Future of Release Management (Firmware/Software Delivery) in the Era of Generative AI

Dominion Nicholas and Bhargav Sharma *

Posted Date: 28 February 2026

doi: 10.20944/preprints202602.1974.v1

Keywords: AI in DevOps; AI-powered CI/CD; firmware release automation; firmware release automation

 doi:10.20944/preprints202602.1974.v1

Article

Future of Release Management (Firmware/Software Delivery) in the Era of Generative AI

Dominion Nicholas [1] and Bhargav Sharma [2,*]

[1] Independent Researcher, India

[2] Independent Researcher, USA

* Correspondence: prosper1149j@gmail.com

Abstract

In the era of connected and cyber-physical systems, firmware and software releases management in industries like IoT, smart grids, medical devices, UAVs, satellites, and vehicles has been growing more complex. These systems require constant, secure, and reliable updates to keep up with increasing user expectations and regulatory requirements. However, the growing complexity of release pipelines, as well as increasing security risks and tight regulatory demands, make it a challenging process. The emergence of Generative AI (GenAI), and Large Language Models (LLMs) in particular, provides a transformative opportunity for innovating traditional release management processes and provides the ability to automate documentation, triaging defects, creating risk assessment, and compliance. This article is aimed at discussing a GenAI-augmented vision for release management in critical systems that provides a conceptual framework built on current literature in OTAs updates, cyber security, and AI in product development. Through a conceptual and integrative review, we provide a number of use case scenarios, in which GenAI can be advantageously used to strengthen release pipelines, such as automation for root cause analysis (RCA) reports, intelligent defect triage, and risk mitigation strategies. We also address the ethical considerations and reliability risks associated with integrating AI in safety-critical environments, including emphasizing the importance of human oversight and governance.

Keywords: AI in DevOps; AI-powered CI/CD; firmware release automation; firmware release automation

1. Introduction

The rapid proliferation of connected and cyber-physical systems has fundamentally transformed the landscape of firmware and software delivery. Industries such as Internet of Things (IoT), smart grids, autonomous vehicles, medical devices, unmanned aerial vehicles (UAVs), satellites, and advanced manufacturing now depend on continuous, secure, and reliable updates to maintain functionality, address vulnerabilities, comply with evolving regulations, and meet rising user expectations. Unlike traditional software environments, these systems operate within tightly coupled hardware–software ecosystems, often under strict real-time, safety, and reliability constraints. Consequently, release management has evolved from a periodic deployment function into a mission-critical discipline that directly impacts system safety, resilience, and business continuity.

Modern release pipelines must orchestrate complex workflows that include firmware builds, validation testing, security scanning, compliance checks, risk assessment, documentation, and over-the-air (OTA) deployment across geographically distributed fleets of devices. The growing scale and heterogeneity of these environments introduce significant operational and governance challenges. Security threats such as supply chain attacks, firmware tampering, and communication-layer exploits further complicate release operations, particularly in safety-critical domains where failures can have severe societal consequences. In parallel, regulatory expectations around traceability, auditability, and quality assurance continue to intensify, demanding comprehensive documentation and

 doi:10.20944/preprints202602.1974.v1

evidence-driven decision-making throughout the release lifecycle.Against this backdrop, Generative Artificial Intelligence (GenAI), particularly Large Language Models (LLMs), presents a transformative opportunity to reimagine release management. While AI has already demonstrated value in areas such as anomaly detection, intrusion detection systems, and predictive maintenance, the emergence of GenAI introduces capabilities that extend beyond pattern recognition. GenAI systems can synthesize technical documentation, analyze historical defect data, assist in root cause analysis (RCA), generate risk assessments, summarize compliance evidence, and support intelligent triage of defects. These capabilities enable a shift from reactive, manual release workflows toward adaptive, data-driven, and semi-autonomous release orchestration.

However, integrating GenAI into firmware and software delivery pipelines—especially in safety-critical and regulated environments—requires careful consideration. The probabilistic nature of LLM outputs, potential hallucinations, bias risks, and explainability limitations introduce new reliability and governance concerns. In contexts such as medical devices, energy systems, and aerospace platforms, erroneous AI-generated insights could propagate operational or safety risks if not properly supervised. Therefore, the future of release management in the GenAI era must balance automation with robust human oversight, transparent governance models, and clearly defined accountability structures.This article presents a forward-looking vision for GenAI-augmented release management in connected and critical systems. Through an integrative review of literature across OTA updates, cybersecurity, AI-driven software engineering, and regulatory compliance, we develop a conceptual framework for embedding GenAI capabilities into release pipelines. We identify high-impact use cases—including automated RCA reporting, intelligent defect classification, compliance documentation generation, risk scoring, and deployment readiness assessments—and analyze their potential operational and strategic benefits.

In addition, we examine the ethical, security, and regulatory implications of AI-augmented release processes and propose governance mechanisms to ensure safe adoption. The discussion also explores how professional roles within release organizations—such as Technical Program Managers (TPMs), release engineers, quality assurance leaders, and cybersecurity specialists—are likely to evolve in AI-augmented environments. Rather than replacing human expertise, GenAI is positioned as a cognitive co-pilot that enhances decision quality, reduces cycle time, and enables more resilient release ecosystems.

As connected systems continue to evolve toward continuous, software-defined architectures, release management will become increasingly strategic. Organizations that successfully integrate GenAI into their release workflows—while maintaining rigorous safety and governance standards—will be better positioned to deliver secure, compliant, and high-quality updates at scale. This work contributes a structured perspective on that transition and outlines a roadmap for enterprises seeking to responsibly harness GenAI in the next generation of firmware and software delivery systems.

2. Literature Review

The field of release management for firmware and software delivery has advanced far with the introduction of modern technologies, such as the Internet of Things (IoT), smart grids and connected devices. As the complexity of firmware updates, deployment and management has increased in the critical infrastructure, so has the requirement for secure, reliable and efficient firmware update management. In this section, we make a cursory review of key literature about release management practices and cybersecurity in connected systems and the use of AI, especially Generative AI (GenAI) in improving release pipelines.

2.1. Firmware /Software Delivery and OTA Updates in Connected Systems

Firmware and software updates are becoming even more of what's on the air for devices that are connected systems. Firmware and software updates, particularly in connected systems, are increasingly over-the-air (OTA). These systems range from IoT devices to satellites and automotive systems; all of these systems rely on secured and efficient update mechanisms to keep these systems

 doi:10.20944/preprints202602.1974.v1

functional and secure. OTA updates make it possible to remotely distribute updates of firmware and software, and don't require physical access to devices, making them essential for remote maintenance of large-scale, geographically dispersed networks.

A remarkable example of OTA updates in the IoT sector is the use of NB-IoT over the use of GEO satellite to deliver firmware updates to machine-type terminals (Bas & Dowhuszko, 2021). Similarly, CubeSats, small satellites carrying low-power payloads need secure software updates in orbit in order to ensure mission success (Molina et al., 2023). These update systems have to deal with both the technical limitations of the devices (like low power, bandwidth) and security risks of remote updates (Solomon et al., 2019).

Furthermore, the secure technologies for Firmware IoT Device - Online Firmware Update (MQTT) for Root Certificate's and Physics of Digital Twin in Smart Home, degrading the life and update cycle during consumer products is a & presented. (Wang & Chien, 2023) & Digital Twin architectures in smart home, degrading the longevity of consumer digital twin in smart home (Zdankin et al., 2022). The ability to handle firmware updates securely in such environments is the key to the adoption of firmware updates in safety-critical industries.

2.2. Security and Privacy Threats in Connected and Critical Infrastructure

IoT devices and other connected systems adoption has made the commodities attack surface of modern infrastructure an enormous one, with more vulnerabilities in cybersecurity. This section outlines the critical security issues experienced in the delivery of software updates, especially in the domain of critical infrastructure such as smart grids, health devices, and unmanned aerial vehicles (UAVs).

In IoT in health devices, Affia et al (2023) discussed the security risks of connected medical devices, pointing out the possible vulnerabilities of medical device hacking (disclosure of patient data without, or unauthorized access to the system) and malware infections. Similarly, the smart grid infrastructure is also exposed to certain challenges of firmware to update security. Mashima et al. (2024) address cybersecurity strategies for the smart grid and how they need to consider the latest threats, such as cyber-attacks related to communication protocols and firmware on the power grid. These challenges are compounded by the need for real-time information on failures in order to prevent their disastrous consequences.

The integration of UAV systems in critical operations (e.g. surveillance, delivery, and military applications) adds other levels of complexity around managing their release. Cordill et al., 2025 offer a thorough survey of privacy and security issues that are endemic in UAV systems-based system design, favoring secure OTA delivery mechanisms to prevent hijacking and unauthorized access to control systems.

For all these sectors, suitable security practices must ensure the validity and integrity of the software updates because of potential attacks such as man in the middle (MITM) or over the airway programming vulnerabilities (Carter, 2016). These systems need to be designed with strong authentication, encryption, and secure booting protocols in place to prevent unauthorised manipulation of the firmware.

2.3. Artificial Intelligence and Machine learning in Cyber Security and System Monitoring.

Artificial Intelligence (AI) and Machine Learning (ML) are being used more frequently in the field of Cybersecurity to make our systems for detection and mitigation less complex, by using AI and ML technologies. In terms of release management, artificial intelligence can provide more security in OTA updates with continuous monitoring of the system, analyzing telemetry data, and predicting potential failure.

For instance, Jamshidi et al. (2025) explains the use of machine learning techniques for Intrusion Detection Systems (IDS) in IoT environments to detect anomaly or predict cyber threats before it causes damage using AI Models. Similarly, Ibrahim & Kashef (2025) analyze the application of large

 doi:10.20944/preprints202602.1974.v1

language models (LLMs) in the cybersecurity context and in particular their use to detect vulnerabilities and recommend strategies to mitigate those vulnerabilities in smart grid systems.

Additionally, Generative AI (GenAI) could be used to augment automation of root cause analysis (RCA) and in defect triage of software release pipelines. LLMs can be configured to become aware of repeated problems in the system's performance by training on past data and give insights to the possible causes of the problem that may have occurred, due to which the software delivery process may save a huge deal of time. These tools can also help create complex release notes, impact assessments, and other documentation that are normally manual input with vast amounts of time and resources being saved.

While promising, this application of AI and ML in cybersecurity and release management introduce new challenges of model interpretability, bias, and the attribution of hallucinated conclusions which may potentially lead to wrong decision-making, particularly in safety-critical systems (Ibrahim & Kashef, 2025).

Generative AI, Software Engineering, Digital Physical Product Development 2.4

Generative AI is not only disrupting the world of cybersecurity, but it is also bringing a revolution in the software engineering practices as well. The integration of digital and physical product development, like what we have been seeing at the intersection of hardware and software for smart devices, are areas that GenAI can provide a lot of value.

Kim et al. (2020) have demonstrated how code auto-completion with machine learning can boost firmware development to speed up the process of writing and debugging code. This capability can be extended to OTA updates, with GenAI being able to help generate patches for the code, identify system vulnerabilities, and automate the deployment to devices.

