Article

Navigate Change and Innovation: Strategically Integrating Emerging Technologies in Release Management

Bishnu Shankar Satapathy * , Ashish Sharma , Madhusmita Dash , S. Ibotombi Singh * , Siddhartha Sankar Satapathy , Joya Chakraborty

Posted Date: 17 April 2024

doi: 10.20944/preprints202404.1184.v1

Keywords: Release Management; Release Readiness Strategy; Process Management; Change Management; Industry 5.0; Software as Service

 doi:10.20944/preprints202404.1184.v1

Article

Navigate Change and Innovation: Strategically Integrating Emerging Technologies in Release Management

Bishnu Shankar Satapathy [1,*], Ashish Sharma [2], Madhusmita Dash [3], S. Ibotombi Singh [4,*], Siddhartha Sankar Satapathy [4] and Joya Chakraborty [5]

[1] Center for Multidisciplinary Research
[2] Birla Institute of Management and Technology, Greater Noida (NCR)-201306, Uttar Pradesh, India
[3] Department of Electronics and Communication Engineering, NIT, Jote-791113, Arunachal Pradesh, India
[4] Department of Computer Science and Engineering
[5] Department of Mass Communication and Journalism, Tezpur University, Napaam-784028, Assam, India
* Correspondence: bishnu_mdp21@tezu.ernet.in (B.S.S.); sis@tezu.ernet.in (S.I.S.); Tel.: +91-9951993400 (B.S.S.); +91-3712-275106 (S.I.S.)

Abstract: The enterprise software sector is experiencing a significant transformation driven by rapid technological progress and evolving business needs. An effective release management process is crucial for the solution providers and customers, as they incorporate the principles of Industry 4.0 and 5.0. With the growing dependence on software solutions and services, ensuring their quality, reliability, and security is highly necessary. The release management process incorporates strategic planning, regression testing, accurate deployment, and continuous monitoring of software updates. However, the rapid advancements in artificial intelligence (AI) and machine learning (ML) create a unique challenge to the release management process. The existing research needs to address the necessary adaptations for Industry 5.0, mainly in product-based software organizations, to the release management process without a change management system. After working with several software solution stakeholders, based on the responses from SaaS project stakeholders, we have presented a revised software release readiness workflow by incorporating a change management module. The preliminary result based on data over a limited period of time indicates that the revised workflow has the potential to improve the management of slippage, particularly by more effectively addressing NFR-related issues, thereby resulting in more predictable and reliable release cycles.

Keywords: release management; release readiness strategy; process management; change management; Industry 5.0; software as service

1. Introduction

Software release management is the process of planning, testing, deploying, and monitoring software updates in a systematic and coordinated way. It intends to ensure the quality, reliability, and security of software products and services. Software release management is a necessary process that needs careful planning, coordination, and execution to deliver updates to end users effectively. It involves detailed planning, scheduling, scoping releases, establishing timelines, and distributing resources. New features and bug fixes, implemented during development through version control systems and continuous integration practices, were released to the customer after end-to-end testing was conducted using various procedures to ensure software quality and compliance with requirements. Non-functional requirements (NFRs), modern technologies, and release management are imperative in modeling user experience, system performance, and the overall accomplishment of software applications. It is essential to have a well-planned release management approach, including best practices, to minimize risks and interruption and maintain customer satisfaction during software releases [1].

 doi:10.20944/preprints202404.1184.v1

Owing to its crucial role in ensuring the quality, reliability, and security of software products and services, release management may help keep up with the promptly emerging artificial intelligence (AI) and machine learning (ML) technologies.

The development and delivery of Native SaaS projects is a sophisticated process. The irregular non-readiness of software solution functionalities is a common issue. In addition, the other issue is that the engineers involved in developing a SaaS product can only extend support for heritage solutions when the SaaS-based solution is fully ready. Engineers' confidence may be impacted if they do not understand the implications of customer constraints. If the SaaS-based solution provider fails to deliver novel solutions on time, other technology partners may hesitate to collaborate with customers. Empirical challenges have also been observed in the release readiness process when an on-prem solution gets migrated/developed on a cloud platform.

This article illuminates the significance of Release Management (RM) and Organizational Change Management (OCM) in digital transformation, specifically in moving from traditional on-premises solutions to enterprise cloud-based SaaS solutions. Change management is necessary for successful software development, ensuring the implementation of new software solutions within organizations by easing resistance and proposing practical change management approaches [2].

The primary objective of release management is to ensure that software updates are released efficiently and effectively. As technology advances, release management also adapts, incorporating new methods to meet higher quality standards and accommodate rapid industry changes [3]. Industry 5.0 brings a new dimension to this field. It emphasizes the well-being of workers and sustainable practices, using technology to benefit society. Understanding how release management fits into this new industrial paradigm is crucial for future software development [5,6].

Release Readiness Strategy (Figure 1) is pivotal for orchestrating software development planning, scheduling, and control within each cycle and across diverse platforms, including on-premises and cloud environments. This strategy is instrumental in preserving the application's robustness upon deployment to the customer. It encompasses a comprehensive plan that guides all stakeholders through the software development life cycle (SDLC) stages, ensuring a cohesive and predictable release process [7].

