International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

Strategic IT Delivery and Operations in Large-Scale Enterprises

Srikanthudu Avancha[1]
[1]*Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.*
Emails: *srikaanth.avancha@gmail.com*[1]

Abstract

In large-scale enterprises, IT delivery and operations form the backbone of the digital strategy as technology now delivers reliable, scalable, flexible, intelligent, and governed business value. This review examines the interaction among IT governance, large-scale agile delivery, DevOps, continuous delivery, enterprise architecture management, IT enabled agility, digital transformation strategy and emerging AI/ML delivery and operations. The literature indicates that enterprise IT performance extends beyond a single technology such as portfolio governance, enterprise architecture, pipelines, operational reliability, AI-assisted observability, MLOps, AIOps, and responsible AI governance; it can be any mix of these technologies that are orchestrated. The literature indicates that there are conflicts between speed and control, standardization and autonomy, resilience and speed, automation with AI and accountability, and human effort and accountability. Whilst automation and cross-team delivery might help enterprise IT to perform better, there are still legacy constraints, distributed accountability, inadequate measurement, model and data dependency, AI governance risks and there is limited historical evidence of the relationship between delivery and operational and strategic outcomes. Further studies are needed to establish links among traditional DevOps and IT operations, AI-driven testing, incident forecasting, monitoring model performance, automated root cause analysis, and model-risk management. This review summarizes some of the major trends, gaps and research directions in strategic IT delivery and operations in complex enterprise environments.

Keywords: *continuous delivery; DevOps; enterprise architecture; IT governance; large-scale agile; operational resilience; AIOps; MLOps; AI governance; machine learning operations.*

1. Introduction

Strategic IT delivery and operations is about how an enterprise is able to translate digital intentions into dependable technology delivery and manage the architecture, services, regulatory compliance, and change pace in large-scale enterprises. New information systems research views enterprise IT not as a back office tool but as a strategic operating capability that is part of transformation programmes, platform renewal and digitally enabled value creation. Gregory et al. have clearly shown that IT transformation programs in mature enterprises are influenced by paradoxes of exploration and exploitation; that is, the delivery and operation of IT systems must be able to facilitate innovation while maintaining stability of the core systems [1]. The significance of this discovery is that large enterprises seldom run a clean technical slate; strategic IT activity happens in the context of legacy architectures, a complex network of vendors, risk management oversight, and a multi-level governance framework. The topic is also related to information systems strategy, software engineering, operations management and organizational design. Research on governance indicates that mechanisms of strategic alignment have an effect on organizational performance, but alignment is not a matter of formal committees or control mechanisms [2]. For large scale enterprises, it is important to balance local autonomy with enterprise-wide standards, as delivery teams need the right to make fast decisions, operational domains need to have shared controls, data standards, resilience targets, and discipline for the infrastructure. The strategic contribution of IT is not the question, as much as its translation to delivery and operation capabilities within the constraints of scale, interdependence, and institutional governance. Large-scale agile research also shows that enterprise transformation is not enterprise level agile routines but also dependency management, distributed coordination, ownership of the architecture and

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

changes of enterprise management roles [3]. This is relevant to the strategic IT delivery literature because in large organizations, there is an expectation of more frequent releases without significant redesign of the operating model, funding model, assurance model, incident response model, and operational accountability structure. Also, digital transformation reviews show that digital investments redefine organizational structures, value propositions and capabilities; and evidence is inconsistent in terms of how digital transformation strategy is implemented and deliver consistent IT performance [4]. AI/ML has further expanded this discussion in more recent times as enterprise IT delivery now encompasses delivery of AI-enabled systems as well as use of AI/ML for delivery and operations. The engineering practice of managing, testing, quality assurance, monitoring, maintenance and dependability of the AI-based systems are additional areas of the engineering practice [15,16]. In that case, one could not have a complete discussion of strategic IT delivery and operations without looking at the capabilities that AI/ML affords. There are a number of gaps that drive this review. First, the literature often treats delivery and operations as two separate technical or organizational topics, but increasingly, delivery and operations are one strategic enterprise capability that is governed, encompassed by DevOps, enterprise architecture, continuous delivery, IT agility, digital transformation strategy and operations powered by AI/ML. Second, there is a lot of evidence that remains case based, survey based or method based and relatively few common operational metrics across firms, sectors, the technology stack and the maturity of AI/ML. Thirdly, many studies focus on tool adoption and not on failures like tool chain fragmentation, incident learning, architecture erosion, compliance bottlenecks, model drift, data quality degradation, operational debt or weak model governance. Fourth, and even more importantly, the attention paid to large-scale enterprise is under-theorized, since many of the ideas of digital delivery originate in a digital-native world and so may not be readily translatable to regulated, multi-unit, legacy-adopting organizations. There's also been a lack of literature that integrates traditional IT delivery/operations with AIOps, MLOps, AI-assisted software engineering, AI-aware operations and responsible AI governance, which is something future research should need to prioritize in the future. The focus of this review is to synthesize critically the body of work and highlight gaps since 2015, covering traditional IT delivery and operations (ITDO) research as well as the latest research on AI/ML for software engineering, AIOps, MLOps, and AI governance. It summarizes results from the various study streams, discusses methodological trends, presents qualitative tables and analytical figures, and outlines future research directions that include the need for enterprise-level evidence, technical and operational measurement, AI/ML operationalization and enhanced strategic governance and delivery execution integration.

