The Digital Twin Of The Enterprise (DTE): Leveraging AI To Simulate Cloud Migration And Platform Modernization Scenarios

Rajasingh Gandhi Ramdas

Senior Technical Architect, USA.

Abstract

Substantial organizational technology transformation efforts, particularly those addressing cloud platform transitions, system unification projects, and obsolete infrastructure elimination, introduce considerable operational and financial exposure to enterprises. The Digital Twin of the Enterprise presents an advancement in methodology: a detailed, continuously updated, and functionally operational virtual construct representing organizational technology assets and operational processes. By maintaining perpetual information synchronization and employing intelligent analytical forecasting, the DTE permits technology strategists to assess complex renovation pathways with empirical assurance, enabling uninterrupted operational transitions while minimizing unanticipated technical and budgetary complications. This article addresses the structural elements supporting the DTE, demonstrates functional applications in assessing renovation alternatives, and chronicles a platform elimination and unification project that exemplifies the revolutionary potential of this technique in technology governance structures.

Keywords: Digital Twin, Cloud Migration, Platform Modernization, Artificial Intelligence, Enterprise Architecture.

Introduction

1. Understanding Transformation Complexity and Associated Risks

Modern organizational contexts necessitate rapid technological evolution, requiring enterprises to update legacy infrastructure, relocate processing workloads to distributed computing environments, and integrate fragmented technological ecosystems. Such transformation projects represent among the most precarious ventures within organizational technology administration, commonly demanding significant financial resources while introducing substantial operational interruption possibilities. Observational data demonstrates that major information technology initiatives routinely encounter critical failure modes, appearing as unexpected system performance decline after platform transitions, errors in component interdependency identification that expand into comprehensive system failures, and dramatic budget overruns that far exceed original financial projections. Traditional change management structures, which depend heavily on static documentation, limited testing frameworks, and reactive problem resolution, have shown inadequate capacity for managing the complexity and interconnected characteristics present in modern enterprise technology architectures.

1.1 Inadequacies of Traditional Planning Approaches

Standard transformation planning methodologies typically employ point-in-time architectural reviews, narrow-scope testing in development environments, and manual dependency charting activities that quickly lose accuracy as systems evolve [1][2]. Such methods operate on essentially reactive principles, identifying issues only after manifestation in production settings, when correction costs reach peak levels and business impact becomes most critical. The gap separating planning assumptions from operational reality creates a persistent risk source that undermines transformation success probabilities. Standard load testing, for example, cannot adequately replicate the complex interaction patterns and emergent behaviors characterizing production workloads across interconnected systems. Similarly, dependency documentation maintained manually inevitably diverges from actual system states, creating visibility gaps in impact assessment and migration planning.

Table 1: Comparison of Traditional vs. DTE-Based Transformation Approaches [1][2]

Planning Dimension	Traditional Approach	DTE-Based Approach
Architectural Assessment	Point-in-time static documentation	Continuous real-time synchronization
Testing Environment	Limited-scope non-production settings	Full-scale production-replica simulation
Dependency Mapping	Manual documentation exercises	Automated graph-based discovery
Issue Detection Timing	Post-implementation in production	Pre-implementation in simulation
Risk Identification	Reactive after deployment	Proactive before deployment
Documentation Currency	Rapidly becomes outdated	Perpetually current
Impact Analysis Scope	Narrow technical focus	Comprehensive technical and operational
Validation Methodology	Sample-based testing	Complete workload simulation

1.2 The Digital Twin Framework for Enterprises

The Digital Twin of the Enterprise appears as an innovative answer to these continuing challenges, representing a paradigm shift from reactive change management to proactive, prediction-based transformation planning through high-precision simulation [3][4]. Unlike static architectural documentation or narrow-scope testing environments, the DTE constitutes a living, operational model of the entire enterprise technology landscape, perpetually synchronized with production systems and capable of simulating complex scenarios before implementation in actual infrastructure. This approach adapts established digital twin principles from manufacturing and industrial automation, customizing them for the specific needs of enterprise IT architecture. The DTE utilizes artificial intelligence and machine learning to create predictive models that forecast performance, stability, and cost implications of proposed changes with remarkable accuracy, transforming architectural decision-making from an experience-based art into a data-driven science.

