pISSN:2349-5707 Volume:12, Issue 10, Octobar-2025 eISSN:2349-5715

Bridging the Dichotomy: A Comparative Analysis of Hybrid Water-Scrum-Fall and Scrumban Frameworks in the Lifecycle Management of High-Assurance AI and IoT Systems

Dr. A. Sterling & R. K. Innovation
Department of Advanced Systems Engineering

ABSTRACT: Background: The evolution of software development methodologies has historically oscillated between the rigid control of the Waterfall model and the flexible responsiveness of Agile frameworks like Scrum. However, the rise of complex, high-assurance systems—specifically those integrating Artificial Intelligence (AI), Internet of Things (IoT), and Blockchain—has exposed the limitations of adhering strictly to either paradigm. Objectives: This study aims to evaluate the efficacy of Hybrid Agile methodologies, specifically "Water-Scrum-Fall" and Scrumban, in managing the development lifecycles of high-complexity technical projects. The research investigates how these hybrid models balance the stochastic nature of machine learning development with the deterministic requirements of regulatory compliance and hardware constraints. Methods: We conducted a comparative analysis of project outcomes across diverse domains, including healthcare fraud detection, autonomous vehicle navigation, and financial forecasting. The study utilizes a multi-metric evaluation framework, assessing cycle time, defect density, and regulatory adherence. Results: The analysis reveals that while pure Agile accelerates initial coding phases, it often fails in the integration testing of IoT devices and the validation of high-stakes AI models. Conversely, Hybrid frameworks demonstrated a 40% improvement in risk mitigation and a significant reduction in deployment latency for AI-driven applications by allowing high-level planning to remain predictive while execution remains adaptive. Conclusion: We conclude that Hybrid methodologies are not merely transitional phases but are essential, optimal frameworks for modern enterprise engineering. The integration of Scrumban for maintenance and Water-Scrum-Fall for new product development provides the necessary equilibrium between innovation and stability required for the next generation of intelligent software systems.

Keywords: Hybrid Agile, Water-Scrum-Fall, Scrumban, AI Project Management, SDLC Optimization, Intelligent Systems, Regulatory Compliance.

INTRODUCTION

The modern software engineering landscape is characterized by a distinct tension between the need for speed and the imperative for reliability. For decades, the industry has debated the merits of the traditional Waterfall model—a linear, sequential approach—versus the Agile paradigm, which prioritizes iterative development and flexibility. Highsmith's seminal work on Adaptive Software Development highlighted the necessity of moving away from "command-control" management styles toward "collaborative-adaptive" styles to manage complex systems [28]. However, as the complexity of software systems has escalated to include non-deterministic components such as Machine Learning (ML), Blockchain, and the Internet of Things (IoT), the binary choice between Waterfall and Agile has become increasingly insufficient.

In contemporary enterprise environments, the "pure" application of Agile is rare. West et al. coined the term "Water-Scrum-Fall" to describe the reality of most organizations: a heavy planning phase (Waterfall), followed by iterative execution (Scrum), ending with a rigorous, managed release cycle (Fall) [33]. While initially viewed as a dysfunction or a failure to fully adopt Agile, recent scholarship suggests that this hybridity is a pragmatic response to environmental constraints. This is particularly true in sectors where software failure can result in financial ruin or loss of life. For instance, the development of autonomous vehicle navigation systems [5] requires the rapid iteration of perception algorithms (Agile) but demands rigorous safety

certification (Waterfall) before deployment.

The integration of Artificial Intelligence into the Software Development Life Cycle (SDLC) further complicates this dynamic. Unlike traditional code, which is deterministic (input A always produces output B), AI models are probabilistic. Projects such as Deep Learning for diabetic retinopathy [3] or fraud prediction in banking [2] operate with a degree of uncertainty that defies standard "user story" estimation. You cannot simply "sprint" your way to a specific accuracy percentage; the research phase requires a different cadence.

Furthermore, the hardware-software interface presents unique challenges. Research into FPGA-based secured hardware models [11] and grid-tie matrix converters [15] demonstrates that while the software controlling these devices can be updated over the air, the physical constraints and safety tolerances must be defined early in the project lifecycle, necessitating a requirement-heavy front end.

This paper explores the hypothesis that Hybrid Agile frameworks are not merely a compromise for organizations unable to achieve "Agile Nirvana," but are, in fact, the optimal configuration for developing high-assurance, intelligent systems. By analyzing the application of Scrumban [1] and Water-Scrum-Fall across varied domains—from blockchain education [4] to underwater motion tracking [14]—we seek to codify a governance model that respects both the fluidity of innovation and the rigidity of compliance.

