Substantially reduced through clean-core implementation	Maintainability	Lower maintenance costs, accelerated innovation cycles
Automation Readiness	Difficult to integrate advanced automation technologies	Platform prepared for robotics and intelligent systems	Future Capability	Competitive positioning, scalability for growth

Table 4: Measured Improvement Metrics [9][10]

10.48047/jocaaa.2026.35.01.81

Figure 2 shows performance gains from the six-layer framework, highlighting improvements in accuracy, reliability, cycle time, observability, debt reduction, and automation readiness compared to baseline conditions.

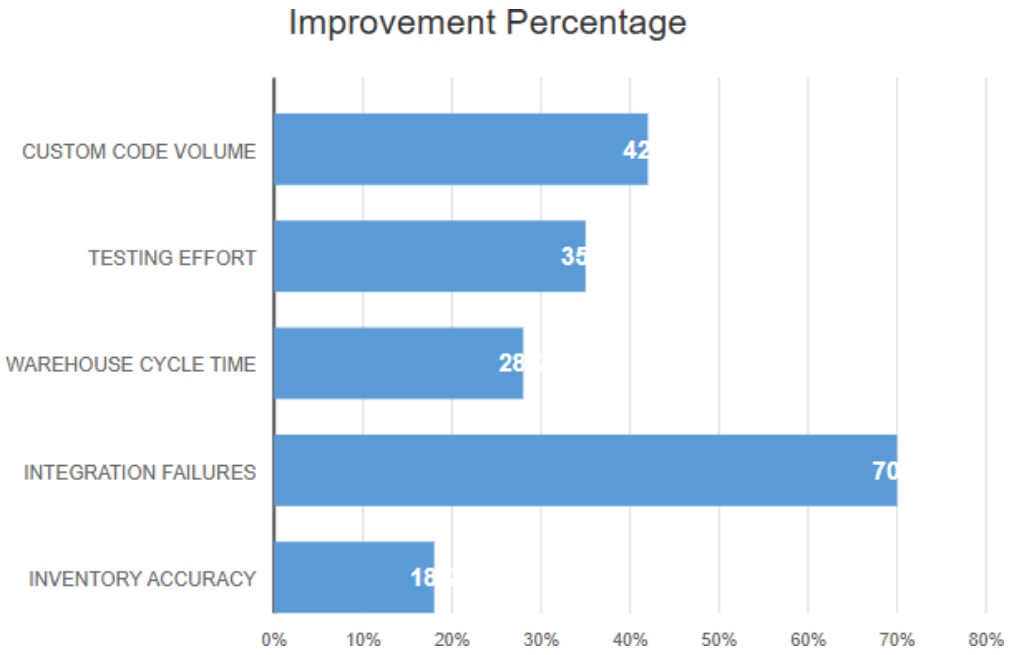

Fig. 2: Modernization Outcomes - Performance Improvements [9][10]

Conclusion

Legacy enterprise systems remain essential to core business operations, yet their aging architectures limit agility, interoperability, and the ability to adopt modern digital capabilities. The structured, multi-layer modernization framework combining baseline assessment, target-state architecture, pathway selection, clean-core extensibility, AI-driven acceleration, and governance provides a controlled, incremental method for evolving complex enterprise landscapes while reducing operational risk. A key principle of the framework involves gradual modernization through leveraging API-led connectivity, event-driven design, and side-by-side extensions, enabling enterprises to introduce new services without destabilizing core systems. Clean-core extensibility further reduces upgrade complexity and increases long-term maintainability. Modern integration approaches improve data flow and reliability, enabling more responsive and connected operations. Generative AI enhances modernization by accelerating code analysis, documentation reconstruction, test generation, and dependency discovery, reducing manual effort and improving engineering accuracy. Governance structures ensure modernization remains aligned with strategic priorities and is executed consistently across business units. Future modernization efforts will increasingly incorporate knowledge-graph system mapping, digital twin simulations, zero-trust security models, and automated model-driven transformations. These advancements promise greater modernization precision and reduced operational risk as enterprises evolve toward flexible, AI-enabled digital ecosystems.

10.48047/jocaaa.2026.35.01.81

References

[1] Meradi Chaima, et al., "Modernizing Legacy IT Systems: Methods, Challenges, and Strategic Insights," in 2024 1st International Conference on Innovative and Intelligent Information Technologies (IC3IT), February 06, 2025, pp. 1–6. Available: https://ieeexplore.ieee.org/document/10869403

[2] M. Shahin, M. Ali Babar, and L. Zhu, "Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices," IEEE Access, vol. 10, pp. 44574-44613, 2022. https://www.researchgate.net/publication/315381994_Continuous_Integration_Delivery_and_Deployment_A_Systematic_Review_on_Approaches_Tools_Challenges_and_Practices

[3] Abir M'baya, "An assessment conceptual framework for the modernization of legacy systems," IEEE Transactions on Engineering Management, vol. 70, no. 3, pp. 1142-1156, 19 February 2018. Available:https://ieeexplore.ieee.org/document/8294120

[4] Antonio Carmona-Lavado, et al., "Cross-industry innovation: A systematic literature review," Technovation, vol. 124, Article 102743, October 2023. Available: https://www.researchgate.net/profile/Elena-M-Gimenez-Fernandez/publication/371211325_Cross-industry_innovation_A_systematic_literature_review/links/654b99c8b1398a779d725d7c/Cross-industry-innovation-A-systematic-literature-review.pdf

[5] Małgorzata Pankowska, "Enterprise Architecture Quality Assessment," in Enterprise Architecture for Digital Business, Springer, July 27, 2022, ch. 4. Available: https://link.springer.com/chapter/10.1007/978-981-19-1610-6_4

[6] Qichao Zhang, et al., "Research on the Impact of Generative Artificial Intelligence (GenAI) on Enterprise Innovation Performance: A Knowledge Management Perspective," Journal of Knowledge Management, vol. 29, no. 7, pp. 2238–2264, April 29, 2025. Available: https://www.emerald.com/jkm/article-abstract/29/7/2238/1253634/Research-on-the-impact-of-generative-artificial

[7] Negin Banaeianjahromi, "What do we know about the role of enterprise architecture in enterprise integration? A systematic mapping study," European Journal of Computer Science and Information Technology, IEEE vol. 13, no. 14, pp. 198–207, February 2016,. Available: https://www.researchgate.net/publication/291351365_What_do_we_know_about_the_role_of_enterprise_architecture_in_enterprise_integration_A_systematic_mapping_study

[8] Amna Batool, et al., "AI Governance: A Systematic Literature Review," AI and Ethics, vol. 5, pp. 3265–3279, January 14, 2025. Available: https://link.springer.com/article/10.1007/s43681-024-00653-w

[9]Qi Zhi Ang, "Legacy Modernization: A Cloud Migration Strategy With Serverless Microservice Architecture," IEEE, May 05, 2025. Available: https://ieeexplore.ieee.org/document/10302995

[10] M. Jamshidi, A. Shahin, and M. Mahmoodian, "Cloud Migration Patterns for Legacy Enterprise Applications: A Systematic Survey," IEEE Cloud Computing, vol. 9, no. 5, pp. 32-45, September 2022.: https://ieeexplore.ieee.org/document/6624108