DISSERTATION

THE APPLICATION OF AGILE TO LARGE-SCALE, SAFETY-CRITICAL, CYBER-PHYSICAL SYSTEMS

Submitted by

Robin Yeman

Department of Systems Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2025

Doctoral Committee:

Advisor: Yashwant Malaiya

James Adams
Steve Simske
Daniel Herber
Erin Arneson

ABSTRACT

THE APPLICATION OF AGILE TO LARGE-SCALE, SAFETY-CRITICAL, CYBER-PHYSICAL SYSTEMS

The increasing complexity of large-scale, safety-critical cyber-physical (LS/SC/CP) systems, characterized by interconnected physical and computational components that must meet stringent safety and regulatory requirements, presents significant challenges to traditional development approaches. Traditional development approaches, such as the waterfall methodology, often struggle to meet adaptability, speed, and continuous assurance demands. This dissertation explores the feasibility of applying and adapting Agile methodologies to LS/SC/CP systems, focusing on challenges like regulatory compliance and rigorous verification, while intending to prove benefits such as improved risk management and faster development cycles. Through case studies and simulations, this research provides empirical validation of Agile's effectiveness in this domain, contributing a framework for adapting Agile practices to meet the unique demands of LS/SC/CP systems.

Employing a mixed-methods approach, the research comprises five key components. First, a systematic literature review (SLR) was conducted to assess the current state of Agile adoption in LS/SC/CP environments. Second, a comparative analysis of the top 10 Agile scaling frameworks was performed to evaluate their suitability for LS/SC/CP system development. Third, a survey of 56 respondents provided both quantitative and qualitative insights into industry trends, adoption patterns, and Agile's impact on LS/SC/CPs. Fourth, 25 one-on-one interviews with industry practitioners further explored the challenges, benefits, and enablers of Agile adoption in these environments. Finally, lifecycle modeling (LML) using Innoslate was utilized to develop a fictional case study, modeling the development of a mid-size low Earth orbit (LEO) satellite using both

NASA's Waterfall approach (Phase A-D) and an Agile approach with a series of Minimum Viable Products (MVPs).

Findings reveal that Agile methodologies, when adapted for LS/SC/CP systems, enable accelerated development cycles, reducing development time by a factor of 2.5 compared to Waterfall while maintaining safety and regulatory compliance. A key contribution of this study is the introduction of a Continuous Assurance Plugin, which integrates continuous validation within Agile's iterative processes, effectively addressing compliance and safety requirements traditionally managed through phase-gated reviews in Waterfall. Additionally, this research provides:

1. Empirical validation of Agile Scaling Frameworks and their suitability for delivering LS/SC/CP systems.

2. Quantitative and qualitative analysis of Agile's current state and impact in LS/SC/CP environments.

3. Evaluation of key enabling technologies such as Model-Based Systems Engineering (MBSE), Digital Twins, and Continuous Integration/Continuous Deployment (CI/CD) that facilitate Agile adoption for LS/SC/CP systems.

This dissertation advances the understanding of Agile's role in LS/SC/CP system development, providing actionable insights and practical adaptations for organizations seeking to implement Agile in complex, safety-critical domains.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Dr. Yashwant Malaiya, for guiding me through this journey with wisdom and encouragement. Your mentorship in developing rigorous research methodologies has been instrumental in shaping my research approach.

I am also sincerely grateful to my committee members: Dr. James Adams, for your insightful advice on modeling using Innoslate; Dr. Steve Simske, for your incredibly fast and thoughtful feedback on my writing; Dr. Daniel Herber, for your support; and Dr. Erin Arneson, for your encouragement throughout this process.

A special thank you to Ms. Ingrid Bridge, who answered so many questions and kept me on track to meet important deadlines.

I am also thankful to INCOSE, Agile and Systems Engineering Working group, for their regular reviews and guidance, and to NDIA Architecture working group, for reviewing my work and providing valuable feedback, especially around architecture.

Your collective guidance and support have made this work possible, and I truly appreciate the time and effort you have invested in my growth as a researcher.

DEDICATION

This dissertation is dedicated, with deepest gratitude, to my wonderful family. To my mother, who instilled in me the belief that I can accomplish anything, even when faced with adversity; to my father, who has always declared me 'wow' material; to my sister, who has been patient through my distractions, offering a listening ear and support; to my children, who supported me every step of the way; and to my husband, my mentor and inspiration, who believed in me even when I doubted myself, through countless weekends. Your belief in me made this achievement possible. This journey would not have been possible without your unwavering love and support. I share this accomplishment with you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

The application of Agile methodologies to Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems is becoming more prevalent among various industries seeking to enhance adaptability and efficiency in project delivery while maintaining high safety and quality standards. LS/SC/CP systems are characterized by extensive organizational structures, stringent safety requirements, and a profound interplay between computational and physical processes, necessitating tailored approaches to Agile adoption. Dikert defines Large-Scale (LS) systems as organizations with 50 or more people or at least six teams [12]. The delineation of Safety-Critical (SC) systems is essential, as failures in these systems can lead to severe consequences, including loss of life and significant environmental damage [13]. Cyber-physical systems (CPS) integrate computation with physical processes, requiring seamless interactions for monitoring and control through feedback mechanisms, establishing a complex interdependent framework essential for effective system performance [14] [15] [16]. Therefore, this paper defines Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems as entities developed by a collective of over fifty people, where failure can result in significant harm, composed of both computational and physical elements. The current pace of change in LS/SC/CP systems is accelerating significantly due to technological advancements, evolving regulatory landscapes, and increasing customer demands. System development has accelerated dramatically since the advent of the Fourth Industrial Revolution, or Industry 4.0. Industry 4.0 is characterized by the emergence of Cyber-Physical Systems (CPS), which seamlessly combine the physical and virtual worlds through advanced technologies such as the Internet of Things (IoT), artificial intelligence (AI), and big data analytics [17]. Following this, Human-Cyber-Physical Systems (HCPS) facilitate collaboration, where human operators assist in oversight and control of automated systems [18]. As we transition from Industry 4.0 to Industry 5.0, the focus shifts towards a more human-centric approach. Integrating advanced technologies aims to enhance the quality of life, social responsibility, and sustainability [19]. Industry 5.0 builds upon the tech-

nological foundations established in Industry 4.0, emphasizing the importance of human input and environmental considerations in industrial processes. This increasing pace of technological change creates a need for adaptable development methodologies such as Agile.

1.1 Problem Statement

Sectors that LS/SC/CP systems face increased pressure to accelerate time to market, reduce costs, and rapidly adapt to changing needs. In parallel, these systems are growing in complexity due to the number of connected elements and teams [20]. The complexity of building modern automobiles has grown 300% in the last 10 years [21]. These challenges have resulted in many companies exploring moving from traditional Waterfall development to Agile methods to overcome challenges in product development. The Waterfall model, first documented by Benington in 1956 and later modified by Winston Royce in 1970 to include feedback loops, following a linear sequential approach with structured phases such as requirements, design, implementation, verification, and maintenance [22] [23]. Agile relies on iterative and incremental development, which is valued for flexibility and rapid delivery cycles. Barry Boehm's lifecycle cost theory states that the cost of change increases throughout the development lifecycle. Waterfall eliminates change early in the system development lifecycle to reduce costs. Agile assumes change is inevitable and focuses on minimizing the impact and cost of change throughout the lifecycle [24]. Given the industry challenge of needing to reduce time to market, reduce cost, and adapt to changing needs, it is reasonable that the industry should adopt Agile to improve system development. However, there are challenges associated with scaling, safety, and physicality.

Scaling Agile has increased complexity due to the number of components and interdisciplinary teams. Research indicates that Agile practices are inherently more suited for small teams where communication and coordination are more manageable [25]. Agile scaling frameworks, such as Scaled Agile Framework (SAFe) [7], Large Scale Scrum (LeSS) [4]. Disciplined Agile (DA) [26] offers strategies to coordinate and collaborate across large organizations. Still, they are focused on

software development and lack guidance on managing the challenges of LS/SC/CP systems, which include regulatory compliance, safety assurance, and hardware-software integration [27].

Agile applied to safety-critical systems must effectively balance speed with compliance and documentation requirements specified in safety standards [28] [29]. Frameworks like R-Scrum [30] and Safe-scrum [27] have provided adaptations designed specifically for safety-critical projects, such as additional artifacts and roles to support the safety-critical demands [30] [31]. However, they do not address scaling or how to manage physical systems.

Agile applied to physical systems presents distinct constraints due to the physical nature of products, longer lead times for prototyping, and the inherent complexities of hardware-software interactions [32]. Modified Agile for hardware development (MAHD) was developed to support portfolios that combine electronics, mechanical components, and software elements [33]. MAHD follows foundational principles such as short development cycles that are continuously validated and accountable, as well as autonomous teams. However, they do not provide any guidance on how to scale or meet regulatory and safety needs.

While research has addressed scaling, safety, and physical constraints, very little has addressed all three separately.

1.2 Research objectives and Questions

We aim to resolve the challenges of adapting Agile methodologies to the specific demands of LS/SC/CP system development by focusing on how Agile principles can be effectively adapted to support coordination and communication in large organizations, ensure regulatory compliance and safety is built-in, and overcome long lead times in and integration complexity of physical systems. This research comprehensively analyzes Agile methodologies within LS/SC/CP systems, utilizing a systematic literature review, scaling framework analysis, surveys, interviews, and modeling. The results provide actionable insights and practical guidance for the safe and efficient implementation of Agile methodologies in these critical systems, contributing to academic literature and industry applications.

RQ1: What are Agile Scaling Frameworks, and how do they compare? What is their suitability in delivering large-scale, safety-critical, cyber-physical systems?

RQ2: To what extent is Agile applied to large-scale, safety-critical cyber-physical systems? What challenges exist? What proposed adaptations have been used?

RQ3: What is the impact of applying Agile to large-scale, safety-critical cyber-physical systems? Do adaptations resolve challenges?

1.3 Structure of Dissertation

This section outlines the paper's structure to guide the reader through this research. It details the focus of each subsequent chapter and demonstrates how they collectively address the research objectives.

Chapter 1 Introduction

This chapter establishes the context for applying Agile methodologies to LS/SC/CP systems by exploring the challenges and opportunities of this application. It emphasizes the need to carefully consider the unique characteristics of LS/SC/CP systems, which is vital to the success of agile implementation. Finally, the chapter provides a road map of the paper's structure, outlining the content and focus of each subsequent chapter.

Chapter 2 Background

This chapter describes the background, provides an overview of Agile methodologies, and outlines the differences between Agile and Waterfall models. It discusses scaling Agile and addresses the challenges associated with large-scale, safety-critical, cyber-physical systems. Finally, it introduces supporting technologies such as MBSE (Model-Based Systems Engineering), Digital Engineering, Additive Manufacturing, and AI. These discussions provide the necessary context for understanding the application of Agile methodologies within complex LS/SC/CP environments."

Chapter 3

Research and Tasks This chapter describes the research methodology employed to investigate the application of Agile methodologies to Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems. It provides a detailed explanation of the chosen approach, including a mixed-methods design incorporating qualitative and quantitative data collection and analysis techniques. Additionally, the chapter outlines the associated research tasks, such as a systematic literature review, scaling framework analysis, surveys, interviews, and modeling. These tasks are essential for gathering and analyzing data to address the research questions and achieve the study's objectives.

Chapter 4 Literature Review

This chapter presents a systematic literature review examining existing research on applying Agile methodologies to LS/SC/CP systems. This review is crucial for establishing the foundation of this research by understanding the current state of knowledge and identifying gaps and opportunities for further investigation. Adopting Kitchenham's approach ensures a rigorous and reproducible review process. This involves formulating research questions, developing a search strategy for relevant studies, and determining data extraction and synthesis methods. The literature review provides valuable insights into the current state of Agile application within these contexts, including the challenges, benefits, and best practices associated with its adoption in LS/SC/CP systems.

Chapter 5 Investigation of Scaling Frameworks

This chapter addresses **Research Question 1** by investigating existing Agile Scaling Frameworks to analyze their suitability for LS/SC/CP systems. This analysis is crucial for understanding how Agile can be effectively scaled to meet the unique demands of large-scale, safety-critical, and cyber-physical development. Utilizing Digital.ai's 17th Annual State of Agile Survey and an author-developed internal survey, this chapter identifies the 10 most utilized frameworks based on respondent data. Subsequently, it assesses the degree of variation among these frameworks at the principles and practices level. This assessment aims to determine the framework most suitable for

LS/SC/CP development and to identify any gaps requiring attention. The findings provide valuable insights for selecting and adapting Agile Scaling Frameworks in LS/SC/CP contexts.

Chapter 6 To what extent is Agile applied to LS/SC/CP systems

This chapter addresses **Research Question 2** through two key components of the research methodology: online surveys and semi-structured interviews. It commences by detailing the design and implementation of the online surveys, which aim to gather quantitative data from a diverse range of industry professionals engaged in LS/SC/CP development. These quantitative data provide a comprehensive overview of current practices and challenges within the field. Subsequently, the chapter elaborates on the semi-structured interviews conducted with a select subset of survey respondents. The chapter concludes by discussing integrating quantitative and qualitative data to understand the research questions comprehensively. This mixed-methods approach yields insights into the factors influencing Agile adoption and adaptation within LS/SC/CP contexts.

Chapter 7 The impact of Agile when applied to LS/SC/CP systems

This chapter addresses **Research Question 3** by comparing traditional Waterfall and Agile methodologies in the context of LS/SC/CP development. This comparison is crucial for understanding each approach's relative strengths and weaknesses and identifying the potential benefits of Agile adoption. To facilitate this comparison, the chapter presents a detailed model, developed using Innoslate, that sets up a fictional case study simulating the development of a large-scale, safety-critical, cyber-physical system (LS/SC/CP), specifically a satellite. The simulation compares the Waterfall methodology with a proposed Agile approach through a series of Minimum Viable Products (MVPs). By providing a detailed account of the fictional experiment setup, modeling process, simulation, and data analysis conducted using Innoslate, the comparison aims to offer valuable insights into the impact of each methodology on LS/SC/CP development in terms of cost and schedule within the context of the case study. These findings inform decision-making regarding LS/SC/CP development methodologies.

Chapter 8 Conclusion, contribution, limitations, Future Work

This chapter consolidates the research objectives and key findings from the preceding chapters, emphasizing the study's contributions to Agile methodology implementation within LS/SC/CP systems. It revisits the problem statement, highlighting the challenges of LS/SC/CP system development and the current state of Agile application. The chapter demonstrates how the systematic literature review, surveys, interviews, and Innoslate model simulation collectively addressed the research questions. The research's implications, including a deeper understanding of Agile adoption challenges and the development of tailored frameworks, are discussed, showcasing the study's theoretical and practical relevance.

Additionally, the chapter acknowledges the research's limitations, such as the scope of the literature review, potential response bias in survey and interview data, and the fictional nature of the Innoslate model simulation. Finally, it outlines the research's contributions to the body of knowledge and offers recommendations for future work, including exploring emerging Agile techniques and conducting longitudinal studies.

Chapter 2

Background

The background chapter provides a comprehensive foundation for understanding the application of Agile methodologies to LS/SC/CP systems. We establish the context for the research by exploring the fundamental principles of Agile. We discuss the unique characteristics of LS/SC/CP systems, setting the stage for a deeper dive into the challenges and opportunities associated with their integration. Lastly, we discuss other relevant technologies to enable Agile in these environments.

This chapter covers several key themes essential to the research:

- Agile Methodologies: It provides an in-depth overview of Agile principles, values, and practices, including popular frameworks like Scrum and Kanban. This section discusses how Agile promotes flexibility, collaboration, and iterative development, contrasting it with traditional Waterfall approaches.
- LS/SC/CP: It defines and explores the distinctive characteristics of large-scale, safety-critical, and cyber-physical systems.
- Engineering Methodologies: The chapter examines relevant engineering methodologies commonly employed in LS/SC/CP development, such as systems engineering and model-based systems engineering (MBSE).

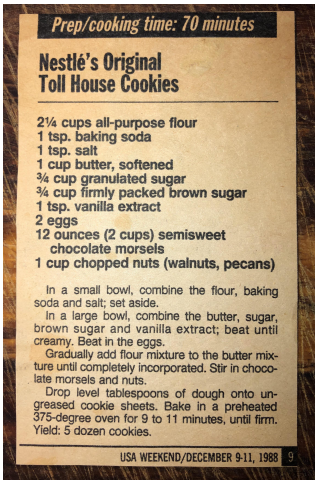
By providing a thorough background on these key themes, the chapter equips the reader with the necessary knowledge to understand the research problem and appreciate the complexities of applying Agile in safety-critical and complex systems development.

2.1 Agile Overview

Agile methodologies, which have evolved significantly since their mainstream adoption in the software industry, can trace their roots back to early implementations across various engineering

fields. Notably, Takeuchi's 1986 paper, "The new new product development game" [34], which introduced one of the most popular Agile methodologies, Scrum, that later became associated with software development.

The Agile Manifesto, introduced in 2001 by a consortium of seventeen software practitioners, emphasizes four fundamental values and twelve guiding principles prioritizing adaptability, collaboration, and customer satisfaction within software development processes. At its core, the manifesto advocates for individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over adherence to a fixed plan [35]. One of the most frequently seen practitioner mistakes is missing the fact stated under the manifesto, "While there is value in the items on the right, we value the items on the left more" [35]. This misunderstanding can lead to the erroneous belief that processes and documentation are entirely dispensable. Such interpretations are detrimental, undermining the practices' strategic role in supporting agile methodologies. Failing to recognize the significance of structured processes may adversely affect a team's competitive edge [36]. Agile methodology focuses on rightsizing documentation, for example, MIL-C-43205G [37] is a documented standard to make chocolate chip cookies, which is 21 pages, and the Agile approach would be similar to how Toll House documented their chocolate chip cookie recipe, shown in Figure 2.1.



Prep/cooking time: 70 minutes

Nestlé's Original Toll House Cookies

2¼ cups all-purpose flour
1 tsp. baking soda
1 tsp. salt
1 cup butter, softened
¾ cup granulated sugar
¾ cup firmly packed brown sugar
1 tsp. vanilla extract
2 eggs
12 ounces (2 cups) semisweet chocolate morsels
1 cup chopped nuts (walnuts, pecans)

In a small bowl, combine the flour, baking soda and salt; set aside.
In a large bowl, combine the butter, sugar, brown sugar and vanilla extract; beat until creamy. Beat in the eggs.
Gradually add flour mixture to the butter mixture until completely incorporated. Stir in chocolate morsels and nuts.
Drop level tablespoons of dough onto ungreased cookie sheets. Bake in a preheated 375-degree oven for 9 to 11 minutes, until firm. Yield: 5 dozen cookies.

USA WEEKEND/DECEMBER 9-11, 1988 9

Figure 2.1: Toll House Cookie Recipes

2.1.1 Differences between Agile and Waterfall

The fundamental differences between Waterfall and Agile, as illustrated in Figure 2.2, revolve around how work is structured and executed during the project lifecycle. The Waterfall model employs a linear, sequential approach where the project is broken down into distinct phases such as requirements gathering, design, implementation, testing, and maintenance. Each of the phases must be completed before the next begins. This method relies heavily on extensive, detailed, upfront planning and documentation, making it less adaptable to changes once the project is underway [38]. Waterfall relies on humans designing and building the system right the first time. Due to the linear approach, Waterfall receives limited feedback until the test phase. This reliance on upfront planning and structured phases requires high confidence in the initial specifications and designs, as any errors may lead to significant downstream issues and necessitate extensive rework if changes are needed [39]. In reality, Winston Royce described this shortcoming in his original paper in 1970 cite royce1970. Royce suggested a more iterative approach to product development, where feedback loops are integrated into the process, allowing teams to adapt and refine their work based on validation at different stages, very similar to what is referred to as Agile today.

Agile decomposes work into small testable batches with rapid feedback, cultivating a culture of transparency and continuous improvement, significantly reducing the risk of misalignment between deliverables and user needs cite Popoola2024. Waterfall is best for simple projects with clear requirements and stable environments, such as developing pencils. Agile is well suited for complex projects with evolving requirements in uncertain environments.

Figure 2.2: The difference between Agile and Waterfall

2.1.2 Agile Foundational Building Blocks

We introduce the foundational Agile methods, the building blocks for many scaling frameworks. Lean, Scrum, Kanban, and extreme programming (XP) have revolutionized development by promoting flexibility, iterative progress, and close customer collaboration. Each method offers a unique approach to Agile development: Lean focuses on waste reduction, Scrum provides a structured framework, Kanban visualizes workflow, and XP emphasizes technical practices. Although these frameworks originated within small, co-located teams and have successfully enhanced productivity and product quality, their application in larger, more complex environments such as LS/SC/CP systems is promising.

Lean: Shown in Figure 2.3, originated from Lean Manufacturing. Lean principles focus on waste reduction, maximizing customer value, and optimizing processes [2]. Mary and Tom Poppendiek adapted lean principles for software development in their Lean software development book [40]. The Lean methodology eliminates non-value-adding activities, enhancing efficiency and delivery speed. Lean methods stress the importance of maximizing customer value while minimizing waste, thus aligning closely with Agile principles. By focusing on delivering value incrementally, teams can prioritize feature development that resonates most with users.