A firm release readiness strategy in a cloud environment is essential to ensure solutions' quality, performance, sustainability, traceability, and security. The Release Readiness Strategy process directs planning, scheduling, and controlling solution releases. Release management ensures application reliability, releasing the right components at the right time with the necessary quality. The release manager ensures that cloud technology solutions meet release validation criteria, certifying the final release gateway. Technology partners define code, test, performance, stability, reliability with scalability (PSR), and security criteria, including additional parameters for deployment, observation, monitoring, internationalization, documentation, and translation. These gates guarantee that the release of a solution complies with DevOps quality processes [4,13].

 doi:10.20944/preprints202404.1184.v1

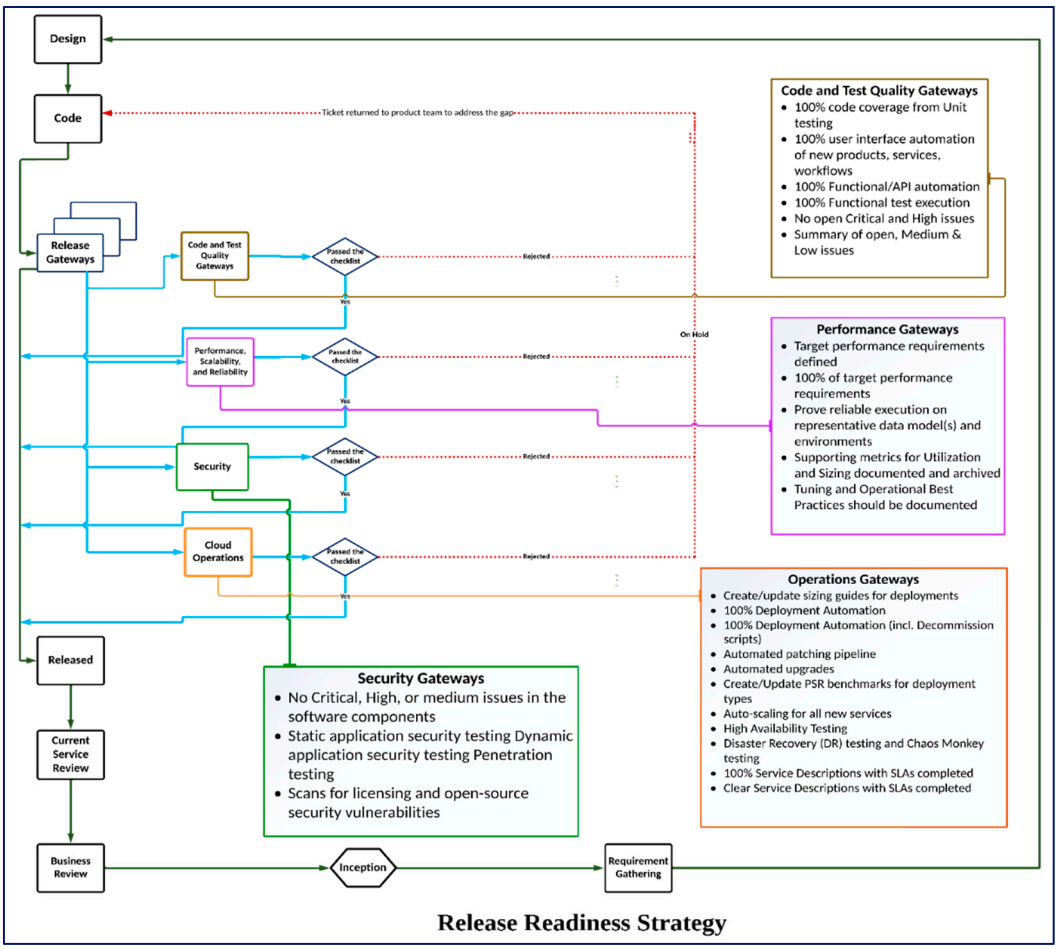

Figure 1. Illustrates the traditional software solution's various gateways before the release without the change control board and release gateway keepers in place**.**

1.1. Release Readiness Strategy

Since the last decade, from the existence of public cloud platforms like AWS, Azure, and GCP, technology partners have been trying to move completely from the traditional on-premises solution to the SaaS business model. Transitioning to a Native SaaS solution involves complexities like customer focus, observability, and clear release roadmaps. It makes release management an integral part of development activities. DevOps now automated and streamlined the release management process with a more sizable number of SaaS-based releases [15]. Various authors related to DevOps say, "You build it. You own it. You run it" [8]. According to the experts, the development teams embrace a DevOps mindset and take responsibility for the SaaS services' build, deployment, and efficient operations in the SaaS world [16]. Starting from the design phase, Figure 1 illustrates the traditional software solution's various gateways before the release without the change control board and release gateway keepers in place [22].