In the development of digital-physical products, Hendler (2021) looks at practices of coordination between software engineers, hardware developers, and product managers, and could be improved. As firmware and software become dependent upon hardware and more closely connected, the role of AI in the management of product development cycles is only likely to increase, enabling more adaptive and iterative product releases.

Furthermore, Lehmann & Recker (2022) believe that digital ventures are adopting an increasingly continuous development approach to innovation, which products are 'ever-in-the-making' after the product has been originally released. This constant development is especially relevant for connected systems such as IoT because OTA updates for the system are essential for maintaining and enhancing system functionality over time.

2.4. Safety Critical Safety Regulatory and Quality Evaluation

As the use of GenAI and AI-driven automation continues to grow, it is important to mitigate the ethical and regulatory issues surrounding these technologies, especially in fields where safety can be of the utmost importance such as the health and energy sectors. Regulatory bodies would need to create frameworks to make sure that AI systems used software release management to comply with the standards of safety and do not harm end users.

Sharma et. al. discusses the regulatory issues of the AI-based drug delivery systems, in which the updates of software have to be within strict standards of patient safety. Similarly, Niu & Lam (2025) address the security problems for automated insulin delivery systems as well as the importance of secure, ethical, and reliable software updates for these lifesaving applications.

In the smart grid sector, Yaddanapudi et al. (2023) suggest the cybersecurity technology road map for the smart grid industry that incorporates guidelines for software updates, patch management as well as vulnerability monitoring for the longevity of the security in the smart grid.

As the integration of GenAI in the release management is more adapted, there is an even significant need to have those regulatory frameworks adapt too for the systems in use to operate in a transparent form and be held accountable for the actions they take since some processes like in industries where safety is vital.

 doi:10.20944/preprints202602.1974.v1

2.5. Synthesis and Gaps

While literature offers great respite in the scope of security risks, artificial intelligence applications and software over-the-air (OTA) delivery mechanism in various sectors, there is still a lack of a complete approach to GenAI augmented release management. Most of the existing work either focuses on specific domains (e.g. smart grid, IoT, Medical devices) or the application of machine learning and AI in isolation without considering the broader implications to be applied for release management in connected and critical systems.

The next step in the literature is the creation of a comprehensive framework integrating GenAI tools in releasing management pipelines considering both technical and ethical issues and implications to security, reliability, and governance. Moreover, there is a need for empirical studies to test validate effectiveness tools driven by AI in real-world scenarios during release management.

3. GenAI-Augmented Release Management Conceptual Framework

In this section, we propose a conceptual framework for GenAI augmented release management, which is focused on enhancing the efficiency, security and reliability of software and firmware update in complex and connected systems. This framework brings Generative Artificial Intelligence (GenAI) capabilities (LLMs in particular) into the existing production short-term schema release management pipeline. The framework presents how GenAI may be applied to tasks such as documentation creation, risk assessment, defect triage, root cause analysis (RCA), or support in decision-making with a solid focus on how AI can be integrated into safety-critical environments that are connected to the internet.

3.1. Multi-Domain System Traditional Release Management

Release management, for complex and distributed systems, is achieved through a multi-step process, which usually consists of the following stages:

Planning: Setting the goals of the release, timelines, and defining the scope of the release.

Build: Making/Compiling and building changes made to the source code into the executable code.

Test: Functional and security testing of the build both automated and manual

Sign: Formal approval for release, according to requirements of quality and security that have been predefined.

Deploy: Transfer the firmware/software to the devices, normally using OTA updates.

Monitor: Observing the effect of the deployment, performance and finding out any post release of defects / incidents.

Rollback: Rolling back to previous version of firmware/software in case of critical failure.

While these steps have served as the basis of release management since, they have become more complex, thanks to the emergence of connected systems, meaning we're looking at IoT systems, smart grid, medical devices and even UAVs which all need extra layers of security and compliance, as well as the ability to make decisions in real-time.

For example, traditional approaches might not be applicable to firmware updates in an environment where security risks and downtime of the system are non-negotiable (e.g., critical infrastructure, smart grids, healthcare devices, aviation systems).

3.2. Generative A.I. Capabilities for Program & Release Management

Generative AI (GenAI) technologies, with LLMs being an example, offer the potential for complementing the conventional approach to release management by automating tasks that are overly time-consuming, error-prone or that involve expertise in multi-domain tasks. Some of the core capabilities of GenAI in release management are given below:

Automated Target of Documentation Creation:

 doi:10.20944/preprints202602.1974.v1

GenAI can take the records of code commits, test results and impact analysis, and automatically generate release notes, change logs and technical documentation by summarizing relevant information about the changes. This capability relieves the work required of release managers and engineers who tend to spend a lot of time writing and going through these documents.

Risk and Impact Analysis:GenAI can analyze data on telemetry, logs, and past release results, and predict potential risks and impacts of future releases. By understanding how the system behaved in the past and the likelihood of future disasters, it can recommend ways to mitigate potential risks and assist teams with prioritizing these testing needs and related concerns based on the most likely solutions.

Automated root cause analysis (RCA): One of the most time-consuming parts of release management is determining the cause of post-release issues. GenAI can help by analyzing logs, system data, and historical events to create unassisted RCA reports to link code changes to failures in operations. This can help to speed up the diagnosis of any defects and help to identify areas for improvement for future releases.

Intelligent Defect Triage:

Using the content of bug reports, code reviews, and historical data, GenAI can suggest the most likely hypothesis of defects, and triage them according to the severity and impact. This will help ensure that the most critical issues are dealt with first, which can effectively help release managers to prioritize their efforts.

Compliance and Regulatory Inspections:

In industries such as healthcare, energy, and aerospace, there are strict standards with firmware/software updates since the updates must comply with strict regulations. GenAI can automate compliance checkups, ensuring that updates are in total agreement with safety, security, and privacy needs. It can produce reports on the status of compliance and show where areas require attention.

Automated Decision Aid for the Release Approval:

GenAI can be used to help in the process of deciding releases, where it can gather information from multiple stakeholders, including developers, testers, security experts, and compliance officers. It can equip decision-makers with summarized decision-making on risk assessments, test coverage reports, and compliance status for facilitation of rapid, better, and more informed decision-making.

3.3. Proposed GenAI Augmented Release Management Architecture

In order to successfully integrate GenAI into the release management process, we propose the following architectural for GenAI-Augmented Release Management:

Data Layer:

This layer is responsible for the gathering and aggregation of data from different sources:

Code Repositories - commit messages, pull requests, merge logs.

Telemetry and Logs: It is the data collected from IoTs, sensors, and other embedded systems.

Test Results: Automated and manual testing results and security vulnerabilities identified in the testing process.

Incident Reports: Posts of defect reports after release and finding of RCA.

GenAI Services Layer:

At the heart of GenAI service architecture, AI services for GenAI leverage Large Language Models (LLMs), and domain-specific AI models, to do their job - summarize release notes, do risk analysis, generate compliance reports, and do defect triage. These models are trained on historical data in order to ensure that they are able to make accurate predictions and provide valuable insights.

Text Summarization: LLMs are used to create brief, clear-cut (and task-specific) release notes.

Risk Prediction: ML models are able to predict the possibility of releasing failures and security vulnerabilities based on past data.

Security & Governance Layer:

 doi:10.20944/preprints202602.1974.v1

This layer is designed to provide the integrity and security of the release management process. It has in place mechanisms for auditing the GenAI decisions and validating the results of their decisions based on predefined rules and compliance with policies associated to specific industries (e.g., HIPAA, GDPR, FDA).

Data Integrity Checks: Verifying the authenticity and accuracy of the input data that is used by GenAI models.

Model Governance: Setting up the control of the processes of updating, validating, and retraining GenAI models to adapt, as they encounter new cases and ever-evolving regulations.

Human-in-the-Loop Decision Points (HITL):

In spite of the automation potential of GenAI, humans in safety-critical environments still remain important in decision-making. GenAI can assist human decision-makers in gaining actionable insights for them, but final software releases should be under human control.

Release Board: A scarcely of experts (security engineers, compliance officers, TPMs) that examine the GenAI-created reports and give final choices on the release of approval.

Release Monitoring: Continuous human surveillance for uncovering unexpected situations and giving corrective measures.

Deployment Layer:

This layer is responsible for the actual deployment of firmware/software updates to devices, either via OTA updates, or as direct system integration based on domain. GenAI ensures that deployment processes are optimized for efficiency to ensure security.

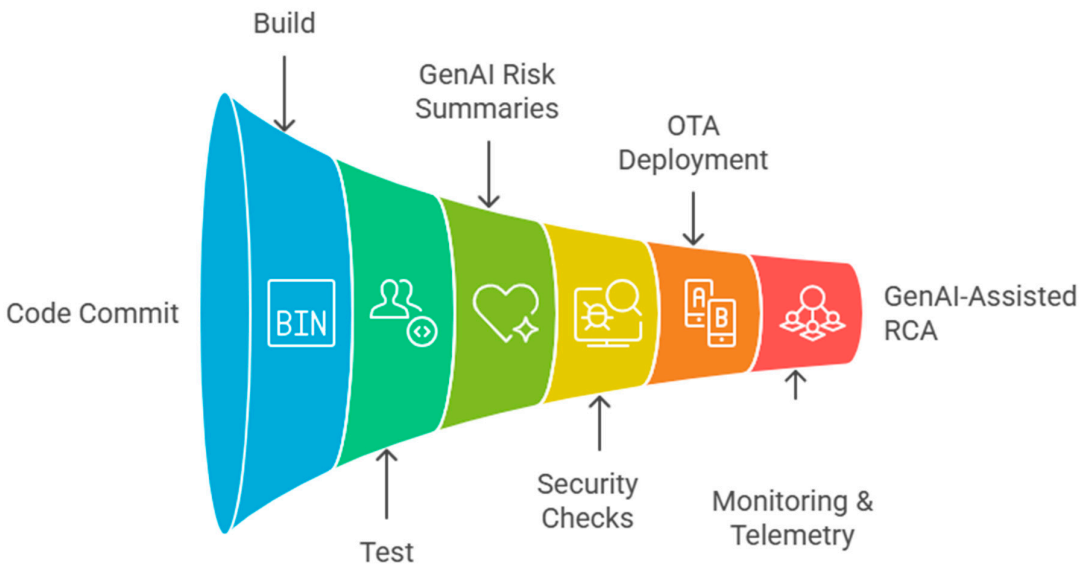
3.4. Positioning within the Existing Cybersecurity and OTA Frameworks

The proposed framework can be integrated with the existing security frameworks and OTA update systems that are used in critical infrastructure sectors. For example, in the case of smart grid systems, IoT health gadgets, UAV systems among other, the integration of GenAI can be used to bolster current cybersecurity policies by automating the real-time telemetry and system logs analysis to identify and respond to potential threats in a faster manner (Maleki; Mashima; Affia). Similarly, GenAI-enable RCA reports can help security engineers better understand the root cause of failures and vulnerabilities in critical firmware/software releases.