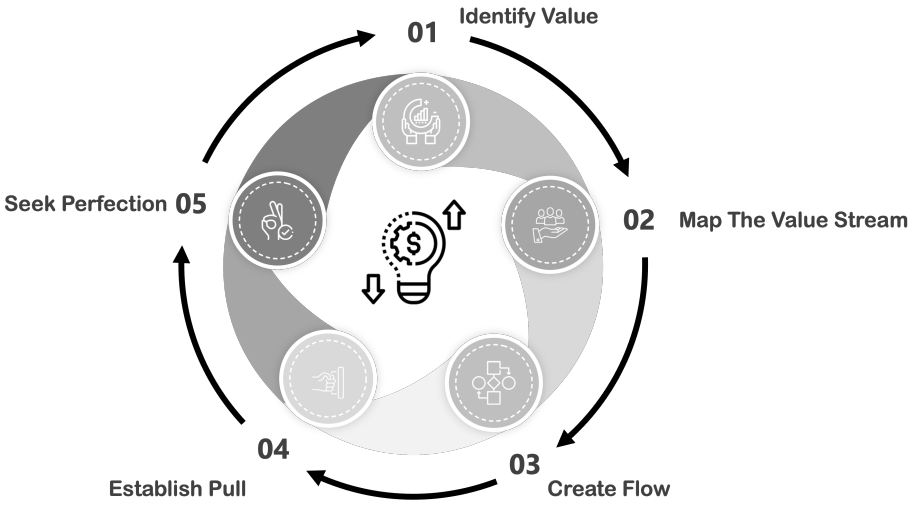


Figure 2.3: Lean Principles

Scrum: is an iterative and incremental framework, illustrated in Figure 2.4, for managing product development. As described earlier, Hirotaka Takeuchi introduced Scrum in 1986 cite takeuchi1986. Takeuchi challenged the sequential approach to product development. Like Lean, Scrum was also adapted for software by Ken Schwaber and Jeff Sutherland. Scrum is structured around small, self-organizing teams that work in time-boxed iterations called sprints, typically lasting two to four weeks. Each sprint begins with a Sprint Planning meeting, where the team selects a set of high-priority tasks from the Product Backlog, a prioritized list of work items maintained by the Product Owner. Daily Scrum meetings (stand-ups) allow the team to discuss progress, identify obstacles, and adjust plans as needed. At the end of each sprint, the team conducts a Sprint Review to showcase completed work and a Sprint Retrospective to reflect on process improvements. The Scrum Master facilitates the process, ensuring adherence to Scrum principles while removing impediments that hinder progress. By emphasizing continuous feedback, adaptability, and team collaboration, Scrum enables organizations to deliver high-quality products efficiently while responding to changing requirements.

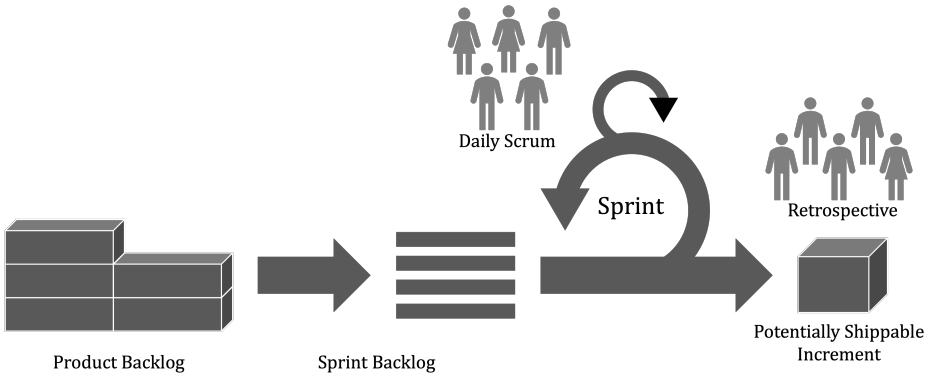


Figure 2.4: Scrum Method

Kanban: Emerged from Lean principles, Kanban emphasizes visualizing work to enhance workflow. Initially developed in the manufacturing sector, it focuses on visualizing the flow of work and managing the work-in-progress (WIP). Like Lean and Scrum, Kanban was adapted to support software development by David Anderson cite anderson2010. Central to Kanban are vi-

sual boards that display tasks at various workflow stages, as illustrated in 2.5. These boards provide teams with real-time insights into progress and bottlenecks. This visual management allows teams to identify and address issues promptly, thus promoting efficiency and continuous delivery. Unlike Scrum, which operates on fixed iterations, Kanban provides greater flexibility in task management, enabling teams to pull new tasks into the workflow as capacity allows [41].

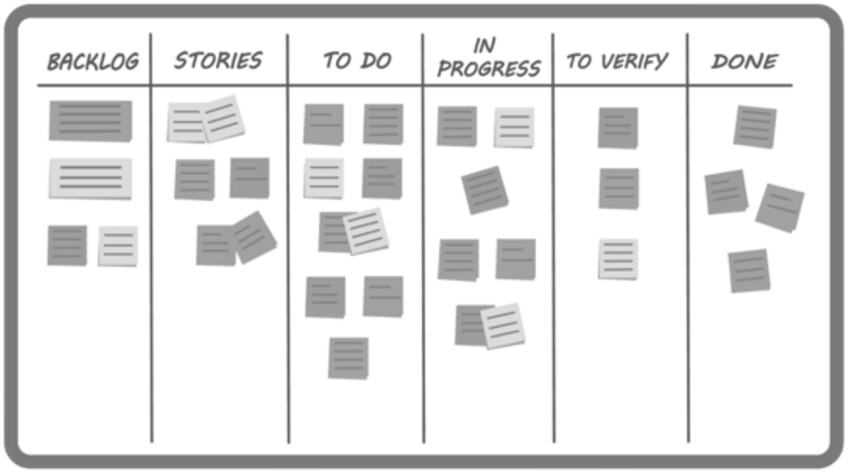


Figure 2.5: Kanban for visualizing work

Extreme Programming (XP): was defined as a software development methodology to improve software quality and responsiveness to changing customer requirements. Introduced by Kent Beck in the late 1990s [42]. Interestingly, XP is the only Agile method that originated with software. XP embodies a set of technical practices illustrated in Figure 2.6, prioritizing technical excellence and customer satisfaction by emphasizing collaboration and the continuous delivery of high-quality software. A fundamental principle of XP is active customer involvement throughout the software development process. Unlike traditional methodologies that treat requirements as fixed at the outset, XP advocates for regular feedback from the customer [43]. This collaborative approach ensures that software development aligns closely with user needs and expectations, enabling teams to adapt quickly to changes and deliver maximum value.

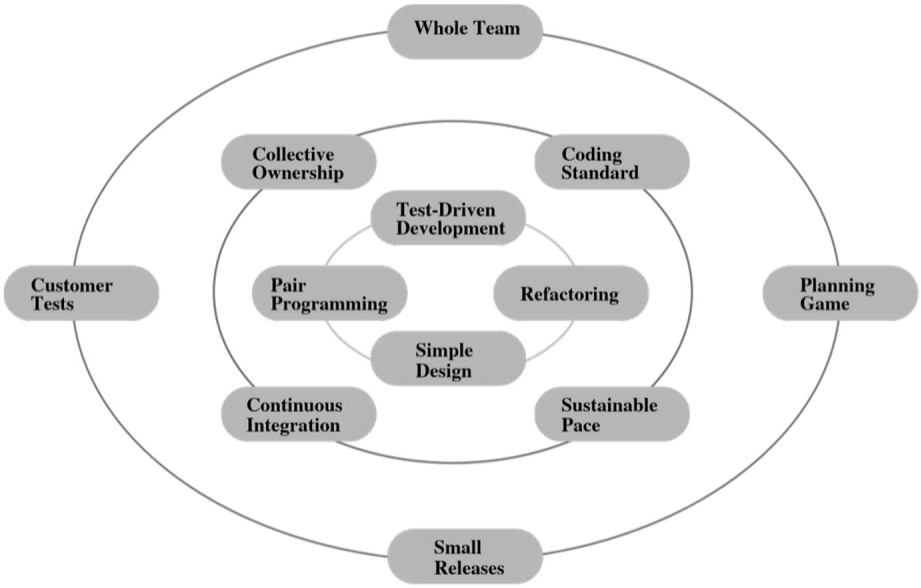


Figure 2.6: Key principles of Extreme Programming (XP)

2.1.3 Scaling Agile

Scaling Agile extends Agile principles, practices, and frameworks beyond a single team to large organizations. While Agile is highly effective at the team level, applying it at scale introduces challenges such as coordination across multiple teams, cross-team communication, dependency management, and alignment with business objectives cite dikert2016. There are several scaling frameworks, such as SAFe (Scaled Agile Framework), LeSS (Large-Scale Scrum), and Disciplined Agile (DA), which provide structured approaches to help organizations scale Agile while maintaining flexibility and adaptability. These frameworks emphasize cross-team collaboration, iterative development, and continuous integration to accommodate enterprise-level software development [44]. Empirical studies describe common challenges such as resistance to change, coordination across multiple teams, communication, and difficulties with stakeholder engagement [45]. Due to the fundamental importance of scaling to this research, a detailed analysis of the top 10 scaling frameworks is presented in Chapter 4.

2.2 Large-Scale, Safety-Critical, Cyber-Physical Systems

A primary characteristic of LS/SC/CP systems is their inherent technological complexity. These systems often comprise intricate networks of computational and physical components, such as distributed sensors and real-time processing units, which interact dynamically to perform tasks [46]. Consequently, these systems require rigorous verification and validation processes to operate safely and effectively. There are several challenges in building these systems, as illustrated in Figure 2.7

Figure 2.7: Challenges for LS/SC/CP Systems

2.2.1 Long lead times

Typically, LS/SC/CP Systems do not get feedback at the same rate as software. Ovensen referred to this as "constraints of Physicality." Which can be summarized as challenges that occur due to being physical [47]. Generally, building a hardware prototype consumes more time than writing and compiling software. In addition, most CPS have components and sub-assemblies from multiple suppliers, causing additional delays [48].

2.2.2 Expensive Test Equipment

LS/SC/CP systems require unique and expensive test equipment. Unlike purely digital systems, these integrated systems must undergo rigorous validation to ensure reliability, safety, and compliance with industry standards. LS/SC/CP systems necessitate specialized hardware-in-the-loop (HIL) simulators, real-time emulation platforms, and environmental testing chambers that replicate operational conditions such as extreme temperatures, vibrations, radiation, and electromagnetic in-

terference. Additionally, high-fidelity sensors and actuators must be tested with precision calibration equipment to verify real-world accuracy. Aerospace, automotive, and industrial automation CPS often require large-scale testbeds, wind tunnels, or anechoic chambers, for example, thermal vacuum chambers needed to test satellites cost millions of dollars, and the test can take months.

2.2.3 Multiple Dependencies

These systems often consist of interconnected subsystems, including hardware components, embedded software, communication networks, and control systems. Multiple dependencies arise between different engineering disciplines, such as mechanical, electrical, and software engineering, requiring cross-domain collaboration and coordinated development efforts. One of the primary challenges associated with these systems is the risk of cascading failures, where an initial, seemingly minor disturbance can precipitate widespread outages across interconnected networks. Islam et al. highlights that cyber-physical inter-dependencies, such as those present in smart grid technologies, heighten the vulnerability of systems to such cascading failures [49]. This interdependence merges the cyber network with physical infrastructures, complicating the modeling, control, and monitoring of system resilience.

2.2.4 Complex Risk Management

The aforementioned multiple dependencies lead to a significant challenge: complex risk management. Systems within domains such as aerospace, defense, healthcare, and industrial automation encounter unique risks encompassing safety, security, reliability, and compliance [50]. Risk management processes for these systems must address technical, schedule, cost, and security risks to ensure successful delivery. It involves continuous stakeholder engagement, resource management, and adherence to compliance standards.

2.2.5 Large Attack Surface

The interconnectedness of LS/SC/CP systems results in an extensive attack surface. For example, satellites contain command and control links, data transmission networks, onboard software,

and supply chain components, all of which present avenues for exploitation by adversaries with malicious intent. Threat vectors targeting these systems vary widely and encompass techniques such as radio frequency (RF) jamming, Global Positioning System (GPS) spoofing, and malware injections. Continuous risk assessments, updated threat models, and the implementation of standardized reporting mechanisms can collectively enhance the resilience of LS/SC/CP against both cyber and physical threats [51].

2.2.6 Safety Requirements

Safety-critical systems are those where failure can lead to severe consequences [13]. Aircraft flight control systems must ensure stable and safe flight; failures can result in crashes. International aviation authorities govern the operational safety of these systems. Compliance with industry standards such as DO-178C (software), DO-254 (hardware), ARP4754A (system development), and ISO 26262 (functional safety for avionics systems) is critical cite Gao2024. The proactive identification and mitigation of hazards in these systems is thus essential. Hazard analysis techniques, such as failure modes and analysis and System-Theoretic Process Analysis (STPA), allow for a thorough examination of hazard scenarios and the control actions that can lead to unsafe states [52] [53].

2.2.7 Regulatory Constraints

The number of regulatory constraints that govern the deployment of LS/SC/CP systems is daunting. Regulatory requirements go beyond safety, adhering to additional laws such as sustainability, export control, and other legal approvals. General Data Protection Regulation (GDPR) is one example of a non-safety-related regulatory compliance requirement. The GDPR imposes specific obligations on entities that process personal data, including obtaining explicit consent from individuals before collecting data, ensuring data minimization, and implementing adequate security measures to protect data against [54].

2.3 Adjacent Engineering technologies

Several adjacent engineering technologies illustrated in Table 2.1 can help overcome LS/SC/CP system challenges, enabling iterative development, rapid prototyping, and continuous integration and validation. Key technologies we explore to support Agile in these environments include Model-Based Systems Engineering (MBSE), Digital Twins (DTw), Digital Engineering (DE), Additive Manufacturing, and Artificial Intelligence(AI).

Table 2.1: Adjacent technologies enable Agile methods

	Technology	**Enable Agile**
1.	Model-Based Systems Engineering (MBSE)	Enables iterative system modeling and traceability
2.	Digital Twins	Enables real-time testing and feedback loops for Agile development
3.	Digital Engineering (DE)	Provides a fully integrated digital development environment
4.	Additive Manufacturing (3D Printing)	Accelerates hardware iteration cycles with rapid prototyping
5.	Artificial Intelligence (AI)	Automates testing, documentation, optimizes designs, and enhances decision-making

MBSE

A Model-Based Systems Engineering (MBSE) methodology can be characterized as the collection of related processes, methods, tools, and environment as used to support the discipline of systems engineering in a "model-based" context [55]. MBSE leverages a combination of architectural views illustrated in Figure 2.8 to engage in critical activities, including requirements management, lifecycle modeling, and system verification and validation, which are foundational

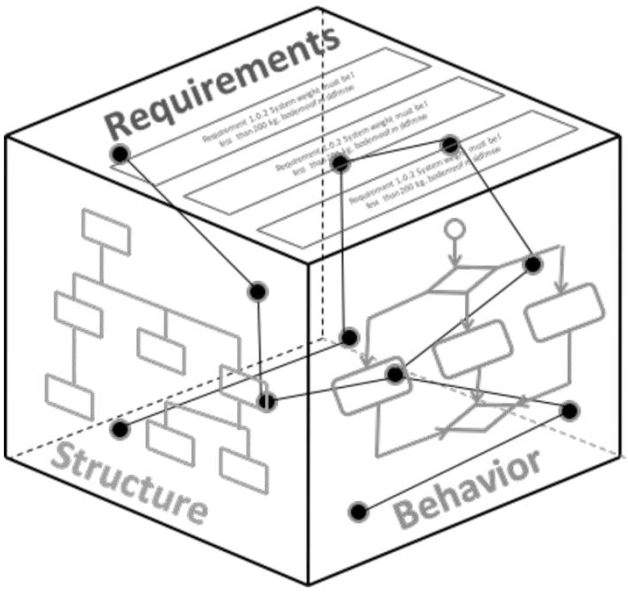


Figure 2.8: Model Based Systems Engineering Views

for understanding complex systems [56]. Models provide the ability to communicate systems before physical implementation [57]. MBSE has four pillars: requirement, structure, behavior, and parametric interrelationships that aid programs in conforming to customer policies while delivering effective, affordable systems and enterprises to their final users [58]. When combined with Agile methodologies, MBSE enhances system engineering efficiency by ensuring transparency and traceability, accelerating the delivery of engineering artifacts [59]. Dr. Bruce Douglass Powel has written several books on Agile, and he has stated that the only difference between Agile MBSE is that the outcome is system specifications instead of software cite douglass2022. While he did a great job introducing the integration between Agile and MBSE, Dr. Powell did not realize that Agile's outcome is that working capabilities and specifications are only one capability element. His implementation has iterative and incremental specifications, but could be implemented further to include MVPs.

Digital Twin

A digital twin is a digital representation of a physical object, person, or process contextualized within a digital version of its environment [60]. Michael Grieves at the University of Michigan

first wrote of the concept using the digital twin terminology in 2002 [61]. This technology leverages real-time data, simulation, machine learning, and reasoning to create living digital simulation models that update and change as their physical counterparts change. Digital twins serve as a mirror to the real world, as illustrated in Figure 2.9. Digital Twins provide a dynamic and evolving model that can be used for various purposes, including monitoring, diagnostics, optimization, and simulation [62]. The key benefit they bring to Agile methods is the ability to get real-time feedback.

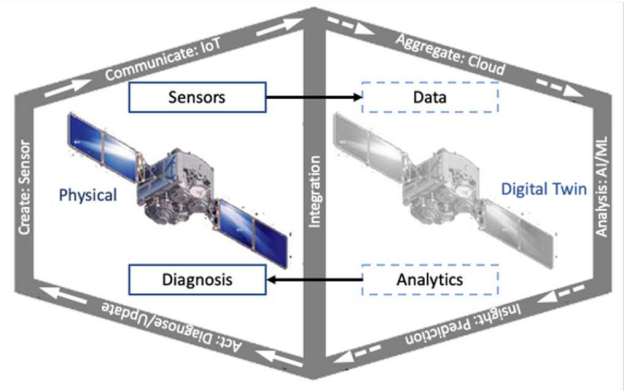


Figure 2.9: Digital Twin

Digital Engineering (DE)

The Department of Defense defines Digital Engineering as "an integrated digital approach that uses authoritative sources of systems' data and models as a continuum across disciplines to support lifecycle activities from concept through disposal" [1]. The approach, as illustrated below in Figure 2.10, incorporates existing model-based tenets (e.g., model-based engineering (MBE), model-based systems engineering (MBSE), digital thread (DT), and digital twin (DTw)).DE fosters better decision-making and enhances the capability to innovate across various safety-critical sectors such as aerospace, defense, automotive, healthcare, and energy [63].

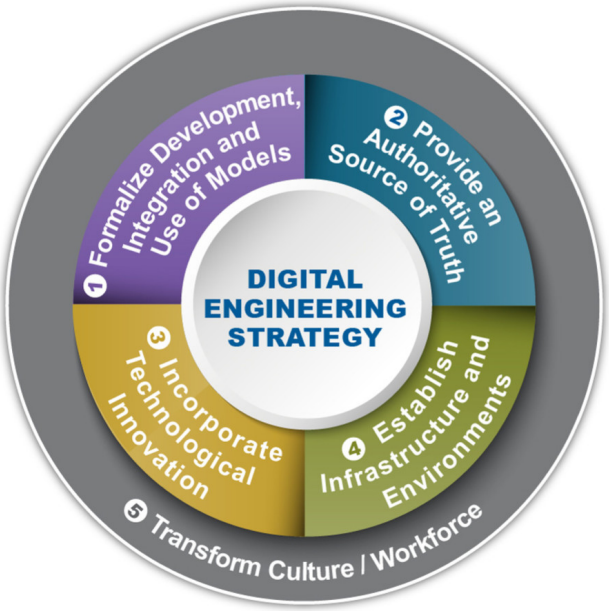


Figure 2.10: Digital Engineering [1]

Additive Manufacturing

Three-dimensional (3D) printing, also known as additive manufacturing, represents a revolutionary approach in the field of manufacturing that fundamentally contrasts with traditional subtractive methods. In additive manufacturing, objects are created layer by layer from digital files, allowing for the fabrication of complex geometries and intricate designs often unachievable through conventional manufacturing techniques [64] [65]. With the rapid turnaround of physical products, we can further reduce the impacts of the supply chain and shorten feedback loops. Reich describes how the early production of prototypes with additive manufacturing reduced the need to collect all requirements at the beginning of product development [66]. His approach is illustrated in Figure 2.11.

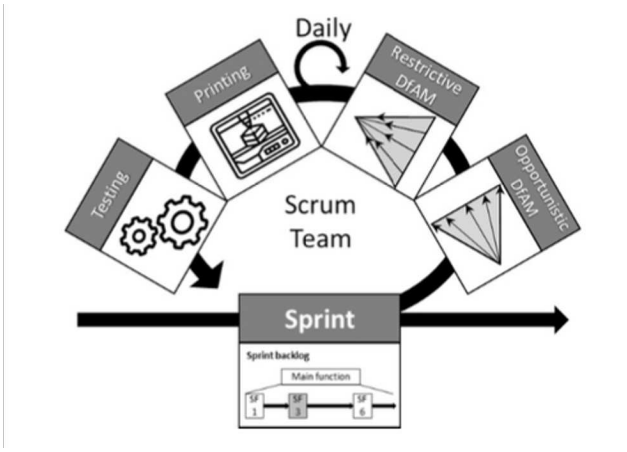


Figure 2.11: Additive Manufacturing with Agile

AI

Artificial Intelligence (AI) is the simulation of human intelligence in machines, enabling them to learn, reason, and make decisions autonomously [67]. This technology has exploded over the last couple of years. The implications of using AI for building LS/SC/CP systems include test automation, document generation, improved decision-making, and more. Historically, engineering design relied heavily on foundational knowledge with time-consuming and resource-intensive methods. AI's applicability in engineering design is driven by its ability to automate complex analyses, identify patterns, and enhance decision-making. It is a powerful tool for tasks where precision, optimization, and adaptability are crucial [68].

Chapter 3

Research and Tasks

3.1 Overview

This dissertation explores the application of Agile methodologies to LS/SC/CP systems. The goal is to evaluate the degree to which Agile principles and practices could be implemented within these systems. To identify the challenges and potential adaptations required to facilitate successful implementation. Furthermore, this research aims to understand the impact of comparing Waterfall and Agile methodologies in the development of LS/SC/CP systems and to determine whether adjacent technologies, such as Model-Based Systems Engineering, Digital Twins for real-time monitoring, Digital Engineering, Additive Manufacturing, or AI assistance for engineering can mitigate these challenges. Ultimately, this research seeks to contribute to the systems engineering body of knowledge and provide actionable insights for organizations to use Agile methodologies at the system level.

