SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Pre-Deployment Risk Reduction Through Digital Twin Simulation in Enterprise Network Change Management

Mitesh kumar Jagdishbhai Patel

Independent Researcher, USA

Abstract
Production-based change validation in enterprise network environments accepts an operational risk that high-consequence infrastructure contexts cannot sustainably absorb. When network changes in manufacturing enterprise environments fail during maintenance windows, the consequences extend beyond service degradation to production line disruption whose financial impact accumulates by the minute. This article argues that digital twin simulation environments, integrated into change management workflows as mandatory pre-deployment validation stages rather than consulted informally, address this risk by converting uncertain production outcomes into confirmed simulation behaviors before any production device is modified. The discussion examines topology synchronisation architecture, workflow integration design, failure scenario coverage, risk-scored change categorisation, and rollback pre-validation as integrated components of a simulation-driven change governance framework. The central argument is that digital twin fidelity and workflow integration together determine the operational value of simulation for change management, that neither alone is sufficient, and that most enterprise digital twin deployments lack the workflow integration that makes fidelity operationally useful.

Keywords: Digital Twins, Network Simulation, Change Management, Risk Reduction, Operational Resilience, Pre-Deployment Validation, Enterprise Networking, Scripting Automation, Configuration Compliance, Governance

I. Introduction

Production change failures in manufacturing enterprise environments carry a cost structure that most IT change management frameworks underweight. A failed network change during a weekend maintenance window at an automotive assembly plant does not merely create a service ticket, it stops production lines whose output loss accumulates at rates that dwarf the cost of the change management process itself [1]. The standard risk mitigation response, maintenance windows, rollback procedures, post-change monitoring, addresses the consequences of failure without reducing its probability. Digital twin simulation addresses the probability directly, by validating change behavior in a production-representative environment before any production device is touched.

What separates a digital twin from a traditional lab environment is synchronisation. A lab built from the same hardware platforms as production approximates the environment; a digital twin that continuously imports production topology state, current device configurations, and traffic behavioral patterns from production systems replicates it [1]. This replication property determines how much a digital twin's change validation results can be trusted as predictors of production behavior, and the fidelity of the replication, which depends on synchronisation architecture quality, is the primary engineering investment that determines the digital twin's operational value. Underinvesting in synchronisation produces a system that looks like a digital twin and validates changes with digital twin process discipline while producing results whose predictive accuracy is no better than a static lab.

Four architectural decisions determine whether digital twin deployment produces consistent change risk reduction: how topology synchronisation is designed and maintained, how simulation is integrated into the change management workflow, which failure scenarios the validation framework is designed to exercise, and how rollback procedures are pre-validated before production change windows open. Each decision independently affects the operational value of the system, and each has a failure mode that produces false validation confidence, the most operationally dangerous outcome possible, because it eliminates the caution that would otherwise prompt additional verification.

This article examines each decision in sequence, with particular attention to the workflow integration question that most enterprise digital twin deployments get wrong: treating the digital twin as an available resource rather than a required process stage, which produces validation coverage that tracks engineer preference rather than change risk.

II. Digital Twin Topology Synchronisation: The Foundation of Simulation Fidelity

Synchronisation fidelity is not a feature of the digital twin, it is the property that makes it a digital twin rather than a static lab. The distinction matters operationally because it determines the category of questions the simulation can answer reliably. A static lab can answer whether a change design is logically correct; a synchronised digital twin can answer whether a change design produces the correct outcome in the specific current state of the production environment, including all the configuration drift, undocumented workarounds, and traffic behavioral quirks that production environments accumulate over time [1]. The second question is the one that change management actually needs answered.

Automated configuration export from production devices, using NETCONF/RESTCONF against Cisco IOS-XE and NX-OS, the APIC REST API for ACI fabric state, and SD-WAN vManage APIs for overlay configuration, provides the configuration synchronisation layer [3]. Routing and topology state, by contrast, is better captured through continuous streaming telemetry, gNMI-based export of BGP and OSPF state changes as they occur, an approach that has matured considerably in production telemetry pipelines [8], since polling-based collection introduces a detection lag that periodic configuration snapshots alone cannot close. Normalisation across vendor-specific schema representations follows the same design pattern as telemetry pipeline normalisation: platform-specific representations of conceptually identical configuration constructs must be mapped to a unified model before they can be applied consistently in the simulation environment. The synchronisation frequency trade-off is real and operationally important: daily synchronisation produces a twin that is always 24 hours stale; continuous synchronisation risks propagating production misconfigurations into validation runs [7].

