



Managing Requirements Volatility in Enterprise Software Projects

Harouna Wendpanga Yann Christian Sankara^{1*}, Aisha I. Abdulwahab²,
Toussida Fatah Tanguy Minoungou³, Olumide Kumuyi⁴

¹Five9, San Ramon, California, USA

²Cummins Inc., Indiana, USA

³DonorsChoose, New York, New York, USA

⁴New York University Abu Dhabi, Abu Dhabi, United Arab Emirates

*Corresponding author: Harouna Wendpanga Yann Christian Sankara, christiansankara@gmail.com

Abstract

Purpose: Requirements volatility is unavoidable in enterprise software programs, yet many organizations still measure it as a raw count of change requests and govern it through uniform approval gates. This review asks how volatility should be measured over time, how the reach of a change should be estimated through traceability, and how baselining and change control should be calibrated so that legitimate change is absorbed without uncontrolled cost, schedule, and defect consequences. **Method:** An integrative review of requirements engineering, traceability, regression testing, and empirical software engineering literature published up to 2025 was conducted under a documented search, eligibility, and appraisal protocol, with synthesis organized by mechanism, context, and outcome. **Findings:** The evidence points to the timing, magnitude, and coupling of changes, rather than their frequency, as the drivers of harm. Trace-based impact analysis is only as useful as the recall of its links, and change control boards add decision latency that can exceed the cost they avert when every request receives equal scrutiny. **Contribution:** The article proposes the TRACE-V framework of timing-weighted rate, reach, amplitude, control latency, and exposure-tiered baselining, plus formal metric definitions, six falsifiable propositions with stated tests, and a longitudinal validation design. **Implications:** Programs should report volatility as a weighted, cause-classified time series, audit trace recall before trusting impact estimates, and route each change to a review tier by exposure.

Keywords: requirements volatility, change impact analysis, requirements traceability, change control board, baselining, regression test selection, enterprise software

DOI: 10.21590/ijed.16.01.06

1 Introduction

Enterprise software programs, such as enterprise resource planning (ERP) rollouts, core banking replacements, customer relationship platforms, and integration layers that connect dozens of legacy systems, are built against requirements that keep moving. Business units reorganize, regulators publish new rules, vendors release new versions of the platforms being configured, and users discover what they need only after they have seen an early build. Requirements volatility, defined here as the rate and extent of additions, modifications, and deletions to an agreed requirements baseline over a defined interval, is therefore a structural property of enterprise delivery rather than a symptom of poor analysis. Reviews of ERP adoption and procurement strategy describe the same pattern from the business side: configuration choices are revisited as process owners learn what the package does, and each revision propagates into data migration, interfaces, reports, and training material (Okonkwo et al., 2024d; Okonkwo et al., 2024e). The practical question is not whether requirements will change but whether an organization can see change early, estimate what it will touch, decide on it quickly, and verify its consequences before release (Gotel et al., 1994; Edivri et al., 2024).

Practice has not kept pace with this framing. Many programs still report volatility as a monthly count of change requests, a figure that treats a corrected label on a report and a redesigned tax-calculation rule as equivalent events. The same programs frequently route every change through a single change control board (CCB) whose weekly cadence imposes identical latency on trivial and consequential requests. Two failure modes follow. When the count is used as a quality signal, teams suppress or batch changes to keep the number low, so the baseline drifts away from what stakeholders actually intend. When the board is



used as the only safeguard, low-risk work queues behind high-risk work and developers begin to make unrecorded changes, which is the precise behavior that traceability and baselining were designed to prevent (Ramesh et al., 2001; ISO/IEC/IEEE, 2018). Studies of KPI-driven governance in other enterprise settings report a similar effect: an indicator that is not tied to a decision rule tends to be managed as a number rather than used as evidence (Sakyi et al., 2022b; Tonoyan et al., 2024b).

The scholarly literature is also fragmented. Requirements engineering research defines volatility measures but rarely connects them to impact analysis. Traceability research has produced sophisticated link-recovery methods, yet it usually evaluates them by precision and recall against an answer set rather than by whether they changed a change-approval decision or prevented a defect (Borg et al., 2014; De Lucia et al., 2007). Regression-testing research optimizes test selection and prioritization, typically taking the set of changed code as given and ignoring the requirement-level change that caused it (Li et al., 2007; Catal et al., 2013). Governance research discusses boards, approval rights, and accountability in general terms without specifying how scrutiny should scale with the exposure of an individual change (Akhigbe et al., 2025; Adebayo et al., 2025). The gap this article addresses is the absence of an integrated, testable account that runs from how volatility is measured, through how its reach is estimated, to how control effort is allocated and how cost, schedule, and defect outcomes respond (Wedraogo et al., 2024; Medon et al., 2024).

1.1 Research Questions and Contributions

Four research questions structure the review. RQ1 asks which volatility metrics capture the properties of change that plausibly drive cost, schedule, and defect outcomes, and how those metrics should be defined so that they are comparable across projects and periods. RQ2 asks how traceability supports the estimation of change propagation and what accuracy is required before an impact estimate can guide approval and re-verification. RQ3 asks how baselining practice and change control boards should be calibrated so that scrutiny is proportionate to the exposure of each change. RQ4 asks what the evidence shows about the effects of volatility and its management on cost, schedule, and delivered quality, and where that evidence conflicts. The questions follow the life of a single change from the moment it is logged to the moment its consequences are observed in production, which is the unit of analysis adopted throughout (Easterbrook et al., 2008; Runeson et al., 2009).

The article makes four contributions. First, it consolidates volatility measurement into a small family of formally defined metrics (a change rate normalized by baseline size, a timing-weighted rate, a size-weighted magnitude, and a baseline drift index) and states the data conditions under which each is valid. Second, it reframes change impact analysis as an estimation problem whose error is measurable, so that impact-set recall and precision become governance quantities rather than research benchmarks. Third, it proposes exposure-tiered change control, in which the review route and the latency target for each change are derived from its estimated reach, amplitude, and phase, and it assembles these elements into the TRACE-V framework (timing-weighted rate, reach, amplitude, control latency, and exposure-tiered baselining). Fourth, it states six falsifiable propositions with explicit tests and disconfirming results, together with a validity plan for a longitudinal change-event study. The remainder of the paper sets out the theoretical grounding (Section 2), the review protocol (Section 3), the synthesis on measurement, propagation, governance, and outcomes (Sections 4 to 7), the framework and propositions (Section 8), the discussion (Section 9), and limitations (Section 10) (Hevner et al., 2004; Wieringa, 2014).

2 Theoretical Grounding

Three bodies of theory explain why the same volume of requirements change can be harmless in one program and ruinous in another. Each contributes a distinct mechanism, and the framework developed in Section 8 depends on all three. The organizational information-processing view explains why change must be absorbed by coordination capacity; the economics of deferred commitment explains why the timing of a change matters more than its existence; and network models of design dependency explain why a small change can reach a large part of the system. Taken together they predict that harm depends on when a change arrives, how far it propagates, and how fast the organization can decide on it, rather than on how many changes arrive (Akin-Oluyomi et al., 2023b; Falemi et al., 2024).

2.1 Volatility as an Information-Processing Load

The information-processing view of organizations holds that uncertainty, measured as the gap between the information required to perform a task and the information already held, must be matched by coordination mechanisms: rules, hierarchy, lateral relations, and slack. A requirements change is a unit of new information that invalidates part of what the team believed.



Absorbing it requires that the right people learn of it, that its consequences are worked out, and that plans and artifacts are revised. When the arrival rate of changes exceeds the processing capacity of these mechanisms, work accumulates in queues, decisions are made on stale information, and inconsistencies accumulate between artifacts. Studies of cross-shift handovers in continuous operations document the same mechanism in a different setting: information that is not transferred through a standardized channel is lost at the boundary, and the loss surfaces later as an operational error (Ekelemu, 2025b). The implication for software is that a CCB, a traceability matrix, and a baseline are coordination mechanisms, and their adequacy should be judged by whether processing capacity keeps pace with the arrival of change (Uchechi et al., 2018).

This lens also clarifies why generic process maturity is a weak predictor. A mature process with a fixed weekly board has a fixed capacity; if arrivals double during a regulatory deadline, latency rises regardless of maturity. The relevant variables are therefore the arrival rate, the service rate, and the variability of each, which queueing theory relates through Little's law, $L = \lambda W$, where L is the number of change requests awaiting a decision, λ is the arrival rate, and W is the mean time in the system. A program that halves W at constant λ halves the backlog of undecided changes, which is the backlog within which unrecorded workarounds tend to appear (Tonoyan et al., 2022b; Anioke et al., 2023a).

2.2 Deferred Commitment and the Timing of Change

The second mechanism concerns timing. A requirement that changes before design has begun invalidates little; the same change after integration testing invalidates design decisions, code, test cases, data migration scripts, and user documentation. Classic cost-of-change reasoning captures this escalation, and it motivates weighting each change by the phase in which it occurs. Real-options reasoning adds a counterweight: early commitment forecloses learning, so there is value in deferring decisions on uncertain requirements until more information is available. Iterative delivery methods exploit this option by keeping the committed scope small and revisiting it each iteration, which is one reason why systematic reviews of agile development report that change is tolerated more easily when commitments are short (Dybå et al., 2008). The two arguments are reconcilable. What matters is the amount of committed downstream work that a change invalidates, and that quantity rises with the phase of the affected items, not with calendar time as such (Ebhojie et al., 2023b; Aliliele et al., 2024).

A useful analogy comes from risk-based regulatory approval, where decision models trade the speed of approval against the quality of evidence and the severity of potential harm, and allow faster routes where exposure is low (Eze et al., 2024b). In requirements management the equivalent is a routing rule that permits a low-exposure change to be accepted quickly while reserving deliberation for changes that invalidate expensive downstream work. The timing weight introduced in Section 4 operationalizes this reasoning. It allows a program to tell whether volatility is concentrated in early, cheap phases (which is healthy learning) or in late, expensive ones (which is a warning), even when the raw count is identical (Eze et al., 2022b).

2.3 Dependency Networks and Change Propagation

The third mechanism concerns propagation. Software artifacts form a directed network in which requirements are refined into design elements, realized in code units, and verified by test cases, with further horizontal dependencies among requirements that share data, interfaces, or quality constraints. A change to one node propagates along the edges of this network. Traceability research has long argued that the network must be recorded explicitly because it cannot be reconstructed reliably after the fact, and that pre-requirements traceability (the origin and rationale of a requirement) is as important as post-requirements traceability (Gotel et al., 1994). Reference models of traceability distinguish low-end users, who record simple satisfiers and verifies links, from high-end users, who record rationale, derivation, and dependency links that support impact analysis (Ramesh et al., 2001). The theoretical implication is that the harm of a change depends on the out-degree and depth of its reachable set, and that an organization can only manage what its link network lets it see (Cleland-Huang et al., 2012; Winkler et al., 2010).

Network reasoning also explains amplification. In security engineering, amplification attacks exploit protocols in which a small request elicits a much larger response (Jimoh et al., 2024). Highly coupled requirements behave similarly: a change to a shared data definition, such as the structure of a customer identifier, can trigger changes in every interface that carries it. Non-functional requirements are a particular source of amplification because security, performance, and privacy constraints cut across many functional requirements at once, and their definitions are themselves contested (Mairiza et al., 2010). Privacy requirements in role-based platforms and bring-your-own-device settings illustrate the point, since a revised access rule touches every screen, report, and integration that exposes the affected data (Badmus et al., 2024; Ladapo et al., 2023b).



3 Review Design and Protocol

The review is integrative rather than systematic in the strict sense. It combines empirical studies, systematic reviews and mapping studies, standards, and fully specified conceptual models in order to develop a framework, and it does not pool effect sizes. This choice reflects the state of the evidence: volatility studies use incompatible metrics, many are single-organization case studies, and some of the most relevant knowledge sits in standards and method texts rather than in controlled studies. The protocol nevertheless follows the reporting logic of the PRISMA 2020 statement for its search, eligibility, and selection stages, so that each decision is documented and could be repeated (Page et al., 2021). Design science guidance informed the framework-building stage, in which constructs are proposed, justified against the synthesized evidence, and exposed to explicit tests (Hevner et al., 2004; Wieringa, 2014). Table 1 summarizes the protocol, and Figure 1 shows the stages.

3.1 Sources and Search Strategy

Six bibliographic sources were searched: IEEE Xplore, the ACM Digital Library, Scopus, Web of Science Core Collection, SpringerLink, and ScienceDirect. The publication window ran from 1990, the period in which requirements traceability emerged as a research problem, to 2025, the year of this review. The core search string combined three concept blocks with Boolean operators: ("requirements volatility" OR "requirements change" OR "requirements creep" OR "scope change" OR "requirements evolution") AND ("impact analysis" OR traceability OR "change control" OR baseline* OR "change propagation") AND (cost OR schedule OR effort OR defect* OR quality OR rework). A second string targeted regression-test selection and prioritization triggered by requirement change, and a third targeted enterprise contexts (ERP, package implementation, systems integration). Standards bodies were searched directly for requirements engineering and software testing process standards, and backward and forward snowballing was applied to the most cited traceability and regression-testing reviews (Borg et al., 2014; Catal et al., 2013). Governance and analytics literature from adjacent enterprise domains was admitted where it addressed measurement design, decision rights, or monitoring of indicators, because volatility governance borrows these mechanisms (Oluwo et al., 2025; Adesuyi et al., 2023).