3.2 Research Focus

This dissertation focuses on:

- Evaluating existing Agile scaling frameworks and their suitability for LS/SC/CP systems.
- Identifying the current state and impact of Agile being applied to LS/SC/CP systems.
- Exploring the role of adjacent technologies in enhancing the efficacy of Agile when developing these systems.
- Assessing the impact of Agile methodologies on the speed and cost while ensuring the safety of LS/SC/CP delivery.

3.3 Research Questions and Task List

3.3.1 Research Question 1

The first research question is to understand the current Agile Scaling frameworks and how they support building LS/SC/CP Systems. The research question is ***"What are Agile Scaling Frameworks, and how do they compare? What is their suitability in delivering large-scale, safety-critical, cyber-physical systems?"*** A series of tasks to answer the question are outlined in 3.1.

Table 3.1: Research Question 1 Tasks

	Title	**Task Description**
1.	Literature Review	Gather data on how to scale Agile effectively.
2.	Bound the Problem	Analyze the top 10 most used frameworks.
3.	Categorize frameworks	Categorize the frameworks to compare
4.	Perform analysis	Analyze the frameworks, compare them against one another, and identify suitability for LS/SC/CP Systems.
5.	Publish Conference Paper	Publish the results of the analysis for conference
6.	Publish a Journal Paper	Publish an extended view of the results in the journal

3.3.2 Research Question 2

The second research question is designed to understand the current state of the practice. The research question is ***"To what extent is Agile being applied to large-scale, safety-critical cyber-***

physical systems, and what challenges exist? What proposed adaptations have been applied?"

A series of tasks to answer the question are outlined in 3.2.

Table 3.2: Research Question 2 Tasks

	Title	Task Description
1.	Design and distribute a survey	Gather real-time, first-hand data directly from participants, ensuring it is relevant, specific, and tailored to experience applying Agile to LS/SC/CP Systems.
2.	Design and Conduct Interview	Gather deeper insights by exploring the "why" behind the answers given in the survey, allowing us to uncover underlying motivations.
3.	Analysis of Survey and Interviews	Validate survey results by cross-checking responses with interviews, performing a triangulation to ensure the accuracy and reliability of their findings.
4.	Update research based on gaps	Based on the data, identify gaps and update research, including data regarding gaps found.
5.	Explore MBSE	Literature review and SME discussions to determine the level of impact.
6.	Explore Digital Twins	Literature review and SME discussions to determine the level of impact
7.	Explore Digital Engineering	Literature review and SME discussions to determine the level of impact.
8.	Explore Additive Manufacturing	Explore Additive Manufacturing
9.	Explore Artificial Intelligence assisted engineering.	Literature review and SME discussions will be used to determine the level of impact.
10.	Publish Paper	Combine survey results (quantitative data) and interviews (qualitative data) and research into a paper.

3.3.3 Research Question 3

The third research question focuses on understanding the impact on the development of LS/SC/CP Systems when applying Agile and the effect of adaptations that may overcome the challenges identified in earlier research. The research question is **"What is the impact of applying Agile to large-scale, safety-critical cyber-physical systems? Do adaptations resolve challenges?"** A series of tasks to answer the question are outlined in 3.3.

Table 3.3: Research Question 3 Tasks

	Title	Task Description
1.	Model Waterfall Development	Create a detailed model of the satellite development process in Innoslate using NASA's documented waterfall method phases A-D.
2.	Model Agile Development	Create a detailed model in Innoslate of the satellite development process using the Agile method, including a series of MVPs.
3.	Perform Monte-Carlo Analysis	Perform a Monte Carlo simulation to compare the timelines of the Waterfall and Agile approaches. Use input parameters such as time estimates and resources to evaluate delivery timelines and labor cost variations.
4.	Analyze results	Analyze the outputs of the two models and compare the impacts on key factors like development speed, adaptability, and cost.
5.	Explore adaptions	Identify adaptations that could overcome challenges identified in building LS/SC/CP Systems and put in the model
6.	Publish Results	Compile the findings into a research paper structured around the impacts of Agile in LS/SC/CP and the role of adjacent technologies in improving efficacy.

3.4 Summary

This chapter outlines the research questions and associated tasks that guide this dissertation. By systematically analyzing each research question, this study aims to investigate how Agile can

be applied to large-scale, safety-critical cyber-physical systems. Can the adaptations provided by adjacent technologies overcome the challenges and enable success? Each research question is detailed below, followed by its specific tasks.

Chapter 4

Literature Review

This chapter performs a systematic literature review to gather, analyze, and combine existing research on Agile's application to LS/SC/CP systems.

4.1 Literature Review Approach

We employed Kitchenham's approach to conduct a systematic literature review (SLR). This method, adapted from the medical field for software engineering research due to its emphasis on empirical evidence and replicability, provides a comprehensive, unbiased, and repeatable aggregation of research evidence [69]. Kitchenham's method is thorough and objective, ensuring credibility and contributing high-quality evidence synthesis to the body of knowledge. This SLR protocol includes the following steps: motivation and research questions, search strategy, inclusion and exclusion criteria, study quality assessment, and data extraction.

4.1.1 Motivation and Research Questions

Agile methodologies are increasingly being adopted in complex systems development. This study examines the application of Agile methodologies to programs characterized as 'Large-Scale,' 'Safety-Critical,' and 'Cyber-Physical Systems' (LS/SC/CP). This intersection, which presents unique challenges and opportunities, remains under-researched. The main objectives are to understand how various domains implement Agile methodologies, to identify the drivers for Agile adoption (e.g., faster time-to-market, increased flexibility), to understand existing challenges, and to identify solutions where Agile methodologies have been adapted to mitigate these challenges. The following research questions will guide this investigation, detailed in Table 4.1.

Table 4.1: Research Questions

	Research Questions
RQ1.	To what extent and in which domains are Agile being applied to LS/SC/CP Systems?
RQ2.	What are the drivers for applying Agile to LS/SC/CP systems?
RQ3.	What are the challenges in applying Agile to LS/SC/CP systems?
RQ4.	Are adaptations needed when applying Agile to LS/SC/CP Systems?

4.1.2 Search Method

The search string for this study was developed using the guidelines provided by Kitchenham et al. [69], which utilize 'population' and 'intervention' concepts. Population refers to the application area, specifically large-scale, safety-critical, and cyber-physical systems (LS/SC/CP). Intervention refers to the treatment being applied, which is Agile methodologies.

The search string was ("Agile" OR "Scrum") AND (("Large-Scale" AND "Safety-Critical") AND ("Cyber-Physical" OR "Mechatronic")).

4.1.3 Inclusion / Exclusion Criteria

Following the protocols described in Kitchenham et al. [69], we applied the inclusion and exclusion criteria, as illustrated in Table 4.2.

Table 4.2: Literature Selection Criteria

Criteria	Inclusion	Exclusion
Material Requirements	peer-reviewed articles, conference papers, and journals	Non-peer reviewed articles
Publication Date	Greater than or equal to 2017	Less than 2017
Study Focus	Agile applied to large-scale & safety-critical & cyber-physical systems	Not utilizing Agile for all three criteria
Language	English	Not English
Availability	Full text Available	Full text not available

4.1.4 Study Selection

The approach taken in this study was methodically designed to ensure a rigorous and transparent selection process for relevant literature. Following the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines, the study selection process was structured to include key identification, screening, eligibility, and inclusion phases. This process is comprehensively illustrated in Figure 4.1. By following the PRISMA framework, the approach maintained high levels of transparency and reproducibility, ensuring the reliability of the systematic review.

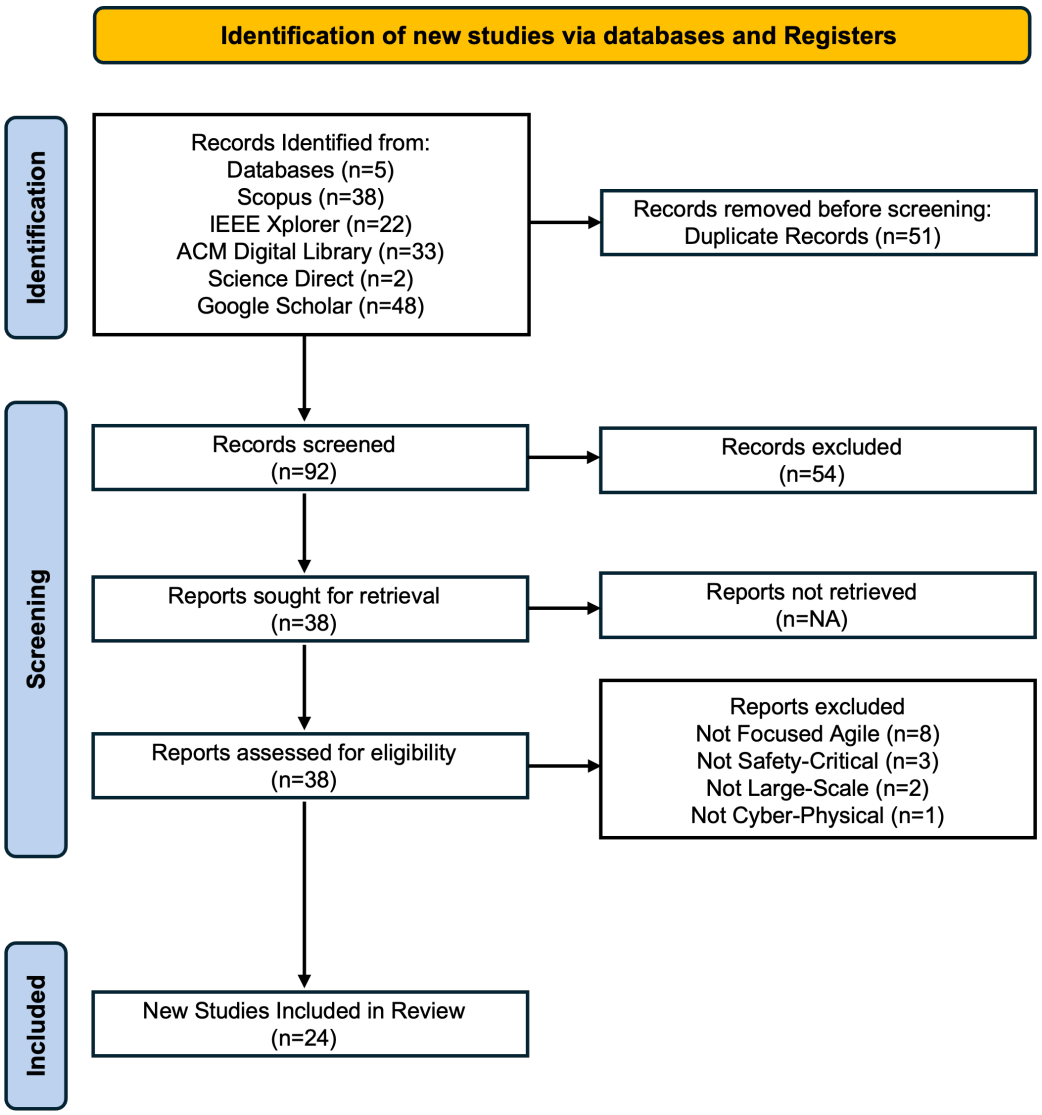


Figure 4.1: Prisma Flow Diagram

4.1.5 Data Extraction

In this step, researchers systematically extracted relevant data using a predefined form. The criteria used are shown in Table 4.3. The data was analyzed using open coding to develop an initial set of labels for each category. Relationships between the data were then developed through axial coding, from which key themes emerged, allowing us to perform selective coding and categorization. For example, in key drivers, labels such as schedule, speed, productivity, and prioritization were categorized as speed, resulting in shorter delivery cycles.

Table 4.3: Data Extraction

Title	Description
Study Information	Capture metadata regarding the article, such as title, authors, publication year, and source.
Methodology	The methodology used in the study, such as case study, survey, experimental, etc.
Scale	Description of scale at which Agile methodologies are applied
Safety-Critical	Identify and describe the safety-critical elements addressed in the study
Cyber-Physical	Detail the characteristics of the cyber-physical systems involved
Reason Agile applied	Description of the reason for selecting Agile
Agile Practices	Enumerate the specific Agile practices used.
Challenges	Description of challenges faced.
Benefits	Reported outcomes or benefits of using Agile
Adaptions	Reported Adaptations to Agile Methods.

4.2 Analysis and Results

The search process encompassed multiple digital libraries, with the resulting studies synthesized using Rayyan. The initial search yielded 147 studies; after duplicate removal and quality assessment, 24 studies remained. As illustrated in Figure 4.2, research on large-scale, safety-critical, cyber-physical systems peaked in 2018 and 2020, with a subsequent decline potentially influenced by the pandemic.

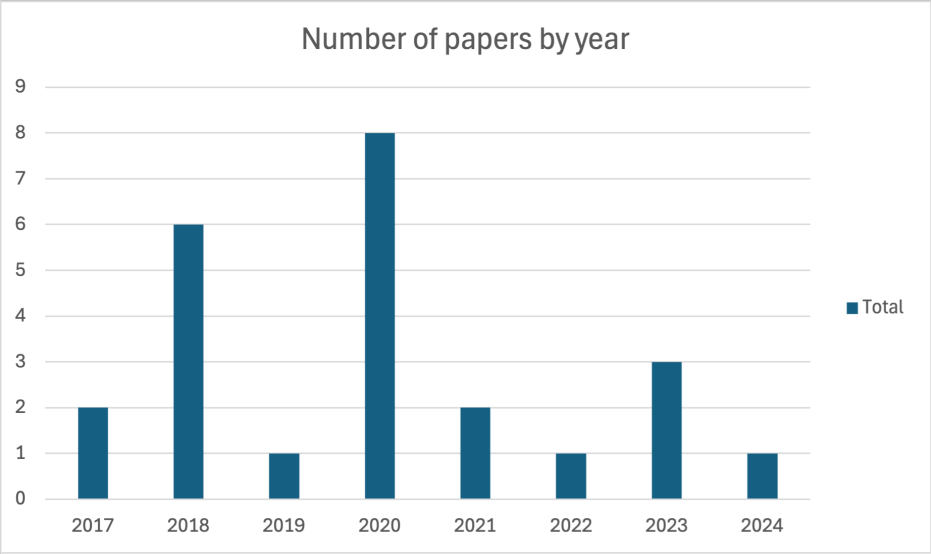


Figure 4.2: Number of papers per year

4.3 Discussion

RQ1: To what extent and in which domains is Agile being applied to LS/SC/CP Systems? The review reveals that the application of Agile has gained significant traction, with varying levels of adoption across a diverse range of domains. As illustrated in Figure 4.3, the literature review encompassed eight distinct domains, including medical, aerospace, robotics, civil, and energy. For example, the aerospace domain shows exceptionally high adoption, while the civil domain lags.

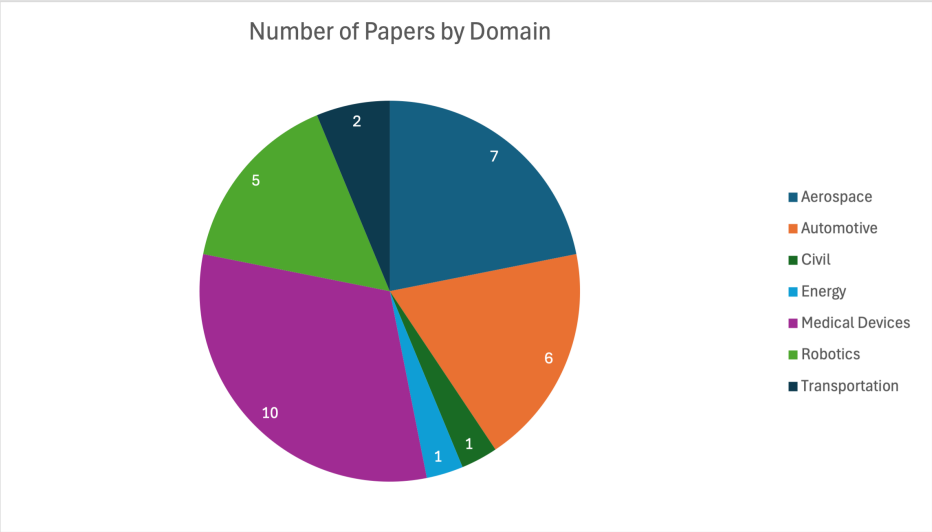


Figure 4.3: Papers Published by Domain

- **Medical** had the largest number of studies (10 of 24), with drivers for Agile adoption, including adaptability, speed, and cost. Strict regulations in the medical device industry require integrating Agile with disciplined practices to ensure both competitiveness and safety.
- **Aerospace** (7 of 24 studies) faced challenges with complexity and long development cycles. Study S2 augmented Agile with a Space Standardization compliance framework and technical readiness levels (TRLs) to ensure safe application in space vehicle development.
- **Automotive** (6 of 24 studies) drivers were similar to those of medical and aerospace, speed, adaptability, and decoupling of hardware/software architecture for agility.
- **Robotics** (5 of 24 studies) emphasized balancing speed with safety. Study S3 proposed augmenting Agile with Reliability, Availability, Maintainability, and Safety (RAMS) processes.
- Other domains like **civil engineering**, **transportation**, and **energy** had fewer studies, possibly due to their greater reliance on physical over digital components. However, their drivers for Agile adoption were consistent with the other domains.

RQ2: What are the drivers for applying Agile to LS/SC/CP Systems? The results showed that the drivers for applying Agile to LS/SC/CP Systems included speed, adaptability, managing

complexity, quality, and cost, as illustrated in Figure 4.4. For this paper, speed is reduced delivery cycles; adaptability equates to minimizing the impact of change; Quality refers to reducing failure or rework; Cost is lifecycle cost; and managing complexity reduces risk and uncertainty through decomposition. The primary drivers for adopting Agile methodologies, as depicted in the pie chart "Agile Drivers," are ranked from highest to lowest as follows: Speed (33%), Adaptability (27%), Quality (19%), Cost (12%), and Managing Complexity (9%).

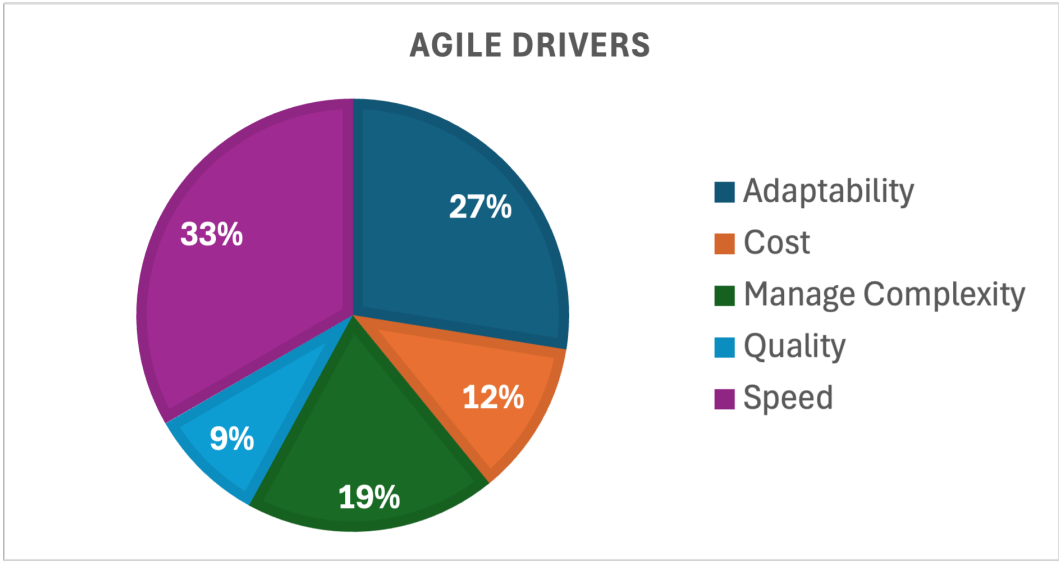


Figure 4.4: Drivers for moving to Agile

Speed is the most significant driver, as discussed in studies (S1, S2, S3, S4, S5, S6, S7, S8, S10, S12, S13, S14, S15, S17, S18, S20, S22, S23, S24). This indicates that organizations must prioritize speed to remain competitive in today's market. Adaptability is a close second, being discussed in studies (S1, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S15, S16, S17, S20, S21, S22, S23, S24), which demonstrates the level of change being experienced in these domains. Adaptability is especially valuable in industries where project requirements are not well-defined at the outset or are expected to evolve. Cost was also a significant driver for transitioning to Agile, as illustrated in studies (S3, S6, S7, S9, S11, S13, S17, and S22). This reflects the drive to reduce costs

and increase value for money across all domains. The remaining drivers, quality and managing complexity, were ranked lower but remain essential across all domains.

RQ3: What are the challenges in applying Agile to LS/SC/CP Systems? The results identified difficulties in the following categories: organizational/team dynamics, process/workflow, regulatory compliance, safety management, and System integration complexity, as shown in Figure 4.5. For this paper, Organization/Team dynamics refers to team structures and relationships; process/workflow, how teams operate; regulatory compliance relates to policies and approval processes; safety management encompasses practices to assure system safety; and system complexity pertains to issues with lead-times, validation, and verification approaches. The key challenges, ranked from highest to lowest in the pie chart "Key Challenges," are Regulatory/Compliance (30%), System Integration Complexity (22%), Organizational/Team Dynamics (19%), Safety Management (16%), and Process/Workflow (13%).