Traffic pattern modeling is the synchronisation dimension that configuration export alone cannot address. Changes whose outcomes depend on traffic-load interactions, QoS policy modifications, ECMP path adjustments, SD-WAN steering rule changes, produce different outcomes at different traffic loads, and a validation run against a configuration-accurate but traffic-absent digital twin will not surface load-sensitive failure modes [1]. NetFlow exports from production devices at appropriate sampling rates, captured during representative operational periods and replayed during validation runs, provide the traffic behavioral context that makes load-sensitive change validation reliable.

Synchronisation failure mode analysis, asking what happens to validation reliability when each synchronisation component degrades, is the engineering exercise that reveals the architecture's operational dependencies. If configuration export fails for a subset of devices, validation runs against stale device state produce results whose accuracy is unknown. If traffic capture fails, load-sensitive validation scenarios give accurate results only for low-load conditions. Building monitoring for each synchronisation component, confirming not just that synchronisation ran but that it produced accurate outputs, is as important to digital twin operational reliability as the synchronisation mechanisms themselves [7].

TABLE I. Digital Twin Topology Synchronisation: Components, Sources, and Failure Risks

Synchronisation Component	Production Data Source	Synchronisation Method	Fidelity Degradation Risk
Device configuration state	NETCONF/RESTCONF, APIC REST, vManage API	Scheduled automated export	Stale baseline misses recent drift
Routing and topology state	Streaming telemetry (BGP, OSPF state)	Continuous gNMI export	Transient topology changes missed
Traffic behavioral patterns	NetFlow/IPFIX at sampled rates	Periodic capture and replay	Load-sensitive changes poorly represented
Multi-vendor normalisation	Platform-specific YANG schemas	Unified model mapping layer	Schema errors contaminate validation

III. Integrating Digital Twin Simulation into Change Management Workflows

The most consequential architectural decision in digital twin deployment for change management is not technical, it is process. A digital twin whose synchronisation is excellent and whose simulation capabilities are sophisticated produces inconsistent operational outcomes when its use is left to individual judgment, because individual judgment systematically underestimates risk for familiar change types and overestimates the

coverage of informal validation [6]. This mirrors a broader shift in network operations toward closed-loop, declarative assurance, where intent-based networking replaces manual validation discretion with automated conformance checking against a stated policy intent rather than relying on operator judgment call by call [2]. Mandatory integration, where changes above a defined risk threshold cannot enter the production execution queue without a completed simulation report, removes this systematic bias and ensures that validation coverage is determined by risk criteria rather than by engineer confidence levels.

TABLE II. Digital Twin Change Management Framework: Integration Maturity Levels

Maturity Level	Digital Twin Usage	Workflow Integration	Governance Output	Operational Outcome
Level 1, Ad hoc	Consulted informally on selected changes	None, engineer discretion	No structured documentation	Inconsistent risk coverage
Level 2, Structured	Used for defined high-risk change categories	Partial, required for some changes	Simulation report for mandated changes	Improved high-risk change outcomes
Level 3, Integrated	Mandatory for all changes above risk threshold	Full, process gate for qualifying changes	Complete governance artifact package	Systematic risk reduction and audit readiness
Level 4, Self-improving	Continuous sync; scenario library updated from incidents	Full integration + feedback loop	Continuous improvement data captured	Progressive reduction in change failure rate

**Maturity-level outcomes in Table II are presented as a conceptual progression intended to describe the direction of expected improvement as integration deepens; they are illustrative characterisations rather than figures drawn from a fielded deployment or longitudinal study.*

Structured simulation workflow stages should be non-negotiable steps rather than advisory guidelines. Change proposal intake captures the change design and provides inputs to risk classification; topology state refresh confirms that the digital twin's baseline reflects current production state within the acceptable staleness window; change execution in the simulation environment produces observable outcomes; success criteria assessment determines whether the simulation results are acceptable; and simulation report generation creates the governance artifact that the change management record requires [4]. A change that bypasses any stage, topology refresh skipped to save time, success criteria assessed informally without documentation, produces a governance record that is incomplete and a validation whose reliability is uncertain.