3.2 Eligibility, Screening, and Appraisal

Four inclusion rules applied. E1 required that a source define or operationalize at least one of the review constructs (a volatility metric, a propagation or impact-analysis method, a baselining or change control practice, or a cost, schedule, or defect outcome linked to change). E2 required a software-intensive or enterprise information-systems context, or a transferable governance mechanism stated in terms that could be mapped to that context. E3 required sufficient methodological detail to appraise the claim. E4 required publication in the window. Four exclusion rules applied: X1 excluded sources that named volatility without measuring or defining it; X2 excluded change counts reported without a denominator or observation window; X3 excluded vendor and marketing material; and X4 excluded traceability tools evaluated without any evidence of link accuracy. Screening proceeded from titles and abstracts to full texts, with a pilot sample screened independently by two reviewers to calibrate the rules before the remainder was screened (Easterbrook et al., 2008; Dybå et al., 2008).

Appraisal used a seven-item instrument adapted to the topic. Each source was rated on whether its volatility or impact metric was formally defined, whether the provenance of change records was reported (tool, recording rule, and completeness), whether trace links were validated against an independent answer set, whether a comparator existed (a prior period, a matched project, or an alternative method), whether plausible confounders such as team experience, system size, and contract type were addressed, whether the delivery model and domain were reported, and whether the strength of the claim was proportionate to the design. Case studies were appraised against established guidance for software engineering case research (Runeson et al., 2009). Standards were treated as specifications of expected controls rather than as evidence that those controls improve outcomes, a distinction that matters because much of the practitioner literature cites standards as if compliance were itself a result (ISO/IEC/IEEE, 2018; Atta et al., 2022).

3.3 Synthesis Logic

Synthesis followed a mechanism, context, and outcome logic. Each included source was coded for the volatility metric used, the propagation model assumed, the control mechanism studied, the delivery context, and the outcome reported. Findings were then grouped by mechanism and compared across contexts. Where studies disagreed, the disagreement was retained and the review asked whether it could be explained by differences in metric definition, recording discipline, delivery model, or design,



rather than averaging it away. This adjudication step is the core of the synthesis in Sections 4 to 7. Adjacent-domain sources were used only for the general mechanism they illustrate (for example, how leading indicators are selected, or how variance is attributed to causes) and never as evidence about software outcomes (Obriki et al., 2025; Arumosoye et al., 2019). The review does not report a count of screened records as a finding, because the purpose was framework construction and the heterogeneity of sources makes such counts uninformative about evidential weight (Sanni et al., 2023).

Table 1 sets out each protocol element, the specification used, and the reason the element matters for research on requirements volatility in particular.

Table 1. Review protocol for requirements volatility, change propagation, and change control

Protocol element	Specification	Reason it matters for volatility research
Sources	IEEE Xplore, ACM DL, Scopus, Web of Science, SpringerLink, ScienceDirect; standards bodies; snowballing	Traceability and regression-testing work is concentrated in software venues; governance work is spread across enterprise journals
Window	1990 to 2025	Covers the emergence of traceability as a research problem through current learning-based link recovery
Search blocks	Volatility terms AND impact or control terms AND cost, schedule, or defect terms	Forces each hit to connect change, its handling, and a consequence
Inclusion (E1 to E4)	Defined construct; software or transferable context; appraisable method; in window	Excludes sources that use volatility as a label without measurement
Exclusion (X1 to X4)	Undefined volatility; counts without denominator or window; vendor material; unvalidated trace tools	Counts without denominators cannot be compared across projects or periods
Appraisal	Seven items: metric definition, record provenance, link validation, comparator, confounders, context, proportionality	Change logs are often incomplete, so provenance determines whether a metric can be trusted
Synthesis	Mechanism, context, outcome matrix; conflicts adjudicated by metric, recording, delivery model, and design	Apparent contradictions often reflect different definitions of the same construct

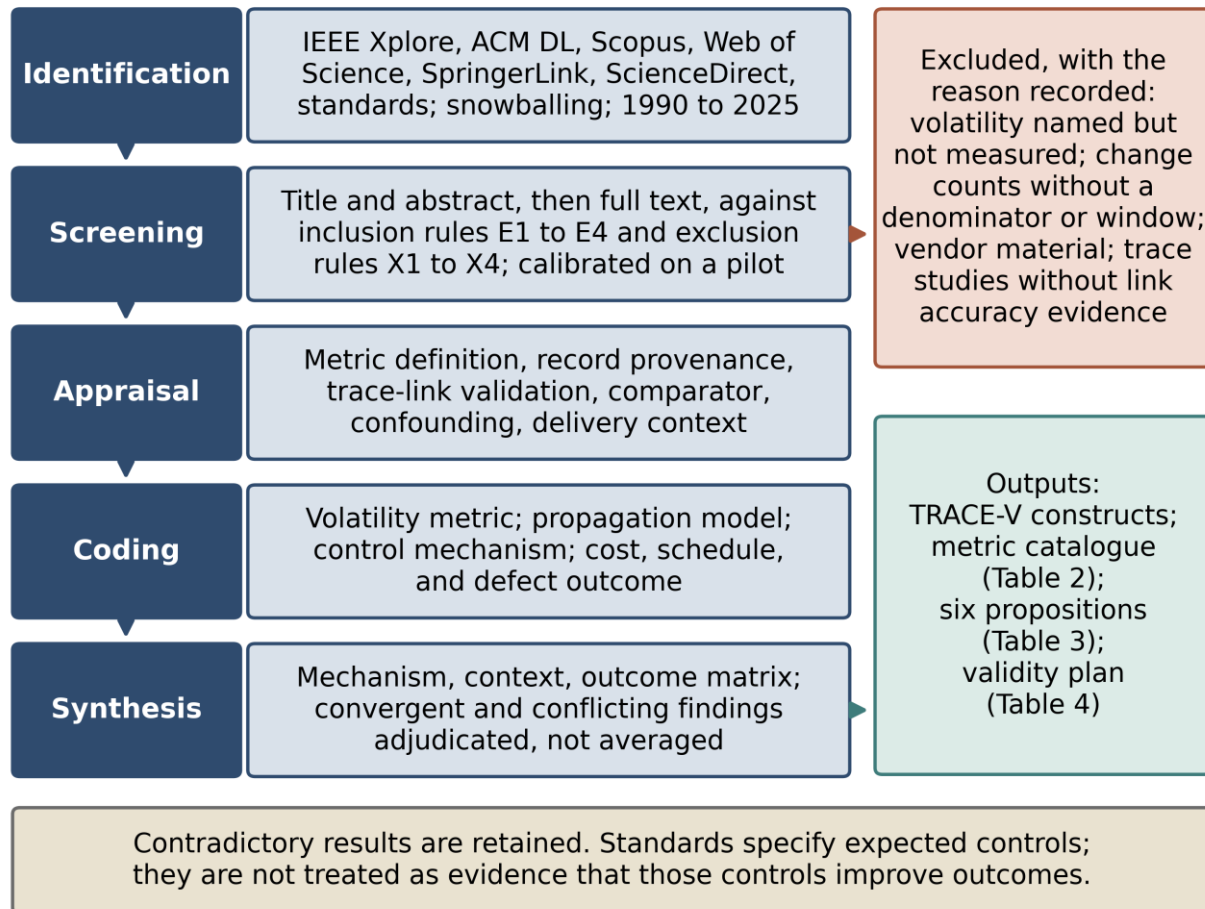


Figure 1. Review protocol stages, exclusion rules, and outputs for the synthesis on requirements volatility

Figure 1 traces the protocol from identification to synthesis. Exclusion reasons are recorded at the screening stage, and the outputs of synthesis are the constructs, the metric catalogue, the propositions, and the validity plan presented later in the article. The bottom panel restates the rule that standards specify controls but do not by themselves demonstrate that controls work, which guards against a common inferential shortcut in the practitioner literature on change management (Eyetsemitan et al., 2023a; Amarahobu et al., 2025a).

4 Measuring Requirements Volatility

Measurement is where most volatility programs go wrong, because the simplest metric (the number of change requests in a month) is also the least informative. This section defines a small family of metrics, each of which captures a distinct property of change, and states the data conditions under which each is valid. The metrics are summarized in Table 2. The general principle, familiar from KPI design in other large organizations, is that an indicator is only useful when its unit, denominator, window, and owner are fixed in advance and when a threshold breach triggers a defined action (Sakyi et al., 2022b; Edivri et al., 2024).

4.1 Change Rate Normalized by Baseline Size

Let B_t denote the number of requirements in the approved baseline at the start of interval t , and let A_t , M_t , and D_t denote the numbers of requirements added, modified, and deleted during the interval. The requirements change rate (RCR) is then defined as follows.



$$RCR_t = \frac{A_t + M_t + D_t}{B_t} \quad (1)$$

Normalization by B_t allows comparison between a program with 300 requirements and one with 3,000, and fixing the interval (typically an iteration or a calendar month) allows a series to be plotted. The cumulative form, the sum of RCR over intervals since the first baseline, indicates how much of the original agreement has been revised. Two refinements are necessary. First, modifications should be counted at the level of the requirement that carries the change rather than the change request, because one request can modify many requirements and many requests can target one requirement. Second, the counting rule for splits and merges must be fixed; splitting one requirement into three is otherwise recorded as two additions and one modification, which inflates the rate. The same problem appears in any predictive monitoring system that depends on event logs: unless the event definition is stable, a trend in the metric may be a trend in recording practice (Adeyelu, 2018; Atakpa et al., 2023).

RCR remains a frequency measure. It says nothing about when changes arrive, how large they are, or why they occur, and empirical programs that report only RCR tend to find weak or inconsistent associations with outcomes. The reason is not that volatility is harmless but that RCR mixes benign clarification with disruptive late change. A program in which ninety percent of modifications are wording clarifications during analysis will show the same RCR as a program in which ninety percent are changes to integration logic after system testing. Measurement guidance from predictive analytics in other domains makes the same point in general terms: a model input that aggregates heterogeneous events loses the signal that distinguishes the cases that matter (Aliliele et al., 2023c; Ogundairo et al., 2023a).

4.2 Timing Weights and Magnitude

The timing-weighted rate (TWR) weights each change by the phase of the most advanced artifact it invalidates. Let c index the changes in interval t , let $\phi(c)$ be the lifecycle phase reached by the affected requirement at the time of change, and let $w(\phi)$ be a phase weight that increases monotonically from analysis to acceptance testing. Then:

$$TWR_t = \frac{\sum_c w(\phi(c))}{B_t} \quad (2)$$

Weights can be set from the program's own rework data, for example as the ratio of the median rework effort for changes at each phase to the median at the analysis phase, which keeps the metric grounded in local cost rather than in textbook multipliers. The size-weighted magnitude (SWM) captures how much of each requirement changes. Where function-point or story-point estimates exist, SWM_t is the sum of the absolute differences in size between the old and new versions of each affected requirement, divided by the total baseline size. Where size estimates are unavailable, a text-based proxy such as the normalized edit distance between successive versions can be used, although it overstates cosmetic rewrites and understates small wording changes that alter business rules. Neither proxy is adequate for non-functional requirements, whose magnitude lies in the threshold they impose (a response time, an encryption standard, a retention period), so a change from a five-second to a two-second response time should be treated as high magnitude regardless of the edit distance (Mairiza et al., 2010; Zhu et al., 1997).

A worked illustration shows why these weights matter. Consider an illustrative ERP finance module with 800 baselined requirements. In month one, 40 requirements change, all during analysis workshops, giving an RCR of 0.05. In month six, 20 requirements change, of which twelve affect items already in system integration testing. RCR has halved, which a count-based dashboard would present as stabilization. With phase weights of 1 for analysis and 6 for integration testing, however, TWR rises from 0.05 to $(8 \times 1 + 12 \times 6) / 800$, or 0.10. The timing-weighted series correctly signals that exposure has doubled even though activity has fallen. Variance analytics in manufacturing cost control rely on the same decomposition, separating volume variance from rate variance so that a falling volume does not mask a rising unit cost (Aliliele et al., 2024; Ebhojie et al., 2023b).

4.3 Classifying the Cause of Change

Cause classification separates volatility that reflects healthy learning from volatility that reflects defects in earlier work. A workable taxonomy for enterprise programs distinguishes five causes: clarification of an ambiguous requirement, correction of an erroneous requirement, external change (regulatory, contractual, or vendor-driven), internal business change



(reorganization, new product, revised process), and technical discovery (a platform constraint or integration limitation found during design or build). Each cause has a different governance implication. A high share of corrections points to weak elicitation or validation; a high share of external change points to the need for modular isolation of regulated rules; and a high share of technical discovery points to architecture spikes that should have happened earlier. Business analysis frameworks that convert ambiguous business problems into structured, automation-ready requirement statements act on the clarification category directly, because they force ambiguity to be resolved before a requirement enters the baseline (Alieme et al., 2024). Regulatory change is a growing driver in enterprise settings, as organizational readiness studies for new artificial intelligence regulation show, because obligations are published in stages and their interpretation matures over time (Ominyi et al., 2024b; Ominyi et al., 2025a).

Cause coding is prone to attribution bias. Analysts who log a change may prefer to classify it as external rather than as a correction of their own earlier work, and business owners may prefer the reverse. Research on attribution modeling in marketing, which faces the problem of assigning credit across many touchpoints, shows that the chosen rule changes the conclusion and that rule-based attribution should be validated against an independent reference (Sanni et al., 2022). Cognitive bias research in audit and risk management supports the same caution, since self-serving attribution is systematic rather than random (Fobellah, 2025a). A practical safeguard is dual coding of a sample of changes by the requester and by an independent analyst, with agreement reported as Cohen's kappa, and periodic recalibration when agreement falls.