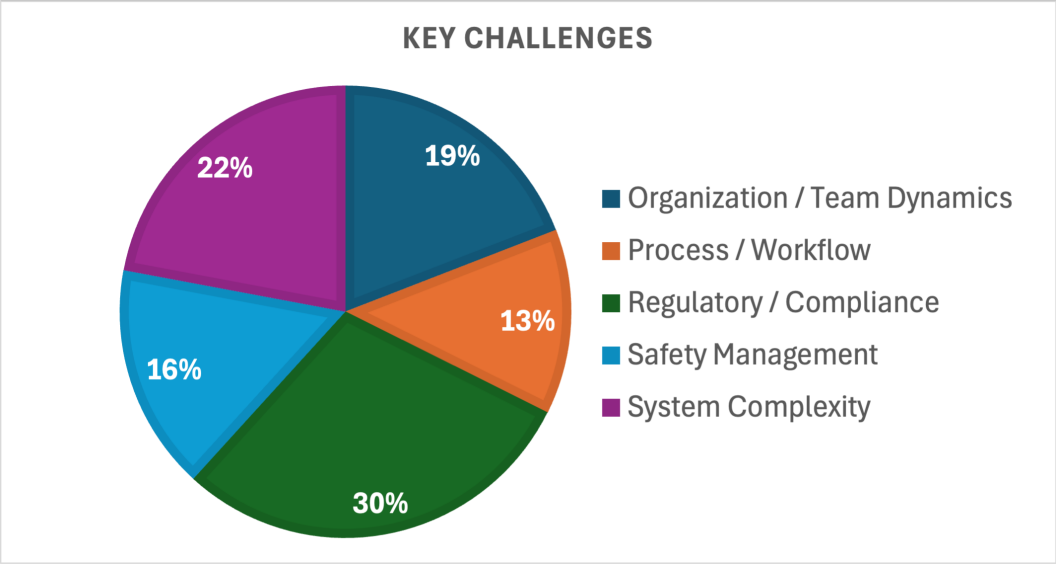


Figure 4.5: Key Challenges in applying Agile to LS/SC/CP Systems

The top challenge, regulatory/compliance, was discussed in studies (S1, S2, S3, S4, S5, S6, S7, S8, S10, S11, S13, S14, S15, S16, S17, S20, S23, and S24). This reflects confusion in demonstrating compliance, often due to validation tied to Waterfall phase gates. A 2017 study from Finland

examined conflicts between European space standards and Agile, finding safety standards were tied to Waterfall milestones [70]. This highlights the need to rethink compliance processes beyond Waterfall.

Studies (S1, S2, S4, S5, S9, S10, S11, S12, S13, S14, S16, S19, S21) noted system integration complexity as both a driver and a challenge. Challenges include dependency management, validation/verification, and lead time in supply chains. Agile decomposition practices help, but long-term planning and understanding dependencies remain problematic.

Studies (S1, S3, S4, S5, S7, S9, S10, S12, S13, S14, S18, S19, and S22) discuss organizational/team dynamics, emphasizing the need for industries to rethink hierarchical structures. Effective communication and team cohesion are essential for Agile success.

Safety management is critical, especially in industries where product safety is paramount. Interestingly, regulatory/compliance was ranked as a more significant challenge than safety, even though many regulations aim to ensure safety. Incremental safety checks could address both concerns.

Finally, process/workflow alignment can impact Agile transformation by influencing efficiency, team dynamics, and consistent value delivery.

RQ4: Are adaptations needed when applying Agile to LS/SC/CP Systems? The Adaptations made by industries to Agile are shown in Figure 4.6. For purposes of this paper, safety frameworks include hazard analysis and assurance; specialized artifacts refer to additional documentation not typically developed in Agile projects; modeling/simulation addresses the use of digital twins, simulators, and emulators; traceability involves bi-directional traceability matrices; Automation relates to automating compliance checks with integrated tool chains; and reimagined programmatics centered on iterative and incremental management techniques to manage teams. The adaptations are ranked as follows: Safety Frameworks (26%), Specialized Artifacts (21%), Modeling/Simulation (18%), Traceability (13%), Automation (11%), and reimagined programmatics (11%).

The adoption of safety frameworks and stories was noted in studies (S2, S3, S4, S8, S9, S12, S13, S17, S20, and S23). This highlights how new Agile applications are in this domain.

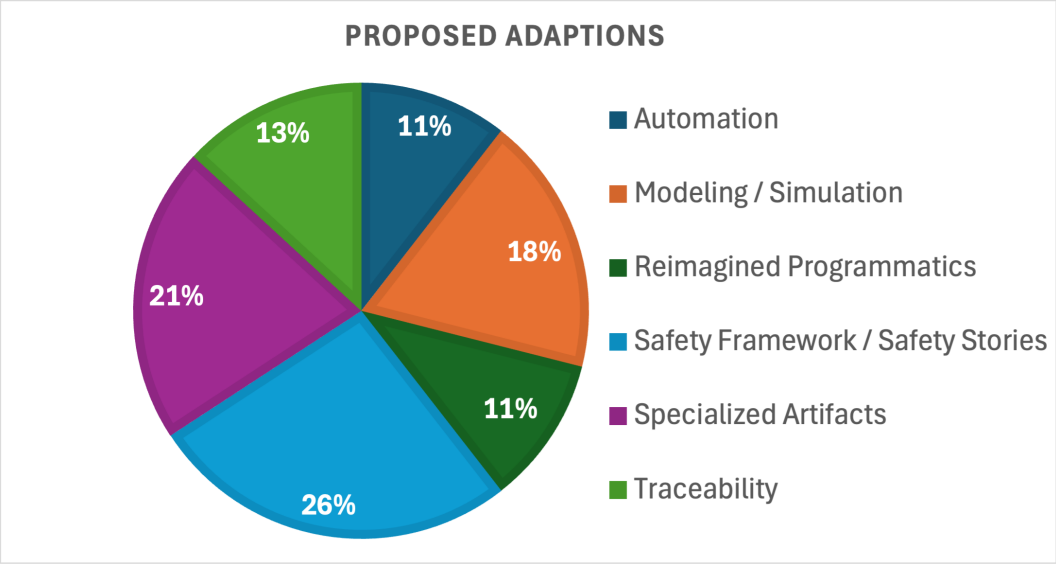


Figure 4.6: Proposed Adaptations to Agile

Safety frameworks like Risk Assessment and Mitigation Management System (RAMMS), Failure Modes and Effects Analysis (FMEA), and Formal Methods enabled the integration of safety stories into product backlogs. Study S2 developed a compliance framework for space vehicle development to integrate Agile, while Study S12 introduced a Safety-Critical Agile Adoption Assessment (SCA3DA) metamodel to support design decisions and improve safety awareness. The Safety Frameworks adaptation category supports both regulatory/compliance and safety challenges.

Agile typically involves lean documentation, but studies (S3, S7, S11, S14, S15, S16, S20) reported the addition of specialized artifacts. Study S7 introduced boundary objects used to collaborate across teams in the automotive industry. Study S11 combined System Theoretic Process Analysis (STPA) with Behavior-Driven Development (BDD) to create STPA BDD artifacts, enhancing regulatory compliance and safety. The Specialized Artifact category supports organization/team dynamics and regulatory compliance.

Modeling/Simulation was incorporated into Agile workflows in studies (S5, S6, S7, S9, S10, S19, and S22) to provide fast feedback on development decisions. This allows updates to physical products in timelines similar to software. Study S22 used a Digital Twin, a real-time virtual representation of a physical system—as a key tool in Agile implementations for cyber-physical

systems. The Modeling/Simulation adaptation category specifically supports the system complexity challenge by making the system easier to visualize and enabling teams to get feedback faster to overcome the impact of supply chain lead times.

Traceability could have been part of specialized artifacts, but it was frequently mentioned as its category, mainly bidirectional traceability matrices required for regulatory compliance. This addition suggests that previous Agile implementations were in non-safety-critical environments. The Traceability category aligns directly with overcoming regulatory/compliance challenges.

Although Agile is known for automation, the review reported it as an addition because many regulatory/compliance and safety checks are manual. The Automation category addresses the need to incorporate these checks into the DevSecOps pipeline.

The final category, reimagined programmatic, refers to new program management practices, including changes in organizational structure, subcontract management, and intentional risk management. This category supports organization/team dynamics and process/workflow.

4.4 Status of proposed adaptations

While incorporating safety frameworks such as RAMMS and FMEA have been adopted in small-scale projects, they are not mature and widely adopted. Adaptations like incorporating bidirectional traceability are still in the proposal stage. They are primarily used in research but have yet to be implemented at scale across the industry.

4.5 Conclusion

This systematic literature review has explored applying Agile principles and practices to large-scale, safety-critical cyber-physical (LS/SC/CP) systems, a domain characterized by complexity, stringent regulatory constraints, and high-reliability requirements. Through a comprehensive review of existing literature, this study has identified the key drivers behind Agile adoption, the significant challenges encountered, and the various adaptations to Agile practices currently being applied in this domain.

The findings show that Agile methodologies have gained notable traction in the LS/SC/CP Systems domain. This trend is driven by the need for faster development cycles, greater adaptability, improved quality, and cost efficiency. It is most pronounced in the medical sector, with aerospace closely behind.

However, transitioning to Agile in LS/SC/CP Systems development is challenging. The most prominent obstacle is regulatory compliance, reflecting the rigorous safety standards that industries like healthcare and aerospace face. Other challenges include managing team dynamics and organizational change, integrating Agile methods into existing workflows, ensuring effective safety management, and coping with the inherent complexity of such systems. These difficulties illustrate the need for adaptations to standard Agile practices.

The literature identifies several enhancements to Agile that are being employed to address these challenges. These include incorporating safety frameworks, creating specialized artifacts, leveraging modeling/simulation, increasing automation, reimagining programming, and implementing bidirectional traceability. These enhancements ensure that Agile practices can be safely and effectively applied to developing LS/SC/CP Systems.

In conclusion, while Agile methodologies offer substantial benefits for developing LS/SC/CP Systems, their successful application requires overcoming considerable challenges and implementing targeted adaptations. Future efforts should incorporate these adaptations into existing Agile frameworks, addressing the unique demands of LS/SC/CP systems. Additionally, these adaptations must be empirically validated in real-world projects. Continued collaboration between academia and industry will be critical to advancing the application of Agile within this complex domain, ensuring innovation while maintaining the highest standards of safety in the development of intricate cyber-physical systems.

4.6 Included Studies

[S1] Ågren, S. M., et al. Agile beyond teams and feedback beyond software in automotive systems. In: IEEE Transactions on Engineering Management, vol. 69, no. 6, pp. 3459-3475 2022.

[S2] Al-Mhdawi, M. K. S., et al. An agile compliance framework for the European Cooperation for Space Standardization. In: 2023 IEEE Aerospace Conference, pp. 1-12. IEEE, 2023.

[S3] Myklebust, T., et al. The Agile RAMSS lifecycle for the future. In: ESREL 2019, Springer, Germany 2019.

[S4] Maqsood, H.A.F.I.Z.A. Agile software development methodologies for safety critical systems. In: Proceedings of the 2022 Conference on Safety Critical Systems, Springer, 2022.

[S5] Demissie, S., Keenan, F., Özcan-Top, Ö., McCaffery, F. Agile usage in embedded software development in safety critical domain–a systematic review. In: Software Process Improvement and Capability Determination: 18th International Conference, SPICE 2018, Thessaloniki, Greece, October 9–10, 2018, Proceedings, vol. 18, pp. 316-326. Springer, 2018.

[S6] Barbareschi, M., Barone, S., Casola, V., Della Torca, S., Lombardi, D.: Automatic Test Generation to Improve Scrum for Safety Agile Methodology. In: Proceedings of the 18th International Conference on Availability, Reliability and Security, pp. 1-6 2023.

[S7] Wohlrab, R., Pelliccione, P., Knauss, E., Larsson, M.: Boundary objects in agile practices: Continuous management of systems engineering artifacts in the automotive domain. In: Proceedings of the 2018 International Conference on Software and System Process, pp. 31-40 2018.

[S8] Vierhauser, M., Mayr-Dorn, C. Breaking the deep freeze: Visualizing change in agile, safety-critical systems. Springer.

[S9] Koren, I., Rinker, F., Meixner, K., Matevska, J., Walter, J. Challenges and opportunities of DevOps in cyber-physical production systems engineering. In: 2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS), pp. 1-6. IEEE, 2023.

[S10]Trauer, J., Schweigert-Recksiek, S., Gövert, K., Mörtl, M., Lindemann, U. Combining agile approaches and risk management for mechatronic product development–a case study. In: Proceedings of the Design Society: DESIGN Conference, vol. 1, pp. 767-776. Cambridge University Press, 2020.

[S11] Wang, Y., Wagner, S. Combining STPA and BDD for safety analysis and verification in agile development: A controlled experiment. In: Agile Processes in Software Engineering and

Extreme Programming: 19th International Conference, XP 2018, Porto, Portugal, May 21–25, 2018, Proceedings, vol. 19, pp. 37-53. Springer, 2018.

[S12] Leite, I. M., Antonino, P. O., Nakagawa, E. Y. From safety requirements to just-enough safety-centered architectural solutions in agile contexts. In: Proceedings of the XXXIV Brazilian Symposium on Software Engineering, pp. 766-771. ACM, 2020.

[S13] Łukasiewicz, K., Górski, J. Introducing agile practices into development processes of safety critical software. In: Proceedings of the 19th International Conference on Agile Software Development: Companion, pp. 1-8. ACM, 2018.

[S14] Badanahatti, A., Pillutla, S. Interleaving software craftsmanship practices in medical device agile development. In: Proceedings of the 13th Innovations in Software Engineering Conference (formerly known as India Software Engineering Conference), pp. 1-5. ACM, 2020.

[S15] Kuiter, E., Krüger, J., Saake, G. Iterative development and changing requirements: drivers of variability in an industrial system for veterinary anesthesia. In: Proceedings of the 25th ACM International Systems and Software Product Line Conference - Volume B, pp. 113-122. ACM, 2021.

[S16] Zaeske, W., Durak, U. Leveraging semi-formal approaches for DepDevOps. In: Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops: DECSoS 2020, DepDevOps 2020, USDAI 2020, and WAISE 2020, Lisbon, Portugal, September 15, 2020, Proceedings, vol. 39, pp. 217-222. Springer, 2020.

[S17] Maqsood, H. M., Guerra, E. M., Wang, X., Bondavalli, A. Patterns for development of safety-critical systems with agile: Trace safety requirements and perform automated testing. In: Proceedings of the European Conference on Pattern Languages of Programs 2020, pp. 1-6.ACM 2020.

[S18] Hostettler, R., Böhmer, A. I., Lindemann, U., Knoll, A. TAF agile framework reducing uncertainty within minimum time and resources. In: 2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC), pp. 767-775. IEEE, 2017.

[S19] Wiecher, C., Japs, S., Kaiser, L., Greenyer, J., Dumitrescu, R., Wolff, C. Scenarios in the loop: integrated requirements analysis and automotive system validation. In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, pp. 1-10. ACM/IEEE, 2020.

[S20] Heeager, L. T., Nielsen, P. A. A conceptual model of agile software development in a safety-critical context: A systematic literature review. In: Information and Software Technology, vol. 103, pp. 22-39. Elsevier, 2018. [S21] Nielsen, P. A., Heeager, L. T. The dynamics of agile practices for safety-critical software development. In: Proceedings of the XP2017 Scientific Workshops, pp. 1-6. ACM, 2017.

[S22] Ugarte Querejeta, M., Etxeberria, L., Sagardui, G. Towards a DevOps approach in cyber physical production systems using digital twins. In: International Conference on Computer Safety, Reliability, and Security, pp. 205-216. Springer, Cham 2020.

[S23] Cleland-Huang, J., Agrawal, A., Vierhauser, M., Mayr-Dorn, C. Visualizing change in agile safety-critical systems. In: IEEE Software, vol. 38, no. 3, pp. 43-51. IEEE, 2020.

[S24]Roy, D., Balszun, M., Heurung, T., Chakraborty, S., Naik, A. Waterfall is too slow, let's go Agile: Multi-domain coupling for synthesizing.

Chapter 5

Scaling Frameworks - Research Question 1

What are Agile Scaling Frameworks, and how do they compare? What is their suitability in delivering large-scale, safety-critical, cyber-physical systems?

Scaling frameworks are crucial in extending Agile methods to LS/SC/CP Systems. This chapter critically examines the current state of Agile scaling practices by decomposing the top 10 Agile scaling frameworks into their constituent elements, illustrated in Figure 5.11, and comparing them. The analysis evaluates their effectiveness in building and deploying LS/SC/CP systems based on their ability to overcome the following challenges, such as regulatory compliance, safety assurance, integration complexity, traceability/documentation, and cultural/organizational barriers. This analysis will provide valuable insights into the strengths and weaknesses of existing frameworks. The research proposes adaptations to enhance existing Agile frameworks' ability to address challenges.

5.1 What are the current Frameworks

For this research, we began with Digital AI's 17th annual Agile assessment [71]. This survey is well respected across the industry and has been published for over 18 years. In addition, we delivered an independent survey to verify if there were similar trends. Both the industry benchmark survey and our survey showed the top 2 choices were the SAFe Framework and Other. However, our survey showed the use of only 3 of the frameworks discussed in Digital.ai. Our survey varied because we focused on safety-critical cyber-physical systems and had a much smaller data set. Table 5.1 illustrates the survey results below.

Table 5.1: Top ten frameworks

Framework	**Methodologist**	**Date**	**Digital.ai Survey**	**Author Survey**
Lean Management	James Womack, Daniel Jones	1990	2%	0%
Enterprise Scrum	Mike Beedle	1997	4%	0%
Large-Scale Scrum (LeSS)	Craig Larman, Bass Vodde	2005	2%	2%
Scrum@Scale / Scrum of Scrums	Jeff Sutherland, Ken Schwaber	2006	19%	0%
Agile Portfolio Management	Jochen Krebs	2008	1%	0%
Scaled Agile Framework (SAFe)	Dean Leffingwell	2010	26%	68%
Disciplined Agility (DA)	Scott Ambler, Mark Lines	2011	3%	2%
Spotify	Henrik Kniberg; Anders Ivarsson	2012	2%	0%
Recipes for Agile Governance (RAGe)	Kevin Thompson	2013	0%	0%
Nexus	Ken Schwaber	2015	1%	0%
Other	N/A	N/A	39%	28%

5.1.1 Overview of top 10 Agile Scaling Frameworks

Agile Scaling Frameworks are structured methods designed to apply agile principles and practices across large, organization-wide environments, beyond the traditional team level [72]. The

frameworks have evolved to support scalability, alignment, and standardization for multi-team development.

Lean

Lean Management, a transformative approach to production and organizational efficiency, was popularized by Womack and Jones in their seminal work "The Machine that Changed the World" [2]. At its core, Lean Management emphasizes maximizing customer value while minimizing waste, thus creating more value for customers with fewer resources [73]. The framework illustrated in Figure 5.1 is to maximize the value delivered. Lean Management was the first book to describe the Toyota Lean Production System. Key concepts of Lean Management include **Value**, **Value Stream**, **Flow**, **Pull**, and **Perfection**. Lean has reduced unnecessary wait times and streamlined operations across multiple domains from Starbucks to emergency department operations [74]. Mary and Tom Poppendieck adapted Lean to support software in 2003 with their Lean Agile toolkit [40]. Lean continues to be successful, recently Katie Anderson took lean principles and applied them to leadership in her book "Learning to Lead, Leading to Learn" [75]. Her work emphasizes the critical role of effective leadership in fostering a culture of continuous improvement and learning in organizations applying Lean principles.

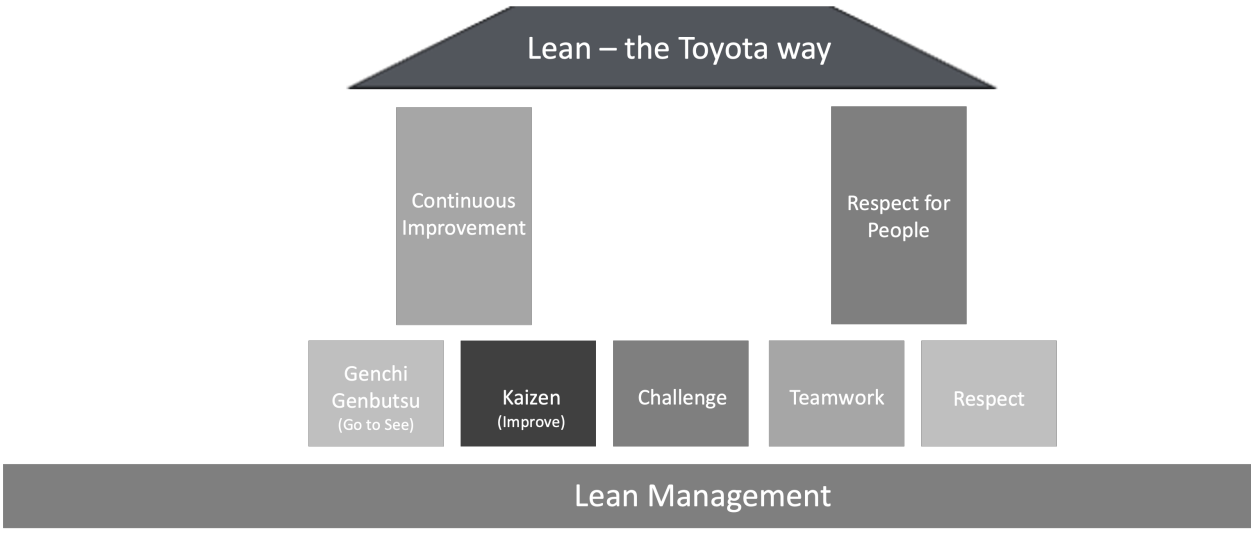


Figure 5.1: Lean Management Framework [2]

Enterprise Scrum

Enterprise Scrum, formalized by Mike Beedle in his 2013 book Enterprise Scrum, An Adaptive Method for Project Success [76], was introduced in 1997, predating the Agile Manifesto [77]. This framework, illustrated in Figure 5.2, leverages a series of tiered business canvases to create alignment across multiple teams. Enterprise Scrum addresses the challenges of scaling Agile in large organizations by providing a structured approach to coordinate and integrate the work of multiple Scrum teams. Enterprise Scrum aims to create coherence and alignment among various Scrum teams, addressing the critical challenge of coordination that often arises in large-scale Agile implementations.