1.3 Investigation Parameters and Research Boundaries

This study examines the architectural foundations, implementation methodologies, and practical applications of the Digital Twin of the Enterprise as a risk-management framework for large-scale digital transformation initiatives. The analysis focuses specifically on scenarios involving cloud platform migrations, system consolidation projects, and legacy infrastructure retirement, where consequences of planning failures are most severe, and the value of predictive simulation is most pronounced. By analyzing both the technical infrastructure required to construct and maintain a DTE and the specific use cases where it generates measurable value, this research establishes a comprehensive framework for implementing digital twin methodologies in enterprise transformation contexts. The examination encompasses data infrastructure requirements, AI and machine learning techniques employed for simulation, and practical workflows through which architects and decision-makers leverage the DTE to validate transformation strategies before committing resources to implementation.

2. Conceptual Foundations and Related Research

The theoretical basis of the Digital Twin of the Enterprise draws from multiple research domains and practice areas, incorporating insights from digital twin theory, enterprise architecture frameworks, and AI-driven predictive analytics. Understanding the theoretical underpinnings and existing knowledge base surrounding these concepts is essential for positioning the DTE within the broader landscape of enterprise transformation methodologies and identifying the unique contributions this approach offers to the field.

2.1 Digital Twin Applications in Enterprise Systems

Digital twin technology has undergone significant evolution from its origins in manufacturing and product lifecycle management to encompass broader applications in enterprise systems and organizational transformation [5][6]. The fundamental concept of a digital twin—a virtual representation of a physical entity that mirrors its state, behavior, and characteristics in real-time—has proven remarkably adaptable to contexts beyond physical manufacturing. In enterprise architecture contexts, the digital twin extends this concept to encompass not just physical infrastructure but the entire socio-technical system, including software applications, data flows, integration patterns, business processes, and organizational structures. Research in industrial digitalization has demonstrated that digital twins provide real-time decision support by maintaining synchronized representations of complex systems and enabling what-if analysis without disrupting operational environments. Applying digital twin concepts to enterprise architecture represents a natural evolution, addressing the growing complexity of distributed systems and the increasing velocity of technological change characterizing modern organizations.

2.2 Transition from Static Documentation to Dynamic Models

Traditional enterprise architecture practice has relied heavily on static documentation artifacts—architectural diagrams, dependency maps, and technical specifications—representing point-in-time snapshots of system state and design intent [7][8]. These static representations suffer fundamental limitations in dynamic, rapidly evolving environments where actual system states diverge from documented states almost immediately following documentation creation. The evolution toward executable models represents a fundamental shift in how architectural knowledge is captured and utilized. Rather than documenting what systems are intended to do or what their state was at some past moment, executable models embody actual system behavior and can be interrogated, tested, and simulated to answer specific questions about system capabilities and constraints. This transition from descriptive to predictive architectural artifacts enables a more scientific approach to transformation planning, where hypotheses about system behavior can be tested and validated before implementation. The digital twin represents the culmination of this evolution, combining real-time synchronization with executable simulation capabilities to create a continuously current, actionable representation of enterprise technology landscapes.

2.3 Artificial Intelligence in Transformation Strategy Development

The integration of artificial intelligence and machine learning into transformation planning methodologies represents a critical enabler for predictive approaches to change management. Historical approaches to capacity planning, performance prediction, and impact analysis have relied on simplified models, expert judgment, and historical analogies that cannot adequately capture the complexity and non-linear behaviors of modern distributed systems. Machine learning techniques offer the capability to learn patterns from historical operational data, identify subtle relationships between system components and performance outcomes, and generate predictions about future behavior under various scenarios. In the context of digital twins, AI serves multiple functions: maintaining currency of the model through automated synchronization, identifying dependencies and relationships that may not be explicitly documented, predicting performance characteristics under hypothetical configurations, and generating insights about optimal transformation strategies. Applying AI to enterprise transformation represents a shift from rule-based, deterministic planning approaches to probabilistic, data-driven methodologies that better account for inherent complexity and uncertainty in large-scale system changes.