4.4 Baseline Drift and the Quality of Change Records

All of the preceding metrics assume that changes are recorded. In practice, a proportion of change is made informally, through conversations, e-mail, or direct edits to user stories, and never passes through a change request. The baseline drift index (BDI) estimates this unrecorded volatility by comparing the approved baseline with the implemented system at a release point. Let R_{rel} be the set of requirements whose accepted implementation differs from the approved text at release, and let R_{cr} be the subset of R_{rel} for which an approved change request exists. Then:

$$BDI = \frac{|R_{rel} \setminus R_{cr}|}{B} \quad (3)$$

A BDI near zero indicates that the recorded change stream is complete; a rising BDI indicates that control has been bypassed, and it is the most direct evidence that a CCB is imposing more latency than teams will accept. BDI is costly to measure because it requires acceptance evidence to be compared with baseline text, but sampling a random subset of requirements at each release gives an unbiased estimate at modest cost. The quality of change records matters for every downstream analysis. Alarm-management research in process industries reports that standing and inhibited alarms erode the reliability of the whole alarm system, because operators learn to ignore signals that no longer mean what they say (Ekelemu, 2025a). An unmaintained change log has the same effect on volatility metrics, and loss-detection programs in liquefaction systems show that systematic reconciliation between recorded and actual flows is needed to locate unrecorded losses (Ekelemu et al., 2025b).

Table 2 brings these metrics together with the impact-analysis and latency measures developed in Sections 5 and 6, showing for each the definition, its data source, its known weakness, and the governance use it supports.

Table 2. Catalogue of volatility, propagation, and control metrics with data sources, weaknesses, and governance use

Metric	Definition	Data source	Known weakness	Governance use
Requirements change rate (RCR)	$(A + M + D) / B$ per interval, Eq. (1)	Requirements tool history; change log	Treats all changes as equal; sensitive to split and merge rules	Trend context only; never a stand-alone target
Timing-weighted rate (TWR)	Phase-weighted sum of changes / B, Eq. (2)	Change log joined to artifact phase status	Weights must be calibrated locally	Early warning of late, expensive change
Size-weighted magnitude (SWM)	Absolute size delta of changed items / total size	Function or story points; version diffs	Weak for non-functional requirements	Re-estimation trigger and contract change pricing
Cause mix	Share of changes by cause class	Coded change requests	Attribution bias in self-coding	Targets elicitation, modularity, or architecture remedies
Baseline drift index (BDI)	Unrecorded differences at release / B, Eq. (3)	Sampled acceptance evidence versus baseline	Costly; requires sampling	Detects control bypass and excessive latency



Metric	Definition	Data source	Known weakness	Governance use
Impact-set recall and precision	$ EIS \cap AIS / AIS $ and $ EIS \cap AIS / EIS $, Eq. (4)	Estimated set at approval; actual set at closure	Actual set is known only after the fact	Decides whether trace estimates can drive approval and testing
Decision latency (p50, p85)	Time from logging to decision, by tier	Workflow timestamps	Averages hide the tail; clock-stopping rules vary	Board capacity and tier cut-off calibration
Propagation defect ratio	Defects traced to an unanalyzed impact / all change-related defects	Defect root-cause records	Depends on root-cause coding quality	Tests whether impact analysis prevents escapes

5 Change Impact Analysis and Traceability-Based Propagation

Impact analysis answers the question that every change decision depends on: what else will this change touch? The answer determines the cost estimate, the reviewers who must be consulted, and the tests that must be rerun. In enterprise programs the question is especially difficult because the affected items span organizational boundaries (configuration owned by one team, interfaces owned by another, reports owned by a third) and because package customizations interact with vendor code that the program does not control. This section examines how trace links are created and maintained, how propagation is estimated from them, and how the estimated impact set is converted into re-verification effort (Cleland-Huang et al., 2012; Aliliele et al., 2024a).

5.1 Trace Models and Link Recovery

A trace model specifies which artifact types exist and which link types connect them. Minimal models record only that a design element satisfies a requirement and that a test verifies it. Richer models add dependency links between requirements, derivation links from business rules to system requirements, and rationale links to the decisions that shaped them. The survey of traceability in requirements engineering and model-driven development shows that richer models support more precise impact analysis but are costlier to maintain, and that model-driven approaches can generate some links automatically as a by-product of transformation (Winkler et al., 2010). For enterprise package implementations the most valuable additions are links from requirements to configuration objects, interface specifications, and data-migration mappings, because these are the artifacts most often missed when impact is estimated by inspection alone (Ramesh et al., 2001; Aliliele et al., 2025b).

Because manually created links decay, information retrieval methods have been developed to recover candidate links by textual similarity. Latent semantic indexing applied to artifact management systems showed that recovered links can reach useful recall, but that precision falls quickly as recall rises, so analysts must vet long candidate lists (De Lucia et al., 2007). A systematic mapping of a decade of information retrieval traceability research found that most evaluations used small, academic datasets and measured precision and recall at fixed cut points, with few studies examining whether recovered links helped engineers perform a task (Borg et al., 2014). More recent work applies machine learning and language models to the same problem, and reviews of artificial intelligence in software testing report similar gains and similar caveats about evaluation realism (Borah et al., 2022). The adjudication for practice is that automated recovery is a maintenance aid, not a substitute for recording links when artifacts are created, and that its value should be judged by the recall of the impact sets it supports.

5.2 Estimating the Impact Set

Impact analysis research distinguishes the starting impact set (SIS), the items directly changed; the estimated impact set (EIS), the items predicted to be affected; and the actual impact set (AIS), the items that were in fact changed or needed to be. The quality of an impact estimate is captured by its recall and precision relative to the actual set:

$$\text{Recall} = \frac{|EIS \cap AIS|}{|AIS|}, \quad \text{Precision} = \frac{|EIS \cap AIS|}{|EIS|} \quad (4)$$

Trace-based estimation computes the EIS as the set of items reachable from the SIS through recorded links, usually with a depth limit, and sometimes with link-type filters (for example, following dependency links between requirements but only satisfies links downward). Figure 2 illustrates the resulting sets. Recall failures arise when a real dependency was never linked

(the dashed chain in the figure); precision failures arise when a link exists but the change does not in fact affect the reached item. The two errors have asymmetric consequences. Low precision wastes review and testing effort, which raises cost and latency; low recall lets affected items escape analysis, which raises the probability of propagation defects in production. For decisions about safety, finance, or regulated data, recall is the binding constraint, a conclusion that echoes risk-based testing guidance in which coverage of high-consequence items is prioritized over efficiency (ISO/IEC/IEEE, 2018; Myers et al., 2011).

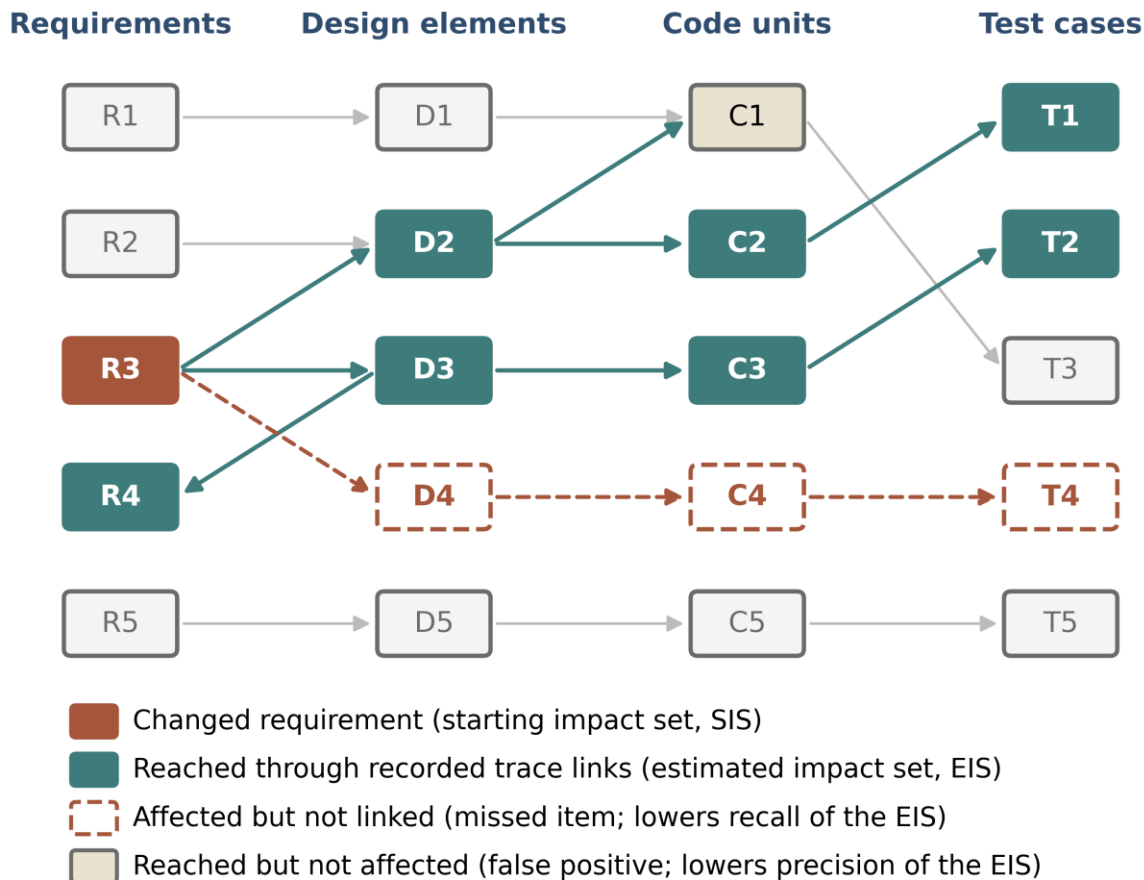


Figure 2. Trace-based propagation of a requirement change, showing the starting, estimated, and missed impact sets

Figure 2 shows a change to requirement R3. Recorded links carry the estimate to two design elements, two code units, and two tests, and horizontally to R4 through a shared design element. A false positive (C1) is reached but unaffected. The dashed chain from D4 to T4 is an actual dependency with no recorded link, so it is missed; if T4 is not rerun, a propagation defect can escape. The figure makes concrete why recall must be audited: the estimate looks complete to anyone who sees only the recorded graph. Digital twin research makes a comparable argument for physical infrastructure, where a fault-detection model is only as good as the fidelity of the dependency model it reasons over (Shittu et al., 2023; Dagodzo et al., 2020).

Because the AIS is known only after the change is implemented, recall cannot be measured at the moment of approval. It can, however, be measured retrospectively for every closed change and tracked as a program-level estimate of how much the trace network misses. If audited recall over the last fifty changes is 0.7, approvers should treat any EIS as covering roughly seventy percent of what will actually be touched and should add contingency or targeted inspection accordingly. This moves impact-analysis accuracy from a research benchmark to an operational parameter, in the same way that predictive maintenance programs track the false-negative rate of their detectors (Atta et al., 2021; Michael et al., 2023c).

5.3 From Impact Set to Re-Verification

The estimated impact set feeds regression testing. Regression test selection chooses the subset of existing tests that exercise affected items, and test case prioritization orders the selected tests so that faults are revealed early. Prioritization effectiveness is conventionally measured by the average percentage of faults detected (APFD). For a test suite of n tests that reveals m faults, where TF_i is the position in the ordered suite of the first test that reveals fault i :

$$APFD = 1 - \frac{TF_1 + TF_2 + \dots + TF_m}{n \times m} + \frac{1}{2n} \quad (5)$$

Search-based prioritization studies compared greedy, additional-greedy, hill-climbing, and genetic algorithms and found that simple additional-greedy strategies performed competitively for coverage-based objectives, with the relative ranking depending on program and suite characteristics (Li et al., 2007). A systematic mapping of test case prioritization found that most techniques are coverage-based and that requirement-based and history-based approaches are less studied, despite their relevance to change-driven testing (Catal et al., 2013). Coverage adequacy criteria provide the underlying rationale: a test suite adequate for the original program is not adequate for a changed one unless it exercises the changed elements and their dependents (Zhu et al., 1997; Ammann et al., 2016).

Requirement-level change adds information that code-level selection lacks. A change to a business rule may require new tests rather than reruns, because the old tests encode the old expectation; a change to a non-functional requirement may require performance or security tests that are not linked to any code unit. Unit testing surveys report that developers often lack time to maintain tests alongside code, which suggests that test obsolescence accumulates exactly where volatility is highest (Daka et al., 2014). Classical testing texts therefore recommend that test design be revisited, not merely rerun, when requirements change (Kaner et al., 1999; Myers et al., 2011). User acceptance testing in lean deployments shows the same need at the business level: acceptance criteria must be re-derived from the changed requirement, or acceptance becomes a confirmation of the superseded baseline (Eyetssemitan et al., 2023b).

6 Baselining and Change Control Boards

Baselines and boards are the formal instruments of volatility governance. A baseline fixes an agreed version of the requirements against which change is measured, contracts are priced, and progress is reported. A board holds the decision right to alter that version. Both instruments are specified in requirements engineering and configuration management standards, and both are routinely criticized as bureaucratic. The synthesis in this section asks under what conditions each instrument adds value, and how their design should vary with the exposure of the change and the delivery model (ISO/IEC/IEEE, 2018; Akhigbe et al., 2025). Enterprise programs complicate the picture because they are multi-party: a systems integrator, several package vendors, and internal teams each own part of the baseline, and governance research on multi-contract industrial projects shows that performance depends on how obligations and decision rights are allocated across contractors rather than on the presence of a central committee alone (Arumosoye et al., 2020). Studies of project governance in multinational infrastructure programs make a comparable point, locating delivery performance in how governance structures coordinate distributed teams and suppliers across jurisdictions (Amayo et al., 2023a). Readiness research adds that control structures must be sized for surge conditions, not average load, which in this context means regulatory deadlines and go-live periods when change arrivals peak (Arumosoye et al., 2023).