An additional benefit is that GenAI's automated support for the decision-making process can be included in compliance checklists, which can decrease the administrative burden of manual compliance carrying out the check of compliance in relation to the regulatory requirements. This can be especially helpful in industries where compliance is highly regulated, like medical devices (Sharma) to automotive systems (Simon & Graham), where meeting high standards of regulation is not negotiable.

Diagram 1: Augmented Release Management Pipeline of GenAI

 doi:10.20944/preprints202602.1974.v1



3.5. Summary

The concept of the proposed GenAI-Augmented Release Management Framework opens a path-breaking method to deal with firmware/software release management of connected & safe-critical systems. By automating tasks such as documentation, risk analysis, and defect triaging, compliance checking, etc., this framework adds considerable value to efficiency, security, and reliability of the release process. Moreover, through the integration of human oversight and AI-based automation, the framework guarantees that the process of release management will be able to keep up with the growing levels of complexity in modern digital-physical ecosystems.

4. Release Management User Cases for GenAI

Generative AI (GenAI) has the ability to revolutionize the management of releases by automating tasks that traditionally need a lot of human involvement, like generating documentation, triaging defects, risk assessment, and decision-making. This section delves into several specific use cases for GenAI in the release management pipeline that illustrate the effect that it has in a variety of areas, including IoT, smart grids, medical devices, UAV, and connected infrastructure.

4.1. Automatic Release Note and Stakeholder Summary Production

One of the things that consumes a lot of time in release management is the process of creating the release notes and other documentation associated with software updates and changes. Traditionally release notes summarized the changes that have been made in a given version of the software, listed fixed defects, and gave instructions for deployment. This task usually needs the input of developers, QA engineers, product managers; it is often a task that is delayed and inconsistent.

GenAI can be used for automating this process by summarizing changes in the code and producing release notes that are tailored to various stakeholders. For example:

Developers may be provided with detailed information regarding the changes made in the code such as the commits, new features, and fixed bugs.

QA engineers can be given more technical information about the areas that can be tested in the light of changes.

Product managers may be given a high-level overview, noting the impact of the release from a strategic perspective, key features, and changes for customers.

 doi:10.20944/preprints202602.1974.v1

With GenAI, this process can be automated, which will ensure less manual work and fewer errors. LLMs (Large Language Models) can be trained off historical data which can look at the patterns and generate coherent summaries from pull requests, commit messages, and bug reports.

Example: For GenAI generated release note it will look like this:

Features Added: "New user authentication system with support for Multi Factor."

Bug Fixes: "Fixed bug relating to payment failure regarding high traffic hours."

Known Issues: "Significant performance decrease for mobile app upon first opening"

GenAI can therefore bring uniformity and correctness to all release documentation and save time spent on manual writing and boost efficiency.

4.2. Intelligent Chatbots to Triaging Bugs for Performing Defect Handling & Developer Assistance

In big release pipelines, the defects are often reported by the users or identified during testing, and they need to be triaged by the development teams to figure out the cause and severity of the defect. Traditionally, this type of task is accompanied by a lot of human power to assign categories and priorities to defects and delegate to the relevant teams. This process can be made even more cumbersome to handle multiple releases at once.

GenAI powered intelligent chatbots can be used to automate this triaging process by analyzing bug reports as they come in and categorizing them based on severity, components affected, and historical data. The chatbot can:

Analyze defect reports: With the help of NLP, GenAI can parse through issue tickets, categorize the defects according to their type (e.g. security, functionality, performance), and automatically assign priority levels.

Suggest resolutions - Based on last tickets, code changes and testing, the chatbot can suggest possible fixes, or who will be the most likely team to fix the issue (i.e. network-related defects can be assigned to network engineering team).

Assist developers: The chatbot can provide developers with quick suggestions or fixes on the basis of historical codebase patterns, like the way through ML-based auto-completion tools to assist in writing code (Kim et al, 2020).

This automated triage process allows developers to work on more complex problems without reducing their time by assigning tickets. Additionally, developers can communicate with such chatbots to get relevant data fast without leaving their Integrated Development Environments (IDEs), leading to improved efficiency.

4.3. Automated Root Cause Analysis (RCA) Report

Root Cause Analysis (RCA) is a very important step in post-release monitoring to determine the underlying cause of defects or system failures. Traditional RCA is a time-consuming one that involves analyzing logs, code changes, system telemetry, and bug reports. GenAI can be used to streamline this process by automatically producing RCA reports that help engineers quickly identify the source of the issue.

GenAI can:

Analyze Historical Logs: By processing logs, error reports, and telemetry data, GenAI can analyze patterns that could be used to identify the root cause of a failure.

Correlate code changes with failures: GenAI can also be used to correlate the data of a release, such as code changes in each version, with the incidents that occurred after the deployment. It can then produce reports on what code modifications are most likely to cause failures.

Generate action steps: Once root cause has been determined, GenAI can make recommendations for corrective actions, such as code changes, configuration, testing, etc. Conducting risk estimation and identifying vulnerabilities: After the root cause of a vulnerability has been determined, the next step is to estimate the probability that it will cause serious incidents, potentially leading to serious consequences.

 doi:10.20944/preprints202602.1974.v1

For example, if an update to the smart grid leads to an interruption in power to a building, GenAI can help quickly analyze logs of the deployment, match it with power flow-dependent data and suggest a specific fix, such as rolling-back a specific firmware module. The automation of this process reduces the downtime and quickens the recovery.

4.4. Security Knowing Release Decision Help

One of the important challenges of release management involves making sure that the updates do not cause a new vulnerability or security risk. Traditionally, security experts check the release of candidates for possible security issues in a manual manner, and this can delay the release process, especially in environments such as smart grids, healthcare devices, and automotive firmware. This is where GenAI can play an important role.

GenAI can be used to help in security-aware decision support by:

Analyzing release candidates for vulnerabilities: Video AI generators like Synthesia can be used for analyzing release candidates for vulnerabilities. GenAI may analyze the codebase, release notes in addition to the test reports to determine any security risks, for example, but not limited to, known vulnerabilities in third-party libraries, exposed endpoints, or improper encryption.

Recommending mitigation actions: Based on known security best practices, GenAI can make recommendations for changing security practices to address security vulnerabilities before the release is deployed.

Simulating possible attack scenarios: GenAI can be used to simulate potential security threats (such as simulated DDoS attacks or unauthorized access attempts) and to evaluate the new release capacity to withstand these threats, using historical data.

For instance, in medical devices, security of firmware is very important that the updates do not harm the patient. GenAI can automatically verify whether the latest update causes any security holes, which could be exploited by the attacker to gain access to sensitive patient data.

4.5. Domain-Specific Scenarios

The use of GenAI in release management can therefore vary widely depending on the domain you are in, with different regulatory standards; technical requirements and risk profiles can exist. Following are a few domains specific scenarios focusing on how GenAI can be used for better release management:

Smart Grid & Energy Systems:

GenAI can help to predict the anticipated effect of firmware updates on grid stability. It can analyze telemetry data to recommend the time for deployment, especially when the components that are critical to the system operation, such as load transformers and voltage regulators, need to be updated, in order to ensure that the updates do not create risks such as power surges or outages (Mashima et al., 2024; Maleki et al., 2025).

Medical Devices & Health IoT:

Tools such as genAI can help to automatically enforce compliance with FDA or HIPAA regulations when support is sent updates for insulin pumps or wearable health monitors. It can be used to analyze past data and identify potential risks and suggest safer methods of deployment (Sharma et al., 2024; Niu & Lam, 2025).

Automotive & Mobility:

In case of the vehicle firmware updates, GenAI can use vehicle performance data as the parameter to recommend the optimal time for firmware fixes to be deployed, ahead of time to prevent any form of failure which could have a detrimental impact on user safety (Simon & Graham, 2017). It can also be used to automate ensuring compliance with vehicle safety standards (e.g. ISO 26262).

Space & UAV:

Offsettable - (OTA) Updates are required by CubeSat systems to ensure operation while in orbit. GenAI can be used to ensure that updates do not interfere with mission-critical systems by simulating potential risks (Molina et al., 2023). In the case of UAVs, it can be used to optimize firmware updates

 doi:10.20944/preprints202602.1974.v1

based on the real-time data of flight to improve the reliability of an autonomous drone commonly used in surveillance or delivery operations (Cordill et al., 2025).

Smart Homes:

For IoT devices in smart homes, GenAI can be used to ensure that firmware updates of smart thermostats, lighting systems, or security cameras do not disrupt system interdependencies to reduce the risk of system failures or poor user experiences (Zdankin et al., 2022).

4.6. Summary of Use Cases Benefits and Limitations

Whilst the potential benefits of GenAI in release management are clear to see, it is important to consider some of the limitations and risks:

Benefits:

Increased efficiency/speed release cycles.

Increased accuracy when making assessments of risks, triaging and documenting:

Increased security with automated vulnerability detection and mitigation; and

Reduction of manual error in the release process, especially on documentation and compliance check

Limitations:

Model bias or hallucinations that may produce erroneous conclusions or defective releases;

Over-reliance on GenAI in a safety-critical environment will perhaps bypass the need for necessary human oversight.

Regulatory and compliance challenges with AI-based decisions in certain sectors not trusted entirely, regulatory and compliance challenges may arise.

Table 1. GenAI Use Cases in Release Management Across Domains.

Domain	GenAI Use Case	Primary Benefits	Key Risks	Representative References
Smart Grid	Risk analysis, firmware impact simulation	Reduced downtime, enhanced grid reliability	Simulation accuracy, data integrity risks	Mashima et al. (2024), Maleki et al. (2025)
Medical IoT	Automated compliance checks, security-aware release decisions	Faster compliance, reduced security risks	Privacy concerns, model bias	Sharma et al. (2024), Niu & Lam (2025)
Automotive	Impact prediction, firmware security checks	Enhanced vehicle safety, timely updates	Critical failure impact, regulatory compliance	Simon & Graham (2017)
UAV & Space	Simulation of risk scenarios for updates	Optimized release schedules, mission integrity	Space-specific constraints, long-term impact	Molina et al. (2023), Cordill et al. (2025)
Smart Homes	Update interdependency management	Reduced failures, optimized	Device fragmentation, vendor lock-in	Zdankin et al. (2022)

 doi:10.20944/preprints202602.1974.v1

	firmware distribution

5. Ethical, Security & Reliability Considerations of GenAI in Release Pipelines 3

The introduction of Generative AI into firmware and software release pipelines is not just adding efficiency to the process, it introduces new and unique failure modes, attack surfaces and ethical dilemmas, particularly in safety critical and highly regulated environments. This section explores the implications of GenAI for how the risk landscape is shifted for the management of its releases in terms of safety and reliability, adversary threat, privacy, accountability, and regulatory alignment.