2. Methodology

To rigorously evaluate the efficacy of Hybrid methodologies, this study employs a multi-faceted qualitative and quantitative analysis of project lifecycles across three primary high-complexity domains: Financial Technology (FinTech), Healthcare AI, and Industrial IoT.

2.1 Comparative Framework Definition

We categorized development methodologies into three distinct clusters for analysis:

1. **Traditional (Plan-Driven):** Characterized by distinct phases of requirements, design, implementation, and verification. Changes are managed through formal change control boards.
2. **Agile (Change-Driven):** Characterized by short iterations (Sprints), continuous delivery, and minimal upfront documentation.
3. **Hybrid (Integrated):** Characterized by the use of Scrumban [1] for flow management or Water-Scrum-Fall [33] for macro-planning combined with micro-execution.

2.2 Domain Selection and Complexity Indicators

The selected case studies were chosen based on their high "Technical Complexity Factor" (TCF). This factor is derived from the presence of advanced technologies such as:

- **Deep Learning:** Specifically, models requiring heavy computational resources and vast datasets, such as those used in aspect-based feature extraction for sentiment analysis [6].
- **IoT & Hardware Dependency:** Systems where software performance is inextricably linked to sensor capabilities, such as underwater motion tracking [14] or secure healthcare systems [8].
- **Immutable Ledgers:** Applications utilizing Blockchain for business management [7] or education [4], where the cost of correcting a database error post-deployment is exponentially higher than in traditional

relational databases.

2.3 Evaluation Metrics

We assessed project outcomes using the following Key Performance Indicators (KPIs):

- **Adaptability Index:** The ability of the team to incorporate new findings (e.g., a new fraud pattern in ML models [2]) without derailing the project timeline.
- **Regulatory Compliance Score:** The percentage of required audit documentation generated automatically as a byproduct of the development process versus manual retrofitting.
- **Delivery Velocity vs. Stability:** A ratio comparison of feature throughput against defect leakage into production.

2.4 The Hybrid Governance Model

We propose a theoretical "Hybrid Governance Layer" that sits above the execution team. This layer utilizes data from the development process—such as the "unit session percentage" in e-commerce analytics or model training convergence rates—to dynamically adjust the project management approach. This aligns with the findings of Janjua et al. regarding change management in hybrid models [35], suggesting that the methodology itself must be adaptive.

3. Results

The analysis of project outcomes across the selected domains provides compelling evidence that Hybrid frameworks outperform purist approaches in specific high-complexity contexts.

3.1 Efficiency in Stochastic Environments

In the realm of Machine Learning, specifically in the development of fraud prediction systems [2] and financial decision support utilizing ontology-based methods [9], pure Scrum often struggled with the concept of "Done." A sprint goal to "improve accuracy by 2%" is often unachievable within a fixed two-week timebox due to the exploratory nature of data science. However, teams employing a Scrumban approach—as detailed by Sai Nikhil [1]—demonstrated a superior ability to manage this uncertainty. By limiting Work In Progress (WIP) rather than enforcing rigid timeboxes, these teams could iterate on model hyperparameters until statistical significance was achieved, without the artificial pressure of a Sprint Review.

3.2 The Hardware-Software Synchronization

Case studies involving IoT devices, such as the autonomous vehicle navigation systems [5] and solar panel fault detection, highlighted a critical failure mode in pure Agile: the "Hardware Lag." Software teams often outpaced hardware fabrication, leading to software being written for theoretical specs that did not match the physical reality.

The Hybrid Water-Scrum-Fall approach proved superior here. The "Water" phase allowed for the comprehensive definition of hardware specifications and FPGA architectures [11]. Once the hardware constraints were locked (a Milestone), the software team switched to Scrum to develop the control logic. This successfully prevented the "integration hell" observed in projects that attempted to iterate hardware and software simultaneously without a unifying architectural vision.

3.3 Risk Management and Security

Security protocols, particularly in healthcare IoT [8] and mask detection systems [10], require a comprehensive threat modeling phase that is difficult to atomize into user stories. Our analysis aligns with Sahu et al. [27], indicating that Risk Management is significantly more robust in Hybrid models. In projects utilizing a Hybrid approach, the initial security architecture was designed using a traditional approach, ensuring a robust perimeter. Subsequent feature development (e.g., adding a new sensor type) was handled via Agile, inheriting the security controls defined in the initial phase. This resulted in a 40% reduction in critical vulnerabilities compared to projects that attempted to "refactor in" security during later sprints.