6.1 Baseline Semantics and Cadence

A baseline is only meaningful if its semantics are clear: what is included, who agreed to it, and what happens when it is revised. The life-cycle requirements standard describes baselined requirements as those that have been validated and placed under configuration control, so that subsequent changes are recorded and evaluated (ISO/IEC/IEEE, 2018). Two failure modes recur. A frozen baseline that is never revised diverges from stakeholder intent, so that the program delivers what was agreed rather than what is needed; a baseline that is revised continuously without versioning loses its function as a reference point, so that volatility cannot be measured at all. Exposure-tiered baselining resolves the tension by applying different cadences to different parts of the baseline. Interface contracts, data definitions, and regulated rules are placed under strict control with explicit versioning, while user-interface details and report layouts are rebaselined every iteration with light review. Studies of



data-sensitivity classification in enterprises use a similar tiering logic, applying stronger controls to data whose exposure is greater (Aliliele et al., 2024a; Mbonu et al., 2018).

Cadence should also follow the delivery model. In plan-driven contracts, the baseline carries commercial weight, and changes after signature trigger price and schedule renegotiation, so strict control of the contracted scope is appropriate. In iterative delivery the product backlog is the living baseline and the iteration commitment is the frozen slice, so the relevant control point is the iteration boundary. Hybrid enterprise programs, which are common in ERP work, need both: a contractual baseline for the program scope and an iteration baseline for detailed requirements, with a defined rule for when detailed change escalates to a contractual change. Cross-functional contract education research reports fewer contract errors when engineering teams understand which changes carry contractual consequences, which supports making the escalation rule explicit to the people who log changes (Anichukwueze et al., 2022; Dybå et al., 2008).

6.2 Decision Rights and Board Latency

A change control board adds value by bringing the right expertise to a decision and by preventing unilateral changes that impose costs on others. It adds cost through latency: every day a change waits for a decision is a day in which the team either stalls, works on assumptions that may be rejected, or makes the change informally. Latency should be measured as the distribution of time from logging to decision, reported at the median and at a tail percentile such as the 85th, because averages hide the long waits that drive bypass behavior. By Little's law, a board that meets weekly and processes a fixed number of items per session has a minimum mean latency of half a week even with no backlog, and any surge in arrivals increases latency linearly with the backlog (Tonoyan et al., 2024b; Sanni et al., 2025).

Process mining of workflow logs offers a direct way to observe this behavior. Problem-oriented process mining reconstructs the actual path of each case from event timestamps and reveals loops, rework, and bypasses that the documented process does not show (Sanni et al., 2025). Applied to change workflows, it can reveal how often changes are implemented before approval, how often they are returned for more impact analysis, and where they wait. The governance literature on enterprise transparency argues that decision rights should be explicit, accountable, and matched to competence, which in the change context means that routine decisions should sit with the people who own the affected component and that the board should reserve its time for decisions that cross ownership boundaries (Akhigbe et al., 2025; Adegbite et al., 2025). Accountability frameworks for large public construction programs make the same distinction between local cost control and program-level approval (Aminu-Ibrahim et al., 2025a).

6.3 Exposure-Tiered Routing

Exposure-tiered routing assigns each change to a review tier according to an exposure score computed from its estimated reach, amplitude, phase, and criticality. A simple multiplicative form is:

$$E(c) = |EIS(c)| \times SWM(c) \times w(\varphi(c)) \times \kappa(c) \quad (6)$$

where $\kappa(c)$ is a criticality multiplier for items that carry regulatory, financial, safety, or privacy obligations. The score is then mapped to tiers with fixed cut-offs, each with its own decision right and latency target, as shown in Figure 3. Tier 0 covers editorial changes with no reach beyond the item and is approved by the author with peer review on the same day. Tier 1 covers changes whose reach stays within one component and is approved by the product owner within two working days. Tier 2 covers changes that touch an interface, a shared data model, or a shared service and goes to the board at its next cycle. Tier 3 covers changes with scope, contract, regulatory, or release-date consequences and goes to a steering group convened as needed. Risk-based approval models in pharmaceutical regulation use the same principle of matching the depth of review to the severity of potential harm (Eze et al., 2024b; Obogo et al., 2019b).

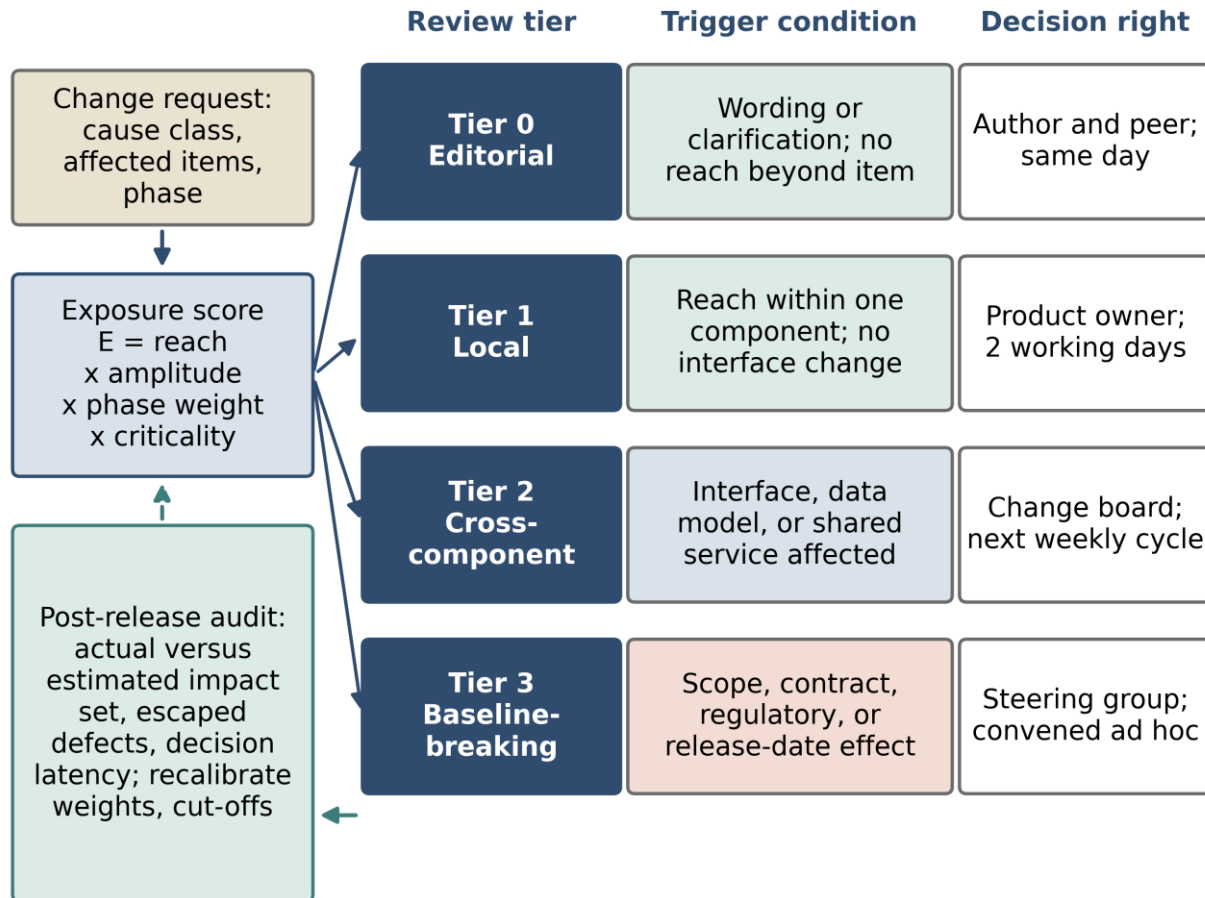


Figure 3. Exposure-tiered change control, with trigger conditions, decision rights, latency targets, and the audit feedback loop

Figure 3 also shows the feedback loop that keeps the tiering honest. After release, each change is audited: the actual impact set is compared with the estimate, escaped propagation defects are traced to their changes, and decision latency is compared with the tier target. If Tier 1 changes produce a disproportionate share of escaped defects, the cut-off is too permissive; if Tier 2 latency routinely exceeds its target while few Tier 2 changes cause defects, the board is over-scrutinizing. Safety-management research on leading and lagging indicators describes the same discipline of adjusting control thresholds from observed outcomes rather than from intuition (Obriki et al., 2018; Arumosoye et al., 2021). A known risk of tiering is gaming, in which requesters understate reach to secure a faster route; the audit of estimated against actual impact sets is the control for that risk, because systematic understatement becomes visible as low recall concentrated among particular requesters (Ahmed et al., 2021).

7 Effects on Cost, Schedule, and Defects

The claim that volatility harms project outcomes is widely repeated, but the evidence behind it is uneven, and some findings appear to conflict. This section compares the evidence on cost, schedule, and defects and adjudicates the conflicts by asking whether they reflect differences in metric, recording, delivery model, or study design. The conclusion anticipates the framework: harm is associated with timing, magnitude, and coupling more than with frequency, and the association is moderated by how well propagation is analyzed and how quickly decisions are taken (Runeson et al., 2009; Easterbrook et al., 2008). The outcomes are treated separately because they respond to different parts of the mechanism, and because a program can improve one while degrading another, for example by holding cost through scope deferral while schedule and delivered



value suffer. Work on operational discipline in environmental risk control makes the general point that outcome measures must be read together, since a gain in one indicator can be purchased with a hidden loss in another (Arumosoye et al., 2025).

7.1 Cost and Schedule

Studies that measure volatility as a frequency often report positive correlations with effort overrun and schedule slip, but the correlations vary widely across samples. Studies that weight change by phase or size tend to report stronger and more consistent associations, which is what the deferred-commitment mechanism predicts: late, large changes invalidate more committed work. The heterogeneity is therefore partly a measurement artifact. There is also a structural reason why frequency can be uncorrelated with cost in iterative programs. When scope is managed as a prioritized backlog with fixed time boxes, change displaces lower-priority work rather than adding effort, so cost stays near plan while delivered scope changes. In that setting the right outcome is not cost variance but value delivered per unit of effort, and a program that reports cost variance alone will conclude, wrongly, that volatility is harmless. Data-led cost governance research in technology-intensive enterprises supports tracking cost and value together for exactly this reason (Adesuyi et al., 2023; Tonoyan et al., 2023).

Schedule effects are more sensitive to decision latency than cost effects are. A change that waits three weeks for a board decision may cost little to implement, but if it sits on the critical path the wait translates directly into slip. Programs with high latency also show a characteristic pattern of end-loaded change, because decisions deferred during build are forced at the release gate. Forecasting research warns that point estimates of schedule impact should be accompanied by uncertainty intervals, since the distribution of change effort is right-skewed and a small number of changes account for most of the overrun (Wedraogo et al., 2024; Tonoyan et al., 2022a). Portfolio-risk models from venture investment offer a related insight: concentration of exposure in a few positions dominates portfolio variance, so monitoring should focus on the tail rather than the mean (Eboh et al., 2023; Eboh et al., 2025b).

7.2 Defects and Propagation Failures

The link from volatility to defects runs through two channels. The first is direct: a changed requirement is implemented incorrectly, often because the change was specified hurriedly. The second is indirect: the change is implemented correctly, but an affected item elsewhere is not updated, so an inconsistency is released. The second channel is the one that impact analysis and traceability are meant to close, and it is the one most clearly associated with late, coupled change. Its signature is a propagation defect, a failure in an item that was not itself changed but depended on one that was. Programs that code defect root causes can compute the propagation defect ratio in Table 2 and test whether it falls when impact-set recall improves. Quality assurance research in pharmaceutical manufacturing argues that assurance should be judged by its effect on supply reliability rather than by the completion of checks, which in this context means judging impact analysis by its effect on escaped defects rather than by the number of impact assessments completed (Asiedu et al., 2024b; Yusuf et al., 2018).

Human-error research adds a qualification. Error causation models in high-risk construction distinguish slips under time pressure from mistakes arising from incorrect mental models, and they show that the remedies differ (Obriki et al., 2020). Direct-channel defects from hurried change specification resemble slips and respond to slack and review; indirect-channel defects from missing links resemble mistakes about system structure and respond to better models, which here means better trace networks. A program that responds to propagation defects only by adding review time will see little improvement, because review cannot find a dependency that nobody knows exists. Studies of hazard identification in urban construction make a comparable point about the limits of inspection when hazards are not represented in the plan being inspected (Obogo et al., 2019; Arumosoye et al., 2022).

7.3 Reconciling Conflicting Findings

Three conflicts deserve explicit adjudication. The first concerns direction of causation. Troubled projects generate more change requests, partly because problems surface as requests for correction, so a correlation between volatility and overrun may reflect reverse causation. Cross-sectional designs cannot separate the two, a limitation that applied research in other fields also notes when associations are drawn from single-time-point data (Amadi et al., 2025). Longitudinal designs that establish whether volatility rises before or after the first signs of overrun are needed, and cause classification helps, since correction-driven change is more likely to be a consequence of trouble than external change is (Onwuzulike et al., 2024).



The second conflict concerns agile evidence. Some studies report that iterative teams absorb volatility with little harm, while others report that iterative enterprise programs suffer from uncontrolled scope growth. The adjudication is that iteration changes where volatility lands, not whether it matters. Change inside an iteration's uncommitted backlog is cheap; change to shared interfaces or data definitions remains expensive under any delivery model, because those artifacts are consumed by other teams on other cadences (Dybå et al., 2008). The third conflict concerns recording discipline. Programs with strict change control report higher measured volatility than programs with weak control, because more of the change is recorded. Comparing their volatility metrics without adjusting for BDI would reward poor recording. Evaluation research on universal programs illustrates the general danger that a uniform metric applied across unlike populations can reverse the apparent ranking (Komi et al., 2025; Komi, 2023).