2. Literature Review

The current state of the art of strategic delivery and IT operations in large enterprises can be grouped into six closely related areas: delivery automation, DevOps operating models, enterprise architecture management, strategic alignment and agility, digital transformation strategy, and AI/ML-driven delivery and operations. This review draws on evidence from the continuous integration and delivery (CI/CD) research [5], enterprise continuous delivery (ECD) analysis [6], qualitative DevOps evidence [7], DevOps concept consolidation [8], DevOps capability case evidence [9], enterprise architecture evidence resulting in benefits for DevOps [10], enterprise architecture measurement for success [11], alignment-agility analysis [12], IT-enabled agility review evidence [13], digital transformation strategy making [14], AI-based software engineering research [15,16], MLOps related literature [17], AIOps literature and LLM-enabled operations research [18], and AI risk-management guidance [19] has been captured as the core evidence shown in Table 1.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

Table 1 Summary of key findings

Ref	Focus	Key Findings
[5]	Continuous integration, delivery, and deployment	Automation improves release repeatability, but testing, environment parity, governance, and tool integration remain persistent barriers.
[6]	Continuous delivery in industrial software settings	Release acceleration creates major benefits, yet legacy architecture, organizational separation, and quality assurance constrain adoption.
[7]	DevOps practice in organizations	DevOps depends on cultural integration, shared responsibility, and automation, not tool adoption alone.
[8]	DevOps concepts and challenges	The field converges around automation, collaboration, measurement, and sharing, but construct boundaries remain inconsistent.
[9]	DevOps capabilities and adoption challenges	Capability maturity depends on deployment automation, monitoring, collaboration, and management support; evidence remains context-specific.
[10]	Enterprise architecture practices and benefits	Architecture management supports coordination, standardization, and decision quality when linked to organizational routines.
[11]	Enterprise architecture management success	Measured success depends on stakeholder involvement, governance integration, and perceived architectural value.
[12]	Strategic alignment and organizational agility	Alignment supports agility when IT and organizational capabilities reinforce responsive action.
[13]	IT and organizational agility literature	IT contributes to agility through sensing, responding, and digital option creation, but causal mechanisms remain dispersed.
[14]	Digital transformation strategy-making	Strategy formation in established firms is iterative, political, and dependent on evolving digital capability structures.
[15]	Software engineering for AI-based systems	AI-based systems require additional engineering attention to data management, dependability, testing, safety, maintenance, and quality assurance beyond conventional software development.
[16]	Machine learning and software development practices	ML introduces uncertainty, data dependency, experimentation, monitoring, and role changes that alter traditional software delivery and maintenance practices.
[17]	MLOps architecture and lifecycle	MLOps supports the operationalization of ML through CI/CD, continuous training, workflow orchestration, reproducibility, versioning of code/data/models, monitoring, and collaboration.
[18]	AIOps and LLM-enabled operations	AI/ML and LLM-based methods support IT operations through log/event analysis, anomaly detection, incident triage, root-cause analysis, failure prediction, and remediation support.
[19]	AI risk management and governance	Trustworthy AI requires governance, mapping, measurement, and management of risks related to reliability, security, transparency, explainability, privacy, and bias.

