

Global Migration System using Agile Methodology: Case Study ServiceNow Version from Vancouver to Washington in Multinational Tobacco Company

Tajghina Qatrunnada Firdaus¹, Teguh Raharjo^{2,4}, Anita Nur Fitriani³

^{1,2,3} Faculty of Computer Science, Universitas Indonesia, Indonesia

⁴ Computer Science Department, Binus Online Learning, Bina Nusantara University, Indonesia

Article Info

Article history:

Received Feb 25, 2026

Revised Apr 26, 2026

Accepted June 29, 2026

Keywords:

Agile Methodology

ServiceNow Migration

Global System Migration

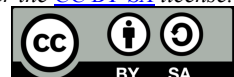
Global Collaboration

IT Service Management

ABSTRACT

This study employs a qualitative case study approach based on project documentation, issue tracking logs, sprint reports, and communication records to examine the application of Agile methodology in a global ServiceNow system migration from the Vancouver to Washington release within a highly regulated multinational tobacco company. The migration involved 1,527 globally distributed team members and required coordinated execution across multiple regions and time zones. Agile's iterative and collaborative approach enabled continuous feedback, cross-regional coordination, and effective management of migration complexities throughout the upgrade process. During the migration, a total of 119 issues were identified and resolved across multiple testing phases, with only six issues (approximately 5%) affecting the production environment. This outcome indicates strong risk containment during pre-production validation stages. The findings demonstrate that Agile governance mechanisms such as iterative sprint cycles, backlog prioritization, and continuous cross-regional communication can support operational continuity and issue containment during large-scale enterprise platform upgrades. This study contributes empirical insights into the role of Agile governance in enterprise-scale ITSM platform migration and proposes an Agile-based global system migration framework that can help organizations mitigate operational risks during complex system upgrade initiatives.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Tajghina Qatrunnada Firdaus

Faculty of Computer Science

Universitas Indonesia,

Jl. Prof. DR. Sudjono D. Puspongoro, Pondok Cina, Kec. Beji, Kota Depok, Jawa Barat, 16425

Email: tajghina.qatrunnada@ui.ac.id

1. INTRODUCTION

Despite the growing body of literature on Agile project management, most prior studies primarily focus on Agile implementation within software development environments or general IT project delivery contexts. Limited attention has been given to the role of Agile governance mechanisms in large-scale enterprise platform migration projects, particularly those involving globally distributed teams, complex system dependencies, and strict operational continuity requirements. Existing studies highlight that Agile practices improve flexibility, coordination, and risk responsiveness in complex IT projects [1], [2]. However, empirical evidence examining how specific Agile governance mechanisms such as sprint-based testing cycles, backlog prioritization, and cross-regional coordination influence operational outcomes during enterprise platform upgrades remains scarce, especially in big-bang migration scenarios where rollback options are limited. Consequently, further empirical investigation is needed to understand how Agile

governance mechanisms support coordination, risk monitoring, and issue containment during large-scale enterprise system migration initiatives.

Agile methodology has increasingly been adopted as a project management approach capable of addressing these challenges due to its emphasis on iterative development, continuous feedback, and cross-functional collaboration [3]. Recent studies demonstrate that Agile governance mechanisms can enhance adaptability, risk responsiveness, and stakeholder alignment in complex IT environments [1], [2]. Agile frameworks have also been shown to improve project transparency and facilitate rapid issue resolution through incremental development cycles and structured communication routines among distributed teams [4].

Furthermore, recent research indicates that Agile practices increasingly function as embedded risk management mechanisms in IT projects, where risk mitigation occurs continuously throughout iterative development cycles rather than through isolated governance phases [5]. This perspective suggests that Agile methodologies may play a critical role in managing uncertainty and operational risks during large-scale enterprise system migrations.

This limitation is particularly relevant in the context of ServiceNow platform upgrades within multinational organizations. ServiceNow serves as a core ITSM infrastructure supporting critical organizational processes such as incident management, change management, configuration management, and service request fulfillment. Upgrading such a platform across globally distributed environments requires coordinated governance mechanisms capable of managing technical dependencies, organizational complexity, and operational risks simultaneously.

Recent literature on Agile-oriented ITSM frameworks highlights the importance of integrating Agile governance with enterprise service management practices to ensure service reliability and operational stability in large-scale digital infrastructures [6]. However, empirical studies demonstrating how Agile practices operate within enterprise-wide ITSM platform migrations remain limited, particularly in highly regulated industries where operational disruptions may have significant business consequences.

To clarify the positioning of this study within existing literature,

Table 1-1 summarizes recent research on Agile governance, IT service management platforms, and enterprise system migration.

Table 1-1 Research Gap in Agile-Based Enterprise System Migration Studies

Study Focus	Context	Key Contribution	Identified Limitation
Agile project management in software development [5], [7]	Software engineering	Demonstrates that Agile practices improve flexibility, iterative development, and team responsiveness in software development projects	Limited discussion of Agile application in enterprise platform migration scenarios
Agile governance in digital transformation [5], [7]	Organizational IT projects	Shows that Agile governance mechanisms support adaptability, stakeholder coordination, and risk monitoring in complex digital transformation initiatives	Limited empirical evidence on Agile governance in ITSM platform upgrade environments
ITSM and ServiceNow implementation [8], [9]	Enterprise service management	Demonstrates that ITSM platforms improve service governance, operational workflows, and organizational coordination in digital infrastructures	Rarely examines large-scale ServiceNow version migration projects involving global coordination
Large-scale system integration studies [10]	Enterprise IT infrastructure	Highlights the importance of governance structures and coordination mechanisms in large-scale system integration initiatives	Limited focus on Agile-driven governance mechanisms during system migration
This study	ServiceNow big-bang migration in multinational organization	Provides empirical analysis of Agile governance mechanisms in a large-scale ITSM platform migration involving globally distributed teams	Addresses the gap in Agile-based governance for enterprise system migration

Although previous studies have examined Agile practices in software development and IT project management, limited empirical research has investigated how Agile governance mechanisms operate during large-scale enterprise platform migration projects, particularly under big-bang upgrade strategies where rollback options are limited.