8 The TRACE-V Framework and Propositions

The synthesis supports a framework with five constructs, three mediating mechanisms, four outcomes, and a set of boundary conditions, as shown in Figure 4. The constructs are the timing-weighted change rate (T), the reach of changes through trace links (R), the amplitude of change (A), the control latency of the decision process (C), and exposure-tiered baselining (E). The first three describe the volatility a program experiences; the last two describe how the program governs it. The mediating mechanisms are impact-set accuracy, proportionate scrutiny, and targeted re-verification. The outcomes are rework and cost variance, schedule slip, escaped propagation defects, and value delivered per change. The framework is a design-science artifact whose constructs are justified by the synthesis and whose propositions remain untested (Hevner et al., 2004; Wieringa, 2014).

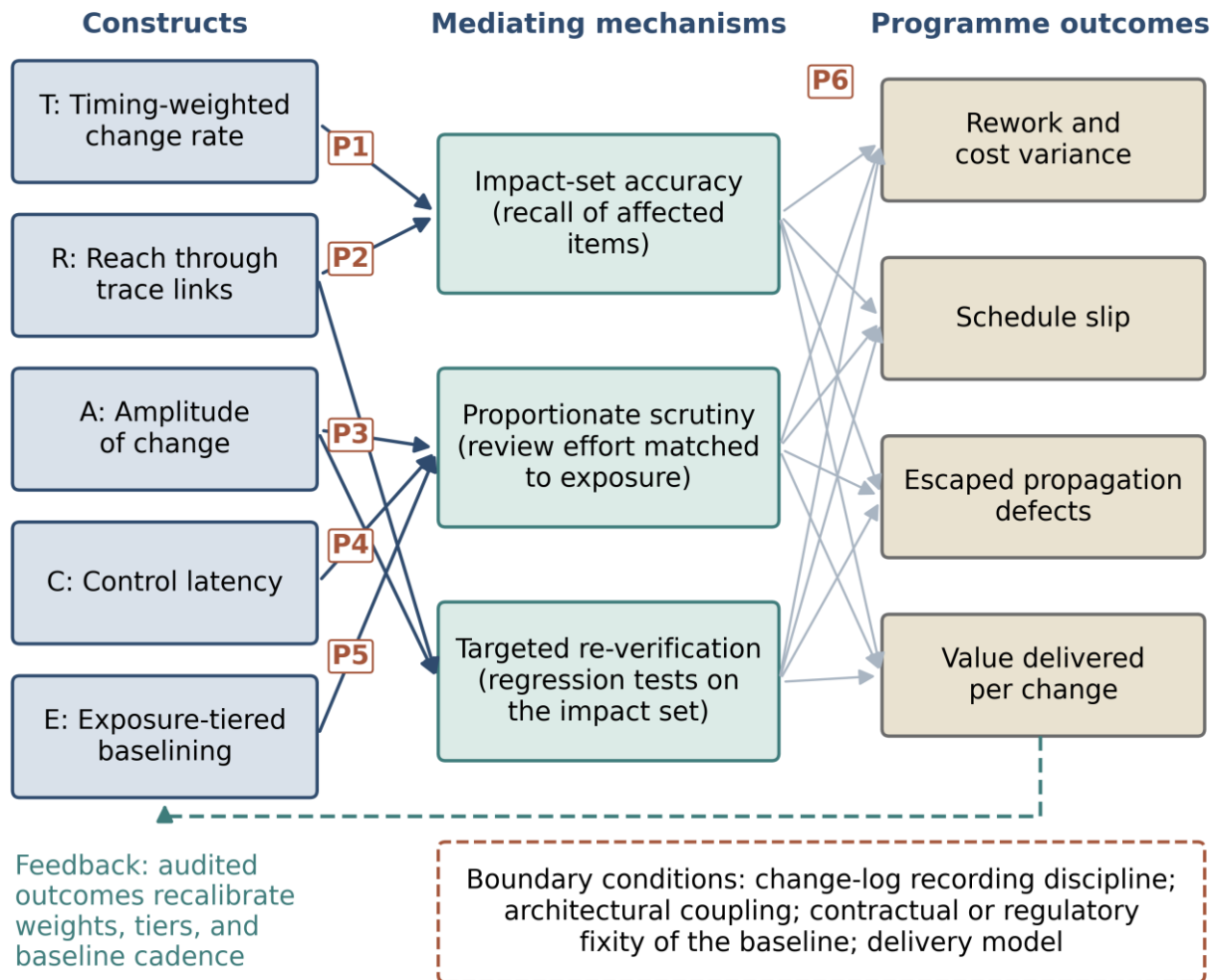


Figure 4. The TRACE-V framework linking volatility and control constructs to outcomes through three mediating mechanisms

8.1 Framework Logic

Figure 4 reads from left to right. Timing and reach determine how accurately the impact of a change can be estimated: late changes reach more committed artifacts, and changes that traverse many links accumulate the recall errors of each link. Amplitude, control latency, and exposure-tiered baselining together determine whether scrutiny is proportionate, meaning that review effort is concentrated on high-exposure changes and that low-exposure changes are decided quickly enough to prevent bypass. Reach and amplitude also determine the scope of targeted re-verification. The three mechanisms jointly drive the outcomes. The feedback path at the bottom of the figure closes the loop: audited outcomes are used to recalibrate phase weights, tier cut-offs, and baseline cadence. Without the loop, the framework degrades into a static classification scheme, a failure that organizational learning research in safety management attributes to indicator systems that are reported but never used to revise practice (Arumosoje et al., 2021; Obriki et al., 2019).

The boundary conditions in the dashed box specify where the framework should not be expected to work. Where change logs are incomplete, every metric is biased, and the framework predicts only that BDI will be high. Where the architecture is loosely coupled, reach is small and the value of trace-based impact analysis falls. Where the baseline is fixed by contract or regulation, the discretion that tiering requires may not exist, because every scope change must go to a formal authority. Where delivery is iterative and interfaces are stable, much volatility is absorbed in the backlog and tiering matters mainly for shared



artifacts. The framework is therefore a conditional theory, and its claims should be tested within these conditions rather than as a universal prescription (Ilodigwe et al., 2024b; Akintola et al., 2025).

8.2 Propositions and Tests

Six propositions follow, each stated so that it can be disconfirmed. P1 holds that the timing-weighted change rate predicts rework effort and schedule slip better than the unweighted change rate, within the same programs and periods. P2 holds that the rate of escaped propagation defects decreases as audited impact-set recall increases, and that recall has a stronger association with escaped defects than precision does. P3 holds that exposure-tiered routing reduces the total cost of processing and implementing changes relative to uniform board routing, without increasing escaped defects among high-exposure changes. P4 holds that the effect of decision latency on the baseline drift index is non-linear, rising sharply once median latency exceeds roughly half the iteration length, because teams then find it cheaper to bypass control than to wait. P5 holds that exposure-tiered baselining, with strict versioning of interfaces and data definitions and iteration-level rebaselining of local details, reduces the divergence between the baseline and stakeholder intent without raising schedule variance. P6 holds that the association between volatility and escaped defects is mediated by re-verification targeting, so that when regression selection is driven by the estimated impact set and APFD on changed items is high, the volatility and defect association weakens (Li et al., 2007; Kumuyi et al., 2024b).

Table 3 states for each proposition the observable prediction, the test design, and the result that would disconfirm it. The tests share a common data spine, a change-event record that joins each change request to its cause class, phase, estimated and actual impact sets, tier, timestamps, tests run, and defects traced to it. Longitudinal data of this kind already exist in most enterprise requirements, workflow, and defect tools, but they are rarely joined, and federated or privacy-preserving designs may be needed when the data span several organizations or contractors (Sanni et al., 2023; Atakpa et al., 2022).

Table 3. TRACE-V propositions, predictions, test designs, and disconfirming results

Proposition	Observable prediction	Test design	Disconfirming result
P1 Timing	TWR explains more variance in rework and slip than RCR	Within-program panel regression by interval; compare nested models by out-of-sample error	TWR adds no predictive accuracy over RCR
P2 Reach	Escaped propagation defects fall as audited recall rises; recall matters more than precision	Change-level logistic model of escaped defect on recall and precision, controlling for size and phase	No association with recall, or precision dominates
P3 Amplitude and tiering	Lower total change cost with no rise in high-exposure escapes	Staged rollout of tiering across programs; difference-in-differences	Cost unchanged, or high-exposure escapes increase
P4 Latency	BDI rises sharply beyond a latency threshold near half an iteration	Segmented regression of BDI on median latency across releases	Linear or flat relationship
P5 Baselining	Lower baseline and intent divergence at stable schedule variance	Interrupted time series around adoption of tiered cadence	Divergence unchanged, or schedule variance rises
P6 Re-verification	Weaker volatility and defect link when selection uses the impact set	Moderated mediation with APFD on changed items as mediator	No attenuation of the volatility and defect association

The propositions differ in how directly they can be tested. P1 and P4 can be tested with archival data from a single large program, because they concern associations across intervals and releases. P2 and P6 require change-level root-cause coding, which many programs do not yet perform consistently. P3 and P5 require an intervention and a comparison, and are best tested through staged adoption across programs or business units, which also allows the analysis to separate the effect of tiering from concurrent improvements in tooling or staffing. Longitudinal and staged designs are preferred throughout because they establish temporal order, which cross-sectional designs cannot (Onwuzulike et al., 2024; Hevner et al., 2004).



9 Discussion

The review set out to connect volatility measurement, impact estimation, change control, and outcomes in a single account. Its central finding is that the harm of volatility is not a property of change volume but of the interaction between when change arrives, how far it reaches, how accurately that reach is known, and how quickly the organization decides. This section draws out the theoretical and practical implications and considers the strongest counter-arguments (Borg et al., 2014; Cleland-Huang et al., 2012). Because enterprise programs are temporary organizations assembled from permanent departments and external suppliers, the implications also concern leadership: research on safety leadership in large temporary project organizations and on leadership-driven culture change shows that control practices take hold only when leaders model them and when the people doing the work see the indicators used to improve conditions rather than to assign blame (Arumosoye et al., 2024; Obogo et al., 2023; Obogo et al., 2019a).

9.1 Theoretical Implications

The first implication is that volatility research should abandon frequency as its primary construct. Frequency conflates learning with disruption, and its weak and inconsistent associations with outcomes are better explained as measurement error than as evidence that volatility is benign. Timing-weighted and size-weighted metrics carry the theoretical content that the deferred-commitment mechanism implies. The second implication is that traceability research and volatility research should share an outcome variable. Traceability studies that report recall against a static answer set, and volatility studies that report correlations with overrun, could be joined by measuring impact-set recall for real changes and relating it to propagation defects. That would turn a methods literature into an explanatory one (De Lucia et al., 2007; Winkler et al., 2010).

The third implication concerns governance theory. Change control is usually theorized as a gate, whose value lies in preventing bad changes. The information-processing view recasts it as a queue, whose value depends on the balance between the harm prevented and the latency imposed, and which can itself cause harm when latency drives bypass. This recasting predicts a non-monotonic relationship between control strictness and outcomes, with too little control producing propagation defects and too much producing drift and slip. It aligns with enterprise risk management frameworks that treat controls as costly interventions whose intensity should follow risk, rather than as free safeguards (Falemi et al., 2024; Fadayomi et al., 2019). Studies of technology adoption and employee readiness add that controls perceived as obstacles are circumvented, which is the behavioral mechanism behind the latency threshold in P4 (Akhigbe et al., 2024).

9.2 Practical Implications

For program leaders the practical sequence is as follows. First, stop reporting raw change counts as a quality signal and replace them with a timing-weighted series, a cause mix, and a periodic drift estimate. Second, record trace links at artifact creation for the high-value link types (requirement to interface, data definition, configuration object, and test), and audit impact-set recall on a sample of closed changes each release. Third, define exposure tiers with explicit cut-offs, decision rights, and latency targets, and publish tier latency at the median and 85th percentile. Fourth, drive regression selection from the estimated impact set and report APFD for changed items. Fifth, review the calibration each release using the feedback loop in Figure 3. Real-time KPI tracking systems make such reporting feasible when the underlying event data are joined (Tonoyan et al., 2024b; Edivri et al., 2024).

A worked scenario illustrates the combined effect. An illustrative ERP program with twelve workstreams routes all changes to a weekly board and observes a median decision latency of nine working days, a BDI of 0.08 at the last release, and a cluster of production defects in the payroll interface traced to unanalyzed changes in the employee master data. Under TRACE-V, the program would first audit impact-set recall for master-data changes and discover that links from data definitions to interfaces were never recorded. It would add those links, place master-data definitions in Tier 2 regardless of apparent size, and move report-layout changes to Tier 0 and Tier 1. The expected result is fewer items at the board, lower latency for the rest, a falling BDI, and fewer propagation defects in interfaces. Whether these expectations hold is exactly what P2 to P5 test. Mentorship and workplace learning research suggests that such changes also require coaching of change requesters, since accurate reach estimates depend on tacit knowledge of the system (Amarahobu et al., 2025a; Amayo et al., 2023c).

Program leaders should also attend to the human side of measurement. Volatility metrics can be read as judgments on the analysts who wrote the original requirements or on the business owners who request changes, which invites the attribution bias



discussed in Section 4. Framing metrics as system properties, reporting them at workstream rather than individual level, and pairing them with value delivered reduces this risk. Studies of diversity and inclusion measurement warn that quantitative indicators applied to people shape behavior in ways their designers did not intend, and that caution applies here (Uchechi et al., 2022; Jones et al., 2021).

9.3 Counter-Arguments and Boundary Conditions

Project leaders may resist the approach for three reasons. The first is that measurement of this kind is too costly for most programs. The response is that the costly elements, drift estimation and recall auditing, can be done by sampling, and that the cheaper elements, timing weights and tier latency, can be computed from data that requirements and workflow tools already hold. The second counter-argument is that tiering invites gaming, since requesters will understate reach. The response is that the recall audit exposes systematic understatement and that tier assignment can be made by the analyst who performs impact analysis rather than by the requester. Security research on agentic workflows makes the related point that any automated routing rule is an attack surface and must be monitored for manipulation (Nwakamma et al., 2024b; Dosunmu et al., 2020).