1.2. Release Management Tools Used at Present by the Technology Partner

The organization picks the tools based on the team's requirements and defined gateways. Release Management tools (Figure 2) are not the sole factors for effectively implementing a release process. Multiple factors influence its success. A well-defined plan, the right tool, an efficient workforce, a business strategy, and a well-defined release process are needed.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 April 2024 doi:10.20944/preprints202404.1184.v1

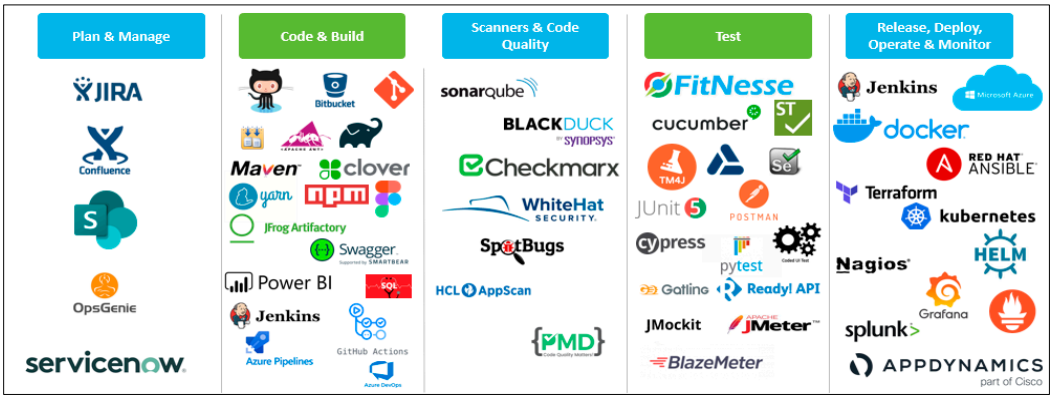

Figure 2. Illustrates various tools used during delivery, starting with planning, coding, scanning, and testing, and during the release of the solutions and monitoring the software solution after deployment.

It is believed in the software industry that more than 30% of releases are affected due to non-compliance with the release gates. Since all these issues are reported towards the end of the release cycle, the pressure on the development increases, affecting the release cycle. The internal survey indicates that most releases affected due to non-compliance with the release gates are often delivered after the due date, and other releases remain on hold as the team has no authority over decision-making. Navigating effective release management in the Industry 5.0 era demands a holistic approach to proactively tackle these challenges and deliver superior software products and services, leveraging the full potential of emerging technologies. This article delves into the impact of Industry 5.0, highlighting customer challenges in migrating from on-premises to cloud platforms. It highlights the under-researched software development release and change management area, bracing for more studies to aid developers and researchers. The crossroads of release management with Industry 5.0 is presented as a critical area for succeeding software delivery. This paper offers a profound view of organizational upgradation through research and interviews, capping in a projected workflow for ensuring software release readiness.

2. Materials and Methods

We did a keyword assessment among available research articles to understand the scale of issues considered in the research manuscripts on software release and change management. We selected the MDPI publisher library as a sample database for our keyword analysis. The library provides an easy user interface for obtaining research article metadata such as title, abstract, and author keywords. We could extract 4526 articles on SaaS-based supply chain software applications published since 2012, about which Industry 4.0 qualified and Industry 5.0 was envisioned. After screening 4526 articles using the inclusion and exclusion criteria, we retained 3900 articles (Supplementary Table S1) for our consideration.

We prepared a Google form-based questionnaire to survey stakeholders associated with different software development life cycle phases and collected feedback to understand technology partners' pain points on full-stack software development. The Google form is available at https://forms.gle/YM4mg5hdA7E4qHTB8 (accessed on). Questionnaires were designed for release planning and coordination issues, user training and support, system performance, security measures, feedback mechanisms, documentation, and continuous improvement.

3. Results

3.1. Research Literature Needs More Focus on Release and Change Management

To gauge the importance of release and change management in automated software development, we conducted a frequency distribution analysis of non-functional requirement (NFR)

 doi:10.20944/preprints202404.1184.v1

keywords across various research articles in Supplementary Table S1. We aimed to identify how emerging technologies and NFRs are prioritized in software development, specifically by analyzing author keywords from MDPI research articles. These keywords included Availability, Observability, Reliability, Scalability, Security, Usability, Resource Utilization, Collaboration, Delivery Strategy, Deployment, Documentation, Ethics, Human Machine Interaction, Change Process, Release Process, DevOps, SaaS, AI/ML, Deep Learning, and ChatGPT [18,10].

By examining these keywords, we aimed to understand the extent to which researchers focus on technology advancements during this transition from Industry 4.0 to Industry 5.0 and whether challenges related to change management and release management are being given the necessary attention. Figure 3 shows that research articles referencing modern technology-related keywords are significantly more frequent than those associated with release management, change management, and software delivery strategy [9,21].