5.1. Safety and Reliability in Critical Systems

In industries that deal with medical devices, energy systems and other critical infrastructure, firmware and software releases are closely linked with safety outcomes. A mistake in decision making around release, a mistake in risk assessment or an incorrect root cause analysis that is generated with GenAI can cause serious harm in the real world.

n the case of automated insulin delivery systems, for example, Niu and Lam (2025) highlight the security failure or faulty logic in control algorithms may cause dangerous dosing of insulin. When GenAI tools are applied to summarizing incident logs, suggesting mitigations or supporting decisions on releases of such systems, erroneous inferences or hallucinations of causal connections can have a direct impact on patient safety if not checked right. Similarly, Sharma et al. (2024) emphasizes the fact that AI-enabled drug delivery and medical devices ecosystem must have strict assurance that software changes do not adversely impact the therapeutic safety or regulatory constraints.

Reliability risks are not only in health. In the case of smart grid environments, firmware updates are in interaction with complex physical processes. Poorly Tested or Misconfigured Updates Mashima et al. (2024) and Maleki et al. (2025) Demono et al. (2018) Maleki et al. (2021) If GenAI is to be used to suggest rollout strategies, evaluate risk logs, or filter alarms, failure in its ability to model edge conditions correctly can provide a contributing factor to grid-wide cascading failures.

From an efficacy of reliability perspective, there is a fundamental issue of GenAI not "understanding" physical risk even if it is able to generate viable text. Without strong guard rails there's a non-undulating chance of GenAI having off-land explanations, RCA stories or "safe to release" summaries written with all that looks convincing but lacks such critical edge cases that could hurt results. In safety critical pipelines this gives rise to a duty to:

GenAI outputs should be regarded as hypotheses and not as facts.

Embed redundant checks (e.g. deterministic safety constraints, hardware-in-the-loop tests) prior to deployment.

Safety cases and validation of artefacts should be ensured to be built without using GenAI-generated narratives in regulated environments.

5.2. New Attack Frontiers Brought in by GenAI

GenAI does not use in vacuum-takes intakes of logs, telemetry, configuration data, historical incidents for where environments have already been under attack. Introducing GenAI to these ecosystems increases the attack surface of the system in at least three ways:

Prompt injection and Context injection

If GenAI models are integrated with ticketing systems or log aggregation or external APIs, malicious inputs could be created to control the responses produced by the model. For example, a device that the attacker has compromised within a smart grid or a UAV fleet (Cordill et al.,2025; Mashima et al.,2024) could plant a log message downplaying or mis-categorizing suspicious activity

 doi:10.20944/preprints202602.1974.v1

which encourage GenAI-assistant classifier to classify serious incidents as benign or mis-f Bant root cause analysis.

Model and Data Poisoning

As it is based on historical data, IDS systems and GenAI assistants learn from historical data (Jamshidi et al., 2025; Ibrahim & Kashef, 2025), so attackers can try to poison datasets with data used for training or fine-tuning purposes, normalizing their attack patterns. In the case of critical infrastructure, where conditions can arise and develop over months or years, the long-term drift in model behaviour can gradually give backseat to detection performance.

Interaction with the Hardware Within and Side Channel Threats

Work done on electromagnetic side-channel attack (EM-SCA) to encryption (Zunaidi et al., 2024) indicate that the attack is possible for the bad actors to use physical leakage and recover secrets. While GenAI is not directly the target for such attacks, the recommendations that it makes regarding cryptographic configuration or firmware rollout may be based on incomplete or compromised evidence if the underlying systems are already under subtle attack.

In combination with advanced persistent threats in industrial control systems (Keliris & Maniatakos, 2017) and the move of critical infrastructure "into space" (Ellis et al., 2023), GenAI-assisted release decisions become another high value target. An attacker who can influence AI-based reports on RCAs and risk summary can have an indirect effect on what firmware gets shipped where and when.

As a result, any use of GenAI in release pipelines must create a system of AI stack itself secured subsystems, with:

Isolation of training and inference environment are good.

Input data sources - a set of integrity checks.

Threat models in a specific and outstanding way for prompt injection, data poisoning and model tampering

5.3. Privacy and Data Protection

Release management has traditionally been based on technical artefacts (source code, binaries, test results) - in a connected systems it is frequently dealing with a different type of data (personal vs. operational):

IoT health devices can be used to record patient behaviors and physiological signals (Affia et al., 2023).

Consumer types of sleep technologies are being used to gather sleep pattern and bedroom environment information in detail (de Zambotti et al. 2025).

Connected vehicles down an aggregate of driver behavior, location history and system usage (Simon & Graham, 2017).

When these logs and telemetry streams are then fed into GenAI for triage, RCA or risk assessment the privacy risk is multiplied. A large language model could therefore accidentally memorise or re-expose sensitive data in generated data, internal cache, fine-tuning checkpoints, etc De Zambotti et al. 2019 then calls for more rigor, context, and collaboration in evaluating consumer sleep technologies; an analogous level of rigor is needed when GenAI is consuming sensitive telemetry from such devices.

There are two particular privacy issues:

Scope Creep of Data Usage

Data that was originally collected for operational monitoring or safety purposes (e.g. device error logs) might be used for GenAI in a new manner-for instance, without explicit consent of the user and without easily accessible opt-outs. This is especially problematic within the medical and automotive fields, where the data minimization and purpose limitation requirements are stringent from a regulatory standpoint (Sharma et al., 2024; Simon & Graham, 2017).

Cross-Context Leakage

 doi:10.20944/preprints202602.1974.v1

A GenAI assistant for multiple product lines or customers may inadvertently reveal some confidential information about one environment while responding to a query about another, if there are no enforced isolation boundaries.

In order to alleviate these concerns GenAI-enabled release pipelines need to follow:

Data minimization strategies Only the required abstractions (e.g. error codes, anonymized metrics) of data are made available for models

Anonymization and Pseudonymization of logs before they are ingested.

Per-tenant and per-domain isolation of context (especially when the IoT is multi-tenant behavior) or multi-tenant edge (Reyes-Acosta et al, 2025)

5.4. Accountability, Traceability and Government

As GenAI begins to contribute to what gets released or not, how quickly it is rolled out and how it is incident-interpreted; questions of accountability are at the forefront. If a GenAI-generated RCA is erroneous and causes an unsafe patch to be released, who is responsible, the engineer supervising the release of the patch, the developers of the models, or the vendor running the GenAI service?

Existing work on secure software updates for resource-constrained IoTs (Solomon et al., 2019), OTA for satellite and radio (Molina et al., 2023; Carter, 2016), and software vulnerability management [18,20] point to the importance of being able to attribute updates to particular sources and to specific individuals who provide a rationale for why a change is necessary. GenAI makes this picture more complex since many of its internal steps of reasoning are opaque.

A GenAI augmented pipeline needs the implementation of governance mechanisms that:

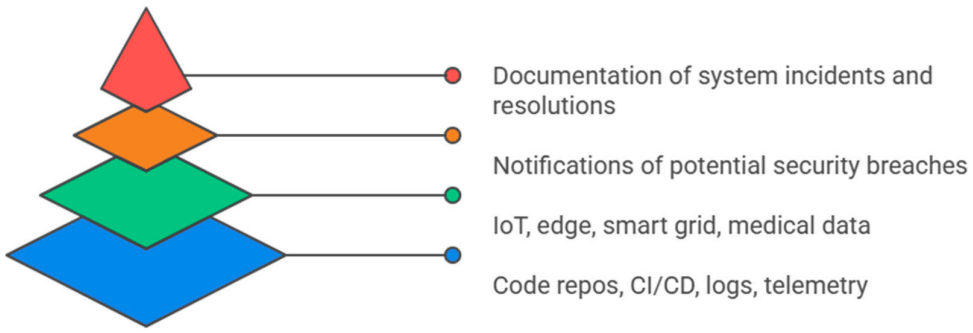
Document all decisions made using AI assistance (e.g. "release is low risk", "rollback not needed") made along with the evidence used to make that decision.

Mexico City denies people the right to sell US pipe manufacturers to them for fear that 'they do not sell to Mexico'

Support post-hoc investigation of possible contribution of a specific AI suggestion to a failure

Here, the development of update frameworks for data management (e.g. Solomon et al., 2019) based on blockchain suggests one direction, i.e. cryptographically provable records of what has been released, why and based on what assessments. When combined with conventional configuration management databases, such mechanisms can provide tamper evident audit trail across the human and AI actors.

Diagram 2: Risk and Assurance Framework for GenAI Augmented Release Management



5.5. Ethical Guidelines and Regulatory Culpability

Finally, ethical implications The use of genai in release pipelines must depend alongside the formation of these newly emerging ethical and regulatory frameworks in various sectors.

In the area of healthcare and medical devices, the research conducted by Sharma et al. (2024)

 doi:10.20944/preprints202602.1974.v1

demonstrates that increasing expectations in reporting, auditing, and robusting against all failure modes to assure patient outcomes are embedded into the system for the responsible regulators. When firmwares decision is carried out with GenAI for infusion pumps, neuromuscular stimulators (Furey et al., 2007) or for wearable devices (de Zambotti et al., 2025) become part of a broader clinical safety case.

Regarding energy and smart grids, cybersecurity roadmaps (Yaddanapudi et al., 2023) focus on the importance of resilience and incident preparedness in these energy systems and secure a way for systems to update securely; GenAI should not compromise these aspects.

In terms of automotive and mobility, privacy and safety concerns about the use of software and firmware in vehicles (Simon & Graham, 2017) require tight controls on both how in-vehicle data is used by AI and how the AI recommendations are converted to changes of driving behaviour or safety systems.

Ethically, there are a number of principles that should be considered when GenAI is being adopted for release management:

Non-Maleficence and First Do No Harm

AI should never be used as an excuse not to test and and check for safety in critical systems, and even make a final review. GenAI is a layer of augmentation and not a replacement for (nor change) good engineering.

Transparency/ Explainability

While full interpretability of LLMs is unrealistic, organizations should be sure to justify and support their release decisions traceably (citing the evidence of measurable metrics passed, risks essential to strategically mitigate) also not just AI narratives.

Decision making by Humans - Contestability of Decisions

Engineers need to be enabled (and expected) to push-back GenAI outputs. Processes should mean that of the methods designed to represent or encapsulate AI recommendations, there are explicit procedures for when they can be overridden and what evidence is required.

Characteristics, Cleaning up, Relational Information, Through Manufacturing

The level of the AI autonomy of the release pipelines should be proportional to the risk in the domain. For low-risk consumer devices you might have a semi-automatic release based on Artificial Intelligence (AI) triage, and - for smart grids or medical devices - GenAI must be strictly decision-support, never a final decision.

While to sum up, making GenAI work for release pipelines is not just a technical optimization but something of a governance/ethics issue. Without careful consideration and design of safeguards, organisations risk building highly efficient pipelines for shipping unsound or unethical decisions. With proper controls, however, GenAI can make the security and reliability fundamentals built by OTA updates, vulnerability management frameworks, and critical infrastructure cybersecurity research even stronger rather than weaker.

6. Implications for Roles, Teams and Organizational Structures

Introduction of GenAI to release management is not merely an upgrade of tooling, it transforms roles and responsibility of whom is doing what, how decisions are made and how digital-physical products are coordinated throughout time. This section unravels the role, team and structural changes when release pipelines become augmented by AI versus person.