Table 2: AI Functions in Digital Twin Operations [3][4]

AI Function Category	Operational Purpose	Technical Implementation	Business Outcome
Automated Synchronization	Maintain model currency	Continuous telemetry processing	Real-time accuracy
Pattern Recognition	Identify implicit dependencies	Graph neural networks	Complete visibility
Performance Prediction	Forecast system behavior	Time-series forecasting models	Proactive planning
Anomaly Detection	Identify potential failures	Unsupervised learning algorithms	Risk mitigation
Load Synthesis	Generate realistic workloads	Generative adversarial networks	Accurate testing
Resource Optimization	Recommend configurations	Reinforcement learning	Cost efficiency
Impact Analysis	Assess change consequences	Causal inference models	Informed decisions
Strategy Generation	Suggest transformation paths	Multi-objective optimization	Optimal outcomes

2.4 Deficiencies in Current Methodologies

Despite advances in enterprise architecture practice, transformation planning methodologies, and testing approaches, significant gaps remain in the organizational ability to predict and manage risks associated with large-scale digital transformation initiatives. Current methodologies cannot continuously validate architectural assumptions against actual production behavior, simulate complex interactions between interdependent systems under realistic load conditions, or quantitatively assess financial and operational impacts of proposed changes before implementation. Testing in non-production environments, while valuable, cannot replicate the full complexity of production workloads, data distributions, and integration patterns. Manual dependency mapping exercises cannot keep pace with the rate of change in modern DevOps and continuous delivery environments. These gaps create persistent blind spots in transformation planning that manifest as unexpected failures, performance issues, and cost overruns during implementation. The Digital Twin of the Enterprise addresses these gaps by providing a continuously current, high-fidelity simulation environment that bridges the divide between architectural intent and operational reality.

3. Structural Components and Information Infrastructure

The Digital Twin of the Enterprise distinguishes itself from mere static documentation through real-time data synchronization and executable nature, representing a fundamental architectural innovation in how enterprise technology landscapes are modeled and analyzed. The foundational architecture of the DTE rests upon three integrated pillars that work in concert to create and maintain a living, actionable representation of organizational technological ecosystems.

3.1 Continuous Telemetry Collection Infrastructure

The foundation of any effective digital twin is the continuous stream of high-fidelity data from systems being modeled, and the DTE's continuous telemetry collection infrastructure serves this critical function by ingesting operational telemetry from production environments [1][2]. This layer continuously captures API call logs, transaction metrics, resource utilization data, including CPU and memory consumption, database query performance, integration latency, and state changes across all monitored systems. Implementing this observability layer requires a robust data infrastructure built on event-driven architecture principles, where state changes and performance metrics are captured and transmitted as discrete events in near-real-time. This architectural pattern enables the DTE to maintain currency without imposing unacceptable overhead on production systems, using asynchronous event streaming to decouple data collection from operational workloads. The observability layer must accommodate heterogeneity of enterprise technology stacks, integrating with diverse monitoring tools, application performance management platforms, and logging systems to create a unified stream of operational intelligence. This real-time data ingestion ensures that the DTE remains a living replica of production environments, far surpassing the utility of static, time-bound architectural assessments that become obsolete almost immediately upon completion.