Risk scoring for change categorisation provides the quantitative foundation for mandatory workflow application. Factors contributing to risk scores in enterprise network change management include the number of devices affected, the existence and tested reliability of redundant paths, the protocol convergence dependencies introduced by the change, the maintenance window timing relative to operational hours, and the historical failure rate for similar change types in the specific network segment [3]. Calibrating score thresholds against the organisation's operational risk tolerance is a governance decision that risk management and network operations teams should make jointly.

Simulation results must be treated as falsifiable predictions, not authoritative verdicts. When a change passes simulation but fails in production, the correct operational response is to analyse whether the failure was a simulation coverage gap, a synchronisation fidelity gap, or a production condition the simulation correctly could not have anticipated, and to update the validation framework accordingly [1]. Organisations that treat digital twin-validated change failures as random occurrences miss the improvement signal that each production deviation from simulation prediction carries.

TABLE III. Change Risk Scoring Framework: Factor Weights and Threshold Routing

Risk Factor	Low Risk Indicator	High Risk Indicator	Validation Tier Triggered
Devices affected	1–5 access-layer devices	10+ core or distribution devices	Tier 1 vs Tier 3
Redundancy availability	Tested redundant paths present	Single path, no redundancy	Expedited vs full simulation
Protocol convergence	Isolated VLAN change	OSPF/BGP policy modification	Nominal vs full scenario coverage

Window timing	Weekend low-traffic window	Weekday production hours	Standard vs extended window
Historical failure rate	<5% for similar changes	>20% for similar changes	CAB waiver vs mandatory CAB review

**The low/high failure-rate bands in Table III are presented qualitatively rather than as fixed percentages. Organisations should derive their own numeric thresholds from internal change-failure history; no external benchmark or dataset is claimed as the source of a specific cutoff.*

IV. Failure Scenario Coverage: What the Digital Twin Must Test

Scenario coverage gaps are the category of digital twin validation failure that is hardest to detect before a production change exposes them. A digital twin that passes every scenario it tests provides no information about the scenarios it does not test, and the production change failures that result from untested interactions are indistinguishable, without careful analysis, from failures caused by incorrect change design [7]. Building a structured scenario coverage framework for each change type category transforms this from an unstructured problem into a manageable engineering discipline.

Redundancy validation is the highest-priority scenario category for changes to enterprise networks where service continuity depends on automated failover mechanisms functioning correctly after configuration changes. HSRP gateway failover, OSPF alternate path activation, BGP next-hop reconvergence, and SD-WAN policy-based path switching must each be exercised under the post-change configuration state to confirm that the change does not inadvertently disable or degrade the failover behavior the production environment relies on [7]. The validation sequence matters: primary path confirmation before failover injection establishes the baseline; failover injection and confirmation determine whether the redundancy mechanism functions; recovery validation confirms primary path restoration.

Protocol interaction testing targets the non-obvious failure modes that change type classification alone does not predict. Routing policy changes whose individual logic is correct may produce unexpected ECMP rebalancing that concentrates traffic on links not designed for that load; ACL modifications may interact with existing NAT translations to block traffic flows that neither the new ACL nor the existing NAT individually would affect; ACI contract changes may interact with existing endpoint group policy states to produce microsegmentation behavior that differs from both the old and new policy intentions in isolation [3].

The scenario library that accumulates from this analysis is a persistent operational asset. As change types that have historically produced failures receive richer scenario coverage, and as new failure modes encountered in production are retroactively added to the library, the validation framework becomes progressively more effective at catching the specific failure types that the organisation's infrastructure is most vulnerable to, particularly when scenario prioritisation is informed by machine learning-based analysis of historical failure patterns rather than manual curation alone [5], [7]. The governance discipline required to capture this data and use it to update the scenario library is the organisational practice that converts digital twin capability into continuously improving operational outcomes.