6.1. Evolution of Technical Program Managers or TPMs and Release Managers

Traditionally, information has flowed from the TPMs and release managers serving as the information integrators throughout the project: gathering statusinformation, managing timelines, negotiating scope and synthesizing risks into executive-ready stories. In a GenAI-augmented environment much of this mechanical aggregation is automated.

GenAI systems can:

Summarize jira tickets, incident reports, testcases test results in good dashboards

 doi:10.20944/preprints202602.1974.v1

One idea is to create draft release notes and risk summaries for various stakeholders.

Highlight dependencies over firmware/ cloud service/ hardware components

As a result of these changes, the role of TPMs and release manager moves in at least three ways:

Status Reporting to Artificial Intelligence (AI) Orchestration

Fast becoming owners to the AI assisted pipeline are TPMs instead of manually aggregating the updates:

Determining the Direction of Data Information into GenAI (Logs, telemetry, tickets, user feedback, etc.).

Calibrating the level of autonomy that the level of autonomy is permitted (e.g. drafting vs. recommending vs. Gating).

Ensuring that the outputs from AI are interpreted in the context and especially where there might be a high level of safety or regulatory exposure.

Developing AI Consumers into Evaluating AI Critical Thinkers & Evaluators

For TPMs to be more effective, they need to have the ability to question GenAI recommendations, not only forward them. When a model results that "the release risk is low" based on historical patterns, it is up to the TPMs to have the literacy to ask:

Various quotes about money: - Which incidents did you think? - "money" =

What biases are you laying in terms of deploy environment?

Are we in some previously unseen configuration, type of device or regulatory regime?

This is especially important in areas such as healthcare and smart grids, which may not be able to capture new vectors for potential threats or infrequent, but catastrophic, failure modes in the past.

From Coordination-Only to Risk Negotiation-Governance

As more "opinions" are injected into the pipeline by GenAI (e.g., "recommending rollbacks, deferrals or partial rollouts"), TPMs have a more explicit role as risk negotiators: Translating AI-augmented risk assessments into trade-offs between availability, security, regulatory exposure and customer impact.

In short, the TPMs and release managers emerge as AI literacy decision facilitators bridging GenAI outputs and the domain constraints & leadership expectations.

6.2. Devops,Sre,AndSecurity EngineeringInAI Augmented Pipelines

DevOps, Site Reliability Engineering, or SRE teams are already in charge of the authorization and dependability themes crosswise pipelines for CI/CD. GenAI opens a new dimension: a probabilistic, non-deterministic dimension, which has an impact on release decisions, triage, and RCA.

SREs: Recommendations: Guardrails for the Reliability of AI

In systems such as smart grids, medical devices, and UAVs, where firmware updates have connections with the physical system, SREs will do more:

Define guardrail policies [consist of not approving a firmware rollout to do all critical substations simultaneously; not detesting hardware-in-the-loop tests, no matter how confident the AI is].

Run chaos and resilience experiments not only on the infrastructure, but on the AI-assisted workflows themselves (e.g. simulate corrupted telemetry, either incomplete logs or some adversarial inputs and see how precondition GenAI-assisted workflow behaves).

Own fallbacks: what happens when GenAI fallbacks to output not being available or being contradictory and quite clearly wrong when a critical incident occurs.

DevOps Engineers: Foreign Direct Investment of AI into Toolchains

The plugging of GenAI into the existing tooling is the responsibility of DevOps teams: CI/CD platforms, incident management systems, and monitoring stacks. Their duties change so they continue moving up such which responsibilities as:

Managing prompt templates, context windows and pipelines of retrieval that determine what GenAI "sees" about a release.

 doi:10.20944/preprints202602.1974.v1

Consistent and repeatable processes - if the same inputs are fed into the system, the AI augmented workflow should result in consistent and auditable outcomes, even if the LLM is probabilistic.

E.g., implementing safety switches-releasing a decision that is AI-generated under the assumption that a human (e.g., systems engineer) will always be able to disable a decision that would otherwise be made by the AI (e.g., gating decisions) but allow that a recommendation is captured without this requirement.

Security Engineers Curators of Security Information & Artificial Intelligence Reason

In environments where, with the help of connections to AI drivers such as AI-based intrusion detection, threat intelligence platform and fauna management, it becomes possible to reduce vulnerabilities even if there is already an international level of hardness (Jamshidi et al., 2025; Ibrahim & Kashef, 2025; Sotiropoulos et al., 2023) resortilation of security engineers is to:

Curate high-quality, representative security datasets to fine-tune or condition GenAI, to ensure that emerging threats reflect in the "mental map" of the model (e.g. load-altering attacks, smart grids, EM-SCA attacks on encrypted embedded systems, exploiting UAVs

Validate AI security reasoning Once GenAI makes a statement like "does not materially increase attack surface" security teams need to be able to validate that against concrete controls and threat models.

To avoid any unidirectional machining of GenAI for projecting severe observations less serious treat their confidentiality or redact or to block any proactive action from established incident response workflows, you must define separation of duties so that GenAI has no silent release to downgrade the severity or mute the alerts.

Collectively, DevOps, SRE and security teams operate from running automation scripts toward running a socio-technical system where the combination of human judgment and GenAI recommendations happens simultaneously and in snowball effect.

6.3. Coordination Practices during Digital - Physical Product Development

Digital-physical product development - in which firmware, embedded systems and cloud services co-evolve - already requires complex coordinated development. Hendler (2021) demonstrates that coordination practices are not limited only to tools and processes, but also how actors negotiate responsibilities and coordinate cycles between the hardware and software.

GenAI adds to these dynamics in a number of ways:

Shared Views of Work Mediated by AI

GenAI have the potential to create cross-disciplinary summaries - e.g. a single view explaining to hardware teams to which firmware modules they should all be looking for changes, to cloud teams which APIs are being changed, and to product teams what the behavior change will be at a user-level. This might reduce friction, but it has the proviso.

Misleading summaries from AI can skew an entire team at the same time.

Teams may have too much faith in what AI-generated "single sources of truth" come up with instead of checking with cross-checked with domain-specific artefacts.

Continuous "Ever-In-The-Making" Offerings with increased Velocity

Lehmann and Recker discuss digital offerings that are "ever-in-the-making" and that, once launched, develops changing thereafter. GenAI accelerates this by:

Reducing friction in generating experiments, variants, and localized releases.

Making it easier to triage and RCA, promoting more aggressive rollouts patterns (e.g. cannary releases across IoT fleets, feature flags in medical smart homes, smart homes).

But in firmware oriented areas: smart grids, medical devices, satellites, you have real physical limitations, real regulatory limitations. Organizations must bargain about where they really want "ever-in-the-making" behavior and where they want slow and conservative change with lots of manual review.

The Challenge of Deciding Who Should Own What Decisions

 doi:10.20944/preprints202602.1974.v1

As GenAI suggests changes that must be made working across firmware, backend and device behavior, you have more cross-team decision points:

Who signs off on when an AI-suggested change results in improvements of reliability for one subsystem but increases system complexity for additional system regulations for another?

How do teams deal with conflicts between opting for a recommended optimization by AI (e.g. faster OTA cadence) and operational realities (e.g. field technicians need to have stable baselines)?

In practice, explicit coordination forums (e.g. AI-augmented release councils) where engineers, TPMs, product leads, and compliance officers are directed to review GenAI outputs together, rather than in separate silos, will have to be established by organizations.

6.4. Skills and Capability Requirements

GenAI-Augmented release management elevates the bar of what people and teams should know. Four capability clusters become of key importance:

Data and AI Literacy

Everyone interacting with GenAI outputs -TPMs, SREs, engineers, security staff have to have a working understanding of:

How training data, prompt design, and retrieval of choices influence the behavior of the model.

Common failure modes: Hallucination, Bias, Over Confidence and Brittleness Out of Distribution.

Prompt & Context Engineering for operation.

Release pipelines will require specialists (or at least power users) who can:

Design reliable prompts and working processes that provide stable and useful outputs of noisy operations.

Compose retrieval contexts with the right logs, code diff, and domain policies for each and every use.

Stress-testing against the adversaries or corrupted inputs, and in particular, against hostile environments in cyber security, e.g., smart grids, UAV fleets and edge computing (Maleki et al., 2025; Reyes-Acosta et al., 2025).

Compliance and Safety Compliance Information Related to Sector

Because identical GenAI mechanisms will be applied across domains, people setting up and authorizing such systems need to have a sense of their regulatory and safety context:

In healthcare, understanding of the regulations for devices, clinical risk management, and data protection.

In the area of energy and smart grids, insight into rules such as reliability standards, rules for the protection of critical infrastructure, and the obligation of reporting incidents.

In the automotive and mobility sectors, knowledge of functional safety requirements, over-the-air update requirements and privacy requirements

GenAI cannot replace this domain of expertise and serves as a multiplier for teams that already are familiar with their constraints, at best.

Organizational Change and Organizational Governance Competence

At last, organizations require leaders and architects that are able to:

Governance structures designed around GenAI (who gives the thumbs up to models, who monitors drift, who is allowed to change the prompts).

Working Example to define your interpretations Make clear lines of accountability when decisions are made with the help of AI go wrong.

Manage cultural changes to not blindly believe in or ignore AI in their teams, but rather the state in between - informed skepticism supported by process.

 doi:10.20944/preprints202602.1974.v1

7. Roadmap for GenAI in Release Management Adoption at the Enterprise

Moving from "cool demo" to production ready, safety aware GenAI, in release pipelines, requires a staged and disciplined roadmap. This section sets out a model of maturity, important technical elements that provide these capabilities, governance and policyAlegy, important technical enablers, as well as KPIs that can be measured and used to ensure that large enterprises can adopt GenAI without surrendering their release process to a free-for-all.

7.1. GenAI Augmented Release Management Maturity Model

Stage 0 -Manual, Ad-Hoc Release Management

At this level, organizations use email, spreadsheets, and manual change boards to coordinate releases. Documentation is written by hand; incident reports and root cause analyses are not formal. OTA updates (if present) are often infrequent and scheduled manually, a risky situation in fields such as smart grid, IoT=health devices, or UAVs, where the attack surface is increasing (Affia et al., 2023; Cordill et al., 2025; Mashima et al., 2024).

Characteristics: Scattered tooling cannot have a total view about the release of risk, not in terms of traceability.

Risk: Human error is prevalent, = Slow response to vulnerabilities / Transparency (audit trail) lacks

Stage 1 - Analytics Driven Dashboards and Partial Automation

Organizations introduce elementary CI/CD, log aggregation and monitoring. They deploy analytics dashboards that provide summaries of deployment frequency, rates of failures and trends of errors. OTA update workflows are partly automated regarding IoT devices, satellites or smart homes (Bas & Dowhuszko, 2021; Molina et al., 2023; Zdankin et al., 2022).

Capabilities:

Automated builds and tests; scripted deploys to some environments.

Dashboards for incidents and system health; basic vulnerability scanning (bt decrease and enhance stacking vulnerabilities, drying up and enhancing more complicated system vulnerabilities) 3.Dashboards for incidents and system health basic vulnerability scanning (Sotiropoulos et al., 2023)

GenAI role: None or experimental Maybe LLM is an offline technology that is used to write a small number of documents but is not connected into the pipeline.