Strategic IT delivery is built on delivery automation studies. While Shahin et al. highlight the research on the concepts of Continuous Integration, Continuous Delivery, and Continuous Deployment, the concepts have been limited in their uptake at scale in enterprise environments by environment, data dependency, legacy systems, and governance checkpoints [5]. Industrial analysis leads to a similar conclusion for Chen about how continuous delivery can reduce the risk of releases, shorten the feedback cycles, and need architectural decoupling, automated testing, and organizational readiness [6]. This has shown that there is no simple argument that automation equals speed. In larger organizations, the strategic value of automation is limited unless release pipelines, security assurance, service ownership and operational readiness are reimagined as interwoven capabilities. DevOps research takes this argument further by integrating delivery and operations within a shared operating model. Erich et al. found that DevOps practice is dependent on collaboration, cultural change, and knowledge sharing between development and operations teams [7]. Leite et al. bring the field together under the themes of automation, collaboration, monitoring and measurement, and acknowledge conceptual ambiguity as to what constitutes DevOps [8]. Support from management, technical practices, and cross-functional learning are needed for the successful implementation of DevOps, as demonstrated by capability-level evidence provided by Senapathi et al. [9]. What is common to all of these areas is the predominance of qualitative and interpretive studies. These prove useful to add depth but are generally not sufficient in and of themselves to compare rates of incidents, lead time, change failure, cost efficiency, or service-level stability across large enterprises. The emphasis on delivery speed is balanced with a corrective from enterprise architecture research. Foorthuis et al. demonstrate the positive impacts of enterprise architecture from a structured practice that it has on decision-making, communication and standardization [10]. Lange et al. have created empirical measures of the success of enterprise architecture management, relating the value of the architecture to stakeholder involvement and organisational integration [11]. The indications from these studies suggest that a strategy for IT delivery must be founded on an architecture that can limit local optimisations without stifling innovation. The conundrum is that while it can help establish coherence in architecture management, over-centralization can also slow down the process of delivery, can lead to ‘bypass' behaviour, or can produce governance artifacts that are not related to operational reality. Strategic alignment and agility studies link IT delivery and operations to organizational outcomes. Liang et al. determine that alignment has a reinforcing effect on organizational agility by creating patterns of capabilities [12]. Tallon et al. [13] show that IT-enabled agility requires sensing and responding mechanisms versus spending money on technology. Digital transformation strategy-making research introduces a process aspect: In established firms, digital strategy making is a process of iteration, negotiation, and learning [14]. A general observation can be made across these clusters: strategic IT delivery and operations can be understood as a multi-level capability which combines strategic formulation, architecture discipline, delivery automation, operational learning and adaptive governance. The primary challenge is the lack of effective longitudinal research designs that follow the interaction between these capability elements over time and the measurable enterprise outcomes. Another development is of enterprise IT's strategic capability – AI/ML-powered delivery and operations. Along with code, data, models, training pipelines, feedback loops and production environments are all key components of AI-based systems [15,16]. This represents another layer of operations requirements for data quality monitoring, model validation, data drift detection, model retraining governance, model explainability and model accountability for the model's lifecycle. To address some of these challenges, MLOps adopts DevOps principles to machine learning processes, such as continuous integration, continuous delivery, continuous training, model versioning, reproducibility, monitoring, and collaboration among the data science, software engineering, and operations teams [17]. The operations side of AIOps utilizes AI/ML, and, more recently, large language models, to use operational data such as logs, traces, metrics, tickets, events, and incident reports to detect anomalies, triage incidents, analyse root causes, predict failures, and provide guidance on remediation [18]. While AI-driven automation does offer governance advantages, it also introduces governance challenges such as opacity, biases, security and privacy violations, lack of accountability, and automation decisions that go too far [19].

3. Methodology

The literature examined includes different methodologies that cover various aspects of strategic IT delivery and operations in large enterprises. Systematic reviews and mapping studies map toolchains, practices over wide evidence bases, software engineering challenges and AI-based system properties and operational concerns [5, 8, 13, 15, 18] shown in Table 2.

Table 2 Method comparison

Ref	Method	Strengths	Limitations
[5]	Systematic review of CI/CD literature	Broad mapping of pipeline practices, tool categories, and adoption barriers.	Limited ability to evaluate enterprise outcomes across comparable operational metrics.
[6]	Industrial analytical article	Strong practical interpretation of continuous delivery benefits and constraints.	Limited statistical comparison and sectoral control.
[7]	Qualitative DevOps study	Rich evidence on collaboration, culture, and practice integration.	Context depth limits cross-enterprise generalization.
[8]	Survey-style literature review	Clarifies DevOps concepts, practices, and challenge categories.	Concept boundaries remain dependent on heterogeneous source definitions.
[9]	Case study of DevOps capabilities	Connects technical practices with organizational capability formation.	Single-context evidence limits causal inference.
[10]	Theory-building enterprise architecture study	Explains how architecture practices generate coordination and decision benefits.	Benefits remain difficult to link to hard operational metrics.
[11]	Empirical enterprise architecture success analysis	Provides measurable factors for architecture management performance.	Perceptual measures may understate operational and technical debt effects.
[12]	Quantitative alignment-agility analysis	Tests relationships among alignment, capability, and agility constructs.	Cross-sectional design limits temporal explanation.
[13]	Systematic review of IT-enabled agility	Synthesizes sensing, responding, and digital-option mechanisms across IT-agility research.	Causal mechanisms remain dispersed across heterogeneous studies.
[14]	Interpretive digital strategy case study	Captures iterative strategy formation in established enterprise settings.	Single-firm design restricts sectoral comparison.
[15]	Survey/mapping study of software engineering for AI-based systems	Broadly identifies AI-specific engineering concerns, including dependability, safety, testing, quality, maintenance, and data-related issues.	The field remains emerging, and many practices are not yet mature across sectors.
[16]	Empirical study of ML-related software development practices	Shows how ML changes development through uncertainty, data dependency, experimentation, and revised team practices.	Findings may depend on the maturity and context of ML-adopting organizations.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