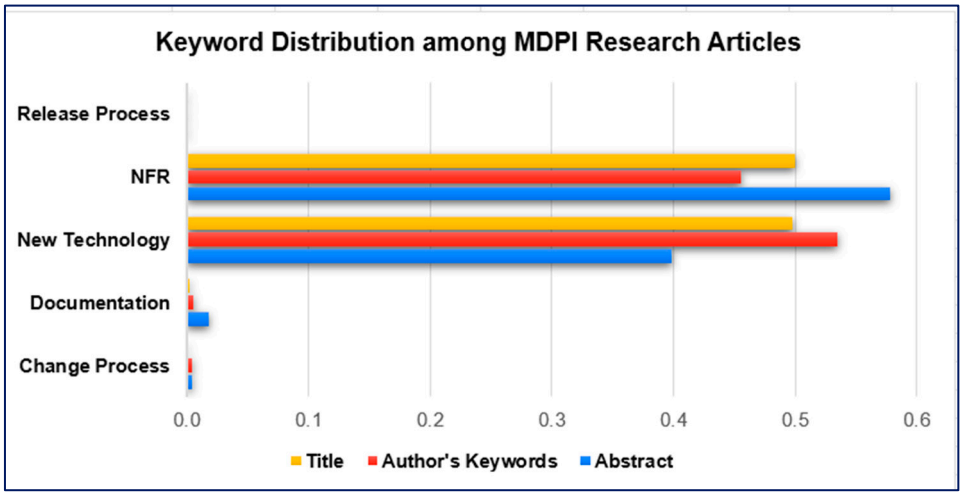

Figure 3. Depicts the frequency of the keywords in the reviewed articles analyzed in this study. The length of the horizontal bars in the figure represents frequencies of keywords of various categories in the title, author's keywords, and abstract of the research articles published in MDPI Journals. Keywords such as Availability, Observability, Reliability, Scalability, Security, Usability, Resource Utilization, Collaboration, Delivery Strategy, Deployment, and Ethics were considered under non-functional requirements (NFR), and Human Machine Interaction, DevOps, SaaS, AI/ML, Deep Learning, ChatGPT were considered under New technologies in this study. The keywords related to release and change management were rare compared to modern technology and NFR in the research articles studied.

3.2. Research Literature Needs More Focus on Release and Change Management

Inadequate software-release oversight can create unnecessary pressure to address any arising risks quickly. However, introducing a change control board (CCB) into the release management process can mitigate these risks and prevent unauthorized changes that may cause disruptions. The release management team is responsible for coordinating several teams, and without proper CCB oversight, errors can occur, and system stability may be compromised, resulting in dissatisfied customers. To better understand the challenges faced by stakeholders in the software industry, we conducted a questionnaire-based survey with those involved in various phases of the software development life cycle, from requirement analysis to maintenance [11,20,23].

To better understand the pain points of our technology partners, we surveyed them over three months, from November 2023 to March 2024. We interviewed skilled stakeholders from various software as a Service (SaaS) projects, and the information we gathered covered everything from simple solutions to complex, multi-modal ones [17,19]. As software development organizations navigate the transformative realm of Industry 5.0, they encounter various challenges and pain points that can impact the seamless adoption of innovative technologies. Our survey results revealed that

 doi:10.20944/preprints202404.1184.v1

many stakeholders faced challenges during development due to the need to adhere to release and change management policies. Respondents emphasized the importance of greater transparency from higher management, particularly regarding release management and change management strategies in adopting modern technologies. The critical concerns of technology partners regarding SaaS solution delivery are summarized in Figure 4.

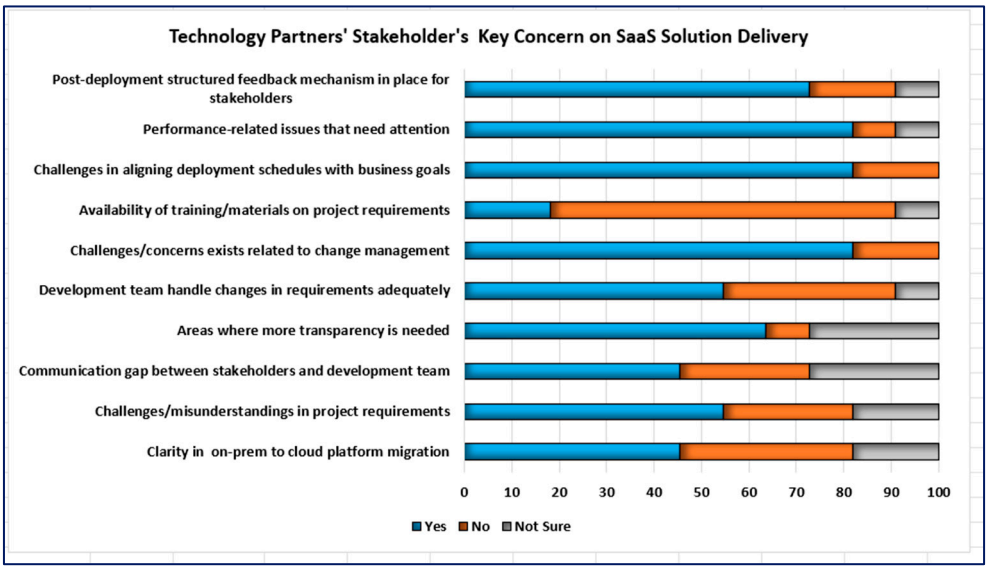

Figure 4. Technology Partners' Stakeholder's Key Concern on SaaS Solution Delivery. The length of the horizontal bars represents the proportion of responses aligning with the legends specified in the chart concerning the survey questionnaire.

There is a pressing need for standardization and regulation in developing and deploying AI and ML systems to tackle these pain points. It involves establishing uniform terminology, metrics, guidelines, and policies. Furthermore, a comprehensive review of cloud computing literature has shown that research should focus on managing and leveraging cloud technology for business purposes. It entails examining field contributions, identifying gaps, and proposing a research agenda aligning with global information management perspectives [14,24].