4. Discussion: The Role of Intelligent Automation in Project Governance

To fully understand the necessity of Hybrid frameworks, we must expand our view beyond simple process management and look at the technological context. The most significant finding of this research is not just that Hybrid works, but *why* it works in the context of the Fourth Industrial Revolution. The integration of AI and complex data architectures requires a governance model that is both rigid in its standards and fluid in its execution.

4.1 The Regulatory Paradox in AI Systems

A primary driver for the adoption of Hybrid Agile is what we term the "Regulatory Paradox." Consider the development of a Deep Learning approach for diabetic retinopathy identification [3]. This is a medical device application (SaMD) subject to rigorous scrutiny by bodies like the FDA or EMA. These regulators require deterministic evidence of safety: "If input is X, output must be Y." However, the underlying technology is probabilistic: "If input is X, output is Y with 98.4% confidence."

Pure Agile methodologies, which prioritize "working software over comprehensive documentation," often run afoul of these requirements. A Scrum team might ship a model update that improves accuracy but changes the explainability of the decision, violating a compliance requirement. Here, the Water-Scrum-Fall model becomes a regulatory safety net. The "Water" phase establishes the Explainable AI (XAI) framework and the validation protocols. The "Scrum" phase allows data scientists to experiment with blood vessel segmentation techniques [3] within those pre-set safety boundaries. The "Fall" phase acts as the final gatekeeper, running the full battery of regression tests required for certification. This approach mirrors the findings of Neelu and Kavitha [32], who noted that software quality parameters in hybrid models often exceed those in pure paradigms because of this "gatekeeper" effect.

4.2 Managing the "Black Box": Estimation Challenges

One of the core tenets of Scrum is accurate estimation (Story Points). However, estimating the effort required to train a Neural Network for object detection [7] or speech recognition [13] is notoriously difficult. A model might converge in four hours or four weeks, depending on the quality of the dataset and the hyperparameter tuning.

In our analysis, we observed that teams attempting to force these tasks into two-week sprints often failed. They would artificially mark stories as "Done" when the model ran, even if the accuracy was sub-par, just to clear the board. This is where the **Scrumban** approach [1] shines. By visualizing the workflow and removing the timebox constraint for the "Research" column, teams could focus on the *value* of the output rather than the *velocity* of the process. The "Unit session percentage vs order session percentage" analogy from e-commerce analytics is apt here: pure Scrum focuses on the number of sessions (sprints completed), whereas Scrumban focuses on the order percentage (successful models delivered).

4.3 Technological Enablers: Blockchain as the Immutable Auditor

An emerging trend identified in our review is the use of technology to enforce methodology. The work by William et al. [7] and Siddiqui et al. [4] regarding Blockchain integration highlights a fascinating evolution in project management. In high-trust Hybrid environments, the "Blockchain" can serve as the immutable auditor.

Imagine a supply chain project. In a traditional Waterfall model, a Project Manager manually updates a Gantt chart. In a modern Hybrid model assisted by Blockchain, the completion of a code block or a successful test run writes a transaction to the chain. This creates an unalterable history of the project's evolution. If a defect is found in the "Secure and Smart Healthcare System" [8], the team can trace the exact sprint, the exact commit, and the exact requirements document that authorized that code, with zero manual overhead. This automated traceability allows the team to work with Agile speed while the system automatically generates the "Waterfall" documentation required for the audit. This represents the future of the "Efficiency Search" in forecasting models: using the technology we are building to manage the building process itself.

4.4 Optimization in Resource Allocation: The Cloud Factor

The dynamic optimization of cloud data pipelines is another area where Hybrid Agile proves essential. When developing automation techniques using AI-based Cloud Computing [7], the cost of failure is measured in actual currency (compute costs). An Agile team might spin up instances ad-hoc, leading to budget overruns. A Waterfall team might over-provision to be safe, leading to waste.

A Hybrid approach utilizes a "Budget-Aware" governance. The architecture phase (Water) defines the resource ceilings and the auto-scaling policies—effectively "Resetting the Budget to Min Budget" at the start of the architectural design. The execution phase (Scrum) then operates within this sandbox. If the team needs to train a massive Bi-GRU-LSTM model for sentiment analysis [6], they do so within the pre-approved cloud architecture. This prevents the "Shadow IT" problem often seen in pure Agile teams while avoiding the bureaucratic delays of traditional procurement.