[17]	Mixed-method MLOps review and architecture study	Provides a structured view of MLOps principles, components, roles, architecture, and workflows.	MLOps terminology and implementation maturity remain uneven across organizations.
[18]	Survey of AIOps and LLM-enabled IT operations	Synthesizes modern AI/ML and LLM-based approaches for operational data analysis, incident management, and remediation support.	The field is rapidly evolving, and standardized evaluation methods remain limited.
[19]	AI risk-management framework	Provides governance categories for trustworthy AI, including reliability, transparency, privacy, security, and accountability.	It is a governance framework rather than an empirical enterprise IT delivery study.

The qualitative case studies provide detailed information about transformation dynamics, role changes, organisational conflicts, DevOps adoption and barriers for implementing DevOps in enterprises [7, 9, 14]. Constructs like ML-related uncertainty, alignment, stakeholder involvement, architectural value, and software engineering practice are more apparent in these areas of governance, alignment, agility, enterprise architecture, and ML-enabled software development, and can be statistically examined or compared [11,12,16].

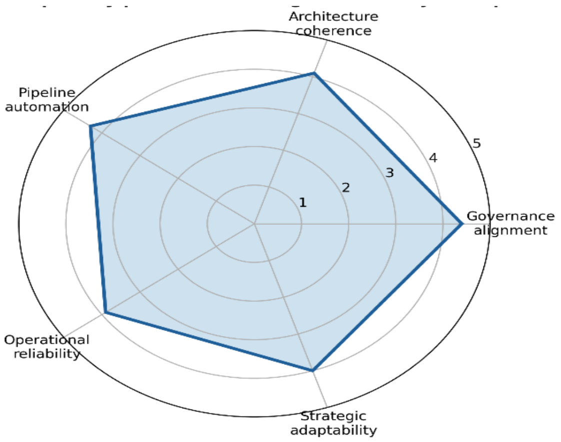

Figure 1 Strategic IT delivery and operations capability profile

In addition to peer-reviewed research, this paper also incorporates mixed-method evidence from literature review, tool review, and expert interviews, as well as the normative governance perspective of guidance for the deployment and operation of trustworthy AI from AI risk-management research [17,19]. Between these two approaches is the industrial experience analysis, which is more practical than purely conceptual, but typically less generalizable than a large-sample empirical design shown in Figure 1. Methodologically, the field is strong in depicting mechanisms but weaker in estimating effects. The lesson from the case studies is why large enterprises find it difficult to integrate delivery and operations – authority is highly distributed, funding cycles remain rigid, platforms are legacy dependent, and operational accountability may be fragmented into application, infrastructure, security and vendor domains. Studies reviewed reveal common practices but may combine evidence on best practices from large incumbent organizations, software vendors, and organizations which are "digital natives". Although quantitative studies test alignment and agility constructs, several are still far from aspects of the technical measures like deployment lead time, mean time to restore, change failure rate, production incident density, or architecture dependency complexity. The Strategic IT delivery and operations capability profile in Figure 1 is a set of five core capability domains that are interconnected: Governance alignment, Architecture coherence, Pipeline automation, Operational reliability and Strategic adaptability. These five domains are still relevant to the delivery of traditional enterprise IT, but the AI/ML enabled operational intelligence capability is a cross cutting capability that affects all five domains. AI accountability, model-risk oversight and decision rights for automated systems need to be added to governance alignment. Consideration of data pipelines, model dependencies, AI services, and enterprise platform integration are among the factors that architecture coherence considers. Pipeline automation now extends beyond software deployment; validation, retraining, monitoring and rollback automation are also becoming commonplace. Operational reliability is enhanced by anomaly, incident triage and root cause analysis powered by AI. AI/ML operational insights to inform feedback on decisions around portfolio, platform and modernization offer strategic adaptability. The capability profile in Figure 1 is a coded version of large-scale enterprise IT delivery and operations. The profile is important because it reveals that strategic delivery takes place when capability development is balanced, not just when delivery automation or governance control is improved. Recurrent capability emphasis is measured by a scale from 1 to 5 based on the review coding of the prominence of the concept in the references listed in [5] – [14].