3.3. Challenges and Potential Solutions in Integrating Change Management

The role of release management is to safeguard efficient software updates. However, it may need to look into stakeholder participation, which may lead to deployment disruptions. It can mainly happen as Release management, in the absence of change management, primarily focuses on the technical aspects of software deployment. The stakeholders involved in the release management process cluster around version control, deployment, and timely software updates. It is aligned to predefined release schedules, allowing competent provision and resource allocation to streamline cycles, leading to predictable outcomes.

Adding a change in management role in the process will increase the engagement among business stakeholders. Appropriate change management processes can help manage the growing interdependency due to the many processes involved while developing SaaS-based solutions. When an organization transitions from an on-premises version of a SaaS-based solution to a more complex one, it becomes more complicated and involves various stakeholders and interdependencies. In this scenario, a relevant change management approach can be crucial in ensuring a smooth transition and overcoming the associated challenges. However, the change in the middle of the cycle may create a risk of disruption due to unexpected issues during deployment. It may be a failure to leverage change-related insights. Change management, however, maintains balance uniqueness with stability, aligning releases with strategic goals and reducing deployment issues. Integrating change management into the release process can enhance software deployment, align it with organizational

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 April 2024 doi:10.20944/preprints202404.1184.v1

changes, and use insights for improvement. The change management strategies can help address this issue by ensuring clear communication and stakeholder involvement [25].

Release management integrated with change management can enhance the release process. Change management considers organizational changes beyond software releases that balance innovation and stability, ensuring alignment with strategic goals. Rigorous approvals minimize disruptions in the development process and help continuous improvement in the overall development process informed by change-related insights. It involves business stakeholders leveraging their insights by aligning with business goals and supporting organizational objectives. However, integrating change management may increase complexity due to the rigorous approvals required that may extend timelines. It may also be resource-intensive and require additional communication efforts.

3.4. Proposed Release Management Workflow Catering Change Management

To mitigate the technology partners' pain points for the stakeholders, we have modified the workflow described in the introduction to ensure better quality delivery (Figure 5). The workflow intends to provide end-to-end visibility to all the project stakeholders involved in the project. In addition, when any bottleneck is created, the decision-making authority plays the change management role to ensure seamless project delivery.

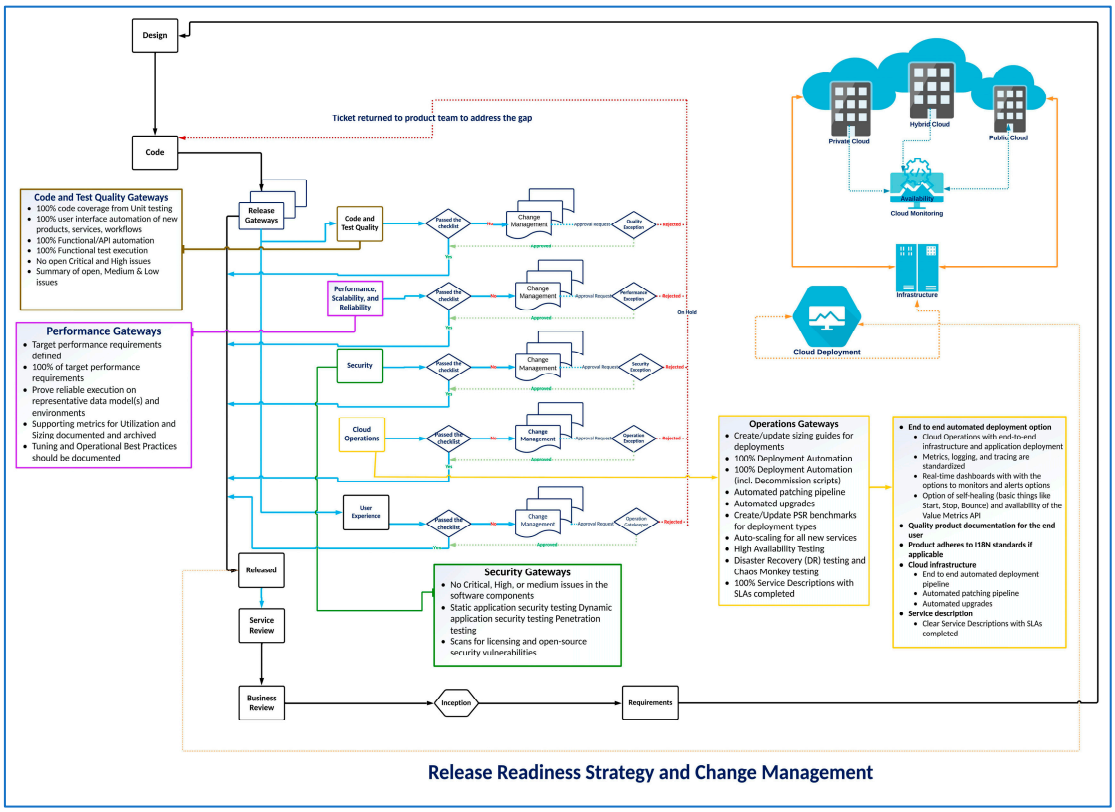

Figure 5. Illustrates the software solution's various gateways before the release, accommodating the change management and additional gatekeepers.