The third counter-argument is that learning-based impact prediction will make trace maintenance unnecessary. Predictive models trained on historical co-change can estimate impact without explicit links, and early results are promising. The review's position is that such models inherit the boundary conditions of their training data: they fail when architecture, regulation, or team structure shifts, exactly the conditions under which impact estimates matter most. Reviews of artificial intelligence as a decision aid in health systems reach a similar conclusion about models that perform well on average but fail on novel cases (Ilodigwe et al., 2025a; Ilodigwe et al., 2024a). Learned predictors should therefore complement recorded links, and their recall should be audited in the same way. Data-governance research adds that such models require their own lineage and privacy controls when trained on cross-team change data (Adegbite et al., 2022; Onche et al., 2023).

10 Limitations and Future Research

The review has four limitations. First, volatility metrics in the literature are heterogeneous, so the synthesis compares mechanisms and directions of effect rather than magnitudes, and it cannot say how large the benefit of any intervention is. Second, much of the evidence comes from case studies of single organizations, whose findings may reflect local recording practice. Third, adjacent-domain governance sources were used for general mechanisms, and their transfer to software rests on the argument made in each case rather than on direct evidence. Fourth, publication bias probably favors studies that find harmful effects of volatility or beneficial effects of traceability, since null findings in these areas are less often reported. Table 4 sets out the threats to validity that a study testing the TRACE-V propositions would face and the mitigations the review recommends (Runeson et al., 2009; Page et al., 2021).

Table 4. Threats to validity in change-event studies of requirements volatility and their mitigations

Threat	How it arises in volatility research	Mitigation
Incomplete change records	Informal changes never logged; metrics understate volatility where control is weak	Estimate BDI by sampling; adjust or stratify comparisons by BDI
Unstable counting rules	Splits, merges, and tool migrations alter counts without real change	Fix and publish counting rules; flag intervals with tool or rule changes
Reverse causation	Troubled programs generate correction requests, inflating volatility after overrun begins	Lagged models; cause classification; test whether volatility precedes overrun
Unknown actual impact set	AIS reconstructed after the fact from commits and defects, missing untouched but affected items	Combine commit, configuration, and defect evidence; audit a sample by inspection
Attribution bias in cause coding	Requesters prefer external causes; owners prefer corrections	Independent dual coding of a sample; report kappa
Confounding by team and system	Experienced teams on simpler systems have lower volatility and better outcomes	Match on system size, coupling, and team tenure; within-program designs
Limited transfer across delivery models	Findings from plan-driven contracts may not hold for iterative programs	Report delivery model; test propositions separately by model



Future research should pursue four lines. The first is a longitudinal change-event study that joins requirements, workflow, code, test, and defect data for several enterprise programs over at least two major releases, so that P1, P2, and P4 can be tested with temporal order established. The second is a staged introduction of exposure-tiered routing across business units or programs, which would allow P3 and P5 to be tested with a difference-in-differences design. The third is methodological work on measuring the actual impact set, which is the weakest link in any evaluation of impact analysis, including the use of configuration and deployment records alongside source code. The fourth is research on learned impact predictors under distribution shift, testing whether their recall degrades after architectural or regulatory change and whether hybrid designs that combine learned predictions with recorded links hold recall better than either alone. Across all four lines, studies should report null and adverse findings and describe implementation fidelity, because a change-control intervention that was adopted in name but not in practice cannot test the mechanism it claims to embody (Oluwadele et al., 2025; Tawose et al., 2023; Miller et al., 2015).

A further agenda concerns the economics of control. Few studies price the latency that a board imposes or the rework that a missed impact causes, so the trade-off at the heart of exposure tiering remains unquantified. Simulation of change workflows, calibrated with process-mining data, would allow programs to estimate the cost of alternative tier cut-offs before adopting them. Cost-control frameworks in infrastructure programs and resilience planning under stress conditions offer templates for this kind of scenario analysis, and lifecycle-evaluation research on purpose-built facilities shows how performance can be tracked over a long service life rather than at handover alone (Aminu-Ibrahim et al., 2025b; Ogbete et al., 2023; Abah et al., 2025). The same long view applies to enterprise software, whose volatility continues after go-live as the business, its regulators, and its vendors keep changing (Mbonu et al., 2021; Adelanwa et al., 2023b; Eboh et al., 2025b).

11 Conclusion

Requirements volatility in enterprise software is best understood as a flow of new information that the organization must absorb, not as a count of failures to be minimized. Its harm depends on when changes arrive, how far they propagate, how accurately that propagation is known, and how quickly decisions are taken. The TRACE-V framework brings these elements together: a timing-weighted rate in place of raw counts, trace-based impact estimates whose recall is audited, amplitude-sensitive scrutiny, measured control latency, and baselines whose strictness follows exposure. Programs that adopt these practices can tell healthy learning from late disruption, concentrate their review effort where it prevents defects, and decide routine changes fast enough that nobody needs to work around the process. The six propositions give researchers a direct path to testing these claims with the change-event data that enterprise programs already generate.

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Competing interests: The authors declare no competing interests.

Data availability: No new data were generated or analysed in this study; all sources are cited in the reference list.

References

- Abah, M. A., Okwah, M. N., Oladosu, M. A., Yohanna, N. R., Blessing, C. T., & Vanessa, O. I. (2025). The impact of lifestyle modifications on cardiovascular health: A review of diet, exercise, and stress reduction intervention. *Drug and Pharmaceutical Science Archives*, 5(3), 26-34. <https://doi.org/10.47587/DAP.2025.5301>
- Adebayo, A., Amadi, C., & Ijagbemi, A. O. (2025). From audit to continuous control: Implementation pathways, governance, and organizational readiness for AI-enabled compliance in critical infrastructure. *International Journal of Computer Science and Mathematical Theory*, 11(8), 88-134. <https://doi.org/10.56201/ijcsmt.vol.11.no8.2025.pg88.134>
- Adebayo, A., Anunagba, C. O., & Ozowara, D. E. (2025). A review of zero trust security models and cost effectiveness in healthcare infrastructure. *International Journal of Advanced Multidisciplinary Research and Studies*, 5(6), 2269-2283. <https://doi.org/10.62225/2583049X.2025.5.6.6051>



Adegbite, M. P., Adebayo, A., & Ahmed, M. O. (2025). A cyber risk quantification and governance architecture for critical infrastructure: From posture measurement to executive reporting. *International Journal of Computer Science and Mathematical Theory*, 11(12), 232-293. <https://doi.org/10.56201/ijcsmt.vol.11.no12.2025.pg232.293>

Adegbite, M. P., Adebayo, A., & Ahmed, M. O. (2022). A security architecture model for IT and OT convergence in regulated energy networks: Design principles and governance alignment. *International Journal of Computer Science and Mathematical Theory*, 8(2), 81-132. <https://doi.org/10.56201/ijcsmt.v8.no2.2022.pg81.132>

Adelanwa, A., Basnet, A., & Anene, U. N. (2023b). Predictive analytics models for financial risk detection and fraud prevention in public systems. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6). <https://doi.org/10.62225/2583049X.2023.3.6.5969>

Adesuyi, M. O., Kalu, A., & Walawalkar, G. (2023). Data-led cost governance in technology-intensive enterprises. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 9(3), 877-896. <https://doi.org/10.32628/IJSRCSEIT>

Adeyelu, O. O. (2018). A predictive compliance monitoring framework for detecting systemic safety risks through aviation consumer complaint data. *Iconic Research and Engineering Journals*, 1(8), 250-276. <https://doi.org/10.64388/IREV118-1718478>

Ahmed, M. O., Adegbite, M. P., & Adebayo, A. (2021). Zero trust architecture for operational technology in North American critical infrastructure: A framework for implementation and resilience optimization. *International Journal of Engineering and Modern Technology*, 7(1), 66-113. <https://doi.org/10.56201/ijemt.vol.7.no1.2021.pg66.113>

Akhigbe, R., Falemi, A., & Akin-Oluyomi, O. T. (2024). An advanced framework for technology adoption that supports employee readiness in modern organizations. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(1), 1637-1659. <https://doi.org/10.62225/2583049X.2024.4.1.5344>

Akhigbe, R., Falemi, A., & Akin-Oluyomi, O. T. (2025). An advanced governance model for improving transparency accountability and strategic execution in complex enterprises. *International Journal of Advanced Multidisciplinary Research and Studies*, 5(6), 1169-1188. <https://doi.org/10.62225/2583049X.2025.5.6.5341>

Akin-Oluyomi, O. T., & Akhigbe, R. (2023b). A conceptual supply chain leadership model for aligning logistics decisions with enterprise strategy. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(1), 1725-1744. <https://doi.org/10.62225/2583049X.2023.3.1.5340>

Akintola, A. S., Fawehinmi, Y. O., Aiyenitaju, O., Chinemerem, B., & Emmanuella, O. (2025). Digital transformation in telehealth: A systematic review of user trust, privacy, and regulatory governance in AI-powered remote monitoring systems. *Journal of Scientific Research and Reports*, 31(11), 163-182. <https://doi.org/10.9734/jsrr/2025/v31i1113658>

Alieme, J. I., Sule, A., & Obukadata, V. O. (2024). From ambiguity to AI-ready requirements: A business analysis framework for structuring business problems for intelligent process automation. *International Journal of Multidisciplinary Research and Growth Evaluation*, 5(6), 2088-2117. <https://doi.org/10.54660/IJMRGE.2024.5.6.2088-2117>

Aliliele, C., Eboh, E. E., & Oluwo, K. (2024). Advances in variance analytics for cost control in manufacturing and industrial enterprises. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 3163-3185. <https://doi.org/10.62225/2583049X.2024.4.6.6012>



- Aliliele, C., Mbonu, I. S., & Iwuanyanwu, U. (2024a). A conceptual framework for enterprise data sensitivity classification and regulatory traceability mechanisms. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 3103-3124. <https://doi.org/10.62225/2583049X.2024.4.6.5991>
- Aliliele, C., Mbonu, I. S., & Iwuanyanwu, U. (2023c). Advances in predictive analytics models for student retention and institutional risk management systems. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6), 2692-2711. <https://doi.org/10.62225/2583049X.2023.3.6.5990>
- Aliliele, C., Mbonu, I. S., Uzoka, E., & Iwuanyanwu, U. (2025b). Advances in data lakehouse governance architectures for enterprise data loss prevention and compliance assurance. *Shodhshauryam, International Scientific Refereed Research Journal*, 8(4), 171-213. <https://doi.org/10.32628/SHISRRJ258474>
- Amadi, C. E., Okorafor, U. C., Okorafor, C. I., Okwah, M. N., Akanbi, P. O., Okam, O. V., Udenze, C. I., Mbakwem, A. C., & Ajuluchukwu, J. N. (2025). Association between triglyceride-glucose index and estimated 10-year risk of cardiovascular disease in Nigerian non-diabetic hypertensives: A cross-sectional study. *PLOS Global Public Health*, 5(7), e0004760. <https://doi.org/10.1371/journal.pgph.0004760>
- Amarahobu, C., Ekelemu, O., Akano, O. A., & Adikwu, F. E. (2025a). Structured mentorship models in on-the-job technician development: A review of advancement and promotion coaching schemes, workplace learning theory and the leadership interface in continuous processing environments. *International Journal of Social Sciences and Management Research*, 11(8), 626-656. <https://doi.org/10.56201/ijssmr.vol.11no8.2025.pg626.656>
- Amayo, E. B., Owulade, O. A., & Isi, L. R. (2023a). Optimizing project governance in multinational infrastructure projects: Insights from General Electric's global operations. *International Journal of Multidisciplinary Research and Growth Evaluation*, 4(1), 975-983. <https://doi.org/10.54660/IJMRGE.2023.4.1.975-983>
- Amayo, E. B., Owulade, O. A., & Isi, L. R. (2023c). The role of diversity and inclusion in enhancing project team performance and delivery in multinational environments. *Journal of Frontiers in Multidisciplinary Research*, 4(1), 48-58. <https://doi.org/10.54660/IJFMR.2023.4.1.48-58>
- Aminu-Ibrahim, A. Y., Ogbete, J. C., & Iwuanyanwu, O. C. (2025a). Cost control and financial accountability frameworks for national healthcare construction programs. *International Journal of Advanced Multidisciplinary Research and Studies*, 5(6), 1970-1990. <https://doi.org/10.62225/2583049X.2025.5.6.5700>
- Aminu-Ibrahim, A. Y., Ogbete, J. C., & Iwuanyanwu, O. C. (2025b). Infrastructure resilience planning for national diagnostic systems under public health stress conditions. *Gyanshauryam, International Scientific Refereed Research Journal*, 8(1), 340-381. <https://doi.org/10.32628/GISRRJ2582311>
- Ammann, P., & Offutt, J. (2016). *Introduction to Software Testing* (2nd ed.). Cambridge University Press.
- Anichukwueze, C. C., Ominyi, M., Uzougbo, N. S., & Jones, B. C. (2022). Cross-functional legal education as risk mitigation: Empirical analysis of contract error reduction in engineering teams. *International Journal of Engineering and Modern Technology*, 8(5), 142-180. <https://doi.org/10.56201/ijemt.v8.no5.2022.pg142.180>
- Anioke, S. C., & Atima, M. E. (2023a). Public health governance models using process optimization and performance metrics for regulatory oversight. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6), 2534-2548. <https://doi.org/10.62225/2583049X.2023.3.6.5491>



Arumosoye, O. M., & Obriki, O. D. (2020). A governance-oriented conceptual model for contractor safety performance in multi-contract industrial projects. *International Journal of Multidisciplinary Research and Growth Evaluation*, 1(5), 728-740. <https://doi.org/10.54660/IJMRGE.2020.1.5.728-740>