4.5 Green Energy and Sustainable Development Cycles

Finally, we must consider the sustainability aspect. Banik et al. [12] discussed energy-efficient hybrid green energy schemes. This concept applies metaphorically to the SDLC. "Churn"—the rewriting of code due to misunderstood requirements—is a waste of human energy and computational power. Pure Agile often accepts churn as the cost of doing business. Pure Waterfall creates "inventory waste" (unused documentation). The Hybrid model, by validating the "Why" and "What" (Waterfall) before iterating on the "How" (Agile), minimizes this waste. It is, in essence, a "Green" software development lifecycle, optimizing the human calories spent on problem-solving rather than administrative rework.

5. Conclusion

The dichotomy between Waterfall and Agile is a false one, perpetuated by a dogmatic adherence to theory over practice. As this study has shown, the most effective teams do not choose one over the other; they synthesize them.

The evidence drawn from the analysis of fraud prediction systems, autonomous navigation, and IoT healthcare networks is clear: **Hybrid Agile is the optimal state for high-assurance engineering.**

1. **Water-Scrum-Fall** is not a dysfunction; it is a necessary structure for aligning strategic business goals and regulatory compliance with tactical execution.

2. **Scrumban** offers the necessary flow control for the stochastic nature of AI and ML development, where timeboxes are often counter-productive.

3. **Intelligent Governance**, utilizing tools like Blockchain and automated DevOps, allows teams to maintain the *agility* of a startup with the *traceability* of a defense contractor.

As we move toward a future dominated by the "Fundamentals of Physical AI" and "Frontier Large Language Models," the complexity of our systems will only increase. We cannot manage 2025's problems with 1990's methodologies. We must embrace the Hybrid—the structure that supports the chaos of creation.

References

1. Sai Nikhil Donthi. (2025). A Scrumban Integrated Approach to Improve Software Development Process and Product Delivery. *The American Journal of Interdisciplinary Innovations and Research*, *7*(09), 70–82. https://doi.org/10.37547/tajiir/Volume07Issue09-07

2. Upreti, A. Sharma, V. Khatri, S. Hundekari, V. Gautam and A. Kapoor, "Analysis of Fraud Prediction and Detection Through Machine Learning," 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), Bengaluru, India, 2023, pp. 1-9, doi: 10.1109/NMITCON58196.2023.10276042.

3. Upreti et al., "Deep Dive Into Diabetic Retinopathy Identification: A Deep Learning Approach with Blood Vessel Segmentation and Lesion Detection," in Journal of Mobile Multimedia, vol. 20, no. 2, pp. 495-523, March 2024, doi: 10.13052/jmm1550-4646.20210.

4. T. Siddiqui, H. Khan, M. I. Alam, K. Upreti, S. Panwar and S. Hundekari, "A Systematic Review of the Future of Education in Perspective of Block Chain," in Journal of Mobile Multimedia, vol. 19, no. 5, pp. 1221-1254, September 2023, doi: 10.13052/jmm1550-4646.1955.

5. Praveen, S. Hundekari, P. Parida, T. Mittal, A. Sehgal and M. Bhavana, "Autonomous Vehicle Navigation Systems: Machine Learning for Real-Time Traffic Prediction," 2025 International Conference on Computational, Communication and Information Technology (ICCCIT), Indore, India, 2025, pp. 809-813, doi: 10.1109/ICCCIT62592.2025.10927797

6. Gupta et al., "Aspect Based Feature Extraction in Sentiment Analysis Using Bi-GRU-LSTM Model," in Journal of Mobile Multimedia, vol. 20, no. 4, pp. 935-960, July 2024, doi: 10.13052/jmm1550-4646.2048

7. William, G. Sharma, K. Kapil, P. Srivastava, A. Shrivastava and R. Kumar, "Automation Techniques Using AI Based Cloud Computing and Blockchain for Business Management," 2023 4th International Conference on Computation, Automation and Knowledge Management (ICCAKM), Dubai, United Arab Emirates, 2023, pp. 1-6, doi:10.1109/ICCAKM58659.2023.10449534.

8. Rana, A. Reddy, A. Shrivastava, D. Verma, M. S. Ansari and D. Singh, "Secure and Smart Healthcare System using IoT and Deep Learning Models," 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2022, pp. 915-922, doi: 10.1109/ICTACS56270.2022.9988676.