Stage 2 - Taken-In GenAI Assistants for Documentation and Triage

Here, GenAI has an active role to play in low-risk, non-decisional parts of the release process:

Documentation: LLMs create a draft note of release, summaries of changes, and stakeholder specific briefings based on the commit logs and test results similar in dimension to ML assisted firmware development (Kim et al., 2020).

Defect Triage Classify incidents -- provide likely components/teams - Identify Similar Issues past, GenAI still human engineers own prioritization

This is a good stage to use consumer IoT, non-critical SaaS applications, or internal applications, but it could also be carefully piloted for more critical areas with higher levels of oversight.

Benefits: Lessening manual overhead, information flow speed up.

Risks: Summarizing hallucinates or is incomplete which may mislead teams when not critically reviewed.

Stage 3 - AI- In the Loop Decision Support in Non-Safety Critical Systems

At Stage 3, GenAI enters into the field of live decision-making, but it is mostly applied for the system where the failure cost is high, but not life-threatening or grid-threatening (e.g., some enterprise platforms; non-critical industrial IoT; smart homes, etc.).

Use Cases:

risk scoring using artificial intelligence to provide comparisons of releases based on telemetry, past releases and similarities of environment.

 doi:10.20944/preprints202602.1974.v1

Suggesting rollout strategies (canary, phased, regional) Roll back triggers.

Automated incidents recurring RCA drafts

Human release boards have to make the final call, but increasingly they are using GenAI to pre-digest massive information volumes. In the form of environments akin to smart homes or non-critical IoT this can have a materially faster release cycle (Zdankin et al., 2022) without unacceptable risk.

Stage 4 - Carefully Managed AI Preferred as 'AI-in-the-Loop' in Safety-Critical Areas

The highest level of maturity is dedicated to smart grids, medical devices, UAVs autonomy, critical infrastructure in the orbit and automotive firmware (Sharma et al., 2024; Niu & Lam, 2025; Cordill et al., 2025; Ellis et al., 2023; Simon & Graham, 2017). Here, GenAI influences - but does not unilaterally take into consideration - decisions for safety-critical releases:

Capabilities:

AI-derived summaries of risks are explicitly linked to domain-specific cases of use, such as safety (e.g. for insulin delivery systems) or firmware safety (e.g. for the smart grid).

GenAI used for cross-checking consistency between logs and test results and compliance checklists for example, but cannot go around hard safety gates

AI-assisted what-if analyses on telemetry and grid/device models to support expert judgment, Mashima, H., Shahzad, M., Maleki, R., et al. "Remote monitoring and control of inverters": in preparation

Controls:

Strict governance of models, stringent isolation of data and formal validation procedures

External audits and scrutiny by regulators about those AI's role in the release process (.Sharma et al. 2024 de Zambotti et al 2025)

The critical insight, then, is that stage 4 is not "more automation" of the same thing. It's a different regime with explicitly regulatory [sic], ethical and safety conditions built into the structure of the design.

7.2. Technical Enablers

Reaching Stages 2-4 requires that you have a good foundation in the technical areas, so attempting to "jump ahead" without these puts you in trouble.

Orcadia subsidy scheme includes "under 1.5 degC of global warming in physical risk, BEV investment, etc." Oxygenation Years: Union Wind Power. (a) Data Infrastructure: Telemetry, Log Aggregation, Observability

GenAI is only as good as the data that it sees. For firmware/software release management in IoT health devices, smart grids, UAVs, satellites and smart houses, this means:

Unified Telemetry Pipelines from devices / edge nodes(c) Affia, T., Venkat Narayan, S. Nair, S. H. Visalven, and A. Tambu, eds. 2023 Unified telemetry pipelines from devices and edge nodes.

Log aggregation in CI/CD, systems of deployment and run time environments.

High-quality observability (metrics, traces, logs), which identifies normal variability from the actual anomalies.

If telemetry information from a UAV fleet or CubeSat payloads is incomplete, biased or delayed (Molina et al., 2023; Cordill et al., 2025), risk assessments using GenAI will be similarly losing.

(b) Implement MLOps / AIOps Pipelines

In order to apply GenAI tools safely in the release management sciences, enterprises require secure MLOps/AIOps capabilities:

Versioned Models, Prompts and Retrieval Configurations.

Controlled releases of AI services into production using roll back and canaries, like any other key service.

Testing of models against possible adversarial examples and poisoned data, in particular, where IDS and security analytics data might be used as inputs for GenAI workflows (Jamshidi et al., 2025; Ibrahim & Kashef, 2025).

 doi:10.20944/preprints202602.1974.v1

These pipelines themselves are sure to need to follow secure-by-design principles, or else we end up with the artificial intelligence layer as a new weakest link in environments already struggling to handle APTs and complicated cyber-physical threats (Keliris & Maniatakos, 2017; Ellis et al., 2023).

(c) Hardware Accelerators and Edge-ai capabilities

Some release decisions and validations have to take place at or close to the edge - for example in smart meters, UAVs or satellite payloads. Here, hardware accelerators come in the picture:

On-device NPUs as in AMD XDNATM NPU that is available in RyzenTM AI processors (Rico et al., 2024) can be used to realize real-time inference for anomaly detection or local policy enforcement without round trip latency to the cloud.

Edge AI can be used to validate OTA updates against local conditions which could be accepted fully, rather than only in environments with limited bandwidth or those with mission-critical requirements (Bas & Dowhuszko, 2021; Molina et al., 2023).

However, pushing GenAI to the edge has to be balanced in terms of power, security and overcomplicating updates; not every device needs or has the capability of it.

7.3. Coordination of Governance, Policy and Compliance

Technical capability without governance is the way you end up with AI-driven incidents in which the auditors are finding out after the fact. A realistic roadmap must conform to GenAI use with sectoral ethical expectations and cybersecurity.

A realistic roadmap could and must conform to GenAI use with sectoral regulations, expectant, with ethics and cybersecurity frameworks.

Matching the Use Cases to the Regulatory Regimes

Different domains have different preferences criticality of AI autonomy:

Medical Devices & Digital Therapeutics.

The use of firmware shift development for automated insulin delivery or neuromuscular stimulator is enclosed to stringent regulations of a medical device (Sharma et al., 2024; Niu & Lam, 2025; Furey et al., 2007).

GenAI can write documentation, suggest ways to mitigate a risk, and pre-screen logs, but final approval for release has to be within the safety case regulating the process.

Consumer Health & Sleeping Technologies

Evaluation of consumer wearables and sleep tech already is an issue of variable rigor (de Zambotti et al., 2025).

The introduction of GenAI requires particular care in reviewing the impact of false reassurance: AI analysis of stability that appears rigorous and simply premature based on poor data.

Smart Grids and Critical Infrastructure Facilities

Cybersecurity roadmaps for smart grids focus on many areas, including resilience, secure updates, and response protocol (Yaddanapudi et al., 2023; Mashima et al., 2024; Maleki et al., 2025).

GenAI-assisted release strategies therefore need to fit into the existing mechanisms of NERC-type or national regulation and not work around them.

Space, UAV, and Edge IoT

Critical infrastructure "in space" and UAV fleets are prone to specific safety and sovereignty issues (Cordill et al., 2025; Ellis et al., 2023).

AI based OTA management has to comply with the sector specific regulations and in some cases also military or dual use export controls.

Capitalizing on Cybersecurity Conceptual Frameworks

Cybersecurity frameworks for edge computing, the Internet of Things and smart grids (Reyes-Acosta et al. 2025; Yaddanapudi et al. 2023) can be lobbied in reusing organizing GenAIs adoption:

Extend threat models that already exist, adding GenAI as a new component Prompt injection, Data poisoning, Model Theft

Take GenAI decisions into consideration as an attack surface and get the monitoring in place for an anonymous artificial behavior.

 doi:10.20944/preprints202602.1974.v1

Require systems that must be released into safety-critical applications of any kind such as traceability, reviewability and contestability.

Regulators are unlikely to accept "the AI said it was safe" as an argument - organizations will need to be in a position to show that human experts, sound evidence and documented processes were still in charge.

7.4. Metrics and KPIs of Measuring Impact

To prevent themselves from being lost in AI hype, enterprises need hard-nosed metrics to prove whether GenAI is indeed improving the release management process without having any negative impact on safety or security.

We can categorize KPIs based on the different categories:

(a) Flow and Speed Metrics

Deployment Frequency How frequently there is a successful release (that includes firmware).

Lead Time for Changes: Time of codes committed to running in production.

Modeling other extremes for EMI/EMC failing at Sometimes other tools can model information on circadian failures for EMI/EMC, but the best tool is Distribution analysis to detect.

Goal: see improvements made where appropriate (e.g. SaaS or at least some non-critical IoT) with unacceptable trade-off (e.g., safety-critical domains)

Metrics for Quality and Reliability required(b)

Change Failure Rate: Percentage of the releases, the result of release which resulted in an incident, rollback or hot fix.

Mean Time to Resolve (MTTR) intend incidents, especially those that need RCA and cross team coordination.

Post-Release Defect Density: Post-application release- Specific, Number of defects found after a release.

That here, GenAI should demonstrably reduce MTTR, and increase the quality of the RCA, not just "generate more text."

(c) Security Metrics

Number and intensity of security incidents related to the releases (vulnerabilities sorted out, misconfigured elements).

Time to Patch High severity vulnerabilities once discovered.

Coverage of Security Checks: proportion of the releases put through automated security analysis and human review(Sotiropoulos et al., 2023; Mashima et al., 2024)

If GenAI is indeed "security-aware", these metrics should improve - or at least not get worse - as AI adoption increases.

(d) Counseling Risk an Governance Metrics unique to A.I.

Override Rate: What percentage of times human experts have overridden GenAI recommendations (both "approve" and "reject".

Incident Attribution: Number of significant incidents that that flawed GenAI recommendations were a contributing factor in

Model Drift and Performance; Periodic testing of outputs of GenAI algorithms on benchmark scenarios to detect possible performance degradation or model drift in the output.

Tracking these metrics helps different organizations adjust how autonomous AI should be, and promptly take appropriate decisions regarding when/where to move along the maturity model.

 doi:10.20944/preprints202602.1974.v1

Table 2. Maturity Stages, Key Activities, Risks, and KPIs.