3.2 Graph-Based Dependency Representation

At the conceptual and technical heart of the DTE lies the modeling layer, which constructs and maintains a vast, interconnected graph database that maps every component of the enterprise technology landscape to its relationships, dependencies, and associated business processes [3][4]. This digital graph captures not merely a catalog of systems and components but the rich network of dependencies characterizing modern distributed architectures. Every software component, configuration item, database schema, integration endpoint, and infrastructure element is represented as a node in the graph, with edges representing dependencies, data flows, API relationships, and functional connections. The modeling layer captures both explicitly documented dependencies and implicit relationships discovered through analysis of operational telemetry, using pattern recognition to identify connections that may not be formally documented but are evident in actual system behavior. This comprehensive dependency mapping extends beyond technical components to encompass business process relationships, enabling impact analysis that considers both technical and operational consequences of proposed changes. The graph database approach provides query flexibility and traversal performance necessary to answer complex questions about system interdependencies, impact radius, and cascading effects that would be computationally infeasible with relational data models or static documentation approaches.

Table 3: Graph Database Node and Relationship Types [3][4]

Node Type	Attributes Captured	Relationship Types	Discovery Method
Software Component	Version, language, deployment location	Invokes, depends on, shares data with	Code analysis, runtime monitoring
Configuration Item	Parameters, settings, environment variables	Configures, influences, restricts	Configuration management systems
Database Schema	Tables, fields, indexes, constraints	Stores data for, queries from, replicates to	Schema introspection, query logs
Integration Endpoint	Protocol, authentication, rate limits	Connects to, sends data to, receives from	API documentation, traffic analysis
Infrastructure Element	Capacity, location, specifications	Hosts, provides resources to, scales with	Infrastructure management tools
Business Process	Owner, criticality, SLA requirements	Supported by, triggers, requires	Process documentation, workflow analysis
User Group	Permissions, roles, access patterns	Uses, administers, and depends on	Identity management, usage logs

Data Flow	Volume, frequency, latency	Originates from, transforms, and terminates at	Network monitoring, data lineage tools

Table 4: Financial Impact Components in DTE Analysis [3][4]

Cost Category	Current State Elements	Future State Elements	Calculation Approach	Decision Impact
Platform Licensing	Multiple system licenses	Consolidated licenses	Per-user or capacity-based pricing	Consolidation ROI
Infrastructure	Distributed server costs	Unified infrastructure costs	Resource consumption simulation	Sizing decisions
Maintenance Overhead	Per-system support contracts	Unified support model	Historical maintenance tracking	Vendor selection
Integration Costs	Point-to-point integration maintenance	Unified integration layer	Complexity metrics	Architecture pattern choice
Operational Support	Multiple team allocations	Streamlined team structure	Labor hour analysis	Organizational design
Training Investment	Platform-specific training	Unified platform training	Per-employee training costs	Change management scope
Migration Execution	One-time transformation costs	Zero-downtime approach costs	Resource and timeline estimates	Budget allocation
Risk Contingency	Historical overrun patterns	Simulation-based risk assessment	Probabilistic cost modeling	Reserve determination

4.4 Business Continuity Validation During Transitions

The requirement for business continuity during transformation creates one of the most significant technical challenges in large-scale modernization initiatives, as organizations cannot afford extended outages of critical business systems [5][6]. Zero-downtime migration strategies involve complex orchestration of data synchronization, cutover procedures, fallback mechanisms, and validation processes that must execute flawlessly to avoid business disruption. The DTE enables rigorous validation of these migration strategies through simulation of the complete cutover sequence. Architects can model the proposed migration approach, including any parallel-run periods, data synchronization mechanisms, and cutover triggers, and simulate execution against realistic transaction loads. The simulation tests whether data consistency can be maintained during synchronization periods, whether cutover can be executed within acceptable time windows, and whether fallback procedures will function correctly if issues are detected. By identifying potential failure points in the migration strategy before execution, the DTE enables refinement and validation of procedures that would otherwise be tested for the first time in production, where the consequences of failure are most severe.