V. Rollback Validation and Governance Automation

Confidence in a rollback procedure is operationally meaningful only when grounded in test execution, not in documentation review. A rollback script that has never been executed may contain errors that are invisible in the document but immediately consequential in production; a rollback procedure that has been successfully tested in the digital twin has demonstrated its correctness and provides an execution time estimate that the change window planning can rely on [4]. Treating rollback validation as a required pre-change deliverable, not an optional quality improvement, changes the operational character of change management from risk-acceptance to risk-confirmation: the production execution window opens only when confirmed rollback capability is in place.

Automated rollback procedure generation from digital twin pre-change state snapshots removes the manual documentation burden that makes rollback preparation a bottleneck in change management processes. When the digital twin captures a complete configuration snapshot before change execution begins and generates a rollback execution package from that snapshot automatically, the rollback procedure reflects the actual pre-change state rather than the engineer's documented recollection of the intended pre-change state [7]. Testing this generated rollback in the digital twin before the production change window confirms both correctness and timing.

Governance automation consolidates the documentation artifacts that simulation-integrated change management generates into a structured record that satisfies audit requirements without dedicated documentation effort. The simulation report, change execution log, post-change validation report, and rollback

readiness certificate together document the complete pre-deployment validation process, change execution, outcome confirmation, and rollback preparation for every significant change [7]. This documentation is generated as a byproduct of process execution rather than as a separate activity, which eliminates the retrospective reconstruction quality problems that manual documentation produces.

The feedback loop that governance automation enables is what converts digital twin-integrated change management from a static deployment into an improving system. Simulation results compared against production outcomes reveal synchronisation fidelity gaps; failure scenario coverage analyses identify validation gaps from post-incident data; rollback execution time comparisons between digital twin predictions and production actuals calibrate the timing model's accuracy [4]. Each production change cycle provides data that can improve all three dimensions of the framework.

TABLE IV. Governance Automation: Change Documentation Artifacts and Their Operational Functions

Artifact	Generated At	Audit Function	Improvement Function
Simulation report	Pre-change (post-simulation)	Pre-deployment validation evidence	Simulation vs production outcome comparison
Change execution log	During change window	Change attribution and timing	Execution sequence analysis
Post-change validation report	Post-change (within window)	Confirmed successful execution	Validation coverage gap detection
Rollback readiness certificate	Pre-change (post-rollback-test)	Confirmed rollback capability	Rollback timing model calibration

VI. Limitations

The argument for mandatory digital twin integration should not obscure the cost of building the capability it depends on. Continuous NetFlow capture at production-representative sampling rates, multi-vendor schema normalisation, and ongoing synchronisation monitoring across every device platform in the environment represent a sustained engineering investment, not a one-time deployment cost. Organisations evaluating this framework should weigh that investment against the specific production-line disruption risk it is intended to reduce, since the framework's value proposition depends on operating in a high-consequence context where change failure carries costs proportional to those described in Section I; the same investment is harder to justify in lower-consequence environments.

The most operationally dangerous failure mode this article identifies, false validation confidence produced by an undetected fidelity gap, is only partially addressed by the synchronisation-component monitoring described in Section II. Monitoring confirms that synchronisation processes ran; it does not by itself confirm that the resulting twin state matches production with the accuracy the validation depends on. Periodic reconciliation audits that directly compare sampled twin state against production ground truth, rather than inferring accuracy from process completion alone, and validation reports that carry an explicit confidence indicator reflecting current synchronisation staleness, are safeguards this framework should incorporate rather than optional refinements. Hardening the synchronisation layer itself against undetected data quality degradation, an area where security-enhanced telemetry approaches are relevant, is a related and still-developing concern [9], [10].

Scalability is a further open question this article does not resolve. The synchronisation and normalisation architecture described in Section II has been discussed at the scale of a single enterprise fabric with a bounded, if large, device count; whether the same architecture, and the computational cost of continuous multi-vendor normalisation in particular, holds without degradation at the scale of very large or multi-site enterprise fabrics with tens of thousands of managed devices is not addressed here and would need to be established separately before the framework is generalised to that scale [10].