In this workflow, the software undergoes rigorous release gateways to evaluate deployment readiness while change management processes maintain accountability for any alterations. After clearing the gateways, a business review confirms the software's alignment with strategic objectives and stakeholder expectations. Exception management strategies protect against unforeseen issues during the release. An operational model connecting development and operations is vital for SaaS solutions, along with a formal approval process for production deployment. This structured

 doi:10.20944/preprints202404.1184.v1

approach to release management boosts software quality and reliability, leading to greater user satisfaction and business success.

The Change Control Board (CCB)'s participation is critical in prioritizing changes based on objectives and ensuring transparency and consistency in the process. With their supervision, compliance with regulations may be protected, and organizations may avoid facing penalties. The Change Control Board can validate requests for release that need to clear the gates. If the board approves the request, the release will proceed with an exception. However, if the request is rejected, it will be returned to the development team for further review. The board uses critical parameters like the reason for the change, business case or impact, downstream impact, and the platform and domain of the product concerned to make its determinations.

All requests submitted for review must contain all essential information to be considered by the Change Control Board. Ensure that the request includes the following details: - Reason for Change - Business Case/Impact - Downstream Impact - Platform and Domain of the product being released. The Change Control Board will only evaluate requests that contain all the necessary data.

Ignoring the Change Control Board (CCB) from the Release Management process can result in risks that may lead to interruptions due to unlawful changes. The Release Management team is liable for bringing together various teams, and the absence of CCB oversight can lead to errors and decrease system steadiness, affecting customer fulfillment. To avoid security risks posed by unapproved alterations, the CCB needs to be involved in appropriate assessment and governance. Including the CCB in the release management process is crucial for safeguarding the organization's reputation and ensuring success.

3.5. NFR Compliance and Slippage Analysis

To understand the impact of change management's role in release management, we collected inputs over a limited time period (organization and project details not disclosed here due to confidentiality issues) for the exception slippage analysis, which is a crucial aspect of software release management. It involves examining a software project's progress, looking for deviations from the planned schedule and effort. This analysis helps identify the additional effort required to meet the reliability objectives within the scheduled time or, if necessary, to reschedule the project's delivery.

From the collected data (Figure 6), over the four quarters of the past year, the software development cycle has included four major and twelve minor releases. This report focuses on analyzing the adherence to NFRs and evaluating the slippage based on unresolved NFR issues.

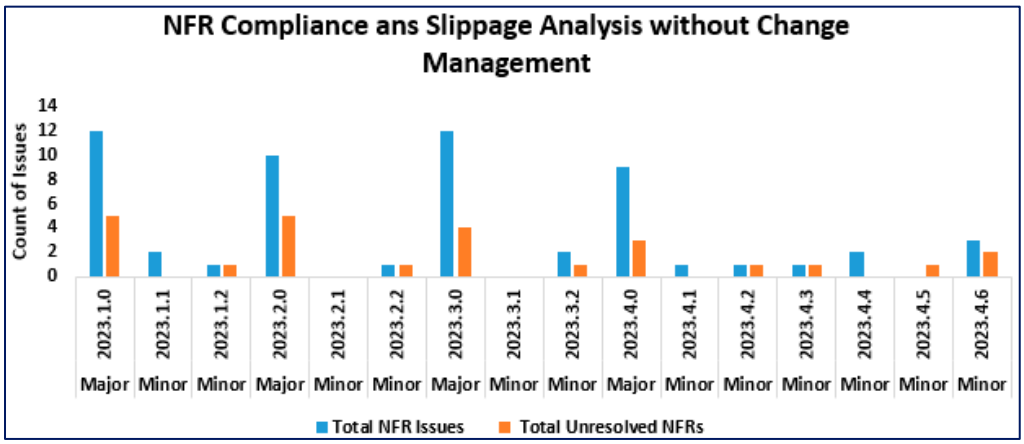

Figure 6. Illustrates the slippage of the NFRs due to the absence of the right authority to manage the issues and make strategic decisions.

The major releases were substantial, incorporating significant feature additions and system overhauls. Twelve NFR issues were reported in the first major release of the last year. Out of which five were left unresolved. The reported unresolved NFRs are security issues mainly reported due to

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 April 2024 doi:10.20944/preprints202404.1184.v1

the code scanning application's last-minute upgrade. During a release, such a slippage can have a negative impact on software quality and reliability. The same slippage trend continued in the second major release also. Out of 10 reported issues, five remain unresolved. The team needed more time to fully adapt it despite the difficulty of integrating new security measures where the new cloud-based application has been introduced to provide a better-secured solution. The stakeholders used an on-premises version of the code scan tool for the previous releases. Three of the reported nine NFR issues remain unresolved in the fourth major release. This shows a high need to improve NFR planning and execution.