In response to these limitations, this study investigates the implementation of Agile methodology in a global ServiceNow migration project from the Vancouver to Washington release

within a multinational tobacco company. The migration involved globally distributed teams and required coordinated execution across multiple regional environments while maintaining operational continuity. Specifically, this study addresses two research questions: (1) how Agile methodology supports complexity management and risk mitigation during a large-scale ServiceNow migration in a multinational enterprise environment, and (2) to what extent Agile governance mechanisms contribute to issue containment and operational stability during the migration process.

By addressing these questions, this research contributes to the existing literature in three ways. First, it provides empirical evidence on the role of Agile governance mechanisms in enterprise-scale ITSM platform migration projects. Second, it demonstrates how iterative coordination and testing practices support risk containment during large-scale system upgrades. Third, the study offers practical insights for organizations planning global system migrations where rollback options may be limited or unavailable.

Unlike prior studies that primarily examine Agile implementation in software development environments, this study provides empirical evidence on how Agile governance mechanisms operate in large-scale enterprise ITSM platform migration projects involving globally distributed operational teams.

1.1 Agile Governance in Enterprise IT Projects

Agile methodology has evolved from a software development paradigm into a broader governance approach for managing complex digital transformation initiatives. Agile frameworks emphasize iterative development, cross-functional collaboration, and continuous feedback mechanisms that enable organizations to adapt to rapidly changing technological and business environments [11].

Recent studies highlight that Agile governance mechanisms can improve coordination and risk responsiveness in large-scale IT projects. Agile project structures facilitate incremental problem detection and resolution through iterative development cycles, allowing project teams to continuously refine system configurations and integration processes throughout the project lifecycle [2].

In addition, Agile methodologies increasingly incorporate risk management as an embedded governance mechanism rather than treating risk mitigation as a separate project phase. The Agile Governance and Performance (AGP) model suggests that risk mitigation occurs through continuous monitoring, iterative testing, and collaborative problem solving among project stakeholders [5].

In large-scale enterprise environments, Agile governance structures also support coordination across distributed teams. Research on large-scale Agile implementations indicates that structured communication routines and synchronized iteration cycles play an essential role in maintaining alignment among geographically dispersed project teams [12].

These characteristics make Agile governance particularly suitable for enterprise system migration projects where requirements evolve dynamically and system stability must be preserved during infrastructure upgrades.

1.2 IT Service Management Platforms and ServiceNow in Global Enterprise Environments

Enterprise digital transformation increasingly relies on integrated IT service management platforms capable of coordinating operational workflows, governance mechanisms, and service delivery processes across distributed organizational units. By integrating service delivery processes, operational governance, and performance monitoring mechanisms into a unified digital environment, ITSM platforms enable organizations to consolidate operational data, workflows, and infrastructure management processes within centralized systems. This integration improves service visibility, accelerates incident resolution, and enhances decision-making across distributed operational environments. Empirical studies further indicate that the implementation of ITSM platforms contributes to improved governance consistency, operational resilience, and coordination among organizational units undergoing large-scale digital transformation initiatives [8], [13].

The importance of integrated ITSM platforms becomes particularly evident in multinational organizations where operational activities are distributed across multiple geographical regions and time zones. In such environments, enterprise service platforms must

support not only operational workflows but also governance mechanisms that ensure service reliability, regulatory compliance, and coordination among geographically dispersed teams. Centralized ITSM platforms therefore function as critical coordination infrastructures that enable organizations to maintain operational stability, service continuity, and consistent governance across global enterprise operations [9], [13].

ServiceNow supports enterprise service management through several core operational modules. These modules collectively enable organizations to manage service disruptions, control infrastructure changes, investigate recurring operational problems, and process service requests from users across organizational units.