4.5 Technical Debt Impact Quantification

Legacy systems often accumulate substantial technical debt in the form of inefficient code patterns, complex customizations, and architectures that no longer align with current best practices or platform capabilities [7][8]. A critical question in modernization planning is which technical debt must be addressed before migration and which can be tolerated or addressed later. The DTE enables quantitative assessment of technical debt impacts by simulating both current-state performance with existing technical debt and future-state performance after proposed refactoring. For instance, when evaluating whether to refactor complex legacy code before migration, the DTE can predict specific performance improvements that would result from refactoring—quantifying reduction in CPU timeouts, decrease in database query overhead, and improvement in user experience metrics. This quantitative analysis transforms technical debt discussions from subjective architectural preferences into evidence-based decisions about where refactoring investment will deliver meaningful returns. The simulation can also identify scenarios where technical debt creates critical risks: if current code approaches platform governor limits under peak load, the simulation can demonstrate that migration without refactoring would result in production failures, providing clear justification for investment in code modernization.

5. Practical Implementation: Platform Retirement and Unification

The practical value of the Digital Twin of the Enterprise is best illustrated through examination of a complex, high-stakes transformation scenario involving strategic consolidation of fragmented legacy systems into a unified platform architecture. This case study examines the application of DTE capabilities to a multi-phase consolidation initiative involving the retirement of vintage enterprise systems and unification of disparate business processes onto a modern, scalable Quote-to-Cash platform.

5.1 Business Context and Strategic Imperatives

The organization faced a common challenge in enterprise IT: decades of organic growth and acquisition activity had resulted in a fragmented technology landscape with multiple redundant systems serving similar business functions. Two vintage Salesforce Orgs existed in parallel, each supporting different business units with overlapping functionality but incompatible data models and custom logic. Additional systems including QuickBooks for financial management and Dynamics for certain customer relationship processes created further fragmentation, with complex point-to-point integrations attempting to maintain data consistency across systems. This fragmentation resulted in substantial operational overhead, data quality issues from a lack of a single source of truth, difficulty extracting enterprise-wide analytics and insights, and high ongoing costs from maintaining multiple platforms and their associated integrations. The strategic imperative was to consolidate these disparate systems into a unified Quote-to-Cash platform that would serve as the enterprise system of record, eliminate redundancy, and provide a foundation for future growth and innovation.

5.2 Legacy Code Modernization Validation

Before executing consolidation, the organization needed to address substantial technical debt in legacy systems, particularly complex custom code that would not perform acceptably if simply migrated to the consolidated platform [1][2]. The challenge was determining which customizations required refactoring before migration and quantifying the expected benefits of that investment. The DTE was employed to simulate the enforcement of a low-customization strategy that emphasized standardized components and modern development patterns. The simulation analyzed the performance history of complex legacy code under production load conditions, identifying specific patterns that resulted in CPU timeouts, governor limit violations, and poor user experience. The AI models then simulated the behavior of this logic when refactored into modern, standardized components using Lightning Web Components and declarative Flow automation rather than complex procedural code. The simulation provided quantitative predictions of the exact reduction in CPU consumption, elimination of SOQL query limit violations, and improvement in transaction throughput that would result from refactoring. These predictions enabled the organization to build a business case for refactoring investment, demonstrating that the cost of lost workdays from timeout failures in the current state far exceeded the investment required to modernize the code. By validating the refactoring initiative before committing resources, the DTE transformed what would otherwise be a speculative architectural preference into a data-driven investment decision.

5.3 Seamless Transition Strategy Testing

The most critical risk in the consolidation initiative was ensuring business continuity during the transition from legacy systems to the unified platform [3][4]. The organization could not afford extended downtime of order management, billing, or customer service functions, necessitating a sophisticated zero-downtime migration strategy. The DTE enabled a comprehensive simulation of the migration approach before execution. The simulation modeled proposed cutover sequences, including periods of parallel operation where transactions would be processed in both legacy and target systems, data synchronization mechanisms that would maintain consistency between systems during this period, and validation procedures that would confirm functional equivalence before final cutover. The digital twin tested the compatibility of the new, unified data model against the full historical transactional load of decommissioned systems, processing representative samples of actual production transactions through simulated target architecture. This simulation identified several critical issues that would have resulted in data loss or functional failures if discovered during actual migration: subtle incompatibilities between legacy data models and unified schema that would have caused data mapping failures, integration conflicts with core systems like SAP S/4HANA and Master Data Management platforms that arose from changes in API contracts, and performance bottlenecks in data synchronization process that would have prevented real-time consistency. By identifying these issues in simulation, the team was able to refine the migration approach, adjust data model design, and implement additional validation mechanisms before actual cutover, dramatically reducing the risk of business disruption.