Arumosoye, O. M., & Obriki, O. D. (2023). Conceptual model for emergency response readiness and capability in energy and process facilities. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 9(3), 897-917. <https://doi.org/10.32628/CSEIT25112791>

Arumosoye, O. M., & Obriki, O. D. (2024). Conceptual model of safety leadership influence in large temporary project organizations. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 3034-3047. <https://doi.org/10.62225/2583049X.2024.4.6.5895>

Arumosoye, O. M., & Obriki, O. D. (2022). Conceptual risk pathway model for lifting and rigging operations in heavy industrial construction. *Gyanshauryam, International Scientific Refereed Research Journal*, 5(5), 346-369. <https://doi.org/10.32628/GISRRJ2256237>

Arumosoye, O. M., & Obriki, O. D. (2021). Organizational learning-based conceptual maturity model for continuous safety performance improvement. *International Journal of Multidisciplinary Research and Growth Evaluation*, 2(1), 970-980. <https://doi.org/10.54660/IJMRGE.2021.2.1.970-980>

Arumosoye, O. M., Obriki, O. D., & Ozobu, C. O. (2025). Conceptual framework for environmental risk control and ESG performance through waste handling and operational discipline. *Gyanshauryam, International Scientific Refereed Research Journal*, 8(4), 177-198. <https://doi.org/10.32628/GISRRJ258367>

Arumosoye, O. M., & Obriki, O. D. (2019). Systematic review of near-miss and hazard observation data utilization in industrial safety management. *Iconic Research and Engineering Journals*, 3(2), 981-999. <https://doi.org/10.64388/IREV312-1714417>

Asiedu, C. S., & Asiedu, A. A. (2024b). Reconceptualizing quality assurance in pharmaceutical manufacturing as a determinant of supply reliability. *International Journal of Health and Pharmaceutical Research*, 9(5), 148-173. <https://doi.org/10.56201/ijhpr.v9.no5.2024.pg148.173>

Atakpa, M. I., & Abolaji, T. O. (2022). A privacy-preserving data architecture model for regulated industry analytics under GDPR and HIPAA compliance. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 8(3), 781-808. <https://doi.org/10.32628/CSEIT22558>

Atakpa, M. I., & Fobellah, A. N. (2023). Anomaly detection in financial time-series data: A conceptual model for healthcare and banking applications. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 9(2), 967-997. <https://doi.org/10.32628/CSEIT2342441>

Atta, S., Adjaklo, E., Asomani, E., & Adenuga, O. M. (2022). Advances in welding inspection standards and quality assurance practices in large-scale power generation projects. *International Journal of Engineering and Modern Technology*, 8(5), 101-124. <https://doi.org/10.56201/ijemt.v8.no5.2022.pg101.124>

Atta, S., Tofah, P., & Adenuga, O. M. (2021). Advances in non-destructive testing methods for weld integrity evaluation in high-capacity power infrastructure. *International Journal of Engineering and Modern Technology*, 7(1), 20-47. <https://doi.org/10.56201/ijemt.vol.7.no1.2021.pg20.47>



- Badmus, O., Jooda, D., & Anunagba, C. O. (2024). A privacy-by-design framework for role-based security architecture in Salesforce environments handling sensitive personal data. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 3244-3254. <https://doi.org/10.62225/2583049X.2024.4.6.6243>
- Borah, S., Aliliele, K. C., Rakshit, S., & Vajjhala, N. R. (2022). Applications of artificial intelligence in software testing. In P. K. Mallick et al. (Eds.), *Cognitive informatics and soft computing (Lecture Notes in Networks and Systems, Vol. 375)*, pp. 727-736. Springer. https://doi.org/10.1007/978-981-16-8763-1_60
- Borg, M., Runeson, P., & Ardö, A. (2014). Recovering from a decade: A systematic mapping of information retrieval approaches to software traceability. *Empirical Software Engineering*, 19, 1565-1616. <https://doi.org/10.1007/s10664-013-9255-y>
- Catal, C., & Mishra, D. (2013). Test case prioritization: A systematic mapping study. *Software Quality Journal*, 21, 445-478. <https://doi.org/10.1007/s11219-012-9181-z>
- Cleland-Huang, J., Gotel, O., & Zisman, A. (Eds.). (2012). *Software and Systems Traceability*. Springer. <https://doi.org/10.1007/978-1-4471-2239-5>
- Dagodzo, D., & Ahiaeke Patrick, M. C. (2020). UAV-based pipeline and corridor monitoring: A review of current practices and emerging technologies. *Iconic Research and Engineering Journals*, 3(10), 574-597. <https://doi.org/10.64388/IREV3110-1716084>
- Daka, E., & Fraser, G. (2014). A survey on unit testing practices and problems. *IEEE International Symposium on Software Reliability Engineering Workshops*, 201-211. <https://doi.org/10.1109/ISSREW.2014.11>
- De Lucia, A., Fasano, F., Oliveto, R., & Tortora, G. (2007). Recovering traceability links in software artifact management systems using information retrieval methods. *ACM Transactions on Software Engineering and Methodology*, 16(4), Article 13. <https://doi.org/10.1145/1276933.1276934>
- Dosunmu, A. A., & Ogundele, P. O. (2020). Intrusion detection and prevention models for enhancing organizational cyber defense effectiveness. *Iconic Research and Engineering Journals*, 4(6), 310-324. <https://doi.org/10.64388/IREV4I6-1713226>
- Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9-10), 833-859. <https://doi.org/10.1016/j.infsof.2008.01.006>
- Easterbrook, S., Singer, J., Storey, M., & Damian, D. (2008). Selecting empirical methods for software engineering research. In *Guide to Advanced Empirical Software Engineering* (pp. 285-311). Springer.
- Ebhojie, O., Dogbatsey, E. A., & Oyeleye, A. O. (2023b). Simulation-based tools in graduate accounting education: A conceptual framework for bridging variance analysis theory and practice. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6), 2851-2862. <https://doi.org/10.62225/2583049X.2023.3.6.6201>
- Eboh, E. E., & Aliliele, C. (2025b). Predictive analytics systems for investment risk monitoring in SME FinTech companies and cross-border capital markets. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 11(2), 3969-4010. <https://doi.org/10.32628/CSEIT25113398>
- Eboh, E. E., & Aliliele, C. (2023). Portfolio risk optimization models for venture capital investment strategy and fraud exposure analysis. *Iconic Research and Engineering Journals*, 7(4), 759-792. <https://doi.org/10.64388/IREV7I4-1715305>



Edivri, J., & Oteri, O. (2024). A conceptual KPI-driven decision and optimization framework for IT service delivery, portfolio performance, and adoption. *Gyanshauryam, International Scientific Refereed Research Journal*, 7(1), 167-187. <https://doi.org/10.32628/GISRRJ246643>

Ekelemu, O. (2025a). Control room alarm management and human factors in large liquefied natural gas facilities: A comprehensive review of rationalization frameworks, operator response, and the elimination of standing and inhibited alarms. *International Journal of Engineering and Modern Technology*, 11(8), 352-396. <https://doi.org/10.56201/ijemt.vol.11.no8.2025.pg352.396>

Ekelemu, O., & Akano, O. A. (2025b). Loss detection and integrity surveying in liquefaction header systems: Flare hunting, attribution and programme design for unintended flow to flare. *International Journal of Engineering and Modern Technology*, 11(8), 458-487. <https://doi.org/10.56201/ijemt.vol.11.no8.2025.pg458.487>

Ekelemu, O. (2025b). Cross-shift information loss and the standardization of logbook handovers: A critical review of communication failure and continuity protocols in continuous processing operations. *International Journal of Engineering and Modern Technology*, 11(8), 397-424. <https://doi.org/10.56201/ijemt.vol.11.no8.2025.pg397.424>

Eyetsemitan, R. A., Ambali, K. B., Oyeleye, A. O., & Fadayomi, O. (2023a). Change management in small business digital transformation: A systematic review and lean change adoption framework. *Gyanshauryam, International Scientific Refereed Research Journal*, 6(6), 521-550. <https://doi.org/10.32628/GISRRJ236648>

Eyetsemitan, R. A., Ambali, K. B., Oyeleye, A. O., & Fadayomi, O. (2023b). User acceptance testing in small business technology deployment: A structured validation framework for lean operational environments. *International Journal of Multidisciplinary Research and Growth Evaluation*, 4(6), 1512-1531. <https://doi.org/10.54660/IJMRGE.2023.4.6.1512-1531>

Eze, F. I., Akinleye, O. K., & Anene, U. N. (2024b). Development of a risk-based accelerated approval decision model balancing speed, safety, and evidence quality in pharmaceutical regulatory science. *Research Journal of Pure Science and Technology*, 7(6), 120-161. <https://doi.org/10.56201/rjpst.v7.no6.2024.pg120.161>

Eze, F. I., Anene, U. N., & Akinleye, O. K. (2022b). Design of an integrated regulatory readiness maturity model for pharmaceutical companies seeking global market entry. *International Journal of Health and Pharmaceutical Research*, 7(1), 77-112. <https://doi.org/10.56201/ijhpr.v7.no1.2022.pg77.112>

Fadayomi, O., Abolaji, T. O., Edivri, J., Ogbole, J. I., Okoruwa, P. O., & Akeju, B. (2019). Risk-based cybersecurity assurance and data availability: Limitations, advances and future research opportunities. *Iconic Research and Engineering Journals*, 2(12), 602-617. <https://doi.org/10.64388/IREV2I12-1713779>

Fadayomi, O., Abolaji, T. O., Edivri, J., Ogbole, J. I., Okoruwa, P. O., & Akeju, B. (2019). Risk-based cybersecurity assurance and data availability limitations, advances and future research opportunities. *Iconic Research and Engineering Journals*, 2(12), 602-617. <https://doi.org/10.64388/IREV2I12-1713779>

Fadayomi, O., Abolaji, T. O., Edivri, J., Ogbole, J. I., Okoruwa, P. O., & Akeju, B. (2019). Risk-based cybersecurity assurance and data availability: Limitations, advances and future research opportunities. *IRE Journals*, 2(12). <https://doi.org/10.64388/IREV2I12-1713779>

Falemi, A., Akhigbe, R., & Akin-Oluyomi, O. T. (2024). A conceptual enterprise risk management model that integrates HR strategy with operational analytics. *International Journal of Multidisciplinary Research and Growth Evaluation*, 5(6), 1730-1753. <https://doi.org/10.54660/ijmrge.2024.5.6.1730-1753>



- Fobellah, A. N. (2025a). Cognitive biases in financial decision-making: Implications for audit and risk management in large corporations: A conceptual review. *International Journal of Science and Research Archive*, 16(2), 17-22. <https://doi.org/10.30574/ijrsra.2025.16.2.2273>
- Gotel, O. C. Z., & Finkelstein, A. C. W. (1994). An analysis of the requirements traceability problem. *IEEE International Conference on Requirements Engineering*, 94-101. <https://doi.org/10.1109/ICRE.1994.292398>
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75-105. <https://doi.org/10.2307/25148625>
- Ilodigwe, L., & Adesemoye, A. C. (2024a). Advances in real-world evidence, predictive analytics, and data-driven decision making for improving health system performance and patient outcomes. *International Journal of Health and Pharmaceutical Research*, 9(5), 174-197. <https://doi.org/10.56201/ijhpr.v9.no5.2024.pg174.197>
- Ilodigwe, L., & Adesemoye, A. C. (2025a). Advances, risks, and implementation challenges of artificial intelligence as a force multiplier for clinical decision making and health system efficiency. *International Journal of Medical Evaluation and Physical Report*, 9(7), 165-185. <https://doi.org/10.56201/ijmepr.v9.no7.2025.pg165.185>
- Ilodigwe, L., & Adesemoye, A. C. (2024b). Beyond technology: A strategic framework for building sustainable and resilient health systems through organizational strategy, operational excellence, and workforce capability. *International Journal of Medical Evaluation and Physical Report*, 8(6), 299-319. <https://doi.org/10.56201/ijmepr.v8.no6.2024.pg299.319>
- ISO/IEC/IEEE. (2018). ISO/IEC/IEEE 29119-2 Software and systems engineering: Software testing processes. International Organization for Standardization.
- ISO/IEC/IEEE. (2018). ISO/IEC/IEEE 29148 Systems and software engineering: Life cycle processes and requirements engineering. International Organization for Standardization.
- Jimoh, H. O., & Ahmed, M. O. (2024). Analyzing Network Time Protocol (NTP) based amplification DDoS attack and its mitigation techniques. *Journal of Digital Innovations and Contemporary Research in Science, Engineering and Technology*, 12(2), 17-24. <https://doi.org/10.22624/AIMS/DIGITAL/V11N2P2x>
- Jones, B. C., & Ominyi, M. (2021). Developing a conceptual framework linking West African agricultural value chains to household economic stability. *International Journal of Economics and Financial Management*, 6(2), 89-114. <https://doi.org/10.56201/ijefm.v6.no2.2021.pg89.114>
- Kaner, C., Falk, J., & Nguyen, H. Q. (1999). *Testing Computer Software* (2nd ed.). Wiley.
- Komi, N. M., & Ganiu, O. S. (2025). When universal programs deepen inequality: A critical review of intersectional frameworks for climate justice and energy poverty. *International Journal of Advanced Multidisciplinary Research and Studies*, 5(6), 2380-2412. <https://doi.org/10.62225/2583049X.2025.5.6.6458>
- Komi, N. M. (2023). When accurate forecasts are not fair forecasts: Deep learning load prediction and equity across districts. *Shodhshauryam, International Scientific Refereed Research Journal*, 6(6), 476-546. <https://doi.org/10.32628/SHISRRJ247102>
- Kumuyi, O., Uzoka, E., Akeju, B., & Ozowara, D. E. (2024b). Framework for privacy-focused digital identity verification supporting financial inclusion in Africa. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 3150-3162. <https://doi.org/10.62225/2583049X.2024.4.6.6005>