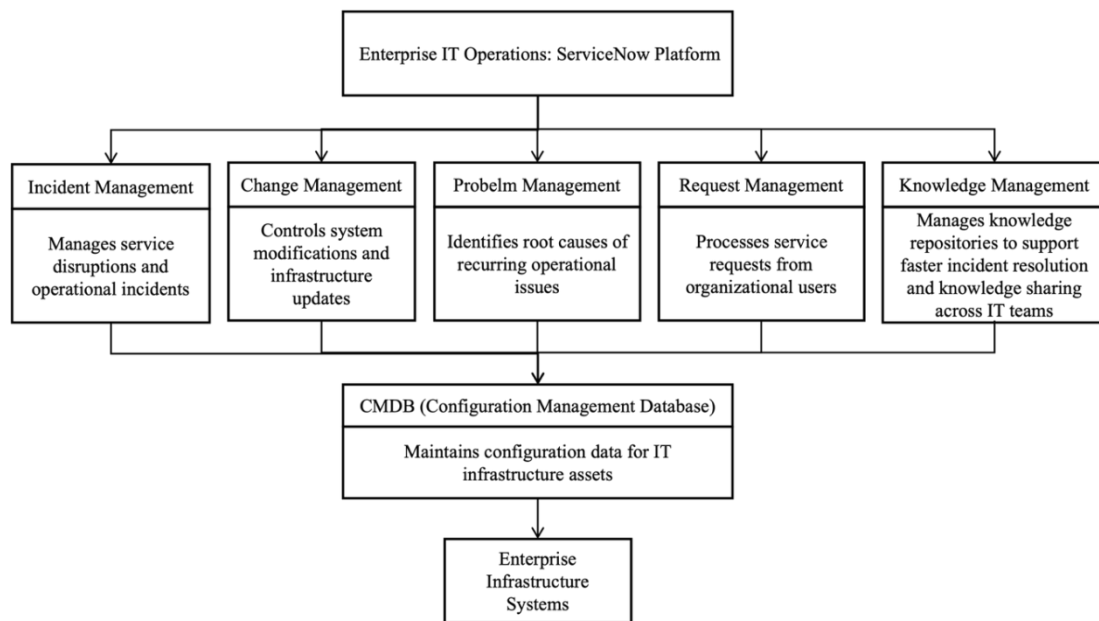


Figure 1. ServiceNow ITSM Architecture in Global Enterprise Environments

These modules collectively form the operational backbone of enterprise service management by ensuring that incidents are addressed promptly, infrastructure changes are controlled through governance mechanisms, and system configurations remain transparent across organizational units. In particular, the Configuration Management Database (CMDB) plays a critical role in maintaining configuration data and dependency relationships among enterprise IT assets, enabling organizations to conduct impact analysis during system upgrades and infrastructure changes [9].

Despite the advantages offered by enterprise ITSM platforms, upgrading such systems presents significant technical and organizational challenges. Platform upgrades frequently involve complex dependencies between system configurations, integration interfaces, and operational workflows. Large-scale system migration research highlights that enterprise platform upgrades often require extensive coordination among development teams, testing teams, and operational stakeholders to ensure that critical organizational services remain uninterrupted during system transitions [10].

These challenges become even more pronounced in multinational environments where system configurations, integrations, and user operations span multiple geographical regions. System upgrades must therefore be executed through structured governance mechanisms that enable coordinated testing, issue monitoring, and operational validation prior to production deployment.

The centralized architecture illustrated in

Figure 1 enables organizations to coordinate service operations, monitor infrastructure changes, and maintain system reliability across distributed environments. By integrating service

management processes into a unified digital platform, organizations can enhance operational transparency and improve governance over IT service delivery.

However, the centralized nature of such platforms also introduces additional challenges during system upgrades. Because multiple organizational services depend on the platform simultaneously, system migration must be carefully coordinated to prevent disruptions to operational workflows.

For multinational enterprises operating in highly regulated industries, these challenges become even more critical. System upgrades must maintain compliance with operational governance requirements while ensuring that service delivery processes remain uninterrupted during the transition period.

Therefore, understanding how governance mechanisms such as Agile methodologies support the coordination and risk management of large-scale platform migrations becomes an important area of investigation. This study examines how Agile practices were implemented during a global ServiceNow platform migration and how these practices contributed to managing operational risks during the upgrade process.

1.3 Global ServiceNow Migration Governance Model

To synthesize the theoretical perspectives discussed in the previous sections, this study proposes a conceptual framework referred to as the Global ServiceNow Migration Governance Model. The model illustrates how Agile governance mechanisms facilitate coordination, risk management, and issue containment during large-scale enterprise system migration.

Large-scale platform migration projects involve multiple layers of complexity including technical dependencies, organizational coordination, and operational risk management. In multinational environments, these complexities are amplified by geographical dispersion, time-zone differences, and integration dependencies between enterprise systems. As a result, migration governance mechanisms must ensure that system upgrades are executed systematically while minimizing disruptions to operational services.

Agile methodologies provide governance structures that support iterative coordination and continuous risk monitoring during complex IT projects. Research on Agile governance highlights that iterative development cycles, structured communication routines, and cross-functional collaboration enable organizations to identify and resolve system issues early in the project lifecycle [2], [5].

In enterprise IT service management environments, these governance mechanisms are particularly important because system platforms support critical organizational services such as incident management, infrastructure monitoring, and service request processing. Consequently, migration processes must ensure that system upgrades maintain service reliability while enabling new platform capabilities.

The proposed governance model conceptualizes the migration process as an interaction between four key dimensions: Agile governance practices, migration coordination mechanisms, migration execution processes, and operational outcomes. Agile practices provide the operational foundation for iterative development and testing cycles, while governance mechanisms ensure coordination across globally distributed teams [14]. These elements collectively support the execution of migration activities and influence the final operational outcomes of the migration project.

1.3.1 Agile Governance Practices

Agile governance practices represent the operational principles guiding project execution. These practices include sprint-based iteration cycles, backlog prioritization mechanisms, continuous testing routines, and collaborative problem-solving across project teams. Such practices enable organizations to respond quickly to emerging technical issues during system migration.

1.3.2 Migration Governance Mechanisms

Migration governance mechanisms refer to the coordination structures that ensure alignment among project stakeholders during the migration process. These mechanisms include risk monitoring procedures, issue tracking systems, and structured communication channels that support collaboration across distributed teams.

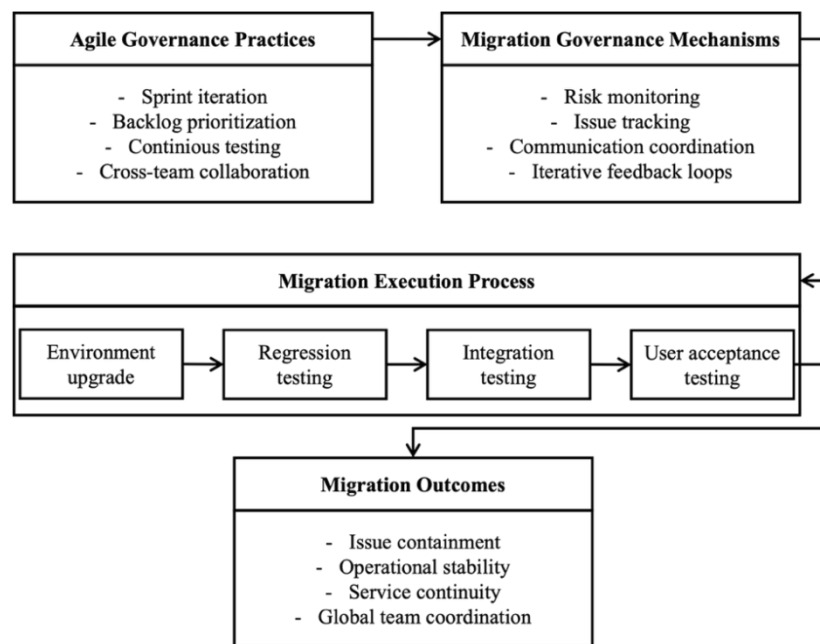


Figure 2. Global ServiceNow Migration Governance Model

1.3.3 Migration Execution Process

The migration execution process represents the operational activities performed during the system upgrade. These activities typically include environment cloning, system upgrade procedures, regression testing, integration validation, and user acceptance testing.