5.4 Capacity Planning Through Predictive Analysis

Traditional load and stress testing approaches are inherently reactive, testing systems after they are built to determine whether they meet performance requirements. The DTE enabled a predictive approach where the consolidated platform's performance characteristics were forecasted before implementation [5][6]. After the target architecture was modeled in the digital twin, the AI simulation engine subjected it to a combined peak load profile of all systems being retired, adjusted for forecasted business growth over the next planning period. This simulation considered not just aggregate transaction volume but complex interaction patterns between different functional areas—sales processes triggering service workflows, order creation driving billing processes, and customer interactions spanning multiple channels and touchpoints. The simulation predicted the platform's steady-state performance under this combined load, identifying specific bottlenecks and scalability constraints that would emerge as a unified system absorbed full workload. One critical finding was that unification of Sales, Service, and Integration functions on a single platform would create contention for shared resources during peak periods, requiring infrastructure scaling beyond initial estimates. The simulation enabled the team to proactively address these scalability concerns through architectural adjustments and infrastructure provisioning before going live, rather than discovering capacity limitations through production incidents. This predictive validation provided confidence that the consolidated platform would not merely function but would deliver acceptable performance under realistic operational conditions.

5.5 Continuous Refinement Through Simulation Cycles

Perhaps the most valuable aspect of the DTE in this consolidation initiative was the ability to iterate on transformation plans within the simulation environment, refining approaches based on predicted outcomes without cost and risk of actual implementation cycles [7][8]. When initial simulation revealed performance concerns or compatibility issues, architects could modify the target architecture, adjust migration procedures, or implement additional optimization measures and immediately re-simulate to validate the effectiveness of those changes. This iterative refinement process continued until simulation demonstrated that all critical requirements would be met: functional equivalence with legacy systems, acceptable performance under peak load, successful zero-downtime migration, and achievement of cost reduction targets. By replacing speculation and hope with predictive certainty, the DTE enabled the organization to execute consolidation with confidence, having validated every critical aspect of transformation in simulation before committing to actual implementation. The result was a successful consolidation that achieved its business objectives without disruption, cost overruns, or performance failures that frequently characterize initiatives of this scale and complexity.

Conclusion

The Digital Twin of the Enterprise represents a fundamental evolution in how organizations approach large-scale digital transformation, cloud migration, and platform modernization initiatives. By creating a living, executable model of the enterprise technology landscape and leveraging artificial intelligence for predictive simulation, the DTE transforms transformation planning from a reactive, risk-laden process into a proactive, evidence-based discipline. The architectural foundations of the DTE, encompassing real-time observability, graph-based dependency modeling, and AI-powered simulation, enable capabilities that were previously unattainable: predicting performance characteristics before implementation, validating complex migration strategies without business disruption, quantifying the impact of technical debt remediation, and iteratively refining transformation plans based on simulated outcomes. The case study of platform consolidation demonstrates the practical value of these capabilities in mitigating substantial risks associated with enterprise-scale transformation initiatives.

The benefits of the DTE approach extend beyond individual transformation projects to encompass ongoing architectural governance and strategic planning. Rather than being a one-time assessment tool, the DTE serves as a continuous asset that maintains currency with an evolving production environment and enables rapid evaluation of new opportunities and changing business requirements. As organizations adopt continuous delivery practices and accelerate the pace of technological change, the DTE provides a mechanism for ensuring that architectural quality and system integrity are maintained even as the velocity of change increases. The predictive certainty enabled by DTE simulation supports more aggressive transformation strategies and more efficient resource allocation, as organizations can pursue optimization opportunities with confidence rather than excessive conservatism driven by uncertainty.