The minor releases that support specific functionalities and bug fixes have fewer NFR issues reported. However, the ratio of slippage that occurred was lower. Still, unresolved NFRs can impact the release quality. Overall analysis shows that while minor releases upheld good NFR compliance, major releases battled with unresolved NFR issues, leading to a higher range of slippage. It indicates a need for a tactical evaluation of NFR management, especially for major releases. It will ensure better arrangement between planned and actual efforts and minimize the impact of unresolved NFRs on release schedules. Incorporating a Change Management Board (CMB) into the release process can be instrumental in improving project timelines and reducing slippage. The CMB's role is administering and approving changes, ensuring that each release is thoroughly assessed for prospective safety and quality hazards and impacts customer satisfaction.

The chart (Figure 7) demonstrates the positive effect of CMB involvement on the slippage of major and minor releases over three months. The data compares releases before and after the CMB was introduced into the process.

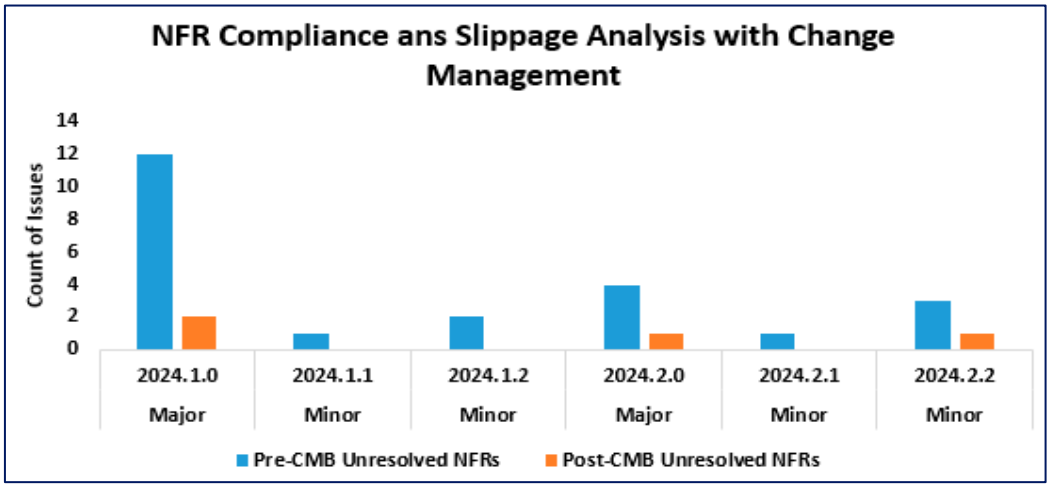

Figure 7. Illustrates the slippage of the NFRs after the appointment of the right authority as CBM to manage the issues and make a strategic decision.

This chart provides a concise overview of the improvement in slippage in the release process due to the CMB's involvement. The 'Pre-CMB Unresolved NFRs' column shows the number of unresolved non-functional requirements before the CMB's integration, while the 'Post-CMB Unresolved NFRs' column reflects the number after the CMB's involvement. The 'Slippage Improvement' column highlights the progress made from one level of slippage to a more favorable one. The CMB's influence is evident in the major releases, where unresolved NFRs decreased significantly, improving slippage from 'High' to 'Moderate' or 'Low.' The minor releases also benefited from the CMB's oversight, with all the listed releases moving from having unresolved NFRs to none, indicating a successful mitigation of slippage. This preliminary result based on data from a limited period suggests that the CMB's active role in the release process has the potential to improve the management of slippage, particularly by more effectively addressing NFR-related issues. It has resulted in more predictable and reliable release cycles.

 doi:10.20944/preprints202404.1184.v1

4. Conclusions

Product quality is essential; collaboration between release management and emerging technologies like AI and ML is critical. The change management process is crucial in technology adoption, but a gap between these areas can negatively impact software quality and an organization's reputation. To bridge this gap, organizations may need to employ tools for release management and emerging technologies, ensuring industry standards and best practices. Addressing these challenges requires a holistic transformation, including upgrading organizational capability, engaging stakeholders, and establishing a software release readiness workflow. Finding a suitable platform, aligning teams with a shared vision, fostering a collaborative culture, and leveraging best practices can all be challenging. However, effective collaboration provides significant benefits that are well worth the effort. It can lead to delivering high-quality software products and services, thus finding the best balance and approach for the organization.

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org. Supplementary Material has been provided in the form of a spreadsheet (Supplementary Table S1).

Author Contributions: Bishnu Shankar Satapathy: Conceived the study, performed research, developed required methodology, analyzed data, and authored the paper; Ashish Sharma, Madhusmita Dash, and Siddhartha Sankar Satapathy: analyzed data and authored the paper; S. Ibotombi Singh and Joya Chakraborty: authored the paper and supervised the research.

Funding: This research received no external funding.

Data Availability Statement: All the data has been shared as Supplementary Material.