1.3.4 Migration Outcomes

The final dimension of the model represents the operational outcomes of the migration process. Successful governance and execution mechanisms contribute to outcomes such as reduced production-level issues, improved operational stability, and effective coordination across global teams.

2. RESEARCH METHOD

1.4 Research Design

This study adopts a qualitative case study approach to investigate the implementation of Agile methodology during a large-scale enterprise system migration. Case study research is widely used in information systems research to analyze complex organizational phenomena where technological, organizational, and governance factors interact within real-world environments [15].

Large-scale enterprise system migration projects typically involve complex coordination among multiple stakeholders, technological dependencies, and operational governance mechanisms. Case study methodology enables researchers to examine such complexities in depth by analyzing empirical evidence derived from organizational artifacts and operational processes. Recent studies also emphasize the relevance of case study methods in analyzing digital transformation initiatives and large-scale IT infrastructure projects within organizational contexts [16].

In this research, the case study focuses on the migration of a global ServiceNow platform from the Vancouver release to the Washington release within a multinational tobacco company. The migration followed a big-bang upgrade strategy, where the new system version was deployed simultaneously across the organization without maintaining parallel production environments. Such migration strategies require rigorous coordination and extensive testing to ensure operational continuity during system deployment.

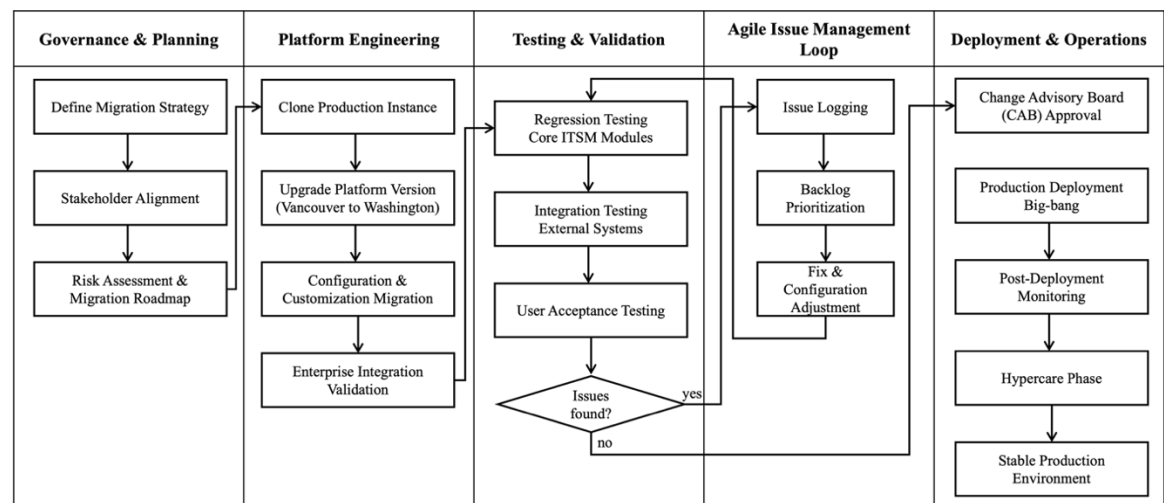


Figure 3. Enterprise ServiceNow Migration Architecture and Governance Workflow

Figure 3 illustrates the architectural process used during the ServiceNow platform migration from the Vancouver release to the Washington release. The migration followed a structured upgrade workflow consisting of environment cloning, configuration migration, and multiple stages of system validation prior to production deployment.

The migration process began with the cloning of the legacy ServiceNow environment to create a controlled upgrade environment. This stage ensured that the migration activities could be performed without affecting the operational production system. Following the environment cloning process, the platform upgrade and configuration migration were conducted to transition the system from the Vancouver release to the Washington release.

After the upgrade process, the system underwent multiple validation stages including regression testing, integration testing, and user acceptance testing. These testing phases ensured that core IT service management modules such as Incident Management, Change Management, Problem Management, Request Fulfillment, and the Configuration Management Database continued to function correctly after the upgrade.

Once the testing phases were successfully completed and critical issues were resolved, the upgraded platform was deployed to the production environment using a big-bang deployment strategy, where the new system version replaced the previous version across the organization simultaneously. Following the deployment, a monitoring and hypercare phase was conducted to observe system stability and ensure operational continuity.

This migration architecture reflects structured governance and validation mechanisms commonly recommended for enterprise system upgrades in complex IT infrastructures [10].

1.5 System Migration Context

The migration project involved 1,527 team members distributed across multiple global regions, including the Americas, Europe, Asia, and the Middle East. The project brought together IT service management specialists, system engineers, and operational stakeholders responsible for coordinating the migration process across distributed organizational environments.

The migration process was executed over approximately six agile sprint cycles, each lasting about one week within 1.5 months and consisted of several iterative phases including system preparation, environment upgrade, regression testing, integration validation, and user acceptance testing. Agile coordination practices such as iterative feedback loops, collaborative issue resolution, and continuous communication across regional teams were implemented to maintain alignment during the migration lifecycle.