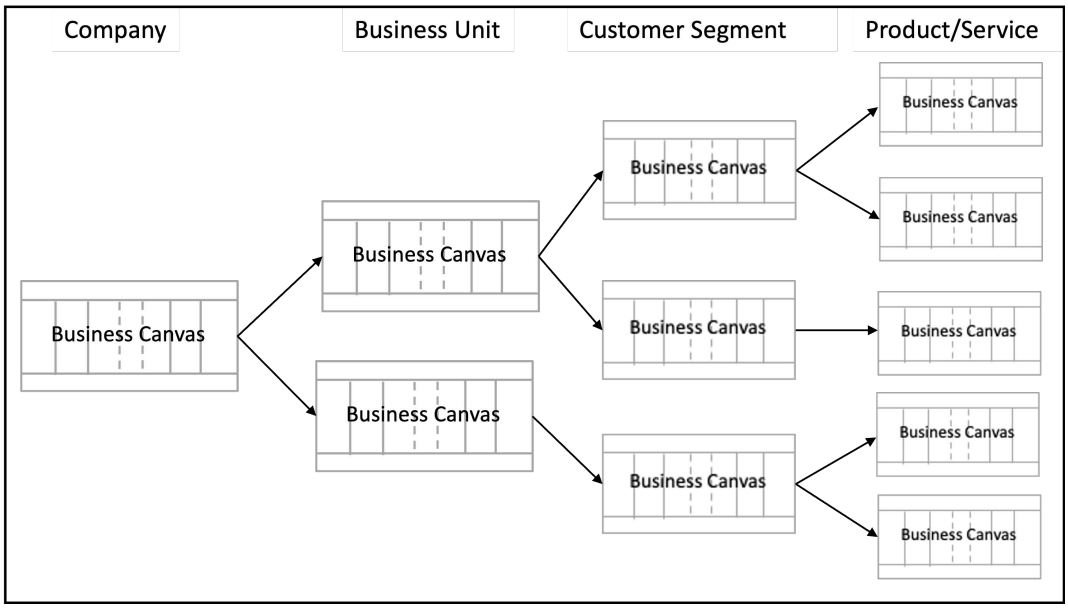


Figure 5.2: Enterprise Scrum Framework [3]

Large-Scale Scrum (LeSS)

Large-Scale Scrum (LeSS), released in 2005 by Craig Larman and Bas Vodde, builds upon their work in Agile and Iterative Development: A Manager's Guide [78]. LeSS supports up to eight teams and was conceived to address the challenges with offshoring large projects. LeSS, illustrated in Figure 5.3, emphasizes iterative development, team autonomy, and customer collaboration, ensuring that scaling does not dilute these essential Agile tenets [4]. For projects requiring

more than eight teams, LeSS provides additional guidance called LeSS Huge, further extending its scalability and adaptability. The LeSS Huge framework introduces specific roles, events, and artifacts shaped to accommodate the coordination among numerous teams and the interdependencies that arise when scaling Agile.

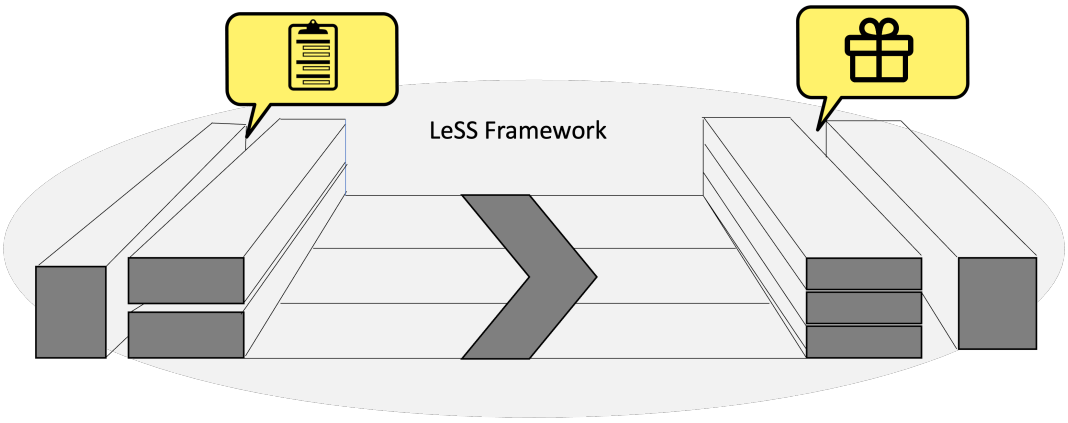


Figure 5.3: Large-Scale Scrum LeSS [4]

Scrum@Scale / Scrum of Scrums

Introduced in 2006 by Jeff Sutherland and Ken Schwaber, authors of the Scrum Guide [79], Scrum@Scale (illustrated in Figure 5.4) aims to support the coordination of multiple business units. This framework focuses on organizational structure and business rhythms to scale Agile effectively across various teams. The framework is a lightweight, flexible approach to scaling Agile across enterprises. As described in Sutherland's book, the framework prioritizes team autonomy, decentralized decision-making, and continuous improvement while maintaining alignment through interconnected Scrum teams [80]. It allows organizations to scale at their own pace while preserving Agile values and minimizing bureaucracy. The framework has two main components: the Executive Action Team (EAT) and the Scrum of Scrums (SoS). The EAT consists of high-level stakeholders responsible for ensuring that the overall strategy aligns with the organization's goals and that teams have adequate resources to succeed.

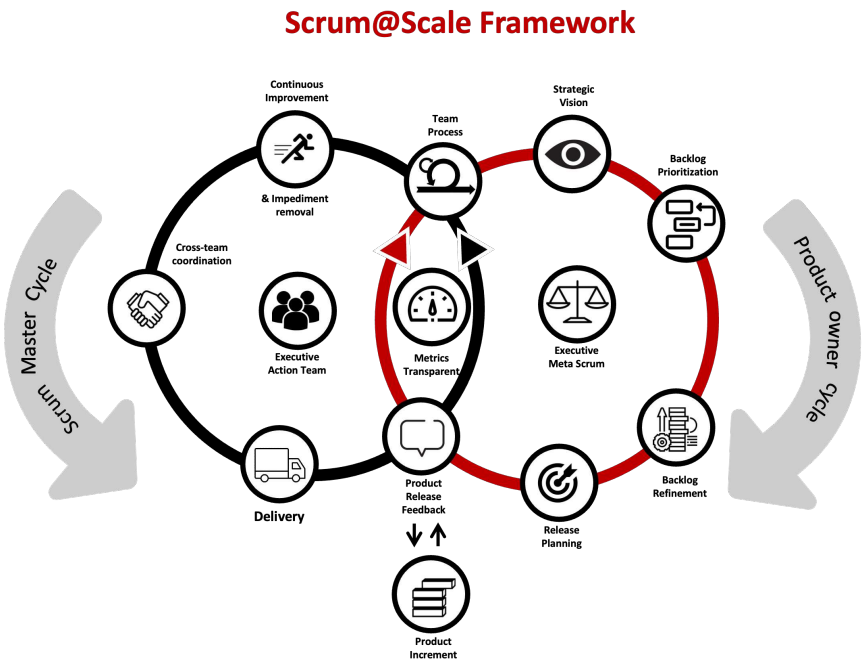


Figure 5.4: Scrum of Scrums / Scrum@Scale [5]

Agile Portfolio Management

Agile Portfolio Management (APM), introduced in 2008 by Jason Krebs [6], was developed to bridge the gap between project teams and executives. Illustrated in Figure 5.5, APM enhances the alignment of Agile initiatives with overall business objectives by facilitating effective prioritization, resource allocation, and risk management across multiple Agile projects. APM is supported through three key bodies of knowledge: program management (PMBoK), unified process models, and Scrum [81]. By setting strategic direction and managing portfolios, APM fosters transparency and collaboration between teams and executives, ensuring that Agile development efforts contribute to the organization's strategic objectives.

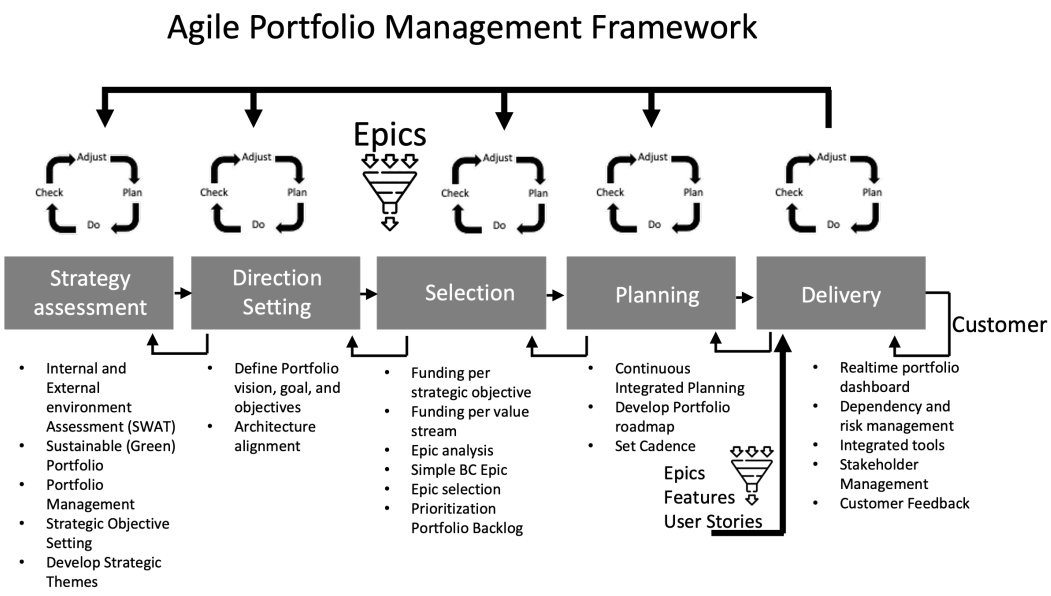


Figure 5.5: Agile Portfolio Management (APM) [6]

Scaled Agile Framework (SAFe)

The Scaled Agile Framework (SAFe), introduced in 2010 by Dean Leffingwell, author of Agile Software Requirements [82], is a comprehensive framework for scaling Agile across the enterprise. Illustrated in Figure 5.6, SAFe has three tiers: Portfolio, Agile Release Train, and Team. This structure enables effective coordination and alignment across multiple levels of the organization. SAFe also addresses common issues in enterprise Agile adoption, such as inter-team synchronization and governance, without sacrificing flexibility [83]. By providing a clear roadmap for scaling Agile, SAFe has become the most widely adopted framework for large-scale Agile transformations.SAFe effectively incorporates agile methodologies, including Lean, Scrum, Kanban, and Extreme Programming (XP). This multi-faceted approach enhances the framework's versatility in addressing the complexities [84].

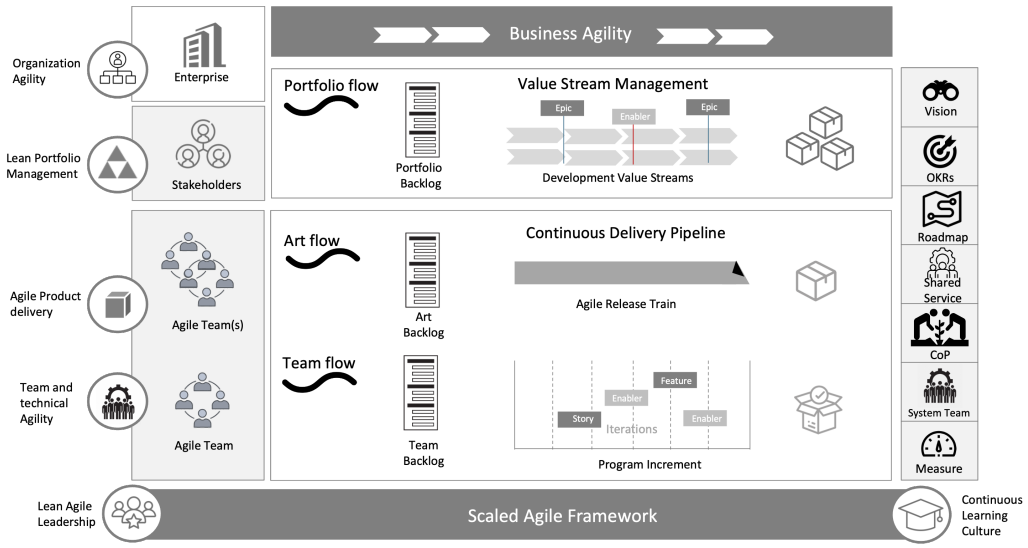


Figure 5.6: Scaled Agile Framework [7]

Disciplined Agility (DA)

Disciplined Agile Delivery (DaD), introduced in 2012 by Scott Ambler [26], stems from Unified Process, Extreme Programming (XP), and Kanban. This framework offers a hybrid approach to Agile, scaling both tactically and strategically. In 2015, DaD evolved into Disciplined Agile (DA), illustrated in Figure 5.7, and was later acquired by the Project Management Institute (PMI) in 2019. By incorporating elements from various Agile and Lean approaches, DA provides a flexible and comprehensive toolkit for tailoring Agile implementations to specific organizational needs. This adaptability makes DA particularly valuable for organizations seeking to optimize their Agile practices and achieve greater agility. DA leverages principles from multiple agile methodologies such as Scrum, Lean, Kanban, and Extreme Programming (XP) [85].

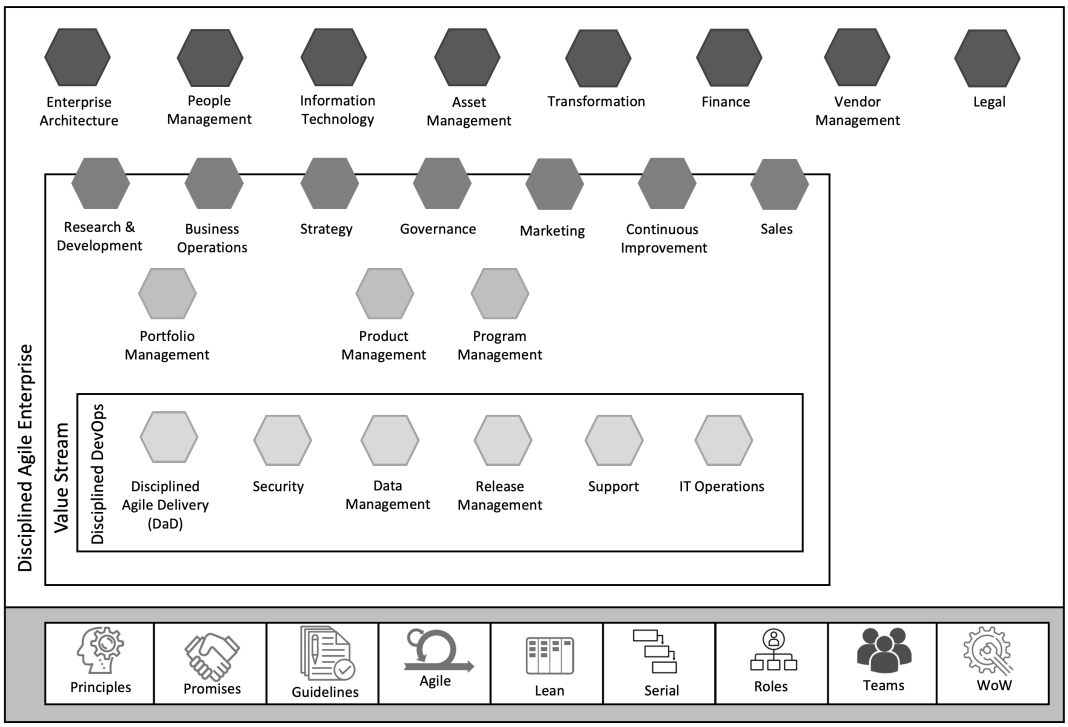


Figure 5.7: Disciplined Agility [8]

Spotify

The Spotify scaling model, introduced in 2012 by Henrik Kniberg and Anders Ivarsson [9], offers a unique approach to organizing Agile teams. Illustrated in Figure 5.8, the Spotify tribe engineering model consists of seven organizational elements, with the chief architect playing a crucial role in maintaining system integrity. This model, which began as an internal organizational blueprint for Spotify, emphasizes autonomy, minimal governance, and a flat organizational structure. By fostering a culture of trust and empowerment, the Spotify model enables teams to operate independently and make quick decisions, promoting agility and responsiveness to change. While the authors themselves do not continue to update the framework, many teams adopt it due to its flexibility [72] [86].

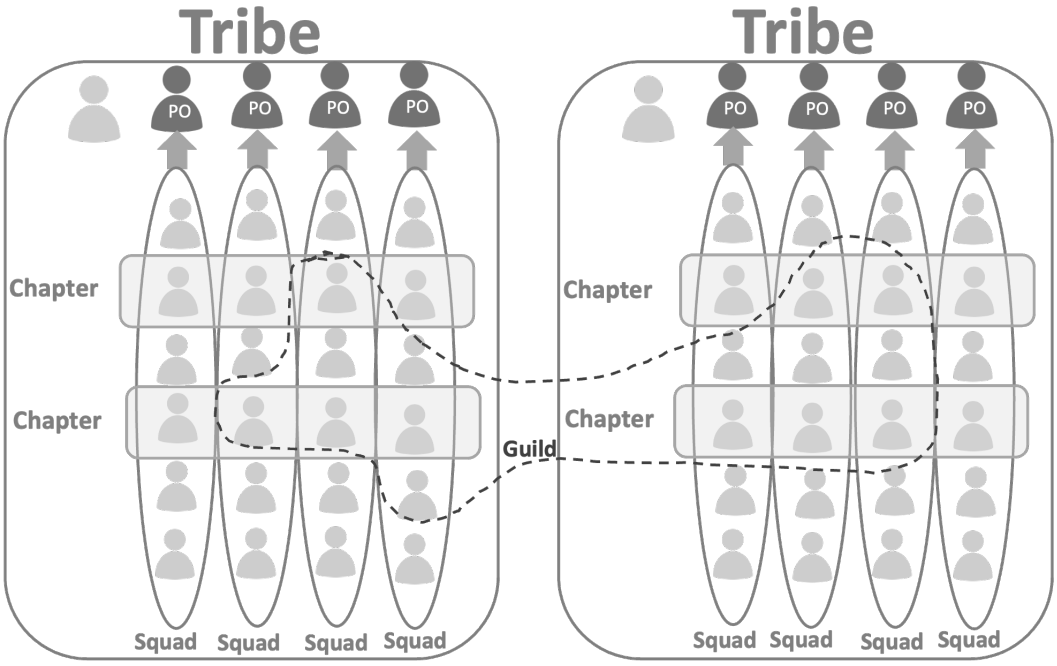


Figure 5.8: Spotify [9]

Nexus

The Nexus framework, introduced in 2015 by Ken Schwaber [87], provides a streamlined approach to scaling Scrum. It stays close to pure Scrum with the addition of the Nexus Integration Team (NIT), which enables coordination across multiple teams. Specifically, the NIT is accountable for integrating all the teams' work, ensuring alignment and cohesion across the project. By providing a clear structure for managing dependencies and integrating work, Nexus helps organizations scale Scrum effectively while maintaining the framework's simplicity and focus on collaboration, thereby enhancing productivity and collaboration [88]. Central to the Nexus framework is its emphasis on reducing unnecessary overhead in communication between teams, which is crucial for maintaining agility. Dumitriu et al. highlights that large projects often suffer from increased developer burden due to complex inter-team dependencies, suggesting that frameworks like Nexus can alleviate these challenges by promoting more streamlined interactions across teams [89]. Nexus stays close to pure Scrum.

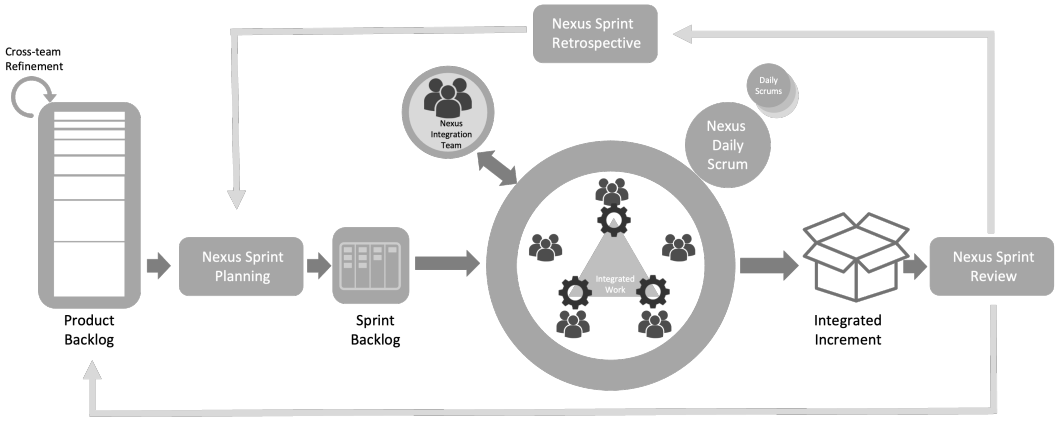


Figure 5.9: Nexus Framework [10]

Recipes for Agile Governance (RAGe)

The Recipes for Agile Governance (Rage) framework, developed in 2013 by Kevin Thompson from CPrime [11], draws from Scrum and Kanban principles. This framework focuses on establishing tiered governance, structured roles and responsibilities, and performance benchmarks. By providing a clear framework for governance, Rage helps organizations maintain control and oversight while implementing Agile methodologies. This ensures that Agile adoption aligns with organizational goals and regulatory requirements. The emphasis on structured roles and responsibilities promotes accountability and transparency, while performance benchmarks enable continuous improvement and optimization of Agile practices.