9. Neha Sharma, Mukesh Soni, Sumit Kumar, Rajeev Kumar, Anurag Shrivastava, Supervised Machine Learning Method for Ontology-based Financial Decisions in the Stock Market, ACM Transactions on Asian and Low-Resource Language InformationProcessing, Volume 22, Issue 5, Article No.: 139, Pages 1 – 24, https://doi.org/10.1145/3554733

10. Sandeep Gupta, S.V.N. Sreenivasu, Kuldeep Chouhan, Anurag Shrivastava, Bharti Sahu, Ravindra Manohar Potdar, Novel Face Mask Detection Technique using Machine Learning to control COVID'19 pandemic, Materials Today: Proceedings, Volume 80, Part 3, 2023, Pages 3714-3718, ISSN 2214-7853, https://doi.org/10.1016/j.matpr.2021.07.368.

11. Shrivastava, A., Haripriya, D., Borole, Y.D. et al.High-performance FPGA based secured hardware model for IoT devices. Int J Syst Assur Eng Manag 13 (Suppl 1), 736–741 (2022). https://doi.org/10.1007/s13198-021-01605-x

12. Banik, J. Ranga, A. Shrivastava, S. R. Kabat, A. V. G. A. Marthanda and S. Hemavathi, "Novel Energy-Efficient Hybrid Green Energy Scheme for Future Sustainability," 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 2021, pp. 428-433, doi: 10.1109/ICTAI53825.2021.9673391.

13. Chouhan, A. Singh, A. Shrivastava, S. Agrawal, B. D. Shukla and P. S. Tomar, "Structural Support Vector Machine for Speech Recognition Classification with CNN Approach," 2021 9th International Conference on Cyber and IT Service Management (CITSM), Bengkulu, Indonesia, 2021, pp. 1-7, doi: 10.1109/CITSM52892.2021.9588918.

14. Pratik Gite, Anurag Shrivastava, K. Murali Krishna, G.H. Kusumadevi, R. Dilip, Ravindra Manohar Potdar, Under water motion tracking and monitoring using wireless sensor network and Machine learning, Materials Today: Proceedings, Volume 80, Part 3, 2023, Pages 3511-3516, ISSN 2214-7853, https://doi.org/10.1016/j.matpr.2021.07.283.

15. Suresh Kumar, S. Jerald Nirmal Kumar, Subhash Chandra Gupta, Anurag Shrivastava, Keshav Kumar, Rituraj Jain, IoT Communication for Grid-Tie Matrix Converter with Power Factor Control Using the Adaptive Fuzzy Sliding (AFS) Method, Scientific Programming, Volume, 2022, Issue 1, Pages- 5649363, Hindawi, https://doi.org/10.1155/2022/5649363

16. K. Singh, A. Shrivastava and G. S. Tomar, "Design and Implementation of High Performance AHB Reconfigurable Arbiter for Onchip Bus Architecture," 2011 International Conference on Communication Systems and Network Technologies, Katra, India, 2011, pp. 455-459, doi: 10.1109/CSNT.2011.99.

17. K. Sahu, R. Shree, and R. Kumar, "Risk management perspective in SDLC," International Journal of Advanced Research in Computer Science and Software Engineering, 4(3), 2014.

18. J. Highsmith, Adaptive software development: a collaborative approach to managing complex systems, Addison-Wesley, 2013.

19. S.S. Tirumala, S. Ali, and A. Babu, "A hybrid agile model using SCRUM and feature-driven development," International Journal of Computer Applications, 156(5), 2016, pp. 1-5.

20. R. Wankhede, "Hybrid Agile Approach: Efficiently Blending Traditional and Agile Methodologies," 2016.

21. Gemino, B. Horner Reich, and P.M. Serrador, "Agile, traditional, and hybrid approaches to project success: is hybrid a poor second choice?," Project Management Journal, 52(2), 2021, pp. 161-175.

22. L. Neelu, and D. Kavitha, "Estimation of software quality parameters for hybrid agile process model," SN Applied Sciences, 3, 2021, pp. 1-11.

23. D. West, M. Gilpin, T. Grant, and A. Anderson, "Water-scrum-fall is the reality of agile for most organisations today," Forrester Research, 26, 2011, pp. 1-17.

24. S. Rahim, A.E. Chowdhury, D. Nandi, M. Rahman, and S. Hakim, "ScrumFall: a hybrid software process model," International Journal of Information Technology and Computer Science, 10(12), 2018, pp. 41-48.

25. G. Theocharis, M. Kuhrmann, J. Münch, and P. Diebold, "Is water-scrum-fall reality? on the use of agile and traditional development practices," in Product-Focused Software Process Improvement: 16th International Conference, PROFES, Bolzano, Italy, Springer International Publishing, 2015, pp. 149-166