The migration primarily involved several core modules of the ServiceNow IT Service Management (ITSM) platform that support enterprise service operations. Enterprise ITSM platforms such as ServiceNow play a critical role in enabling operational coordination, service governance, and infrastructure visibility in large organizations. Research shows that centralized

ITSM platforms improve operational transparency, service reliability, and cross-team coordination within enterprise digital infrastructures (Sarwar et al., 2023; Dande et al., 2024).

1.6 Migration Cutover Plan and Contingency Strategy

Because the migration followed a big-bang deployment strategy without maintaining parallel production environments, a structured cutover governance plan was implemented to minimize operational risks during the final deployment stage. The cutover process consisted of several coordinated steps conducted immediately prior to production release [8], [10].

First, a final system validation was performed after the completion of regression testing, integration testing, and user acceptance testing to ensure that all critical issues had been resolved. Only issues categorized as low severity were permitted to remain open prior to deployment. Second, a formal go-live approval checkpoint was conducted involving migration coordinators, system administrators, and regional IT stakeholders to confirm operational readiness across global environments [10].

During the deployment window, the migration team implemented a coordinated cutover schedule to synchronize configuration updates and system activation across distributed regions. To address potential operational risks after deployment, a dedicated hypercare monitoring phase was initiated immediately following the go-live event. A centralized incident monitoring channel and rapid response coordination team were established to detect and resolve potential post-deployment issues [5].

Although a technical rollback option was not available due to the nature of the platform upgrade, contingency measures were prepared through rapid configuration fixes and hot-patch deployment procedures to maintain service continuity [5], [10].

1.7 Data Sources

This study uses multiple sources of empirical evidence derived from organizational artifacts generated during the migration project. The use of multiple data sources enables data triangulation, which strengthens the validity and reliability of case study findings [15]. The primary data sources used in this study are summarized in Table 2-1.

Table 2-1. Data Sources Used in the Study

Data Source	Example Artifacts	Data Extracted	Analytical Purpose
Project documentation	Migration roadmap, governance documents, upgrade schedules	Migration phases, governance checkpoints, deployment timeline	Understanding migration governance and planning structure
Issue tracking logs	Defect records recorded in ServiceNow issue tracker	Issue ID, severity level, detection phase, resolution status	Identifying issue distribution and resolution patterns
Sprint reports	Agile sprint summaries and iteration reports	Sprint duration, issue backlog, resolution cycles	Analyzing Agile iteration processes during migration
Communication records	Meeting minutes, coordination messages, migration updates	Decision logs, issue escalation, coordination patterns	Validating cross-regional collaboration and governance practices

The use of multiple organizational artifacts enables data triangulation, which improves the reliability and validity of case study findings. Triangulation across project documentation, issue logs, and coordination records allows the study to validate migration activities, issue resolution processes, and governance mechanisms implemented during the platform upgrade [12], [15].

The analyzed project documentation included the ServiceNow migration roadmap, upgrade schedule documents, sprint planning reports, and governance meeting minutes generated during the migration period. Issue logs were extracted from the ServiceNow issue tracking system used by the migration team. Communication records included coordination messages and meeting summaries exchanged through internal collaboration channels during the migration process.

1.8 Data Extraction Procedure

The data extraction process was conducted by the research team following the completion of the migration project. Project artifacts were reviewed between January and February 2025. Issue records were extracted from the ServiceNow issue tracking system and compiled into a structured dataset containing issue identifiers, detection phase, severity classification, resolution status, and production impact status. Although issue records include information about the detection phase, detailed quantitative distribution across individual testing phases was not consistently recorded in all project artifacts. Therefore, the analysis focuses primarily on overall issue resolution patterns and production impact outcomes.

1.8.1 Artifact Identification

Project artifacts relevant to the migration process were identified, including migration governance documentation, issue tracking logs, sprint reports, and coordination records generated throughout the migration lifecycle.

1.8.2 Issue Dataset Compilation

Issue records generated during development testing, regression testing, and user acceptance testing were extracted from the issue tracking system. Each issue record included the following attributes: (1) issue identifier, (2) detection phase, (3) severity classification, (4) resolution status, and (5) production impact status. A total of 119 issue records were compiled and analyzed in this study.

1.8.3 Data Verification

To ensure data consistency, issue statistics were cross validated with sprint reports and migration governance documentation. Communication records were also examined to verify issue resolution timelines and coordination activities among global teams.

1.9 Issue Classification and Measurement Criteria

To ensure consistency in data analysis, issues recorded during the migration project were categorized according to predefined severity classifications commonly used in enterprise IT service management.

Table 2-2. Issue Severity Classification Criteria

Severity Level	Definition
Critical	Issue causing system failure or preventing core functionality
High	Issue causing major disruption to system functionality
Normal	Issue with moderate operational impact
Low	Minor issue such as interface inconsistencies or documentation errors

Severity-based issue classification is commonly applied in IT service management environments to prioritize issue resolution and assess operational risks during system implementation and migration activities [8].

Duplicate issue reports referring to the same root cause were consolidated into a single issue record during dataset preparation to avoid inflation of issue counts. In this study, an issue refers to any defect, configuration error, integration failure, or functional inconsistency identified during development testing, regression testing, or user acceptance testing phases and recorded in the project issue tracking system.

1.10 Definition of Production Impact

For analytical clarity, this study defines production impact as an issue that meets at least one of the following conditions after system deployment: (1) the issue is detected after the system is released to the production environment, (2) the issue disrupts operational system usage, and (3) the issue requires corrective action following deployment. In practical terms, production-impact issues may include incidents reported by end users after deployment, system monitoring alerts indicating service disruption, or defects requiring post-deployment configuration fixes.

Based on this definition, six issues (approximately 5% of the total recorded issues) were identified as affecting the production environment, while the remaining issues were resolved during pre-production testing phases.