Acknowledgments: Bishnu Shankar Satapathy acknowledges CMDR, Tezpur University for providing a research facility. The Grammarly (AI writing assistance) tool is used in this paper to check the language issues to make the sentences more meaningful and cohesive.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Smith, J., & Johnson, R. (2020). Release Management Best Practices: Ensuring Successful Software Updates. Journal of Software Engineering, 45(3), 123–138.
2. García, M., & Martínez, L. (2019). Change Management Strategies for Successful Software Development. International Journal of Information Technology, 32(2), 87-102.
3. Asana. (2024). Release Management: 5 Steps to Success. Retrieved from https://asana.com/resources/release-management.
4. Satapathy, B.S., Satapathy, S.S., Singh, S.I., Chakraborty, J. (2023). Continuous Integration and Continuous Deployment (CI/CD) Pipeline for the SaaS Documentation Delivery. In: Hasteer, N., McLoone, S., Khari, M., Sharma, P. (eds) Decision Intelligence Solutions. InCITe 2023. Lecture Notes in Electrical Engineering, vol 1080. Springer, Singapore.
5. Kraaijenbrink, J. (2022). What Is Industry 5.0 And How It Will Radically Change Your Business Strategy? Forbes. Retrieved from https://www.forbes.com/sites/jeroenkraaijenbrink/2022/05/24/what-is-industry-50-and-how-it-will-radically-change-your-business-strategy/
6. Mannam, P. K. (2023). How New Tech Elevates Release Management's Quality Standards. DevOps.com. Retrieved from https://devops.com/how-new-tech-elevates-release-managements-quality-standards/
7. Doe, J. (2024). Effective Strategies for Release Readiness in Software Development. Journal of Software Management, 29(3), 45–59. doi:10.1234/jsm. v29i3.456
8. Vogels, W. (2006). Amazon's Approach to IT. Interview with Werner Vogels. Retrieved from Atlassian website: https://www.atlassian.com/incident-management/devops/you-built-it-you-run-it1
9. Jagli, D., Purohit, S., & Chandra, N. S. (2018). SAASQUAL: A Quality Model for Evaluating SaaS in the Cloud Computing Environment. In V. B. Aggarwal et al. (Eds.), Big Data Analytics, Advances in Intelligent Systems and Computing, 654, 429-437. Springer
10. Kaur, H., & Mustafa, M. Z. (2019). Evaluating the Impact of Change Control Board (CCB) in Release Management. Journal of Software: Evolution and Process, 31(3), e2147

11. Mohamed, S. (2016). Software Release Management Evolution-Comparative Analysis across Agile and DevOpsContinuous Delivery. International Journal of Emerging Trends & Technology in Computer Science. 3. 2349–6495.
12. Barqawi, Neda & Syed, Kamran & Mathiassen, Lars. (2016). Applying service-dominant logic to the recurrent release of software: an action research study. Journal of Business & Industrial Marketing. 31. 928-940. 10.1108/JBIM-02-2015-0030.
13. DOI Smith, D. R. (2022). Creation of a Unified Cloud Readiness Assessment Model to Improve Digital Transformation Strategy. International Journal of Data Science and Analysis, 8(1). DOI: 10.11648/j.ijdsa.20220801.
14. Omurgonulsen, M., & Yiğit Kazançoğlu, M. (2021). Cloud Computing: A Systematic Literature Review and Future Agenda. Journal of Global Information Management. DOI: 10.4018/JGIM.20230101.oa1
15. Suescun Monsalve, Elizabeth & Calvache, César & Muñoz, Sergio & Uribe, Alejandro. (2021). DevOps in Industry 4.0: A Systematic Mapping. Revista Facultad de Ingeniería. 30. 1-16. 10.19053/01211129.v30.n57.2021.13314.
16. Feng, Haiyang and Jiang, Zhengrui and Liu, Dengpan, (2017). Quality, Pricing, and Release Time: Optimal Market Entry Strategy for New Software-as-a-Service Vendors
17. Feng, H., Jiang, Z. and Liu, D., (2018). Quality, Pricing, and Release Time: Optimal Market Entry Strategy for Software-as-a-Service Vendors. MIS Quarterly, 42(1), pp.333-353., Available at SSRN: https://ssrn.com/abstract=2663868 or http://dx.doi.org/10.2139/ssrn.2663868.
18. Mohamed, S. I. (2017). Software Release Management Evolution-Comparative Analysis across Agile and DevOpsContinuous Delivery. International Journal of Emerging Trends & Technology in Computer Science1
19. (2019). Release cycle management: an action research study into a software company. International Journal of Business Information Systems, 30(2), 152–1762
20. Krishna Kaiser, A. (2023). Release Management in DevOps. In: Reinventing ITIL® and DevOps with Digital Transformation. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-9072-9_10
21. Montero Pérez, O. (2021). Adapting a quality model for a Big Data application: The case of a feature prediction system. https://core.ac.uk/download/481424509.pdf
22. Tan, B., Zhu, Z., Pan, J., & Wang, X. (2023). Modeling Multi-Generation Product Diffusion in the Context of Dual-Brand Competition and Sustainable Improvement. Sustainability, 15(17), 12920.
23. Bharathi, K., & Sree, K. (2015). Recent Developments on Data Warehouse and Data Mining In Cloud Computing. International Journal of Science, Engineering and Computer Technology, 5(2), 31–34.
24. Almuqrin, A., Almuqrin, A., Alomran, A., & Zhang, J. (2023). Information System Success for Organizational Sustainability: Exploring the Public Institutions in Saudi Arabia. Sustainability, 15(12), 9233.
25. Phillips, J., Klein, J.D. Change Management: From Theory to Practice. TechTrends 67, 189–197 (2023). https://doi.org/10.1007/s11528-022-00775-0