4. Discussion

The evidence reviewed suggests that there is a long-standing speed-control-stability dilemma in strategic IT delivery and operations in large-scale enterprises. Automation can minimize manual handoffs and shorten feedback cycles as reported in continuous delivery and DevOps studies, but enterprise delivery speed is limited by integration testing, environment consistency, release governance, data migration, security controls, and architectural coupling [5, 6, 8]. The main conclusion is that it's not just a team-level engineering challenge to deliver quickly. For enterprises, the flow of releases relies on product ownership, platform standardization, compliance automation and production observability. If these elements are still in disjointed states, automation can make improvements in a small area but not in the enterprise as a whole, with little impact on throughput, reliability, or risk. The second trend identified by DevOps findings is that in every discussion of organizational integration, there is a stress on it being a prerequisite for operational improvement, but measurement is still not well developed. Qualitative research has been linked to the concepts of shared responsibility, cultural transformation, cross-functional collaboration and ongoing learning in DevOps [7], [9]. But in the literature, the adoption of practices is often reported, while operational benefits are inferred rather than directly measured. Monitoring and measurement are important DevOps concepts identified by Leite et al. [8], but the research literature lacks homogeneity in measuring the concepts. This results in a divide between practice rhetoric and scholarly evidence. For large enterprises, metrics need to be aligned with software flow and business impact on quality: Deployment Frequency, Lead Time, Failure Rate, Recovery Time, Incident Recurrence, Security Exceptions Volume, Platform Cost Efficiency. DevOps-inspired interpretations of enterprise architecture are complicated by enterprise architecture research. Both Foorthuis et al. and Lange et al. conclude that enterprise architecture management facilitates coordination and decision quality, and that the value of architecture is reliant on participation, governance connection, and organizational embedding [10, 11]. the evidence reviewed here indicates that architecture does not have to be a hurdle to delivery and the opposite is actually true. In complex businesses, the architecture discipline offers dependency visibility, integration standards, and long-horizon platform direction. Simultaneously, when decisions are made outside of delivery teams, or documentation is used instead of platform services, standards, and automated compliance, architecture can turn into a hindrance. In this case the contradiction is that both delivery autonomy and architecture governance are needed but the literature offers little insight into the point at which to make the decision. Technical-operational findings are complemented with a strategic interpretation through alignment and agility studies. The effects of governance mechanisms and strategic alignment on organizational performance have been found to exist by Wu et al. [2] and agility through capability configurations by Liang et al. [12]. Tallon et al. additionally state that IT contributes to agility not only through investment of resources, but via sensing and responding processes as well [13]. The results suggest that the successful delivery and operation of strategic IT should not only be measured in terms of faster releases, but also in terms of the ability to focus technology effort on changing market, regulatory and operational priorities. A delivery organization may be fast but misaligned, creating features or services that don't help enhance enterprise responsiveness. However, good governance coupled with the lack of delivery flow can create strategic intent that cannot be delivered at market speed. Research on digital transformation strategy further emphasizes this. According to Chanias et al. digital strategy making in established organisations is a process of iteration and of political negotiation [14]. The discovery has a significant impact on large-scale enterprise, as strategic IT delivery and operations need to adapt to evolving requirements, whilst maintaining business continuity. Modernization of platforms, data flows, customer channels, cybersecurity posture and analytics capability can all be necessary for a transformation agenda at the same time. Literature from DevOps, enterprise architecture, and alignment indicates that breakdown is when these streams operate independently: agile teams maximize local increment, architecture teams maximize standards, ops teams maximize availability, and strategy teams maximize portfolio stories. Evidence suggests that these concerns should be linked through integrated operating models with shared measures and shared decision routines. However, with AI/ML comes a new set of dynamics because enterprise IT operations are no longer restricted to human-in-the-loop monitoring, manual triage, and hard-coded automation rules. AIOps can be implemented to analyse the operational data of logs, traces, alerts, configuration, service tickets, and incident history to identify anomalies, predict failures, correlate events, perform root-cause analysis, and provide remediation support [18]. MLOps is another of the extensions of DevOps that incorporates model development, model deployment, data pipelines, continuous training, drift monitoring, and production model governance into the operational lifecycle [17] Shown in Table 3. This progress increases requirements for accountability, explainability, validation, security, and human oversight. In addition to algorithmic performance, the strategic value of AI/ML delivery in a large enterprise also depends on other factors such as service ownership, enterprise architecture, compliance processes, observability solutions, and risk management frameworks shown in Figure 2.