- Ladapo, O. O., Jooda, D., Dosunmu, A. A., & Abolaji, T. O. (2023b). Systematic literature review on security access control policies and techniques based on privacy requirements in a BYOD environment. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6), 2890-2904. <https://doi.org/10.62225/2583049X.2023.3.6.6206>
- Li, Z., Harman, M., & Hierons, R. M. (2007). Search algorithms for regression test case prioritization. *IEEE Transactions on Software Engineering*, 33(4), 225-237. <https://doi.org/10.1109/TSE.2007.38>
- Mairiza, D., Zowghi, D., & Nurmuliani, N. (2010). An investigation into the notion of non-functional requirements. *ACM Symposium on Applied Computing*, 311-317. <https://doi.org/10.1145/1774088.1774153>
- Mbonu, I. S., Aliliele, C., Iwuanyanwu, U., & Oluoha, O. M. (2018). A conceptual framework for legal and ethical risk modeling in enterprise data protection governance systems. *Iconic Research and Engineering Journals*, 2(2), 207-226. <https://doi.org/10.64388/IREV2I2-1714911>
- Mbonu, I. S., Iwuanyanwu, U., Aliliele, C., & Uzoka, E. (2021). A review of VoIP forensic analytics models for financial fraud detection and regulatory compliance monitoring. *International Journal of Multidisciplinary Research and Growth Evaluation*, 2(6), 711-730. <https://doi.org/10.54660/IJMRGE.2021.2.6.711-730>
- Medon, J. J., & Oduleye, T. E. (2024). An integrated predictive analytics model for enhancing strategic financial forecasting and decision accuracy. *Gyanshauryam, International Scientific Refereed Research Journal*, 7(3), 258-279. <https://doi.org/10.32628/GISRRJ247322>
- Michael, O. N., & Ogunsola, O. E. (2023c). Development of predictive data models for enhancing crop resilience and climate adaptation in agrarian economies. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(1), 1510-1527. <https://doi.org/10.62225/2583049X.2023.3.1.5143>
- Miller, M., Reichelstein, J., Salas, C., & Zia, B. (2015). Can you help someone become financially capable? *World Bank Research Observer*, 30(2), 220-246. <https://doi.org/10.1093/wbro/lkv009>
- Myers, G. J., Sandler, C., & Badgett, T. (2011). *The Art of Software Testing* (3rd ed.). Wiley.
- Nwakamma, S., Ojukwu, J., & Oyesiji, S. O. (2024b). Securing agentic AI enterprise workflows against prompt injection, tool poisoning, memory manipulation, and excessive agency threats. *World Journal of Innovation and Modern Technology*, 8(6), 217-248. <https://doi.org/10.56201/wjimt.v8.no6.2024.pg217.248>
- Obogo, S. F., Nwafor, M. I., & Ozobu, C. O. (2023). Conceptual leadership model for safety culture development in construction and engineering projects. *International Journal of Scientific Research in Civil Engineering*, 7(6), 121-153. <https://doi.org/10.32628/IJSRCE237554>
- Obogo, S. F., Ozobu, C. O., & Uduokhai, D. O. (2019a). Advances in leadership driven safety culture transformation in large construction workforces. *Iconic Research and Engineering Journals*, 3(5), 507-523. <https://doi.org/10.64388/IREV3I5-1715497>
- Obogo, S. F., Ozobu, C. O., & Uduokhai, D. O. (2019b). Development of a construction safety risk governance model for multi contractor high rise projects. *Iconic Research and Engineering Journals*, 2(9), 502-522. <https://doi.org/10.64388/IREV2I9-1715495>
- Obogo, S. F., Uduokhai, D. O., & Ozobu, C. O. (2019). Systematic review of hazard identification and risk control practices in urban construction projects. *Iconic Research and Engineering Journals*, 2(12), 666-681. <https://doi.org/10.64388/IREV2I12-1715496>



- Obriki, O. D., & Arumosoye, O. M. (2019). A conceptual framework linking management safety walkthrough frequency and coverage to safety culture outcomes in mega projects. *Iconic Research and Engineering Journals*, 2(8), 355-374. <https://doi.org/10.64388/IREV2I8-1714416>
- Obriki, O. D., & Arumosoye, O. M. (2020). Conceptual framework for human error causation in high-risk construction and industrial activities. *International Journal of Multidisciplinary Research and Growth Evaluation*, 1(5), 715-727. <https://doi.org/10.54660/IJMRGE.2020.1.5.715-727>
- Obriki, O. D., & Arumosoye, O. M. (2018). Conceptual modeling of data-driven occupational safety risk control in large-scale energy infrastructure projects. *Iconic Research and Engineering Journals*, 1(7), 169-189. <https://doi.org/10.64388/IREV1I7-1714414>
- Obriki, O. D., Arumosoye, O. M., & Ozobu, C. O. (2025). Conceptual model linking leading safety signals to sustained injury-free project performance. *International Journal of Scientific Research in Humanities and Social Sciences*, 2(3), 233-254. <https://doi.org/10.32628/IJSRHSS252342>
- Ogbete, J. C., Aminu-Ibrahim, A. Y., & Ambali, K. B. (2023). Lifecycle performance evaluation of purpose built diagnostic laboratories supporting long term healthcare delivery. *International Journal of Advanced Multidisciplinary Research and Studies*, 3(6), 2605-2621. <https://doi.org/10.62225/2583049X.2023.3.6.5698>
- Ogundairo, K. M., & Ayivi-Donkor, S. S. (2023a). Machine learning approaches to customer acquisition and retention: A framework for predictive lifecycle management. *IIARD International Journal of Economics and Business Management*, 9(10), 210-246. <https://doi.org/10.56201/ijebm.v9.no10.2023.pg210.246>
- Okonkwo, C. S., Agbabiaka, J., Mayo, W., & Okeke, O. T. (2024d). Review of advances in procurement strategy, ERP adoption, and logistics performance. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 2827-2835. <https://doi.org/10.62225/2583049X.2024.4.6.5495>
- Okonkwo, C. S., Agbabiaka, J., Mayo, W., & Okeke, O. T. (2024e). Review of digital supply chain models for cost control and operational continuity. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 2836-2846. <https://doi.org/10.62225/2583049X.2024.4.6.5496>
- Oluwadele, J. F., Samuel, O., Tawose, O. M., Ekeocha, A. H., & Adika, A. O. (2025). Evaluation of alternative energy sources to replace maize in Marshall broiler diets: Effects on growth performance, meat quality, and serum biochemistry. *EUREKA: Life Sciences*. <https://doi.org/10.21303/2504-5695.2025> [Author list truncated in source; verify full authorship.]
- Oluwo, K., Bola-Sadipe, D., & Aliliele, C. (2025). Conceptual framework for data analytics driven financial decision making in banking sector organizations. *Shodhshauryam, International Scientific Refereed Research Journal*, 8(5), 259-302. <https://doi.org/10.32628/SHISRRJ258529>
- Ominyi, M., Anichukwueze, C. C., & Uzougbo, N. S. (2025a). A conceptual framework for multi-agent AI quality control in the review of regulated documents. *International Journal of Social Sciences and Management Research*, 11(8), 534-557. <https://doi.org/10.56201/ijssmr.vol.11no8.2025.pg534.557>
- Ominyi, M., Anichukwueze, C. C., & Uzougbo, N. S. (2024b). A systematic review of organizational readiness for the EU Artificial Intelligence Act in multinational enterprises. *World Journal of Innovation and Modern Technology*, 8(6), 185-203. <https://doi.org/10.56201/wjimt.v8.no6.2024.pg185.203>



- Onche, V. O., Adegbite, M. P., & Ogbonna, C. S. (2023). BYOD adoption and information-security risk in university administrative offices: Evidence from Nigerian federal universities. *International Journal of Multidisciplinary Research and Growth Evaluation*, 4(6), 1549-1563. <https://doi.org/10.54660/IJMRGE.2023.4.6.1549-1563>
- Onwuzulike, O. C., Uzundu, N. C., & Joseph, O. B. (2024). Impact of strategic management education on professional development: A longitudinal study. *World Journal of Advanced Research and Reviews*, 23(3), 323-337. <https://doi.org/10.30574/wjarr.2024.23.3.2672>
- Page, M. J., et al. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. <https://doi.org/10.1136/bmj.n71>
- Ramesh, B., & Jarke, M. (2001). Toward reference models for requirements traceability. *IEEE Transactions on Software Engineering*, 27(1), 58-93. <https://doi.org/10.1109/32.895989>
- Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14, 131-164. <https://doi.org/10.1007/s10664-008-9102-8>
- Sakyi, J. K., Filani, O. M., Nnabueze, S. B., Okojie, J. S., & Ogedengbe, A. O. (2022b). Developing KPI frameworks to enhance accountability and performance across large-scale commercial organizations. *Journal of Frontiers in Multidisciplinary Research*, 3(2), 81-93. <https://doi.org/10.54660/IJFMR.2022.3.2.81-93>
- Sanni, J. O., Atima, M. E., & Attah, A. (2022). Systematic review of attribution modeling methods resolving bias in multi-touch journeys. *Shodhshauryam, International Scientific Refereed Research Journal*, 5(1), 304-334. <https://doi.org/10.32628/SHISRRJ247135>
- Sanni, J. O., Iwuanyanwu, U. A., Essien, M. A., & Attah, A. (2023). Lifecycle-aware marketing automation using federated learning for secure cross-organizational data management. *Gyanshauryam, International Scientific Refereed Research Journal*, 6(6), 337-364. <https://doi.org/10.32628/GISRRJ236642>
- Sanni, J. O., Iwuanyanwu, U. A., & Essien, M. A. (2025). Problem-oriented process mining for auditable marketing automation lifecycle control. *International Journal of Advanced Multidisciplinary Research and Studies*, 5(6), 1933-1947. <https://doi.org/10.62225/2583049X.2025.5.6.5650>
- Shittu, M. A., Shittu, H. A., Adeleke, O. J., & Adedokun, O. J. (2023). Digital twin modeling for real time monitoring and fault detection in smart substations. *International Journal of Industrial Engineering Research and Development*, 14(2), 25-44. https://doi.org/10.34218/IJIERD_14_02_003
- Tawose, O. M., Oluwadele, J. F., Ekeocha, A. H., Odumboni, A. A., & Egbeyemi, O. S. (2023). Aflatoxin detection and quantification in poultry feeds available in selected areas in Ekiti State. *FUOYE Journal of Agriculture and Human Ecology*, 7(2). <https://doi.org/10.62923/fuojahe.v7i2.302>
- Tonoyan, A., Dada, O., & Ayivi-Donkor, S. S. (2022a). Advances in geospatial analysis: Predictive analytics methods for store location selection and market entry return on investment. *International Journal of Marketing and Communication Studies*, 6(2), 78-105. <https://doi.org/10.56201/ijmcs.v6.no2.2022.pg78.105>
- Tonoyan, A., Dada, O., & Ayivi-Donkor, S. S. (2023). A conceptual framework for retail asset valuation: Integrating quantitative pricing and multi-unit financial performance prediction. *Journal of Accounting and Financial Management*, 9(12), 218-250. <https://doi.org/10.56201/jafm.v9.no12.2023.pg218.250>



- Tonoyan, A., Dada, O., & Ayivi-Donkor, S. S. (2022b). Algorithmic process optimization and cost reduction: A review of simulation modeling and financial impact assessment. *Journal of Accounting and Financial Management*, 8(8), 139-169. <https://doi.org/10.56201/jafm.vol.8.no8.2022.p139.169>
- Tonoyan, A., Dada, O., & Ayivi-Donkor, S. S. (2024b). Real-time KPI tracking systems: A review of automated performance monitoring and data-driven decision making. *World Journal of Innovation and Modern Technology*, 8(6), 247-281. <https://doi.org/10.56201/wjimt.v8.no6.2024.pg247.281>
- Uchechi, N., Ominyi, M., Anichukwueze, C. C., & Uzougbo, N. S. (2022). Advances in quantitative evaluation models for measuring corporate diversity and inclusion investments. *World Journal of Finance and Investment Research*, 6(2), 56-101. <https://doi.org/10.56201/wjfir.v6.no2.2022.pg56.101>
- Uchechi, N., Ominyi, M., & Jones, B. C. (2018). Procedural standardization and workflow optimization in high-volume dispute resolution clinics: A systematic review. *Iconic Research and Engineering Journals*, 2(6). <https://doi.org/10.64388/IREV2I6-1722789>
- Wedraogo, L., & Sanni, J. O. (2024). Machine learning models addressing uncertainty in cross-channel campaign performance forecasting accuracy. *International Journal of Advanced Multidisciplinary Research and Studies*, 4(6), 2875-2890. <https://doi.org/10.62225/2583049X.2024.4.6.5648>
- Wieringa, R. J. (2014). *Design Science Methodology for Information Systems and Software Engineering*. Springer. <https://doi.org/10.1007/978-3-662-43839-8>
- Winkler, S., & von Pilgrim, J. (2010). A survey of traceability in requirements engineering and model-driven development. *Software and Systems Modeling*, 9, 529-565. <https://doi.org/10.1007/s10270-009-0145-0>
- Yusuf, H. D., & Abdulrazaq, K. (2018). A conceptual framework for strengthening diagnostic quality systems in resource-limited clinical laboratories. *Iconic Research and Engineering Journals*, 1(11), 145-179. <https://doi.org/10.64388/IREV1I11-1723090>
- Zhu, H., Hall, P. A. V., & May, J. H. R. (1997). Software unit test coverage and adequacy. *ACM Computing Surveys*, 29(4), 366-427. <https://doi.org/10.1145/267580.267590>