1.11 Bias Mitigation and Validity Considerations

Because this study relies on internal project artifacts generated within the organization being studied, potential risks of selective reporting or optimistic interpretation must be considered. Several validation strategies were applied to mitigate these risks.

First, data triangulation was conducted by comparing issue statistics across multiple project artifacts including issue tracking logs, sprint reports, and migration governance documentation. This process ensured consistency between reported issue metrics and operational records. Second, source cross-validation was performed by examining communication records and meeting documentation to verify issue resolution timelines and coordination practices among distributed teams. Third, artifact consistency verification was applied by including only issue records that appeared consistently across multiple project artifacts in the final dataset.

Triangulation is widely recognized as an effective strategy for improving reliability and validity in qualitative case study research, particularly in studies relying on organizational artifacts as primary data sources [15].

3. RESULTS AND DISCUSSION

The results of the ServiceNow version upgrade project indicate that the migration process followed Agile best practices throughout its execution. Agile principles such as flexibility, iterative development, and close collaboration were consistently applied throughout the migration lifecycle. The upgrade activities were organized into iterative cycles, allowing feedback from testing, development, and integration phases to be continuously incorporated into subsequent iterations.

From the perspective of the proposed Global ServiceNow Migration Governance Model, the migration results demonstrate the interaction between Agile governance practices, migration coordination mechanisms, and operational outcomes. Agile practices such as sprint-based testing cycles and backlog prioritization provided the operational foundation for iterative issue resolution. At the same time, governance mechanisms including issue tracking systems, structured communication channels, and cross-regional coordination supported the monitoring and management of migration activities across globally distributed teams. These combined mechanisms enabled the migration execution process to proceed through controlled validation stages, ultimately contributing to the low proportion of production-level issues observed after deployment.

The project involved 1,527 global team members distributed across the Americas, Europe, Asia, and the Middle East. Despite geographical dispersion, the project progressed through coordinated Agile practices across globally distributed teams. Multiple communication channels including daily calls, emails, group chats, and Kanban boards were used to support cross-regional coordination [17]. Overlapping work schedules across time zones enabled synchronized meetings and continuous issue handling.

During the migration process, 119 issues were identified and logged across development, regression testing, and user acceptance testing phases. Each issue was categorized based on severity (critical, high, normal, and low) and addressed iteratively within Agile sprints. Of the total issues reported, only six issues (approximately 5%) impacted the production environment, indicating effective issue containment prior to deployment. A closer examination of the issue dataset indicates that most issues were detected and resolved during pre-production testing phases, particularly during development testing, regression testing, and user acceptance testing cycles. This distribution suggests that iterative validation processes implemented during the migration effectively prevented most system defects from reaching the production environment. Early issue detection during iterative sprint cycles allowed project teams to progressively refine configurations, integration interfaces, and workflow dependencies before the final deployment stage.

The empirical findings observed in this study align with the proposed Global ServiceNow Migration Governance Model. Agile governance practices such as sprint-based iterations and backlog prioritization enabled early issue detection, while migration coordination mechanisms including issue tracking systems and structured communication channels facilitated cross-regional collaboration. These mechanisms collectively supported the migration execution process and

contributed to the observed operational outcomes, particularly the low proportion of production-level issues following deployment.

The relatively low proportion of production-impact issues (6 out of 119 total issues, or approximately 5%) suggests that most defects were successfully identified and resolved during pre-production validation stages. This outcome indicates that iterative testing and coordinated governance mechanisms played an important role in containing operational risks prior to production deployment.

1.12 Agile Mechanisms and Operational Outcomes

The observed issue containment outcome can be interpreted as the result of several interrelated Agile governance mechanisms implemented during the migration process. Iterative sprint cycles enabled continuous testing and incremental refinement of system configurations, allowing defects to be detected early in the migration lifecycle. In parallel, backlog prioritization mechanisms ensured that critical and high-severity issues were addressed immediately during testing phases, preventing unresolved defects from progressing toward production deployment. These governance practices collectively contributed to reducing the likelihood of production-level disruptions during the system upgrade.

First, the use of iterative sprint cycles enabled continuous testing and incremental issue resolution throughout the migration process. Issues detected during development testing were immediately incorporated into subsequent sprint iterations, allowing teams to progressively refine system configurations before production deployment. Such iterative testing cycles have been shown to improve defect detection and reduce the probability of unresolved issues reaching production environments [2].

Second, the backlog prioritization and issue tracking mechanisms supported structured issue triage during the migration lifecycle. Each issue identified during testing phases was recorded in the issue tracking system and prioritized based on severity classification. This backlog-driven prioritization ensured that critical and high-severity issues were addressed immediately, while lower-priority issues were resolved in subsequent iterations. Agile backlog management has been widely recognized as an effective coordination mechanism for managing defect resolution in complex IT projects [1].

Third, continuous cross-regional communication practices played an important role in accelerating issue resolution. Daily coordination meetings, collaborative messaging channels, and shared Kanban boards allowed globally distributed teams to exchange feedback rapidly and synchronize troubleshooting activities. Previous research on distributed Agile teams highlights that structured communication routines significantly improve coordination efficiency and reduce resolution time in multi-team environments [18].

These mechanisms collectively contributed to early issue detection and progressive defect resolution during pre-production testing phases. As a result, only six issues (approximately 5% of the total recorded issues) reached the production environment, suggesting that iterative Agile governance mechanisms supported effective risk containment during the migration process.

However, it is important to acknowledge that the observed low production issue rate cannot be attributed solely to Agile practices. Several contextual factors may also influence the outcome, including team expertise, maturity of the ServiceNow platform, scope of configuration changes, and the extent of pre-deployment testing coverage. Therefore, the findings of this study should be interpreted as evidence that Agile governance mechanisms supported coordination and iterative issue resolution during the migration process rather than as definitive proof of causal impact.