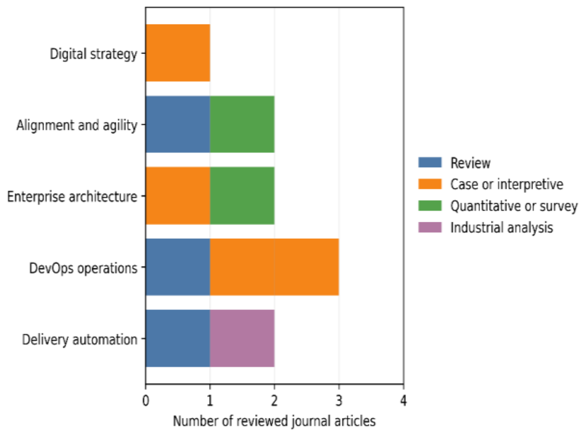

Figure 2 Evidence distribution across method families and enterprise IT domains

Table 3 Results comparison

Ref	System / Context	Metric / Basis of Comparison	Outcome
[5]	CI/CD pipeline research	Build, test, release, deployment, toolchain, and challenge categories	Automation improves repeatability but testing and environment barriers remain major enterprise constraints.
[6]	Industrial continuous delivery	Release speed, quality assurance, architecture readiness	Benefits are high when architecture and testing support frequent release; legacy coupling limits scale.
[7]	DevOps practice settings	Collaboration, automation, culture, shared responsibility	Cross-functional integration is central; tooling is insufficient without role and accountability change.

[8]	DevOps literature corpus	Conceptual categories of automation, collaboration, monitoring, measurement	DevOps is multidimensional, but construct ambiguity weakens cumulative evidence.
[9]	DevOps case context	Capability development, management support, deployment automation	Adoption depends on technical and organizational maturity rather than isolated practice introduction.
[10]	Enterprise architecture practice	Practice-benefit links in coordination and decision support	Architecture contributes value through organizational routines, not through artifacts alone.
[11]	Enterprise architecture management	Success factors and perceived value measures	Stakeholder involvement and governance integration strengthen architecture management outcomes.
[12]	Strategic alignment context	Alignment-agility relationship	Alignment can improve agility when organizational and IT capabilities reinforce adaptive response.
[13]	IT-enabled agility literature	Sensing, responding, digital option mechanisms	IT contributes to agility through dynamic response capability rather than investment volume alone.
[14]	Established financial services firm	Digital strategy formation process	Digital strategy evolves through iterative capability development and internal negotiation.
[15]	AI-based software systems	Software engineering areas, dependability, safety, testing, maintenance, and data-related challenges	AI-based systems require engineering practices that extend beyond conventional software development because system behaviour depends on models and data as well as code.
[16]	ML-enabled software development	Development practices, uncertainty, data dependency, experimentation, testing, and maintenance	ML changes software delivery by increasing uncertainty and requiring new workflows for data, model behaviour, evaluation, and monitoring.
[17]	MLOps lifecycle	CI/CD, continuous training, model versioning, reproducibility, monitoring, collaboration, and governance	MLOps links ML development and operations, but implementation maturity and terminology remain uneven across enterprises.
[18]	AIOps and LLM-enabled operations	Logs, metrics, traces, tickets, anomaly detection, incident triage, root-cause analysis, and remediation support	AI/ML can strengthen IT operations, but evaluation, reliability, explainability, and integration with human decision routines remain major issues.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

[19]	AI governance and risk management	Trustworthiness, reliability, transparency, security, privacy, accountability, bias, and lifecycle governance	AI-enabled delivery and operations require explicit governance and risk-management mechanisms to avoid uncontrolled or unaccountable automation.

Figure 2 represents the distribution of evidence across the method families and topic domains. It is important because a method map determines the field's "legitimation claim": delivery literature is more prevalent in the mapping of practice, while the literature of governance and enterprise architecture is more developed in the areas of construct development and empirical measurement. Figure 2 contrasts the emphasis of method coding by topic areas in the reviewed articles. The visualization is important because it reveals that there is a disparity between the evidence of broad delivery automation, as compared to thinner evidence of operational reliability and transformation strategy. Article classifications (not effect sizes) by dominant method and domain author's coded interpretation [5] – [14] Shown in Figure 3.

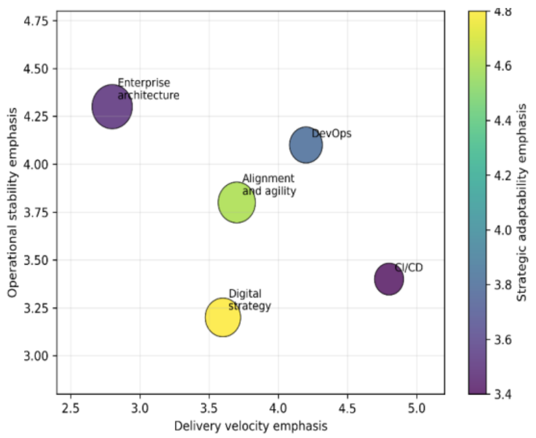

Figure 3 Comparative positioning of study clusters by velocity, stability, and governance emphasis