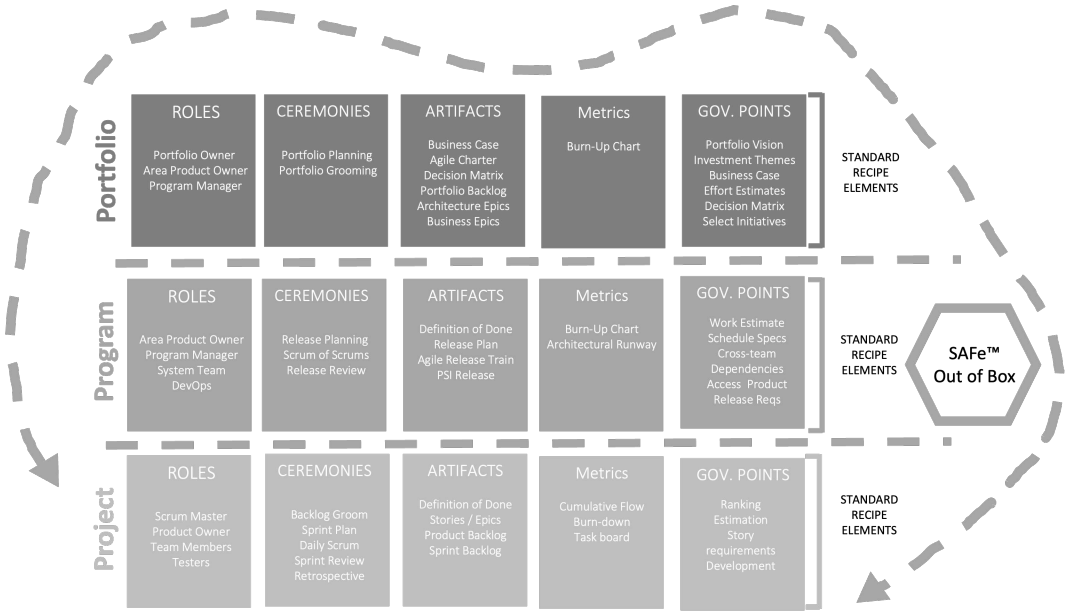


Figure 5.10: Recipes for Agile Governance [11]

5.1.2 Comparative Analysis

We leveraged the dimensions of a business operating model to compare the Agile Scaling frameworks, as both aim to coordinate complex work across an organization. However, as there is no formalized set of dimensions specific to Agile scaling frameworks, applying a business operating model provided a structured comparison. The frameworks were compared across six predefined dimensions: principles, organizational structures, roles, processes, tools, and culture, to identify key differentiators and commonalities, as illustrated in Figure 5.11. By examining these dimensions, the research aims to identify the most suitable frameworks for LS/SC/CP system development, considering the unique challenges and requirements of this domain.

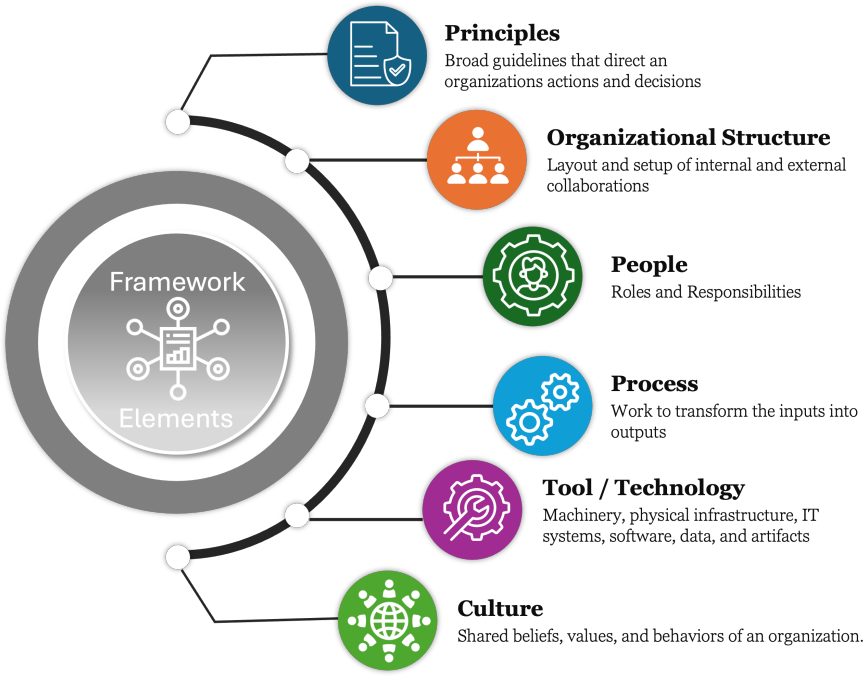


Figure 5.11: Framework Elements

Guiding Principles

We define principles as guidelines that direct organizations' actions and decisions. The analysis categorized these principles into five key themes: Value-Driven Development, Continuous Improvement, Systems Thinking, Servant Leadership, and Flexibility. As shown in Figure 5.12, Continuous Improvement (70%) is the most frequently represented principle, appearing in 7 of the 10 frameworks and indicating the importance of continuous learning and adaptation. Empiricism (40%), Flexibility/Optionality (40%), Systems Thinking (40%), and Value-Driven Development (40%) are moderately represented, suggesting a shared emphasis on these principles across many frameworks. However, Leadership (30%) is present in only 3 of the frameworks, reflecting variations in the leadership models emphasized. The two most comprehensive frameworks are SAFe(100%) and Disciplined Agility (83%), indicating they may have the broadest toolkits to support organizational challenges. These findings highlight the importance of carefully considering the principles emphasized by different frameworks when selecting an approach for LS/SC/CP

development. The varying representation of principles across frameworks can significantly impact organizational culture, leadership styles, and overall effectiveness in adopting Agile practices."

Principles	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Continuous Improvement	1	1	1		1			1	1	1	7	70%
Empiricism					1	1		1	1		4	40%
Flexibility / Optionality	1	1	1					1			4	40%
Servant Leadership		1						1		1	3	30%
Systems Thinking	1	1			1			1			4	40%
Value-Driven Development		1		1				1		1	4	40%
	3	5	2	1	3	1	0	6	2	3	26	
	50%	83%	33%	17%	50%	17%	0%	100%	33%	50%		

Figure 5.12: Guiding Principles

Organizational Structure

Organizational structure refers to the layout of internal and external collaborations. The four most typical structures in Agile frameworks are Teams, Teams of Teams, Communities of Practice, and Executive Leadership Groups. As shown in Figure 5.13, every framework explicitly addresses the team structure, demonstrating the universal importance of teams as the core unit of Agile development. 90% of frameworks address the Team of Teams structure, reflecting the importance of cross-team alignment, synchronization, and strong communication channels for successful scaling. Communities of Practice promote knowledge sharing and learning across the organization, while Executive Leadership Teams provide strategic direction and oversight.

The analysis reveals that SAFe has full coverage (100%) of these organizational structures, while Agile Portfolio Management (APM), Disciplined Agility (DA), Large-Scale Scrum (LeSS), Scrum@Scale, and Spotify have high coverage (75%). The remaining Agile scaling frameworks have moderate to low coverage, suggesting a lower maturity level in addressing complex organizational structures. These findings highlight the importance of selecting a framework that aligns with the organizational complexity and collaboration needs of LS/SC/CP projects. Frameworks

with high coverage may be better suited for complex hierarchies and distributed teams, while those with lower coverage may be more appropriate for smaller, co-located teams.

Organizational Structures	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Communities		1			1			1		1	**4**	40%
Executive Leadership	1							1	1		**3**	30%
Team	1	1	1	1	1	1	1	1	1	1	**10**	100%
Team of Teams	1	1	1		1	1	1	1	1	1	**9**	90%
	3	**3**	**2**	**1**	**3**	**2**	**2**	**4**	**3**	**3**	**26**	
	75%	75%	50%	25%	75%	50%	50%	100%	75%	75%		

Figure 5.13: Organizational Structures

People / Roles

The Agile roles, illustrated in Figure 5.14, include Team Member, Product Owner, Team Coach/Scrum Master, Scaled Product Owner, Scaled Team Coach/Scrum Master, Manager, Architect, and Business Owner. The universally present roles, including Team Member, Team Coach/Scrum Master, and Product Owner, are found at the team level, which is unsurprising, as the team is the core unit in Agile. Scaled Product Owners and Managers are represented in 60% of the organizations, indicating that many organizations face challenges in effectively implementing these roles at scale. Furthermore, the following roles are minimally represented: Scaled Team Coach/Scrum Master, Coordinator, Business Owner, and Architect, at 40% or lower. This result is surprisingly low, given the need for governance, architecture, and coordination to manage the complexity of large systems. It indicates potential challenges for organizations seeking to scale Agile effectively. Notably, Scaled Agile Framework (SAFe) and Disciplined Agile (DA) have the widest range of roles, reflecting their focus on Agile at the enterprise level, which is necessary to manage the complexity above. These findings emphasize the importance of selecting a framework that provides adequate support for the roles required in LS/SC/CP development, considering the complexity and scale of these projects.

Roles	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Architect		1						1			2	20%
Business Owner	1	1						1			3	30%
Coordinator		1		1				1		1	4	40%
Manager	1	1		1	1		1			1	6	60%
Product Owner	1	1	1		1	1	1	1	1	1	9	90%
Scaled Product Owner		1	1		1		1	1	1		6	60%
Scaled Team Coach / Scrum Master			1					1	1	1	4	40%
Team Coach / Scrum Master	1	1	1		1	1	1	1	1	1	9	90%
Team Member	1	1	1	1	1	1	1	1	1	1	10	100%
	5	**8**	**5**	**3**	**5**	**3**	**5**	**8**	**5**	**6**	53	
	56%	89%	56%	33%	56%	33%	56%	89%	56%	67%		

Figure 5.14: People and Roles

Process

Alignment/Coordination/Governance, Retrospective/Lesson Learned, and Planning are the most universally adopted Agile processes, present in 80–100% of the frameworks illustrated in Figure 5.15. This highlights their critical role in fostering collaboration, continuous improvement, and strategic direction within Agile environments. Continuous Delivery and Portfolio/Program Management are moderately represented, reflecting their growing importance in scaling frameworks where managing technical workflows and strategic backlogs is essential.

In contrast, processes like Architecture and Quality Improvement have limited adoption, which could result in technical debt and system fragility. This limited adoption indicates a potential risk to the stability and long-term health of the systems. Frameworks designed for enterprise-scale agility, such as SAFe and DA, cover a broader range of processes to support complex organizational structures. Conversely, lightweight frameworks, like Spotify and Scrum@Scale, focus on simplicity, autonomy, and minimal governance. This contrast illustrates how different frameworks balance scaling needs with Agile's core principles. These findings emphasize the importance of selecting a framework that provides adequate support for the processes required in LS/SC/CP development, considering the complexity and scale of these projects.

Processes	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Retrospective / Lesson Learned		1	1		1	1	1	1	1	1	**8**	80%
Alignment / Coordination / Governance	1	1	1	1	1	1	1	1	1	1	**10**	100%
Architecture		1						1			**2**	20%
Continuous Delivery	1	1	1	1	1	1		1			**7**	70%
Demonstration			1		1	1		1			**4**	40%
Planning	1	1	1		1	1	1	1	1		**8**	80%
Portfolio / Program Management (Backlogs)	1	1					1	1	1		**5**	50%
Quality and Improvement	1			1	1			1			**4**	40%
	5	**6**	**5**	**3**	**6**	**5**	**4**	**8**	**4**	**2**	**48**	
	63%	75%	63%	38%	75%	63%	50%	100%	50%	25%		

Figure 5.15: Processes

Tool/Technology

As shown in Figure 5.16, Backlog/List tools are the most widely adopted across Agile frameworks, with 90% coverage. This highlights their fundamental role in work prioritization, sprint planning, and iterative development. Tools supporting Alignment/Coordination/Governance, Automation/Test, and Work Visualization are also prominent, each present in 60% of frameworks. This reflects their importance in enhancing cross-team collaboration, supporting continuous delivery, and improving workflow transparency. Planning and Progress Tracking/Metrics tools show moderate adoption 50%, indicating that while these functions are critical, the degree of formalization varies depending on the framework's complexity and scale.

Furthermore, tools related to Architecture and Business Case/Need Definition are less common at 40%, typically found in frameworks focused on enterprise agility where strategic alignment and technical governance are essential. Frameworks like SAFe and RAGE exhibit comprehensive tool integration to manage large-scale Agile implementations, supporting governance, automation, and performance tracking. Conversely, lightweight frameworks such as Scrum@Scale, LeSS, and Nexus adopt fewer formal tools, emphasizing flexibility, autonomy, and simplicity, consistent with Agile core principles. The widespread adoption of automation tools reflects the growing influence of DevOps, while the popularity of work visualization tools aligns with Agile's focus on trans-

parency and real-time collaboration. Overall, the results suggest that Agile frameworks balance between structured tool sets for complex environments and adaptive approaches that prioritize team autonomy and lightweight processes. These findings highlight the importance of selecting a framework with appropriate tool support for LS/SC/CP development, considering the specific needs and priorities of the project."

Tool/Technology	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Alignment / Coordination / Governance		1	1	1	1		1	1			6	60%
Architecture		1			1			1	1		4	40%
Automation / Test		1			1		1	1	1	1	6	60%
Backlog / List	1	1	1		1	1	1	1	1	1	9	90%
Buiness Case / Need Definition	1						1	1		1	4	40%
Planning		1	1				1	1			4	40%
Progress Tracking / Metrics		1	1	1			1	1			5	50%
Work Visualization			1	1			1	1		2	6	60%
	2	6	5	3	4	1	7	8	3	5		
	25%	75%	63%	38%	50%	13%	88%	100%	38%	63%		

Figure 5.16: Tool / Technology

Culture

Transparency is Agile frameworks' most emphasized cultural attribute, as shown in Figure 5.17. Its presence in 90% of the frameworks highlights its critical role in fostering trust, collaboration, and continuous improvement. Transparency is achieved by making work visible, promoting open communication, and enabling data-driven decision-making. Customer centricity and empowerment/accountability appear in 50% of the scaling frameworks, reflecting Agile's focus on delivering customer value and empowering self-organizing teams.

In contrast, innovation/rapid learning is the least emphasized attribute, present in only 30% of frameworks. This suggests that while continuous learning is central to Agile, it is often considered an implicit outcome rather than an explicitly defined cultural pillar. Frameworks like SAFe, Scrum@Scale, and Spotify demonstrate comprehensive cultural integration, emphasizing all four attributes to support process agility and mindset transformation. Conversely, frameworks such

as APM and RAGE place minimal focus on cultural attributes, potentially prioritizing technical practices or process optimization over cultural change. This variation implies that while some frameworks embed cultural values deeply into their structure, others rely on organizations to cultivate the necessary cultural environment. These findings highlight the importance of considering the cultural emphasis of different frameworks when selecting an approach for LS/SC/CP development. Organizations should proactively foster a culture that supports transparency, customer focus, empowerment, and continuous learning to maximize the benefits of Agile adoption.

Culture	APM	DA	Enterprise Scrum	Lean Management	LeSS	Nexus	RAGE	SAFe	Scrum@Scale	Spotify	Total	
Customer Centricity		1	1		1			1		1	**5**	50%
Empowerment / Accountability				1		1		1	1	1	**5**	50%
Innovation / Rapid Learning		1						1		1	**3**	30%
Transparency	1	1	1	1	1	1		1	1	1	**9**	90%
	1	**3**	**2**	**2**	**2**	**2**	**0**	**4**	**2**	**4**	**22**	
	25%	75%	50%	50%	50%	50%	0%	100%	50%	100%		

Figure 5.17: Culture

5.2 Frameworks suitability in building LS/SC/CP Systems

Applying Agile methodologies in complex domains such as large-scale, safety-critical, and cyber-physical systems presents multiple challenges. According to the literature review in Chapter 4, these challenges include regulatory compliance, system integration complexity, organizational dynamics, safety management, and existing organization process workflow [90]. Table 5.2 provides a detailed overview. Regulatory compliance often requires extensive documentation and traceability, which can be challenging with Agile's iterative and incremental approach. System integration complexity can lead to difficulty managing dependencies and ensuring consistency across different components. Organizational dynamics, such as resistance to change and cultural mismatches, can hinder Agile adoption. Safety management requires rigorous testing and verification, which may not align with Agile's emphasis on rapid iteration and continuous delivery. Finally, ex-

isting organizational process workflows may need to be adapted to accommodate Agile practices. These challenges highlight the need for careful planning and adaptation when implementing Agile in complex domains.

Table 5.2: Challenges identified in Literature Review

	Challenge	Description
1.	Regulatory Compliance	Safety-critical systems are subject to strict regulatory requirements, such as ISO 26262 for automotive systems or IEC 61508 for industrial safety, ensuring that Agile practices comply with these standards while maintaining The flexibility and speed that Agile offers a significant challenge [91]
2.	Safety Assurance	The need for safety assurance in these systems cannot be overstated. Agile's iterative approach must be carefully integrated with safety assurance practices, ensuring that each iteration does not compromise the system's safety. The safety challenge lies in balancing the need for thorough safety analysis with the desire for fast-paced, iterative development [92]
3.	Integration Complexity	Safety-critical CPS often involve integrating software with hardware components, creating a highly complex system with numerous interdependencies. CI/CD has to be adapted for integrating physical components [30]
4.	Traceability / Documentation	Agile emphasizes working capabilities over comprehensive documentation, safety-critical systems require detailed documentation to ensure traceability of requirements, design decisions, and testing results. Agile teams must find ways to produce the necessary documentation without undermining the efficiency and adaptability of their processes [93]
5.	Cultural / Organizational Barrier	Implementing Agile in environments traditionally relying on more linear, waterfall-based approaches can encounter resistance from teams and management. Successfully implementing Agile in these environments often requires significant cultural and organizational change, including training, shifts in mindset, and modifications to existing processes [12]

5.2.1 Suitability Evaluation Criteria

Regulatory Compliance

This research evaluates scaling frameworks' ability to support regulatory compliance by assessing their capacity to facilitate and manage related activities. These activities include item classification, developing compliance matrices, preparing regulatory document submissions, and conducting audit readiness reviews. Effective execution of these activities requires people, processes, and tools. However, the analysis reveals that all frameworks lack explicit guidance on handling regulatory compliance.

Despite this general lack of explicit support, the Scaled Agile Framework (SAFe) indirectly addresses compliance through its shared services team, including specialized roles that facilitate and manage compliance activities. Similarly, Disciplined Agility (DA) introduces the role of a technical coordinator to support compliance efforts. From a tools perspective, SAFe introduces the solution intent repository, which provides guidance and defines constraints. From a process perspective, both SAFe and DA outline governance structures that support the preparation of regulatory documentation and the execution of audit readiness assessments. These findings highlight the importance of considering the level of support for regulatory compliance when selecting an Agile framework for LS/SC/CP development. Organizations operating in regulated environments may need to adapt and extend existing frameworks to ensure compliance requirements are met effectively.

Safety Assurance

Safety assurance is paramount in large-scale, safety-critical, cyber-physical (LS/SC/CP) systems, which require rigorous processes to ensure reliable operation and prevent harm. Mechanisms to support safety assurance include hazard analysis, risk management, safety case development, human factors, and incident response planning. Typically, multiple types of hazard analysis are employed to ensure system safety. While none of the Agile Scaling Frameworks explicitly priori-

tize safety assurance as a core element, some frameworks provide relevant guidance and practices that can be adapted and extended.

For example, SAFe offers explicit compliance and risk management guidance that can be leveraged to incorporate safety assurance activities. Its emphasis on defining and managing risks aligns well with safety analysis processes, and its focus on continuous integration and testing can incorporate safety checks and validation activities throughout the development life cycle. Other frameworks, such as Disciplined Agile (DA), LeSS, and Scrum@Scale, offer foundational practices that can be adapted to support safety assurance. For instance, Scrum's sprint reviews can be adapted to include specific safety checks and evaluations. At the same time, LeSS's focus on minimizing dependencies and promoting cross-functional teams enhances communication and collaboration on safety-related aspects. DA's emphasis on context-specific process tailoring enables the integration of safety assurance activities into existing workflows. These findings highlight the importance of carefully evaluating and adapting Agile frameworks to ensure safety assurance is effectively addressed in LS/SC/CP development.

Integration Complexity

Integration of systems (SoS) is a complex endeavor that requires careful planning and execution. Mechanisms to support integration complexity include architecture, standardized interfaces, data mapping, robust system infrastructure, and governance. Several Agile frameworks provide explicit or indirect support for managing this complexity.

For example, SAFe and DA explicitly address integration complexity through various roles, processes, and practices. SAFe employs Enterprise and Solution Architecture roles, Agile Release Trains (ARTs), and Lean Portfolio Management (LPM) to manage architectural decisions, data integration, and governance across large-scale systems. Similarly, DA emphasizes Enterprise Awareness, with roles like the Architecture Owner to guide technical coherence, robust data management practices, and a formal Governance Life cycle to balance agility with compliance and risk management. Other frameworks, such as Nexus and Agile Portfolio Management (APM), indirectly support integration complexity. Nexus utilizes the Nexus Integration Team (NIT) to man-

age cross-team dependencies and technical alignment, supporting continuous integration practices. APM aligns architectural decisions with strategic objectives, ensuring data governance and infrastructure investment. These findings highlight the importance of considering the level of support for integration complexity when selecting an Agile framework for LS/SC/CP development, especially for projects with complex system interactions and dependencies.

Traceability / Documentation

Bidirectional traceability ensures that requirements can be traced forward (from requirements to implementation and testing) and backward (from code, design, or test results back to the original requirements). Traceability is critical in large-scale, safety-critical, cyber-physical systems to maintain compliance, manage complexity, and support change impact analysis. While none of the frameworks explicitly prioritize traceability as a core element, some provide relevant guidance and practices that can be adapted and extended.

For example, SAFe integrates traceability through its Requirements Model, connecting epics, capabilities, features, and stories. This provides a clear link between different levels of requirements and enables tracking of changes and their impact across the system. Similarly, DA supports traceability through its Governance Life cycle and emphasizes enterprise awareness. It allows organizations to tailor traceability practices based on context and ensure links between business requirements, architectural decisions, and technical implementations. These findings highlight the importance of considering the level of support for traceability when selecting an Agile framework for LS/SC/CP development, especially for projects with complex requirements and stringent compliance needs.

Cultural and Organizational Barriers

Overcoming cultural and organizational barriers requires effective change management practices, defined communication channels, strong leadership, and respect for diversity. Several Agile frameworks provide explicit or indirect support for addressing these challenges.