1.13 Interpretation and Implications

The findings indicate that Agile governance mechanisms supported effective coordination, iterative validation, and early issue detection across globally distributed teams during the migration process. Through structured sprint cycles, backlog prioritization, and continuous cross-regional communication, project teams were able to progressively identify and resolve system issues prior to production deployment, thereby limiting production-level disruptions. These results align with prior studies demonstrating that iterative governance structures and continuous testing practices improve defect detection and reduce operational risks in large-scale IT implementations [2], [5]. In complex system migration environments characterized by uncertainty, Agile's iterative governance

approach enables continuous monitoring and embedded risk mitigation through feedback loops and incremental delivery, rather than treating risk management as a separate control phase [5], [19]. This integrated approach facilitates early identification and resolution of defects, reducing the likelihood of unresolved issues reaching production and reinforcing the role of Agile governance as an effective mechanism for risk containment in enterprise system upgrades [20].

Furthermore, the ability to coordinate 1,527 globally distributed team members highlights Agile's effectiveness in addressing collaboration challenges in multi-time-zone environments. Large-scale Agile implementations in highly regulated environments are often characterized by governance complexity and structural constraints that require disciplined coordination mechanisms [12]. In parallel, studies on distributed Agile teams emphasize that trust-building practices and structured communication routines are essential for maintaining alignment across geographically dispersed units [18]. The use of multi-channel communication and synchronized scheduling demonstrates how Agile practices can facilitate alignment and accountability in geographically dispersed teams.

This study extends prior research by demonstrating how Agile governance mechanisms operate in a big-bang ServiceNow migration within a highly regulated industry context. While [5] conceptualize Agile practices as integrated risk governance mechanisms, this study provides empirical evidence from a large-scale system migration context.

While prior studies primarily examine Agile practices in software development environments, this study demonstrates how Agile governance mechanisms operate within enterprise-scale ITSM platform migration projects involving globally distributed operational teams.

The observed production-impact rate (5%) is considerably lower than defect leakage rates reported in several large-scale IT implementation studies, which typically range between 10–20% depending on project complexity [2], [10]. This comparison suggests that iterative governance mechanisms implemented during the migration may have contributed to improved defect containment prior to production deployment.

In particular, the findings complement the integrative Agile–ITSM framework proposed by [6] demonstrating how Agile practices function as operational risk mitigation mechanisms during enterprise-wide ServiceNow upgrades. Iterative approaches have also been shown to support smoother system transitions by enabling early issue resolution and reducing operational disruptions during major system changes [10].

4. CONCLUSION

This study demonstrates that Agile methodology can function as an effective governance approach for managing complexity and mitigating operational risks during large-scale enterprise system migrations. The case study of a global ServiceNow upgrade within a multinational tobacco company shows that iterative testing cycles, continuous feedback mechanisms, and cross-regional coordination practices help limit production-level disruptions during big-bang platform deployments. These findings extend existing Agile–ITSM literature by providing empirical evidence that Agile governance mechanisms can support not only software development processes but also enterprise-wide infrastructure migration initiatives in highly regulated organizational environments [2], [5].

From a theoretical perspective, this research extends Agile–ITSM literature by providing evidence that Agile governance mechanisms operate not only in software development contexts but also in enterprise-wide system migrations. From a practical perspective, the study offers actionable insights for CIOs and IT governance leaders in designing migration strategies for large-scale enterprise platform upgrades where rollback options may be unavailable.

This study is based on a single case study examining a global ServiceNow migration within a multinational organization operating in a highly regulated industry. Consequently, the findings should be interpreted with caution when generalizing to other organizational contexts. Case study research typically prioritizes analytical insight and contextual understanding rather than statistical generalization across populations [15].

These findings suggest that organizations planning large-scale enterprise platform upgrades may benefit from integrating Agile governance mechanisms with structured migration validation processes to reduce operational risks during system deployment.

Despite this limitation, the case provides valuable empirical insights into how Agile governance mechanisms operate in large-scale enterprise system migration projects involving globally distributed teams and complex operational dependencies. Such contexts represent environments where coordination complexity and operational risks are particularly high, making them relevant settings for examining Agile-based governance practices in enterprise IT infrastructure transitions.

Therefore, the contribution of this study lies primarily in providing analytical insights into the mechanisms through which Agile practices support coordination, iterative validation, and issue containment during system migration processes. Future research may extend these findings by conducting comparative studies across different organizations, industries, or migration strategies to further evaluate the transferability of the proposed migration governance framework.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to the academic advisors, faculty members, and colleagues who provided valuable guidance, constructive feedback, and support throughout the preparation and completion of this research. The authors also appreciate the contributions of all individuals who assisted in the development and refinement of this study. This study received no specific financial support from any public, commercial, or non-profit funding agency.

CONFLICT OF INTEREST

The authors declare that there are no known conflicts of interest related to this study. The research is presented solely for academic purposes and does not represent or promote the interests of any specific company, organization, or commercial entity.

DATA AVAILABILITY

The data associated with this study are not publicly available due to confidentiality and access restrictions. The materials analyzed in this study were obtained from project-related documentation and records. Further information may be made available by the authors upon reasonable request, subject to applicable restrictions.

AUTHOR CONTRIBUTIONS

Tajghina Qatrunnada Firdaus was responsible for the conceptualization and overall execution of the research, including methodology, data curation, investigation, formal analysis, project administration, funding acquisition, and preparation of the original manuscript. **Teguh Raharjo** contributed to the conceptualization of the study and provided academic supervision and guidance throughout the research process, as well as reviewing and editing the manuscript. **Anita Nur Fitriani** contributed through academic supervision and provided feedback and support in reviewing and refining the manuscript.