The most robust implication of the studies is that strategic IT delivery and operations needs capability complementarity. Without architecture discipline, a large enterprise cannot realize agile delivery; without DevOps alignment, it cannot gain a lasting competitive edge; and without digital strategy operational execution capacity, it cannot achieve digital transformation. In the literature, technical practices become strategic only when they are embedded in decision structures and measurement systems. Automated pipelines can help with regulatory traceability, security evidence, and release assurance, but this only happens when controls are defined as delivery workflows. Likewise, agility can be achieved with enterprise architecture when agility principles are realized through the use of reusable platforms, interfaces, automated standards, and feedback in the operations. Several limitations remain. First, there are many taxonomies that describe the nature of practice, and fewer causal explanations which account for the relationship between capability maturity and operational and strategic outcomes. Second, evidence exists in the enterprise but tends to be industry-specific, especially within financial services, telecom, software-centric industries and very large technology companies. Third, technical-operational measures are not necessarily related to strategic outcomes, and it can therefore be hard to analyse whether the rate of IT delivery has a positive impact on enterprise performance or simply moves risk into the production mode. Fourth, governance challenges, vendor ecosystem, outsourced operations, cloud managed services, platform engineering and third-party AI services – are not adequately explored by the literature. Fifth, there is a lack of attention being given to concepts like failure, rollback, incident learning, operational debt, model drift, data quality decay, and AI-specific technical debt, which are all pertinent to enterprise-level IT operations. Sixth, the research on AI/ML for delivery and operations is not yet well integrated with the broader streams of DevOps, enterprise architecture and governance; there is a lack of synthesis between current practice and comprehensive academic knowledge and understanding.

5. Future Directions

Research on strategic IT delivery and operations in

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

large-scale enterprises should extend beyond the adoption of technologies to comparative capability-outcome analysis. More robust designs would link delivery indicators (e.g. lead time, deployment schedule, automated test coverage, rollback rate, incident frequency, mean time to recover, service level breach frequency, etc.) to strategic outcomes (e.g. responsiveness, compliance effectiveness, customer-channel reliability, platform modernization progress, etc.). There is some existing literature in the fields of CI/CD and DevOps that highlights practices that are relevant to the work [5, 8, 9], but there is a need for better longitudinal connections between these practices and operational and strategic performance. The second priority is research on governance automation. While the present governance and enterprise architecture research is highlighting the importance of alignment, involvement of stakeholders, and architectural routines, future research should investigate the changes to enterprise control models brought about by policy-as-code, security-as-code, automated evidence collection, and platform guardrails. In the case of regulated businesses, which typically have a manual approval board, this direction is particularly significant since it can lead to delays in delivery without resulting in better real-world safety. Studies should compare manual governance, hybrid governance and automated governance by industry for various risk levels.Third, platform operating models are important. In recent years, large enterprises have turned to internal developer platforms, cloud foundations, service catalogues, observability platforms, and site reliability practices to mediate the delivery. The literature reviewed includes discussions about automation and collaboration, and there is a limited amount of direct evidence on platform engineering as a strategic delivery and operations capability. Research needs to focus on the boundaries of ownership within product teams, platform teams, security functions, architecture teams, and operations centres. Fourth, AI/ML capabilities for delivery and operations should be viewed as enterprise capabilities rather than tools. The future research directions that could be explored are incident detection, alert correlation, service degradation prediction, root cause analysis, automated remediation, and post-incident learning, among others. Research efforts for MLOps should more explicitly be tied to enterprise governance through aspects related to model approval, model monitoring and drift detection, triggers for model retraining, rollback, explanation, and auditability. Examples of software engineering practices supported by AI are automatic testing, code review support, change-risk scoring, requirements analysis, release-readiness assessment and architecture-dependency analysis. More comparative studies will be needed to gauge the effect of using AI/ML to decrease mean time to detect (MTTD) and mean time to recover/restore (MTTR), affect failure rate, impact on incident recurrence, amount of security exceptions and compliance burden, as well as determine when and where human oversight is still needed. The fifth priority is to broaden failure-centered research. Most of the literature focuses on successful adoption rather than failed delivery systems, escalation of incidents, architecture erosion, operational debt, failed automation, or model-related operational failures. Further research should explicitly target areas such as: post-incident learning, cascading chains of failure on system changes, hidden system dependencies, model drift, data pipeline breakdowns, AI-assisted remediation failures, and structural trade-offs regarding operational resilience. Finally, knowledge gained by information systems strategy, software engineering metrics, AI/ML engineering, cyber-risk governance and operations analytics and responsible AI governance should be woven and merged to assess strategic IT delivery and operations as an enterprise capability instead of a methodology.