For example, SAFe and DA offer comprehensive support through structured change management practices, defined communication channels, strong leadership development, and a focus on diversity and inclusion. SAFe utilizes the Implementation Roadmap and promotes Lean-Agile Leadership to drive mindset shifts, supported by mechanisms like Agile Release Trains (ARTs) and Inspect & Adapt Workshops for transparent communication. DA emphasizes Enterprise Awareness and situational leadership, with its Transformation Cycle guiding organizations through tailored change initiatives. Both frameworks foster psychological safety, team structures, and inclusive leadership practices to support cultural transformation in large-scale, safety-critical environments. Other frameworks, such as Scrum@Scale, Enterprise Scrum, and RAGE, indirectly support cultural and organizational change. Scrum@Scale leverages the Executive Action Team (EAT) and Meta Scrum forums to promote decentralized decision-making and servant leadership, enhancing cross-team collaboration. Enterprise Scrum focuses on applying Agile principles enterprise-wide, fostering continuous adaptation and inclusion across diverse functions. RAGE supports change in compliance-heavy environments through lightweight governance practices and adaptive leadership models, ensuring diverse stakeholder engagement and cultural alignment. These findings highlight the importance of considering the level of support for artistic and organizational change when selecting an Agile framework for LS/SC/CP development, especially for organizations undergoing significant transformations or facing resistance to Agile adoption.

Analyses of frameworks in overcoming challenges

The analysis of Agile Scaling Frameworks reveals a critical gap in their explicit coverage of key challenges specific to LS/SC/CP systems development. These challenges include regulatory compliance, safety assurance, integration complexity, traceability/documentation, and overcoming cultural and organizational barriers. While the Scaled Agile Framework (SAFe) and Disciplined Agile (DA) provide the most comprehensive coverage, they still require adaptations to deliver LS/SC/CP systems effectively. This lack of comprehensive support demonstrates the need to carefully evaluate and adapt existing frameworks when implementing Agile in LS/SC/CP contexts. Organizations may need to develop supplementary guidance and practices to address these chal-

lenges effectively, ensuring that Agile adoption does not compromise safety, compliance, or system integrity.

5.2.2 Threats to Validity

Our research team has firsthand experience with a subset of Agile Scaling Frameworks, specifically SAFe, Disciplined Agile (DA), Scrum@Scale, Nexus, and Large-Scale Scrum (LeSS). While this practitioner perspective strengthens our ability to evaluate these frameworks with depth and nuance, it may also introduce bias in terms of:

- Favorable interpretation of familiar frameworks, particularly in qualitative assessments or when coding practices.
- Potential underrepresentation or misinterpretation of less familiar frameworks, Spotify Model, or Enterprise Scrum, where analysis was based solely on documentation and secondary sources.

To mitigate this threat, we employed strategies such as cross-referencing framework documentation, independent coding validation, and maintaining transparency about our research experience. Nonetheless, our partial experience base may have influenced the completeness or interpretation of findings, especially in comparative evaluations.

5.3 Conclusion

This research provides a comprehensive comparative analysis of the top Agile scaling frameworks, evaluating their suitability for LS/SC/CP systems. The findings indicate that while Agile frameworks such as SAFe and Disciplined Agile (DA) offer the most comprehensive coverage to support the unique challenges associated with LS/SC/CP systems, they still require adaptations to address specific gaps in regulatory compliance, safety assurance, integration complexity, traceability, and cultural transformation.

The study reveals that neither framework explicitly supports critical safety and regulatory elements. However, SAFe and DA indirectly address these aspects through governance structures,

compliance management roles, and process adaptations. Furthermore, key challenges such as system integration complexity and traceability/documentation are only partially addressed, necessitating strategic enhancements like incorporating Model-Based Systems Engineering (MBSE) tools, continuous compliance mechanisms, and safety assurance techniques such as System-Theoretic Process Analysis (STPA).

Chapter 6

Survey / Interviews

To what extent is Agile applied to large-scale, safety-critical cyber-physical systems? What challenges exist? What proposed adaptations have been used?

This chapter explores the current state of Agile adoption in LS/SC/CP systems, identifying existing challenges and analyzing proposed adaptations, and building upon a comprehensive literature review that identified key challenges ranked as Regulatory/Compliance (30%), System Complexity (22%), Organizational/Team Dynamics (19%), Safety Management (16%), and Process/Workflow (13%). This research employs a mixed-methods approach to gather empirical data. Furthermore, a deep dive evaluation of the top 10 scaling frameworks was conducted to assess their suitability for LS/SC/CP systems. The deep dive findings show no explicit guidance regarding LS/SC/CP systems, but some frameworks would be easier to adapt. First, a survey was conducted on SurveyMonkey, collecting responses from 56 industry professionals involved in LS/SC/CP development. This survey provided a broad overview of Agile adoption, the prevalence of these identified challenges, and the adaptations implemented to address them. Subsequently, semi-structured interviews were conducted with 25 survey respondents, delving deeper into their experiences and perspectives related to these challenges and the practicality of the scaling frameworks in their context. The chapter analyzes the combined data from the surveys and interviews, comparing and contrasting the empirical findings with the initial literature review and the framework evaluation to provide a comprehensive understanding of the extent of Agile adoption, the specific manifestations of the key challenges encountered, and the effectiveness of the proposed adaptations and selected scaling frameworks used in LS/SC/CP contexts.

6.1 Methodology

A mixed-methods approach was employed to comprehensively investigate the state of Agile adoption in large-scale, safety-critical, cyber-physical (LS/SC/CP) systems. This approach com-

bined quantitative and qualitative data collection through surveys and semi-structured interviews, aiming to provide a broad understanding of industry trends and in-depth insights into the experiences of professionals working in this domain.

6.1.1 Quantitative - Survey Method

This survey, conducted online via SurveyMonkey between August and October 2024, aimed to gather objective feedback from professionals knowledgeable in Agile methodologies and working within highly regulated industries, building Life Science/Supply Chain/Critical Process (LS/SC/CP) systems. Participation was anonymous and voluntary. We distributed the survey through professional networks and relevant social media platforms, including LinkedIn.

The survey utilized a combination of multiple-choice and open-ended questions, providing quantitative data for analysis and qualitative insights for deeper context. This approach enhanced data validity, uncovered unexpected insights, engaged respondents, and offered a more comprehensive understanding of the topic. Combining question types gave the survey a more holistic view of the subject matter.

6.1.2 Qualitative - Interview Method

Semi-structured interviews were conducted to gain deeper insights into Agile methodologies within LS/SC/CP systems. Participants were recruited from survey respondents who opted to provide contact information and through network recommendations, ensuring a diverse pool with relevant expertise. Selected participants possessed expertise in system development and involvement in Agile implementation. Interviews, lasting approximately 30-45 minutes, were conducted in-person and virtually via video conferencing, accommodating diverse geographical locations. Open-ended questions were used to elicit detailed responses, and each session was transcribed with participant consent. Transcribed interviews were analyzed thematically using spreadsheets. An inductive coding approach was used to identify recurring patterns, followed by axial coding to establish connections between categories.

6.1.3 Ethical Considerations

All participants provided informed consent before participation, with assurances of confidentiality and anonymity. Data were securely stored, and any identifiable information was removed during transcription. The research adhered to ethical guidelines outlined by the CSU Institutional Review Board (IRB).

6.1.4 Data Analysis

This study employed a mixed-methods approach to analyze the extent of Agile adoption of LS/SC/CP systems, the challenges encountered, and the adaptations made to address these challenges. The data sources include Survey responses from industry professionals and follow-up interviews with selected participants to provide deeper insights into the Survey findings. The analysis follows a structured process, integrating quantitative and qualitative methods to triangulate findings.

Survey Data Analysis

The Survey data was prepared by removing incomplete responses and standardizing categorical data. Quantitative responses were analyzed using descriptive statistics such as mean, median, standard deviation, and frequency distributions to understand the overall trends in Agile adoption. For qualitative Survey responses, thematic analysis was employed. Open coding was used to identify initial themes related to Agile challenges and adaptations. Axial coding was then applied to categorize these themes into broader challenge areas (e.g., regulatory compliance, safety assurance, integration complexity) and adaptation strategies (e.g., model-based systems engineering, digital twins). This thematic structure informed the development of follow-up interview questions.

Interview Data Analysis

The interview data were transcribed to ensure participant privacy and coded using Excel. Open coding identified emerging patterns in how Agile is applied within large-scale, safety-critical, and cyber-physical (LS/SC/CP) systems, as well as the challenges experienced and adjustments teams

make to Agile frameworks. Axial coding grouped responses into overarching categories, aligning with survey findings and ensuring consistency across data sources. A selective coding phase refined these categories into core categories, such as tailored Agile methodologies, hybrid approaches, and domain-specific Agile adaptations.

Integration of quantitative and qualitative analysis

A triangulation approach was used to integrate the quantitative and qualitative findings. Key statistical trends from the survey, such as mean adoption rates and frequency of reported challenges, were cross-referenced with interview data to validate and deepen the understanding of Agile adoption challenges and adaptations. This mixed-methods integration provided a comprehensive analysis, capturing numerical trends and practitioner experiences. The results from both studies are presented in the following section, highlighting the current state of Agile adoption in LS/SC/CP systems, the primary challenges, and innovative strategies to overcome them.

6.1.5 Threats to Validity

There are several threats to validity outlined below:

- **Selection Bias:** Participant recruitment through LinkedIn networks may have skewed domain representation. In addition, the author is a SAFe Fellow, which impacts the recruitment based on the existing social network.
- **Response Bias:** Response bias may arise in how the question's wording influences respondents' answers. Researchers attempted to mitigate this by using neutral language and piloting the survey.
- **Response Bias:** Response bias may arise in how the question's wording influences respondents' answers. Researchers attempted to mitigate this by using neutral language and piloting the survey.

- **Construct Validity:** Thematic coding of interview responses relies on subjective interpretation, potentially introducing researcher bias. An inductive coding approach was applied to mitigate this, followed by axial coding to ensure consistency.
- **External Validity:** The sample size and industry representation constrain external validity.
- **Triangulation:** Triangulation of quantitative and qualitative data strengthens validity through cross-referencing trends with practitioner insights. However, the mixed-methods approach has inherent limitations, such as potential misalignment between survey responses and interview interpretations. Researchers attempted to minimize this by carefully aligning interview questions with survey themes.

6.2 Results / Analysis

The survey, conducted over eight weeks, and subsequent interviews revealed a notable demographic skew. Regarding industry representation, the survey and interview data heavily favored Aerospace, with 66% and 72%, respectively. At the same time, other sectors like Services, Manufacturing, Automotive, Energy, and Medical were represented to a lesser extent. Regarding roles, there was a strong skew towards Executives and Coaches, potentially over-representing high-level strategic and advisory perspectives. Specifically, 30% of survey respondents were Executives, and 25% were Coaches, while 56% of interviewees were Executives and 33% were Coaches. Additionally, a significant portion of both survey respondents and interviewees possessed extensive industry experience, with nearly 50% and 68% respectively having over 21 years of experience. Overall, the data is heavily weighted towards executive leaders in Aerospace with extensive industry experience.

6.2.1 To what extent is Agile being applied to LS/SC/CP Systems

As shown in Figure 6.1, 74% of respondents (41 of 56) apply Agile to Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems sometimes or often, while 16% always apply Agile, and 11% rarely or never utilize Agile. This reflects a growing interest in and experimentation with

Agile in this domain. These findings suggest that Agile adoption is not yet ubiquitous in LS/SC/CP systems. Further research is needed to explore the factors driving and hindering Agile adoption in this context.

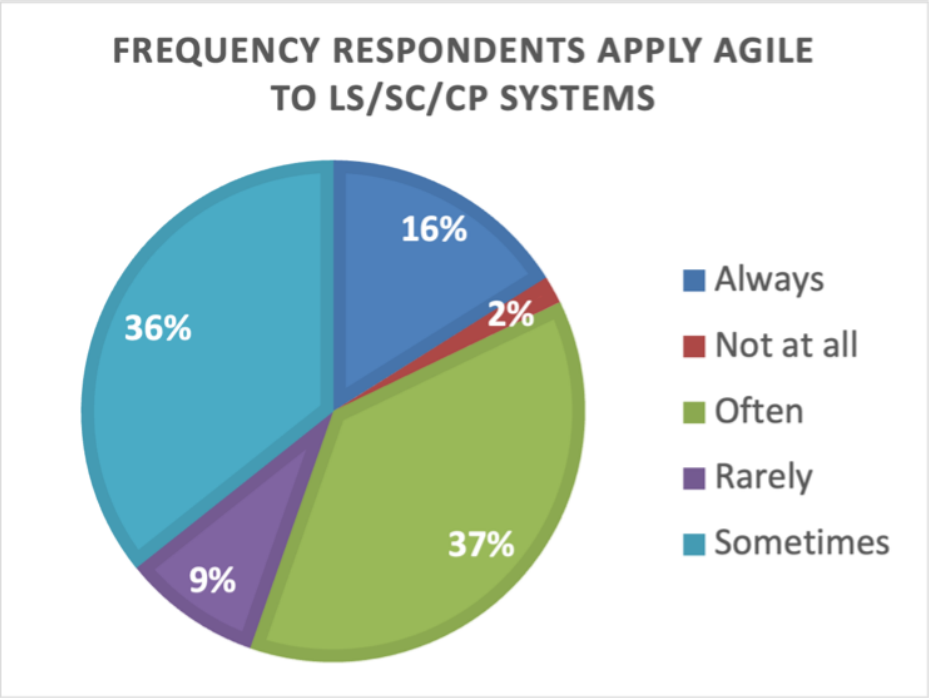


Figure 6.1: Extent Agile applied to LS/SC/CP Systems

Interviewees reported even higher adoption rates than survey respondents, with 56% using Agile on at least 75% of their programs. Interviewee (I09, Program Manager), who noted the priority in the government space to do more with less, is a key driver for jumping in with both feet. 89% of the interviewees were executives and coaches, who are often early adopters of new approaches; this finding may not represent the broader population. If we had a fuller picture of more junior representatives, we might see that the overall depth of Agile adoption across organizations is less mature. This potential disconnect between leadership and other roles highlights the importance of considering the perspectives of all team members when assessing Agile adoption.

According to the survey results shown in Figure 6.2, 41% of the respondents consider Agile integral to their program management, representing a significant industry shift. Previously, many

only considered Agile about software development as opposed to how to manage the program at the system level. This shift may indicate a growing recognition of the benefits of Agile principles for managing complex projects beyond software development. However, it is essential to understand the factors driving this change in perception and whether it translates to effective Agile implementation across all levels of the organization.

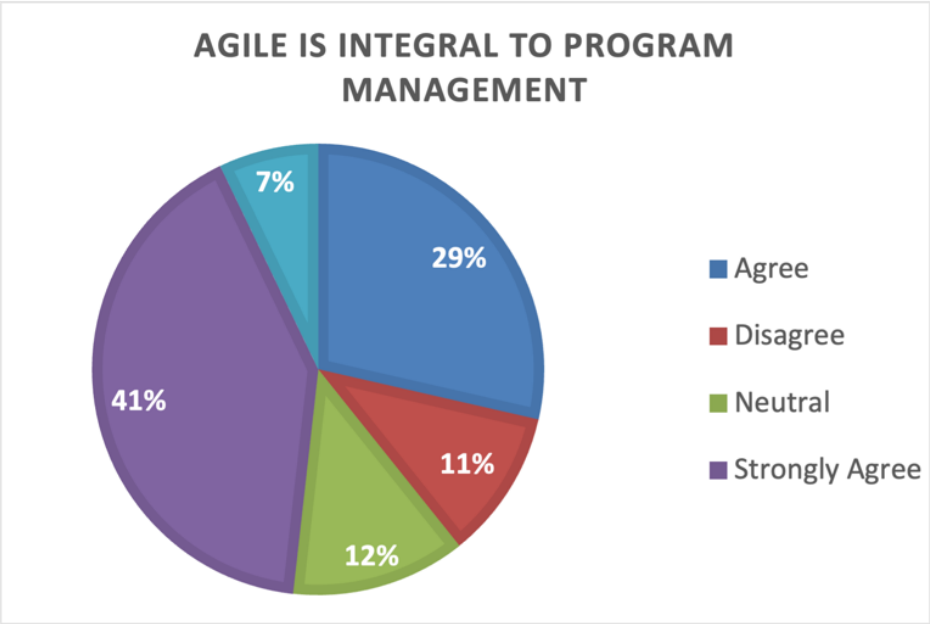


Figure 6.2: Percent who believe Agile is integral to Program management

The survey and interviews revealed that the Scaled Agile Framework (SAFe) was the most commonly used Agile scaling framework, with 71% of survey respondents and 52% of interviewees reporting its use. This may be due to SAFe's comprehensive guidance and support for large organizations. Interviewees I02, I03, I05, I07, I09, I10, I11, I12, I15, and I23 leveraged multiple frameworks, potentially indicating a greater level of Agile maturity and a recognition that no single framework may address all challenges. I15 stated that the "context of the problem determined which practices to use, which frequently crossed frameworks". This suggests that these highly regulated domains need a tailored, context-specific approach. Organizations may need to adapt and

combine elements from different frameworks to address their unique challenges and effectively achieve successful Agile adoption.

6.2.2 Challenges in applying Agile to LS/SC/CP Systems

Survey Results

The survey results highlighted multiple challenges applying Agile to Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems. The most frequently cited challenge was integrating Agile with existing processes (31%), followed by organizational/team dynamics (24%), regulatory compliance challenges (19%), hardware constraints (17%), and safety challenges (9%). These findings demonstrate the complexities of adopting Agile in regulated environments with complex systems. Further research is needed to explore strategies for overcoming these challenges and successfully implementing Agile in LS/SC/CP development.

Interview Results

The interviews validated challenges identified in the literature review and survey, with integrating Agile into existing processes being the most cited (43%). This difficulty stemmed from external agencies and regulatory bodies not being structured for Agile's iterative nature. I10 said that "Agencies are not funded to perform incrementally". Organizational/team dynamics (25%) also presented hurdles, particularly resistance to change and communication barriers across silos. I08 stated, "The White blood cells in their organization were constantly trying to wipe Agile out". Hardware constraints (18%) were another concern, with interviewees I13 and I17 describing their procurement process impacting completing work in sprints or program increments. I23 stated that they needed to develop practical solutions, such as modeling or cardboard prototypes, to balance hardware lead times with iterative development. While less prevalent, regulatory compliance (8%) and safety concerns (6%) were still noteworthy. Regulatory bodies often expect fully documented, preplanned validation, which clashes with Agile's iterative approach. I14 stated that Agile does not provide enough up-front documentation to meet Automotive Spice (ASPICE) process requirements. The safety concerns revealed in the interviews centered on the perceived risks of Agile

iterations in high-assurance environments and a lack of guidance on ensuring safety while implementing Agile. I02 stated that their leadership assumed that experimentation meant a safety risk. These findings highlight the need for adapted Agile methodologies, increased collaboration, and evolved regulatory frameworks to support iterative development in LS/SC/CP systems.

A comparison of the survey and interview results is illustrated in 6.3

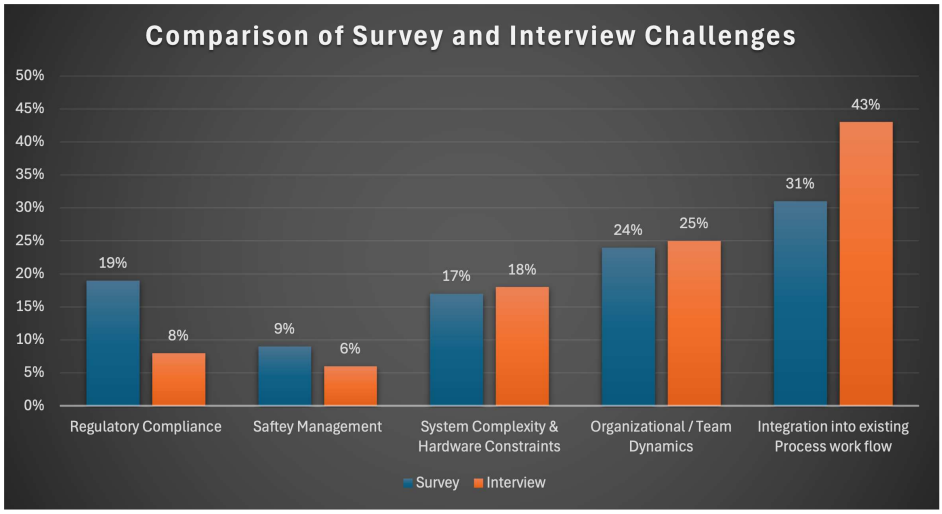


Figure 6.3: Comparing Survey to Interview Results regarding Challenges

6.2.3 Adaptations seen in applying Agile to LS/SC/CP

Survey

The survey revealed that respondents adapted Agile practices to meet their unique needs. The most reported adaptation cited by 27% of respondents was the use of Modeling/Simulation/Digital Twins. Additionally, 26% incorporated specialized artifacts, such as bidirectional traceability matrices. Program management enhancements were reported by 25%, while 12% integrated hazard analysis and system-theoretic Process Analysis (STPA). Finally, 10% utilized safety stories to address safety considerations proactively.