Despite demonstrated value of the DTE approach, several limitations and challenges merit acknowledgment. Construction and maintenance of a high-fidelity digital twin requires substantial investment in data infrastructure, observability tooling, and modeling capabilities. Organizations must balance the cost of implementing and operating a DTE against value delivered through improved transformation outcomes. Accuracy of DTE predictions depends critically on the quality and comprehensiveness of observability data and the sophistication of AI models, creating ongoing requirements for data governance and model refinement. Integration with diverse technology stacks and legacy systems can present technical challenges, particularly where observability capabilities are limited or where systems lack modern instrumentation. Additionally, cultural and organizational changes required to shift from traditional planning approaches to simulation-based methodologies should not be underestimated.

Future research directions for advancing the DTE concept are numerous and promising. Application of generative AI techniques to automatically construct digital twin models from existing documentation, code repositories, and operational telemetry could dramatically reduce the effort required to establish a DTE. Automated generation of test cases and validation scenarios based on production patterns could enhance simulation coverage and reduce manual effort in transformation validation. Integration of DTE capabilities with autonomic computing concepts could enable self-optimizing architectures that automatically adapt to changing conditions based on continuous simulation of alternative configurations. Research into federated digital twin architectures could address the challenge of modeling and simulating complex multi-organization ecosystems and supply chain networks. Development of standardized interfaces and protocols for digital twin interoperability could enable ecosystem-wide simulation and optimization. Investigation of quantum computing

applications to enterprise simulation could overcome current computational limitations in modeling extremely large, complex systems.
As digital transformation remains a strategic imperative for organizations across industries, the need for more sophisticated approaches to managing transformation risk and complexity will only intensify. The Digital Twin of the Enterprise offers a compelling vision for how artificial intelligence and advanced simulation techniques can be harnessed to navigate this complexity, providing architects and decision-makers with predictive capabilities required to execute transformative change with confidence. Continued research, development, and practical application of DTE concepts will be essential for realizing this vision and establishing simulation-based planning as a standard practice in enterprise architecture and transformation management.

References

[1] Zhixiong Chen, et al., "Distributed Digital Twin Migration in Multi-Tier Computing Systems," IEEE Xplore, 26 January 2024. Available: https://ieeexplore.ieee.org/document/10415196
[2] Bruno Ribeiro, et al., "Digital Twin Migration using the OKD platform: A Use-Case for Emergency Vehicles," 2023 33rd International Telecommunication Networks and Applications Conference, 29 December 2023. Available: https://ieeexplore.ieee.org/document/10368494
[3] Ramesh Mahankali, "DIGITAL TWINS AND ENTERPRISE ARCHITECTURE: A Framework for Real-Time Manufacturing Decision Support," International Journal of Computer Engineering and Technology (IJCET), Jan–Feb 2025. Available: https://www.researchgate.net/publication/388105964_DIGITAL_TWINS_AND_ENTERPRISE_ARCHITECTURE_A_FRAMEWORK_FOR_REAL-TIME_MANUFACTURING_DECISION_SUPPORT
[4] Dirk Hartmann, Herman van der Auweraer, "Digital Twins," arXiv preprint, 2020. Available: https://arxiv.org/abs/2001.09747
[5] Digital Twin Consortium Team, "Enterprise Transformation with Digital Twins," Digital Twin Consortium Proceedings, July 24, 2024. Available: https://www.digitaltwinconsortium.org/2023/07/enterprise-transformation-with-digital-twins/
[6] Multiple Authors (PMC Review Team), "A Comprehensive Review of Digital Twin from the Perspective of Industrial Digitalization," PMC Sensors Journal, October 7, 2023. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC10575411/
[7] Mengjin Qu, et al., "Industrial Digital Twins Based on Enterprise Modeling," Journal of Industrial Information Integration, 2024. Available: https://www.sciencedirect.com/science/article/abs/pii/S2452414X25001359
[8] Zhixiong Chen, et al., "Distributed Digital Twin Migration in Multi-Tier Computing Systems," IEEE Xplore, 26 January 2024. Available: https://ieeexplore.ieee.org/document/10415196