Stage	Key Capabilities	Typical Activities	Primary Risks	Core KPIs
0	Manual release processes, siloed tools	Email-based approvals, manual OTA coordination, ad-hoc incident reviews	Human error, slow patching, poor traceability	Deployment frequency (low), change failure rate, time-to-patch
1	CI/CD, dashboards, basic automation	Automated builds/tests, scripted deployments, log aggregation, basic vulnerability scanning	False sense of security from shallow metrics, limited coverage	Deployment frequency, MTTR, basic incident counts, test coverage
2	GenAI for documentation & triage (human-reviewed)	AI-drafted release notes, automated ticket categorization, suggested owners for defects	Misleading summaries, triage mistakes if unchecked	Time spent on documentation, ticket triage time, developer satisfaction, override rate of AI docs
3	AI-in-the-loop decision support (non-critical systems)	AI-assisted risk scores, rollout strategy suggestions, RCA drafts influencing but not replacing decisions	Over-reliance on AI, hidden biases in recommendations	Change failure rate, MTTR, rollback frequency, AI recommendation adoption vs. override rates
4	Regulated AI-in-the-loop for safety-critical environments	AI-supported safety cases, compliance cross-checking, AI-assisted analysis of grid/medical/UAV telemetry	Safety/regulatory breaches if governance fails, complex accountability	Severe incident count, regulatory findings, documented AI-human decision trails, external audit outcomes

8. Discussion and Future Research Guidelines

This article has focused on why Generative AI can be a mass game changer for firmware and software release management and has a great impact on OTA-centric, safety-critical and highly connected environments. At the same time, it has highlighted GenAI as introducing new risks, governance challenges and organisational demands. This section places the proposed framework within the existing theory and presents practical implications as well as some key limitations and directions for future research.

8.1. Theoretical Implications

A first contribution to the literature is theoretical - i.e., with respect to digital-physical product development and coordination. Prior work has shown how digital-physical products require complex practices of coordination and "ever-in-the-making" offerings that continue to change post-launch (Hendler, 2021; Lehmann & Recker, 2022). By foregrounding GenAI as an actor in release pipelines, in this paper, this conversation is extended to beyond coordination amongst teams to who

 doi:10.20944/preprints202602.1974.v1

and what partakes in coordination. Release management becomes a socio-technical process in which GenAI systems aid in building shared narratives of risk, impact and readiness between when firmware, cloud services and even physical devices are involved.

Second, the framework adds to the research literature of OTA security and cyber-physical resilience. Prior work has mainly focused on OTA update mechanisms, secure channels and domain-specific cybersecurity issues, for example, NB-IoT over satellite for firmware updates Bas & Dowhuszko 2021, secure OTA for CubeSats Molina, et al., 2023, smart grid threats Mashima, et al., 2024 & Maleki, et al., 2025 or IoT health device security Affia, et al., 2023. By bringing together these strands together with GenAI assisted decision making, the article reframes OTA, not simply as a transport and cryptography problem, but a decision quality problem, i.e. what is released, when and on what decisions about AI mediated interpretation of telemetry and risk.

Third, the work cuts across emerging research on responsible/operational AI. The existing studies on the application of ML for intrusion detection and smart grid cybersecurity (Jamshidi et al., 2025; Ibrahim & Kashef, 2025) mainly focus on the performance of detection and classification. Instead, this article considers GenAI as an operational workflow embedded (decision support layer) and raises the issues of accountability, explainability and human-in-the-loop governance. It therefore contributes to a change towards model-centric evaluation ("Is the classifier accurate?"), to pipeline-centric evaluation ("What happens to releases, incidents and safety outcomes when AI recommendations have influence on decisions?"

Overall, the conceptual framework opens the door to future theoretical work where GenAI is considered a first-class, part of socio-technical systems for cyber-physical operations; in contrast to a black box addition to existing tools.

8.2. Implications to Stakeholders in Industries

For both tool vendors and ecosystem builders (CI/CD platforms, observability providers, security toolchains), the proposed roadmap addresses a clear definition of how capabilities should be sequenced. Vendors can productize GenAI in stages, which might be from documenting and triage assistants to risk-scoring and RCA-support for non-critical systems and a regulated AI-in-the-loop support for regulated AI, specifically for smart grid, medical, automotive, and aerospace customers. The framework also highlights the need for close connectivity to telemetry, logs, vulnerability management and over-the-air (OTA) mechanisms, and not as an isolated chatbot.

For operators of critical infrastructure - utilities, hospitals, aerospace operators, automotive OEMs - the important thing to consider is that GenAI adoption cannot be left to "the AI team" alone. It has to be handled as a cross-functional effort that includes release management, SRE, cybersecurity, safety engineering, and compliance. The basis for the internal readiness assessment offered by the maturity model is that organizations will be able to position themselves on the stage 0-4 spectrum and to have a vision for achievable steps there with risks in mind rather than jumping right into autonomous AI-gated releases.

For policymakers and regulators working in energy, health, automotive and aerospace, this analysis makes it a compelling point - that GenAI is already advancing into the operational stack as well as (back-office) analytics. Regulatory frameworks for AI in medical devices, smart grids and mobility will have to cover not only model evaluation but also the process evaluation: the discussion of how AI recommendations come into the process of making release decisions, the auditing, and the allocation of responsibility between vendors, operators and AI providers.

Finally, for the leaders and TPM of engineering, the discussion suggests a change in competency profiles. Teams need to develop AI and data literacy, to develop the capacity to interrogate AI-sourced evidence and fluency in risk negotiation over the need to report on simple status. Organizations that do not invest in these capabilities may be risking that they underutilize GenAI (which can be used as a superficial documentation aid but no deeper) or over-trust it in areas where failure (which is rare) is devastating itself.

 doi:10.20944/preprints202602.1974.v1

8.3. Limitations

This work is conceptual and integrative on its own. It is a synthesis of knowledge from multiple areas - IoT health, smart grids, UAVs, satellites, automotive and smart homes for example - but lacks new empirical data and controlled experiments. As a consequence, the framework should be considered a scaffold rather than a validated prescription that is open to hypotheses.

Second, the analysis abstracts away some domain-specific details that are necessary. For example, the regulations and working dynamics of a national-power grid are very different from the working and regulatory ones of consumer wearables or CubeSat payloads. While the approach of the maturity model, and the taxonomy of use cases, are designed to be cross-sectoral, some of the recommendations may require significant adaptation. There is a risk of over-generalization if practitioner's uncertainties by running the framework do not place careful focus on grounding the framework in domain specific safety cases and regulatory obligations.

Third, there is a relatively high level of abstraction used in this article, where "GenAI" and "LLMs" are defined in a similar way together, urging disparate architectures, ways of deployment, and training regimes. It does not distinguish more precisely on the issue of on-premises vs. cloud-hosted, foundation models vs. domain-specific fine-tunes, or retrieval-augmented vs. standalone - all of which can have material differences regarding security, privacy, and operation risk consideration.

Finally, the paper tends to be more process-focused (vs. rigorous quantitative risk modeling) and organizational instead. It lacks, for example, formal probabilistic models of AI-causing failure rates, or comparable information comparing the net risk of an AI augmented versus purely human release pipeline. These omissions are partly intentional - as a reflection of the speed with which GenAI deployment is happening in industry - but they reduce the precision with which cost-benefit trade-offs can be judged.

8.4. Future Research Agenda

A number of lines of inquiry are a direct result from these limitations.

Empirical Yet Empirical Studies of GenAI Augmented Release Pipelines

There is a current and severe need for deep case studies and longitudinal field research in certain areas, such as where smart grids, insulin delivery systems, or worst-case UAV fleets, GenAI have been incorporated into release workflows. Such studies should monitor tangible results, such as change of failure rates, MTTR, severity of incidents, and measurements of safety or compliance results, before and after implementation. Comparative studies between organizations in various states of maturity would be a useful means to validate (or revise) the proposed roadmap.

Safety and Reliability Evaluation using Simulation Based and Digital Twin

Given how hard it is to experiment directly on a critical infrastructure, simulation and digital twin environments seem like a promising avenue for understanding the effect of AI-augmented decisions on the dynamics of systems. Borrowing further from existing research on architectures for digital twin of (smart) homes and grid(s), one could in future research scenario-gen AI-driven release strategies in the presence of adversarial conditions, rare failures or cascading incidents to quantify benefit and risk.

Standardization and Benchmarks for Artificial Intelligence Supported Release Decisions

At the moment, there is no generally recognized benchmark suite for testing AI-aided release management. Research communities and standards bodies (e.g., with the introduction of the specification of cybercrime incident requirements by ISO 31667) and regulatory authorities (e.g., with the introduction of upcoming cybercrime regulations by the EU) may define benchmark scenarios, data sets, and metrics to assess GenAI tools dimensions present by e.g. accuracy of RCA, quality of risk summaries, robustness to corrupted logs, susceptibility to propitiate or data injection, and the need to alert in scenario analysis and propose a classification. This would make it possible to more rigorously compare tools and provide evidence-based regulation.

Human-Prised Interaction and Organizational Studies

 doi:10.20944/preprints202602.1974.v1

Another promising direction is to examine the ways in which the roles of humans change around the pipelines, augmented by AI in practice: how TPMs, SREs, security engineers in practice use, trust, fight, or work around the recommendations of GenAI in practice over time. Ethnographic and mixed methods can enlighten research about failure modes that are not visible in technical assessments (e.g. excessive deference reserved for AI, "checking boxes" to use AI in order to appease managerial pressure, or silence resistance and underutilization).

Model of Formal Risk and Accountability

Finally, it would be interesting to develop more formal frameworks for determining responsibility and liability when GenAI is involved in releasing decisions in future work. This could involve combining methods from safety engineering, legal analysis and algorithmic accountability work to suggest more concrete patterns for audit trails, logging of decisions, and shared responsibility between model providers, system operators and (potentially) regulators.

Taken together, these research directions are intended to help bring the topic from being one of promise, to being one of practice that is grounded and domain specific, and to ensure that the future of GenAI-enabled release management is not only faster and more automated, but also safer, more transparent, and more accountable.

9. Conclusions

9.1. Summary of Key Insights

This article explores the role of Generative AI in the reconfiguration of firmware and software release management in the era of ubiquitous connectivity, OTA delivery, and cyber physical dependency. For GenAI in particular, it has attempted to synthesize work across IoT, smart grids, medical devices, UAVs, satellites, automotive systems, smart homes and made the following case: GenAI is not only a convenience feature for documentation purposes, but a potential decision shaping actor in complex, safety relevant pipelines.

The proposed GenAI-augmented release management framework identifies LLMs and the related models as tools used in automating the release notes, risk and impact summaries, defect triage and RCA generation using rich telemetry and operational data as the groundwork. The use-case analysis has established that such capabilities may save time in the release cycles, enhance the situational awareness and de-cognitive load of the users - provided they are embedded in healthy cybersecurity architectures with OTA-mechanisms.

At the same time, through the analysis of ethics, security and reliability, it is evident that GenAI creates a new attack surface, poses risks to privacy, and introduces failure modes. In areas like safety-critical areas, uncritical adoption will be crazy. The discussion of roles and organizational implications underlines the fact that it is not "adding AI" that should be the main challenge, rather it is an image to rethink socio-technical systems into an interaction between human expertise, regulatory constraints and AI recommendations, in a closed, traceable manner. The maturity model and adoption roadmap are used to translate these insights into a governance-first path for enterprises which is staged.

Overall, the main takeaway is simple: GenAI can clearly have tangible positive effects on the management of release for firmware/software systems, though the benefits are highly limited by the seriousness with which organizations consider security, safety, and management.

9.2. Responding to the Research Questions

RQ1: How can the technique of GenAI be used in the augmentation of firmware/software release management pipelines?