Conclusion

The four architectural decisions examined here, synchronisation design, workflow integration, scenario coverage, and rollback pre-validation, are not independent design choices that can be optimised in isolation; they interact. A digital twin with excellent synchronisation but discretionary workflow integration produces the same inconsistent risk coverage as one with mandatory integration but shallow synchronisation, because in both cases the gap between what the twin could validate and what it actually validates is left to circumstance rather than closed by design. The framework's central claim is accordingly a joint one: fidelity and integration

are each necessary and neither is sufficient, and most enterprise deployments underperform not because either capability is absent but because the two have been built without reference to each other.

What follows from this for organisations already operating a digital twin is an audit question rather than a build question: whether the twin's synchronisation architecture and its position in the change workflow have been designed together, or whether one was added to an environment already shaped by the other. What follows for organisations without one is a sequencing question, since the workflow-integration discipline described in Section III has value even before synchronisation fidelity is fully mature, provided the limitations in Section VI are treated as active risks to be managed rather than caveats to be noted.

Two questions remain open for further work rather than settled by the argument here. The first is measurement: how an organisation validates the digital twin's own synchronisation accuracy on an ongoing basis, at scale, without that validation becoming as labour-intensive as the change review process the twin was meant to streamline. The second is portability: whether failure scenario libraries built from one organisation's incident history transfer usefully to another's infrastructure, or whether scenario coverage is inherently organisation-specific in a way that limits the field's ability to standardise validation practice across enterprises. Progress on either question would extend the framework proposed here beyond a single-organisation governance practice toward a more general basis for evaluating digital twin-integrated change management.

References

[1] P. Almasan et al., "Network Digital Twin: Context, Enabling Technologies, and Opportunities," IEEE Communications Magazine, vol. 60, no. 11, pp. 22–27, 2022. Available: https://ieeexplore.ieee.org/document/9795043

[2] A. Leivadeas and M. Falkner, "A Survey on Intent-Based Networking," IEEE Communications Surveys & Tutorials, vol. 25, no. 1, pp. 625–655, 2023. Available: https://ieeexplore.ieee.org/document/9925251

[3] E. Coronado et al., "Zero Touch Management: A Survey of Network Automation Solutions for 5G and 6G Networks," IEEE Communications Surveys & Tutorials, vol. 24, no. 4, pp. 2535–2578, 2022. Available: https://ieeexplore.ieee.org/document/9913206

[4] Xiaoang Zheng and Aris Leivadeas, "Network Assurance in Intent-Based Networking Data Centers with Machine Learning Techniques," 17th International Conference on Network and Service Management (CNSM), 2021. Available: https://ieeexplore.ieee.org/document/9615580

[5] Baolin Qin, et al., "Machine and Deep Learning for Digital Twin Networks: A Survey," IEEE Internet of Things Journal, vol. 26, no. 3, pp. 1529–1573, 2024. Available: https://ieeexplore.ieee.org/document/10570372

[6] Daniel González-Sánchez, et al., "Toward Building a Digital Twin for Network Operations and Management," IEEE Open Journal of the Communications Society, 2025. DOI: 10.1109/OJCOMS.2025.3549873

[7] Mehdi Kherbache, et al., "Network Digital Twin for the Industrial Internet of Things," IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2022. Available: https://ieeexplore.ieee.org/document/9842757

[8] Wenxin Zhong and Xianhong Zhao, "Research on DCI Streaming Telemetry System Based on gNMI," 4th International Conference on Information Science, Parallel and Distributed Systems (ISPDS), 2023. Available: https://ieeexplore.ieee.org/document/10235592

[9] D. Kong et al., "Toward Security-Enhanced In-Band Network Telemetry in Programmable Networks," IEEE Transactions on Network and Service Management, vol. 20, no. 1, pp. 137–155, 2024. Available: https://ieeexplore.ieee.org/document/10764753

[10] Hiren Patel et al., "A Survey on Digital Twin: From Industrial Applications to Cybersecurity," IEEE Smart World Congress (SWC), 2024. Available: https://ieeexplore.ieee.org/document/10924871