REFERENCES

- [1] S. Avancha, O. Goel, and P. Pandian, "Agile Project Planning and Execution in Large-Scale IT Projects," *Darpan International Research Analysis*, vol. 12, pp. 239–252, Feb. 2024, doi: 10.36676/dira.v12.i3.80.
- [2] C. Koudriachov, C. Tam, and M. Aparicio, "Success with Agile Project Management: Looking back and into the future," *Journal of Systems and Software*, vol. 226, p. 112428, Mar. 2025, doi: 10.1016/j.jss.2025.112428.
- [3] E. Hanser, "Software Process Models," in *Agile Processes – Agile Teams*, E. Hanser, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2026, pp. 1–9. doi: 10.1007/978-3-662-72726-3_1.

- [4] J. Silva, A. Silva, A. Araújo, and A. Silva, "Agile Project Management in Government Software Development: Addressing Challenges in Education Public Policy," in *International Conference on Enterprise Information Systems, ICEIS - Proceedings*, Science and Technology Publications, Lda, 2025, pp. 340–347. doi: 10.5220/0013434800003929.
- [5] S. Singh, G. Madaan, A. Singh, K. Sood, S. Grima, and R. Rupeika-Apoga, "The AGP Model for Risk Management in Agile IT Projects," *Journal of Risk and Financial Management*, vol. 16, no. 2, p. 129, 2023, doi: 10.3390/jrfm16020129.
- [6] M. Mora, J. Marx Gomez, P. Y. Reyes-Delgado, and O. Adelakun, "AN INTEGRATIVE AGILE ITSM FRAMEWORK OF TENETS AND PRACTICES—ITS DESIGN AND EXPLORATORY UTILIZATION," *Journal of Organizational Computing and Electronic Commerce*, vol. 32, no. 2, 2022, doi: 10.1080/10919392.2022.2068905.
- [7] J. Reimon Batmetan, "Agile IT Governance for Sustainable Digital Transformation in Universities," *International Journal of Information Technology and Education (IJITE)*, vol. 4, no. 4, pp. 1–18, 2025, [Online]. Available: <http://ijite.jredu.idhttp://ijite.jredu.id>
- [8] M. I. Sarwar, Q. Abbas, T. Alyas, A. Alzahrani, T. Alghamdi, and Y. Alsaawy, "Digital Transformation of Public Sector Governance With IT Service Management—A Pilot Study," *IEEE Access*, vol. 11, pp. 6490–6512, 2023, doi: 10.1109/ACCESS.2023.3237550.
- [9] F. Dande, X. Li, M. A. Shofoluwe, and A. McLeod, "ITIL Service Configuration Management as a Driver of Enterprise Digital Transformation: A Quantitative Research Study," *Communications of the IBIMA*, pp. 1–8, Jun. 2024, doi: 10.5171/2024.917754.
- [10] B. Johansson, F. Waldau, and O. Ahlström, "What needs making information systems integration successful in the case of mergers and acquisition," in *Procedia Computer Science*, Elsevier B.V., 2023, pp. 619–625. doi: 10.1016/j.procs.2023.01.331.
- [11] E. Hanser, "MAP—Meta Agile Process Model," in *Agile Processes – Agile Teams*, E. Hanser, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2026, pp. 143–181. doi: 10.1007/978-3-662-72726-3_9.
- [12] J. A. Hüllmann, K. Kimathi, and P. Weritz, "Large-Scale Agile Project Management in Safety-Critical Industries: A Case Study on Challenges and Solutions," *Information Systems Management*, vol. 42, no. 2, pp. 138–160, Mar. 2025, doi: 10.1080/10580530.2024.2349886.
- [13] Q. Huang and Y. Tang, "Enterprise Digital Transformation Strategy: The Impact of Digital Platforms," *Entropy*, vol. 27, no. 3, 2025, doi: 10.3390/e27030295.
- [14] S. Saxena and S. Pal, "Agile and Scrum Hybrid Model for SDLC: Efficacy Analysis in Automotive Industry," in *Intelligent Systems*, S. K. Udgate, S. Sethi, G. Ghinea, and S. K. Kuanar, Eds., Singapore: Springer Nature Singapore, 2025, pp. 421–434.
- [15] R. K. Yin, *Case Study Research and Applications: Design and Methods*, 6th ed. SAGE Publications, 2018.
- [16] J. Delisle, C. Marnewick, and A. Romero-Torres, "Exploring temporal dimensions of benefits realisation management in agile IT environments," *International Journal of Information Systems and Project Management*, vol. 13, pp. 1–17, Jul. 2025, doi: 10.12821/ijispm130403.
- [17] O. Mojabi, M. Svahnberg, and M. Unterkalmsteiner, "Navigating Uncertainty and Adaptability: A Survey on the Role of Kanban and Scrum in Software Startups," in *Software Engineering and Advanced Applications: 51st Euromicro Conference, SEAA 2025, Salerno, Italy, September 10–12, 2025, Proceedings, Part III*, Berlin, Heidelberg: Springer-Verlag, 2025, pp. 263–279. doi: 10.1007/978-3-032-04207-1_18.
- [18] S. Tyagi, R. Sibal, and B. Suri, "Empirically developed framework for building trust in distributed agile teams," *Inf. Softw. Technol.*, vol. 145, p. 106828, May 2022, doi: 10.1016/j.infsof.2022.106828.
- [19] B. Li, Z. Tan, A. Arreola-Risa, and Y. Huang, "On the improvement of uncertain cloud service capacity," *Int. J. Prod. Econ.*, vol. 258, p. 108779, Apr. 2023, doi: 10.1016/j.ijpe.2023.108779.

- [20] M. Khasabah, Q. Al-Hammouri, N. Nusairat, S. Freijat, E. Alqararah, and S. AlFraihat, "The impact of risk management and agile methodology on cybersecurity project success: the mediating role of team collaboration," *Journal of Project Management*, vol. 10, pp. 737–744, Feb. 2025, doi: 10.5267/j.jpm.2025.7.004.