GenAI can be used to augment pipelines while automating difficult (cognitively heavy) but structurally repetitive tasks: Automating the generation of stakeholder-specific release notes, Automating the summarization of risk logs and test results, Automating the triaging of defect key points, Generating the RCA narrative generated by language. This is playing a critical role as an

 doi:10.20944/preprints202602.1974.v1

information compression and synthesis layer on top of CI/CD, observability, and security tooling (versus playing the role of replacing these systems).

RQ2: Which are the use cases in important systems and their dangers?

Key use cases include automatic RCA report, intelligent RCA, terminal triage chatbot, security-aware release decision support, domain abstraction-specific simulation of risk and smart grid and medical IoT, UAV, space systems and the automotive firmware. The linked risks tend to congregate around hallucinated or biased evaluations, the possibility of Ernesto of adversarial toy with AI inputs, permeating sensitive telemetry accessibility, and an over-reliance on AI in areas where rare failures can have far-reaching effects.

RQ3: How are jobs such as TPMs, release managers and security engineers going to change?

TPMs and release managers away from the role of manual status aggregation and towards triggering and questioning AI-augmented flows, as risk negotiators and governance agents. DevOps and SRE teams, for example, take on the role of stewards of an AI-administered automation stack reliable to include guard rails and fall backs alongside security engineers, who evolve into a role of responsible custodians and validators of the security information and knowing about the data and thinking that goes into the examples to GenAI. The roles then across vary from AI literacy to data literacy, and domain related regulatory competence becomes core capabilities.

RQ4: What do you think would be the roadmap large enterprises need to adopt GenAI safely and effectively?

Enterprises need to be able to move through a stage maturity model: from manual and siloed processes (Stage 0), to analytics and basic process automation (Stage 1), enabled GenAI documentation and triage (Stage 2), to AI in the loop decision support of non-critical systems (Stage 3), and then tightly controlled AI support of safety-critical spheres (Stage 4). Each stage requires accumulation precedents investment of data infrastructure, secure MLOps, governance and express movement to sectoral regulations.

9.3. Final Remarks

The future of release management with Generative AI will not be determined by how aggressively organizations use automation, but how smarts they are in limiting such automation. Large enterprises flanking GenAI conservation: Let's face it: large enterprises can leverage a mission-oriented GenAI use case - but let's be conservative and proactive, always recognizing the need to measure results rigorously and to then proceed with GenAI-in-the-loop decisions in non-critical systems with great human supervision.

Human-in-the-loop control, transparent governance, and domain-specific rigor is not some flavor option or extra, it is the necessary pre-requisite for handling GenAI to become more than a color substitute (a promising existence as a prototype) to being an element of a now-trustworthy release management fabric for cyber-physical systems.

References

Kim, J., Lee, K., & Choi, S. (2020). Machine learning-based code auto-completion implementation for firmware developers. Applied Sciences, 10(23), 8520. https://doi.org/10.3390/app10238520

Harding, W., Hartmann, A., O'Brien, S., Sinzig, V., Break, G., & Sinzig, N. Securing a Connected Future. https://doi.org/10.1007/978-3-032-07309-9

Jamshidi, S., Nikanjam, A., Wazed, N. K., & Khomh, F. (2025). Leveraging Machine Learning Techniques in Intrusion Detection Systems for Internet of Things. arXiv preprint arXiv:2504.07220. https://doi.org/10.48550/arXiv.2504.07220

Sharma, N., Bisht, R., Sontakke, R., & Vinchurkar, K. (2024). Regulatory Insights into Artificial Intelligence in Drug Delivery and Medical Devices. In AI Innovations in Drug Delivery and Pharmaceutical Sciences; Advancing Therapy through Technology (pp. 199-228). Bentham Science Publishers. https://doi.org/10.2174/9789815305753124010

 doi:10.20944/preprints202602.1974.v1

Niu, Y., & Lam, S. K. (2025). Securing Automated Insulin Delivery Systems: A Review of Security Threats and Protectives Strategies. arXiv preprint arXiv:2503.14006. https://doi.org/10.48550/arXiv.2503.14006

Ibrahim, N., & Kashef, R. (2025). Exploring the emerging role of large language models in smart grid cybersecurity: a survey of attacks, detection mechanisms, and mitigation strategies. Frontiers in Energy Research, 13, 1531655. https://doi.org/10.3389/fenrg.2025.1531655

Affia, A. A. O., Finch, H., Jung, W., Samori, I. A., Potter, L., & Palmer, X. L. (2023). IoT health devices: exploring security risks in the connected landscape. IoT, 4(2), 150-182. https://doi.org/10.3390/iot4020009

Rico, A., Pareek, S., Cabezas, J., Clarke, D., Ozgul, B., Barat, F., ... & Noguera, J. (2024). Amd xdna™ npu in ryzen™ ai processors. IEEE Micro. DOI: 10.1109/MM.2024.3423692

Cordill, B., Fang, D., & Xu, S. (2025). A Comprehensive Survey of Security and Privacy in UAV Systems. IEEE Access. DOI: 10.1109/ACCESS.2025.3583985

Mashima, D., Chen, Y., Roomi, M. M., Lakshminarayana, S., & Chen, D. (2024). Cybersecurity for modern smart grid against emerging threats. arXiv preprint arXiv:2404.04466. https://doi.org/10.48550/arXiv.2404.04466

Maleki, S., Pan, S., Lakshminarayana, S., & Konstantinou, C. (2025). Survey of load-altering attacks against power grids: Attack impact, detection and mitigation. IEEE Open Access Journal of Power and Energy. DOI: 10.1109/OAJPE.2025.3562052

de Zambotti, M., Vallat, R., Pho, G., Goldstein, C., & Patel, S. (2025). Toward better evaluation of consumer sleep technologies: a call for rigor, context, and collaboration. Sleep Advances, 6(4), zpaf063. doi:10.1093/sleepadvances/zpaf063

Yaddanapudi, A., Chaudhary, K., Alabdulaziz, M., Albabtain, M., Raju, N. H., Sirimongkarakorn, T., ... & Daim, T. U. (2023). Cybersecurity Technology Roadmap: Data and Information Security for Smart Grid Industry. In Cybersecurity: A Technology Landscape Analysis (pp. 193-218). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-34843-3_8Hendler, S. (2021).

Exploring coordination practices in digital–physical product development. Journal of Manufacturing Technology Management, 32(3), 742-771. https://doi.org/10.1108/JMTM-06-2019-0229

Lehmann, J., & Recker, J. (2022). Offerings That are "Ever-in-the-Making" How Digital Ventures Continuously Develop Their Products After Launch. Business & Information Systems Engineering, 64(1), 69-89. https://doi.org/10.1007/s12599-021-00730-y

Zunaidi, M. R., Sayakkara, A., & Scanlon, M. (2024). Systematic Literature Review of EM-SCA Attacks on Encryption. arXiv preprint arXiv:2402.10030. https://doi.org/10.48550/arXiv.2402.10030

Thomasian, A. (2024). RAID Organizations for Improved Reliability and Performance: A Not Entirely Unbiased Tutorial (1st revision). arXiv preprint arXiv:2401.03235. https://doi.org/10.48550/arXiv.2401.03235

Karim, M. R., Haque, M. A., Ahmed, S., Reza, M. N., Lee, K. D., Kang, Y. H., & Chung, S. O. (2025). Effects of Sensor Speed and Height on Proximal Canopy Reflectance Data Variation for Rice Vegetation Monitoring. Agronomy, 15(3), 618. https://doi.org/10.3390/agronomy15030618

Molina, F. X., Baccelli, E., Zandberg, K., Donsez, D., & Alphand, O. (2023, September). Cubedate: Securing Software Updates in Orbit for Low-Power Payloads Hosted on CubeSats. In 2023 12th IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN) (pp. 1-6). IEEE. DOI: 10.23919/PEMWN58813.2023.10304910

Reyes-Acosta, R. E., Mendoza-González, R., Oswaldo Diaz, E., Vargas Martin, M., Luna Rosas, F. J., Martínez Romo, J. C., & Mendoza-González, A. (2025). Cybersecurity Conceptual Framework Applied to Edge Computing and Internet of Things Environments. Electronics, 14(11), 2109. https://doi.org/10.3390/electronics14112109

Campell, C., Graber, J., Tashakkori, R., & O'Brien, W. (2022, October). IoT Apiary Fleet Management with Jenkins. In 2022 IEEE 13th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) (pp. 0328-0335). IEEE. DOI: 10.1109/UEMCON54665.2022.9965635

Bas, J., & Dowhuszko, A. A. (2021). On the use of NB-IoT over GEO satellite systems with time-packed optical feeder links for over-the-air firmware/software updates of machine-type terminals. Sensors, 21(12), 3952. https://doi.org/10.3390/s21123952

 doi:10.20944/preprints202602.1974.v1

Furey, K., Conway, R., O'Keeffe, D., & Lyons, G. M. (2007). The application of a use case/task based approach in the development of software for a portable neuromuscular stimulator device. Medical engineering & physics, 29(7), 765-774. https://doi.org/10.1016/j.medengphy.2006.08.016

Keliris, A., & Maniatakos, M. (2017). Demystifying advanced persistent threats for industrial control systems. Mechanical Engineering, 139(03), S13-S17. https://doi.org/10.1115/1.2017-Mar-6

Solomon, G. J., Zhang, P., Liu, Y., & Brooks, R. (2019, December). Blockchain Based Owner-Controlled Secure Software Updates for Resource-Constrained IoT. In International Conference on Network and System Security (pp. 371-386). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-36938-5_22

Zdankin, P., Picone, M., Mamei, M., & Weis, T. (2022, July). A digital-twin based architecture for software longevity in smart homes. In 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS) (pp. 669-679). IEEE. DOI: 10.1109/ICDCS54860.2022.00070

Carter, J., Grommon, E., & Harris, P. (2016). From conceptual to operational: Over-the-air-programming of land mobile radios. Physical Communication, 19, 18-29. https://doi.org/10.1016/j.phycom.2016.01.007

Simon, P., & Graham, S. (2017, June). Potential Privacy Ramifications of Modern Vehicle Software and Firmware. In 16th European Conference on Cyber Warfare and Security (ECCWS) (pp. 452-458). DOI: 10.1109/JIOT.2023.3327447

Wang, N. Z., & Chien, H. Y. (2023). Design and implementation of MQTT-based over-the-air updating against curious brokers. IEEE Internet of Things Journal, 11(6), 10768-10777. DOI: 10.1109/JIOT.2023.3327447

Sotiropoulos, P., Mathas, C. M., Vassilakis, C., & Kolokotronis, N. (2023). A software vulnerability management framework for the minimization of system attack surface and risk. Electronics, 12(10), 2278. https://doi.org/10.3390/electronics12102278

Ellis, T., Hitaj, B., Lindqvist, U., Shands, D., Tinnel, L., & DeBruhl, B. (2023). Critical infrastructure security goes to space: Leveraging lessons learned on the ground. arXiv preprint arXiv:2309.15232. https://doi.org/10.48550/arXiv.2309.15232