Conclusion

Strategic IT delivery and operations is best viewed as an enterprise capability, which is backed by governance alignment, architectural coherence, delivery automation, operational reliability, strategic adaptability and crosscutting AI/ML-enabled intelligence. The literature reviewed demonstrates that technical acceleration is useful only when if accompanied by decision rights, service ownership, architectural discipline, measurable operational learning and responsible automation. For large enterprises, delivery and operations are also impacted by legacy systems, regulatory requirements, decentralization, multiple vendors, integrations with cloud and platforms, and more recently, AI/ML systems that require responsible human oversight, explainability, model monitoring, data governance, drift management, and additional complexities. The central argument is that the various capability domains must work together in order to implement enterprise IT. While these are all components of the solution, none of them independently solves for complex organizations' strategic technology execution problem. Significant gaps remain in longitudinal measurement, causal attribution, automated governance, platform operating models, operationalization of AI/ML, model risk management and failure centred operational evidence. In the future, technical delivery metrics, AI/ML operational metrics and strategic outcomes should be seamlessly integrated into scholarship, where they can help to make sense of the challenges of translating digital and AI-enabled goals into reliable, flexible, consistent, and accountable technology operations.

References

[1]. Gregory, R. W., Keil, M., Muntermann, J., & Mähring, M. (2015). Paradoxes and the nature of ambidexterity in IT transformation programs. Information Systems Research, 26(1), 57–80.

[2]. Wu, S. P.-J., Straub, D. W., & Liang, T.-P. (2015). How information technology governance mechanisms and strategic alignment influence organizational performance: Insights from a matched survey of business and IT managers. MIS Quarterly, 39(2), 497–518.

[3]. Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software, 119, 87–108.

[4]. Vial, G. (2019). Understanding digital transformation: A review and a research agenda. Journal of Strategic Information Systems, 28(2), 118–144.

[5]. Shahin, M., Babar, M. A., & Zhu, L. (2017). Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices. IEEE Access, 5, 3909–3943.

[6]. Chen, L. (2015). Continuous delivery: Huge benefits, but challenges too. IEEE Software, 32(2), 50–54.

[7]. Erich, F. M. A., Amrit, C., & Daneva, M. (2017). A qualitative study of DevOps usage in practice. Journal of Software: Evolution and Process, 29(6), e1885.

[8]. Leite, L., Rocha, C., Kon, F., Milojicic, D., & Meirelles, P. (2019). A survey of DevOps concepts and challenges. ACM Computing Surveys, 52(6), Article 127.

[9]. Senapathi, M., Buchan, J., & Osman, H. (2018). DevOps capabilities, practices, and challenges: Insights from a case study. Journal of Systems and Software, 135, 57–67.

[10]. Foorthuis, R., Steenbergen, M., Brinkkemper, S., & Bruls, W. (2016). A theory building study of enterprise architecture practices and benefits. Information Systems Frontiers, 18(3), 541–564.

[11]. Lange, M., Mendling, J., & Recker, J. (2016). An empirical analysis of the factors and measures of enterprise architecture management success. European Journal of Information Systems, 25(5), 411–431.

[12]. Liang, H., Wang, N., Xue, Y., & Ge, S. (2017). Unraveling the alignment paradox: How does business–IT alignment shape organizational agility? Information Systems Research, 28(4), 863–879.

[13]. Tallon, P. P., Queiroz, M., Coltman, T., & Sharma, R. (2019). Information technology and the search for organizational agility: A systematic review with future research possibilities. Journal of Strategic Information Systems, 28(2), 218–237.

[14]. Chanias, S., Myers, M. D., & Hess, T. (2019). Digital transformation strategy making in pre-digital organizations: The case of a financial services provider. Journal of Strategic Information Systems, 28(1), 17–33.

[15]. Martínez-Fernández, S., Bogner, J., Franch, X., Oriol, M., Siebert, J., Trendowicz, A., Vollmer, A. M., & Wagner, S. (2022).

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 5070-5082
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0667

Software engineering for AI-based systems: A survey. ACM Transactions on Software Engineering and Methodology, 31(2), Article 37, 1–59.

[16]. Wan, Z., Xia, X., Lo, D., & Murphy, G. C. (2021). How does machine learning change software development practices? IEEE Transactions on Software Engineering, 47(9), 1857–1871.

[17]. Kreuzberger, D., Kühl, N., & Hirschl, S. (2023). Machine learning operations (MLOps): Overview, definition, and architecture. IEEE Access, 11, 31866–31879.

[18]. Zhang, L., Jia, T., Jia, M., Wu, Y., Liu, A., Yang, Y., Wu, Z., Hu, X., Yu, P. S., & Li, Y. (2025). A survey of AIOps in the era of large language models. ACM Computing Surveys, 58(2), Article 44, 1–35.

[19]. National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1.