Interviews

The interviews revealed similar adaptations to the survey findings, though with slight differences in wording. For example, while the survey referenced Modeling/Simulation/Digital Twins, the interviews focused on Model-Based Systems Engineering (MBSE), with 23% of the interviewees emphasizing using models and simulators to visualize work and gather feedback. I10 described how modeling helps to identify safety requirements early. 10% of the interviews highlighted bi-directional traceability matrices as essential artifacts for demonstrating regulatory compliance. I16 stated that modeling was the only way they could manage the impact of change because of the complexity of the legacy system. Furthermore, 54% of the interviews emphasized improved program management and organizational structure updates, a significantly higher percentage than revealed in the survey. I07 stressed the importance of granting program managers independence to give them greater control over delivery. 13% of the interviews discussed using safety stories to ensure safety needs were met. I20 described team mobbing with compliance subject matter experts to streamline regulatory processes. The interviews also revealed additional themes not identified in the survey. One key finding was the shift in planning and review approaches; rather than conducting a single Preliminary Design Review (PDR), some programs adopted incremental review processes. I03, I11, I12, and I25 discussed that the iterative reviews increased the usable design feedback. In summary, the interviews reinforced several themes identified in the survey, revealing a greater emphasis on program management, regulatory compliance, and evolving review processes. A comparison of the survey and interview results is illustrated in 6.4

6.3 Discussion

The research on applying Agile methodologies to LS/SC/CP systems reveals a complex landscape characterized by enthusiasm for Agile practices and significant challenges in their implementation. The quantitative data collected through surveys and qualitative insights from structured interviews provide a detailed understanding of the current state of Agile adoption in these contexts. The findings indicate a substantial interest in Agile methodologies, with 74% of survey respondents

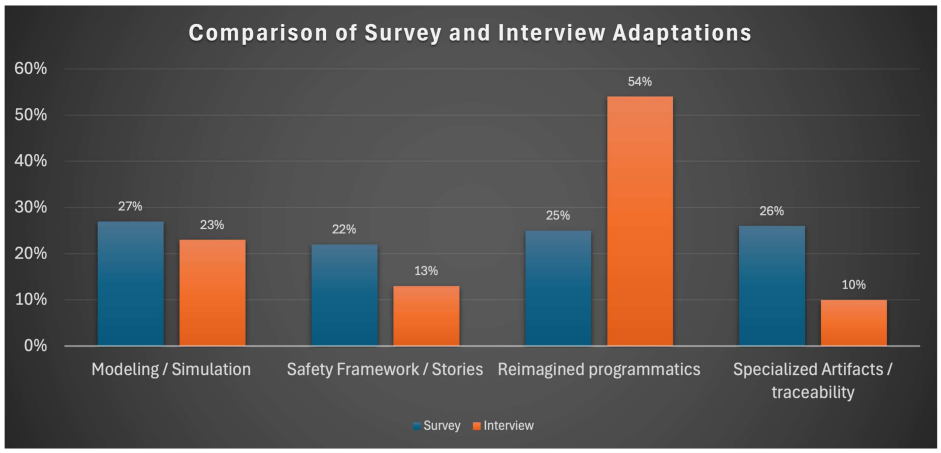


Figure 6.4: Comparing Survey to Interview Results regarding Adaptations

utilizing Agile either sometimes or often. Furthermore, 41% consider Agile integral to program management, suggesting a growing recognition of its potential benefits in enhancing project flexibility and responsiveness [94]. However, the skew towards responses from executives and Agile coaches may lead to an over-representation of early adopters and a lack of insight into lower-level practitioners' hands-on challenges. This bias highlights the need for further investigation into the experiences of technical teams who may encounter different obstacles in Agile adoption. The challenges in applying Agile to LS/SC/CP systems are multifaceted. Many organizations struggle with the inconsistent application of Agile practices, hindering their effectiveness. As noted by Laanti, the more extended experience with Agile methods positively influences perceptions of their usefulness, indicating that organizations may need time and support to fully integrate Agile into their operations [95]. Additionally, the need for tailored approaches that consider the unique demands of safety-critical environments is crucial, as generic Agile practices may not adequately address the complexities involved [96]. Organizations are increasingly adopting hybrid approaches that combine various Agile frameworks to overcome these challenges. The Integration of Agile practices into existing processes, as discussed by Wang, is essential for achieving consistent infusion of Agile methodologies into organizational culture [97]. This suggests that organizations should focus on developing a comprehensive understanding of Agile principles and practices, allowing them to adapt and refine their approaches based on specific project needs and contexts [98]. The qualitative

data from interviews reveal that while there is a strong endorsement of Agile methods, the practical implementation has shown gaps in understanding and execution. This discrepancy demonstrates the importance of fostering a culture of continuous improvement and learning within organizations, as highlighted by the need for regular feedback loops and collaborative practices [94].

6.3.1 To what extent are Agile methods being applied to LS/SC/CP Systems?

The findings from the survey and interviews regarding the application of Agile methods in LS/SC/CP systems reveal significant implications for the adoption, approach, maturity, and results of Agile methodologies in these contexts. Illustrated in table 6.1, the reported high levels of Agile adoption (54% from the survey and 56% from interviews) indicate a strong recognition of Agile practices within these environments, aligning with the broader trend of organizations increasingly embracing Agile frameworks to enhance flexibility and responsiveness [44]. However, the notable divergence in moderate and low adoption rates between the survey and interviews suggests that while many organizations are successfully implementing Agile, a substantial number are still grappling with its application in their specific contexts [84] [12].

Table 6.1: Agile adoption for LS/SC/CP systems

Adoption Level	**Survey** (To what extent are you applying Agile?)	**Interview** (Percentage of your programs are using Agile)
High Adoption	54%	56%
Moderate Adoption	36%	16%
Low Adoption	11%	28%

The predominance of frameworks such as the Scaled Agile Framework (SAFe), which is reported as the most widely adopted (75% in surveys and 52% in interviews), demonstrates the structured governance and compliance mechanisms that these frameworks provide, which are crucial for managing the complexities of safety-critical environments [86] [99] [30]. The interviews

revealed that 40% of participants utilize multiple frameworks, further suggesting a growing maturity in Agile practices, as organizations recognize the need for hybrid approaches to effectively navigate the intricacies of large-scale implementations [100] [101]. This aligns with findings highlighting the importance of tailoring Agile methods to fit organizational needs, particularly in complex environments [102] [103]. The perception of Agile as integral to program management, with 70% of respondents affirming its importance, indicates a shift towards embedding Agile principles throughout organizational operations rather than confining them to software development [104]. This holistic Integration fosters a more adaptive and responsive approach to project management, essential for navigating the challenges posed by LS/SC/CP systems [30] [105]. The strong endorsement of Agile practices, reflected in high median scores (79) and a mean of 70.4, suggests that organizations that view Agile as a core component of their operations are more likely to achieve successful outcomes in complex project environments [106] [45].

In summary, the implications of these findings highlight a dual narrative. While there is a robust movement towards Agile adoption in LS/SC/CP systems, significant challenges remain that organizations must address to realize the full benefits of Agile methodologies. The variance in adoption levels indicates that many organizations are still in the early stages of their Agile journey, necessitating a focus on developing a coherent Agile culture and mindset to support successful transformations [107] [12]. As organizations continue to explore and implement Agile frameworks, the need for tailored approaches that consider the unique demands of safety-critical systems will be paramount for achieving sustained success.

6.3.2 What are the current challenges being experienced in this domain?

There are significant challenges in applying Agile to LS/SC/CP systems, first identified by Ovesen in an analysis of multiple Danish companies that develop physical products [47]. Ovensen categorized these challenges into five key groups: constraints of physicality, paradigm perplexity, designer's dissent, team distribution dilemma, and education. In 2019, Atzberger reassessed these challenges to determine their persistence, identifying constraints of physicality, mindset, scaling,

and team distribution as the most pressing issues [108] More recently, a systematic literature review highlighted additional key challenges, including organizational and team dynamics, process and workflow limitations, regulatory compliance, safety management, and system complexity [90]. To explore these challenges further, we conducted a survey that presented respondents with five multiple-choice options: hardware constraints, Integration with existing processes, resistance to change, regulatory compliance, and safety concerns. Initially, we expected hardware constraints, regulatory compliance, and safety to be the most significant barriers to Agile adoption. However, the results showed that resistance to change (24%) and Integration with current processes (31%) together accounted for 55% of the reported challenges, outweighing the combined impact of hardware constraints, regulatory compliance, and safety. This finding suggests that organizational resistance to change is the primary obstacle to Agile implementation. To gain deeper insights, we conducted open-ended interviews regarding Agile adoption challenges. Notably, fewer than 27% of interviewees cited hardware, regulatory compliance, or safety as significant issues. Instead, 73% of responses emphasized resistance to change and a lack of Agile understanding as the primary difficulties. One interviewee likened the resistance to an immune system response: "The white blood cells are trying to kick out Agile methods constantly." These findings align with Kumar's study, which also identified resistance to change and Integration with existing processes as the dominant challenges [109]. In follow-up interviews, we examined the impact of these challenges, as illustrated in Figure ref fig: Impacts; the most frequently reported consequence was schedule delays, followed by increased costs. The research results indicate that Agile adoption in LS/SC/CP environments is less about overcoming physical or regulatory constraints and more about addressing human and organizational factors. Companies working for a successful Agile implementation should prioritize cultural change, process alignment, and education.

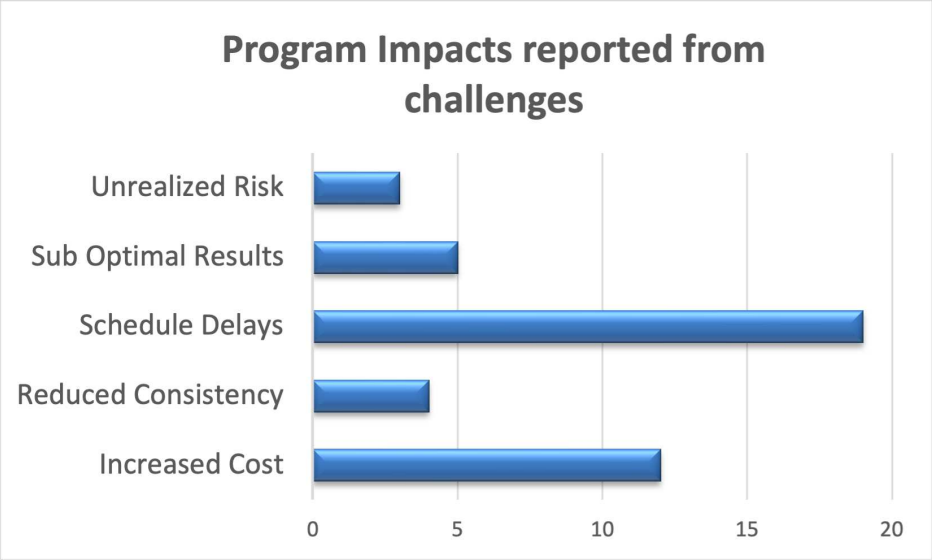


Figure 6.5: Impacts to programs seen by respondents

6.3.3 What adaptations are being made to Agile to overcome challenges?

The survey results and interviews regarding adaptations in large-scale, safety-critical, cyber-physical systems reveal significant insights into the evolving practices within Agile methodologies. The survey indicated that 27% of respondents are utilizing Modeling, Simulation, and Digital Twins, which align with the principles of Model-Based Systems Engineering (MBSE). This approach allows teams to visualize complex systems and gather feedback effectively, enhancing safety and compliance by better understanding and representing system behaviors. The emphasis on MBSE is crucial as it facilitates integrating safety considerations early in the design process, which is essential for safety-critical systems [110]. Furthermore, the incorporation of specialized artifacts such as traceability matrices, reported by 26% of respondents, shows the importance of maintaining regulatory compliance and ensuring that safety requirements are met throughout the development lifecycle. The interviews highlighted the necessity of bi-directional traceability matrices, which are vital for demonstrating compliance with safety regulations and facilitating stakeholder communication. This aligns with findings from previous studies that emphasize the role of structured documentation in enhancing safety management practices [111]. MBSE can be a so-

lution for supporting documentation and traceability [112]. Program management enhancements, reported by 25% of survey participants, reflect a broader trend towards improving organizational structures to support Agile practices in safety-critical environments. The interviews revealed a strong consensus on the need for program managers to have greater autonomy, which can lead to more effective decision-making and delivery outcomes. Additionally, embedding compliance engineers within teams was identified as a strategy to streamline regulatory processes, thereby reducing bureaucratic overhead and fostering a safety culture. The Integration of hazard analysis techniques, such as Failure Modes, Effects, and Analysis (FMEA) and System-Theoretic Process Analysis (STPA), reported by 12% of respondents, highlights the proactive measures being taken to identify and mitigate risks associated with system failures. These methodologies are critical in safety-critical domains where understanding potential failure modes can significantly enhance system reliability and safety. The interviews further revealed a shift towards incremental reviews rather than traditional Preliminary Design Reviews (PDR), indicating a more adaptive approach to project management that allows for continuous improvement and responsiveness to emerging safety concerns. Lastly, the call for increased investment in digital tools and automation, emphasized by many interviewees, reflects a recognition of the need for advanced technologies to meet regulatory compliance and safety requirements effectively. This aligns with the broader industry trend towards leveraging automation and real-time data monitoring to enhance safety management and reduce human error. The findings suggest that organizations are increasingly aware of integrating technological advancements into their safety practices to foster a more efficient and compliant operational environment [113].

6.4 Conclusion

This study provides empirical evidence of Agile adoption in Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems, shedding light on the extent of implementation, key challenges, and adaptations necessary for success. The findings indicate that while Agile is increasingly recognized as valuable in these environments, its adoption remains inconsistent due to industry-specific con-

straints, regulatory requirements, and organizational resistance to change. The research highlights that the most significant challenges are not technical but often stem from culture. Organizations are adapting Agile with techniques such as Model-Based Systems Engineering (MBSE), digital twins, living traceability matrices, and hybrid governance models to address these challenges. These adaptations enable teams to balance the iterative nature of Agile with the need for regulatory compliance and safety assurance.

Furthermore, the study reinforces the necessity of leadership buy-in, stakeholder collaboration, and investment in automation to facilitate Agile adoption in LS/SC/CP domains. This research suggests that Agile methodologies can enhance efficiency, improve collaboration, and accelerate delivery timelines in safety-critical environments when tailored appropriately. However, achieving widespread adoption requires a shift in organizational mindset, establishing clear compliance strategies, and integrating Agile principles into system engineering practices. Future research should analyze the sociotechnical factors impacting Agile adoption and performance, including organizational culture, team dynamics, and technological infrastructure. In addition, evaluate integrating advanced technologies, developing continuous safety certification processes, and establishing standardized frameworks. As the field evolves, further empirical studies and theoretical explorations will be essential to refine these methodologies and address the unique challenges of safety-critical environments.

Chapter 7

Satellite Case Study

What is the impact of applying Agile to large-scale, safety-critical cyber-physical systems? Do adaptations resolve challenges?

7.1 Introduction

This chapter presents a comparative analysis of two distinct methodologies for satellite system development: NASA's Waterfall approach and an Agile approach. By modeling the development of a hypothetical satellite using both methods, we aim to provide a comprehensive understanding of their respective impacts on development timelines, costs, and risk mitigation. This analysis is facilitated by using Lifecycle Modeling Language (LML), which enables a structured and visual comparison of the two approaches. This chapter's findings will answer the research question: "What is the impact of applying Agile to large-scale, safety-critical cyber-physical systems, and do adaptations effectively resolve the associated challenges?".

7.1.1 Why a Satellite

The timelines for satellite development and deployment vary greatly. NASA's James Webb Space Telescope took about 25 years from concept to launch, while SpaceX can build and deploy a Starlink satellite in under a year and currently is [114]. Military satellites often take 5-10 years to develop and deploy. Many government-built satellites tend to have schedule delays and cost overruns. For example, costs for the Advanced Extremely High Frequency (AEHF) satellite program grew by 118 percent, and its first Satellite was launched more than 3.5 years late. Costs for the Space Based Infrared System (SBIRS) grew nearly 300 percent, and its scheduled launch was delayed roughly 9 years [115]. Currently, The Space Command and Control (Space C2)is also reporting persistent delays [116]

7.2 Lifecycle Modeling

Lifecycle modeling is a structured approach to visualizing, analyzing, and managing the development, deployment, operation, and retirement of complex systems [117]. It enables engineers and project managers to model the entire system lifecycle using standardized methodologies such as SysML, LML, and UAF, ensuring alignment with industry standards. To effectively compare the Waterfall and Agile lifecycles. Lifecycle modeling in Innoslate is valuable for our fictional case study because it provides an integrated modeling environment capable of clearly visualizing, simulating, and analyzing differences in these methodologies. LML covers the entire system's lifecycle, from conceptual development to disposal. The approach involved creating detailed activity and action diagrams, running simulations, and evaluating outcomes to objectively determine the effectiveness and suitability of Agile versus Waterfall for satellite development. Innoslate is a cloud-based and on-premises platform that supports requirements management, modeling and simulation, verification and validation, risk analysis, and collaboration within a single digital environment.

7.3 Satellite Example

The mid-size Low-Earth Orbit (LEO) Satellite under development is designed to provide high-resolution Earth imagery and weather monitoring capabilities. With a launch mass of 250 kg, this Satellite is equipped with a 1 kW solar array to support its operational power needs. It also features dual-band communication via S-band (125 Kbps uplink, 2 Mbps downlink) and X-band (650 Mbps downlink) for efficient data transmission.

The Satellite's mission objectives focus on capturing Earth imagery for environmental monitoring, disaster response, and resource management while supporting weather observation and atmospheric data collection. The spacecraft is designed to operate in LEO, optimizing its orbital characteristics for frequent revisit times and continuous global coverage. This Satellite aims to deliver critical data to researchers, meteorologists, and government agencies by leveraging advanced

sensor payloads and high-speed communication links, contributing to improved forecasting, climate studies, and geospatial intelligence.

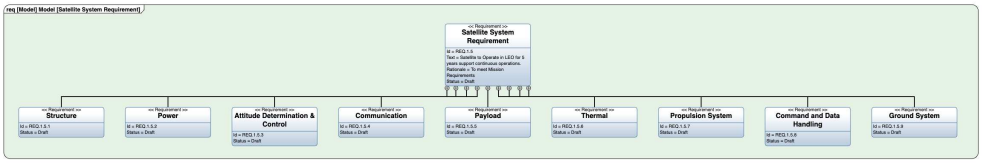


Figure 7.1: Satellite Requirements Diagram

7.3.1 Establishing Inputs and Outputs for Both Models

Each model started with a detailed breakdown of each subsystem's inputs (components, requirements) and outputs (verified functionality). The inputs described in Table 7.1 were identical for both the Agile and Waterfall models to maintain consistency:

Table 7.1: Modeled Subsystems

	Subsystem	**Inputs**	**Outputs**
1.	Structure	Primary & Secondary Structures	Verified Structural Integrity
2.	Power	Battery, Solar Arrays	Power Distribution Verified
3.	Attitude Determination and Control	Reaction Wheels, Star Trackers, Software	Attitude accuracy verified
4.	Communications	Transmitters, Receivers, Antennae	Reliable communication link established
5.	Payload	Scientific Instruments, Payload Specifications	Data collection and processing operational
6.	Thermal Control	Radiators, Heaters, Insulation, sensors	Thermal Controls Verified
7.	Propulsion	Thrusters, Fuel Thanks, Piping	Basic maneuver capability established
8.	Command & Data handling	Onboard Computer, Software, Sensors	Command/Data Handling Verified

7.3.2 Subsystems

Developing a mid-size Low-Earth Orbit (LEO) satellite requires integrating multiple interdependent subsystems, each critical to mission success. These subsystems work together to provide structural integrity, power generation, attitude control, communication, payload operation, thermal regulation, propulsion, and command and data handling.

Structure

The structural subsystem is the Satellite's backbone, providing mechanical support and protection for all internal components. It is designed to withstand the stresses of launch, the space environment, and on-orbit operations. The primary and secondary structures are lightweight yet durable materials, such as aluminum alloys and composite materials, ensuring rigidity and strength while minimizing mass. The structure also houses the payload and ensures proper alignment of sensors and antennas.

Power

The power subsystem generates, stores, and distributes electrical power to all onboard systems. The Satellite has a 1 kW solar array, which collects and converts solar energy into electrical power. Lithium-ion batteries store excess energy during eclipse periods when the Satellite is not exposed to sunlight. A Power Distribution Unit (PDU) regulates and distributes power efficiently, ensuring uninterrupted operation of critical subsystems.

Attitude Determination and Control System (ADCS)

The ADCS ensures the Satellite's precise orientation and stability to maintain proper pointing for imaging, communication, and orbital maneuvers. The system includes reaction wheels, magnetometers, gyroscopes, and star trackers for attitude sensing and control. Magnetorquers or thrusters may be used for momentum management and stabilization after deployment. The ADCS enables accurate positioning for Earth observation and data transmission, ensuring optimal performance of the payload and antennas.

Communication

The communication subsystem provides command, telemetry, and data transmission capabilities between the Satellite and ground stations. It operates in S-band (125 Kbps uplink, 2 Mbps downlink) for telemetry, tracking, and control (TT&C) and X-band (650 Mbps downlink) for high-data-rate payload transmission. The subsystem consists of high-gain and low-gain antennas and

software-defined radios (SDRs) for efficient frequency modulation and adaptability to mission requirements.

Payload

The payload subsystem comprises high-resolution imaging sensors and weather monitoring instruments designed for Earth observation. The imaging system captures multispectral and thermal imagery for environmental monitoring, disaster response, and resource management. Weather instruments collect atmospheric data, cloud cover, and temperature variations, contributing to meteorological forecasting and climate studies. The payload is optimized for high spatial and temporal resolution to maximize scientific and operational benefits.

Thermal Control

The thermal control subsystem ensures that all components operate within their required temperature ranges in the extreme space environment. It includes passive thermal elements such as radiators, multi-layer insulation (MLI), coatings, and active thermal management using heaters and heat pipes. The system prevents electronic components from overheating and ensures that the payload, batteries, and communication systems function reliably across day-night temperature cycles in LEO.

Propulsion

The propulsion subsystem provides orbital maneuvering, attitude corrections, and station-keeping capabilities. It consists of thrusters, fuel tanks, piping, and valves for controlled thrust generation. The propulsion system supports collision avoidance maneuvers, deorbiting, and precise station adjustments, extending the Satellite's operational lifetime and ensuring compliance with space debris mitigation guidelines.

Command & Data Handling

The C&DH subsystem acts as the Satellite's central processing unit, managing data flow between subsystems and executing mission operations. It includes an onboard computer, data storage

units, and fault-tolerant software. The system processes telemetry data, executes onboard autonomy algorithms and ensures real-time decision-making. It also interfaces with the ground control center, executing commands and coordinating data collection, storage, and transmission.

7.3.3 Assumptions made for this development

Overall Assumptions

Table 7.2 summarizes key assumptions underlying the satellite development models using both Waterfall and Agile methodologies. Multiple assumptions were made to simplify the modeling process and compare these two distinct approaches effectively. These assumptions focus on critical aspects such as requirements management, workflow structure, resource availability, integration, testing, compliance, and risk management. Clearly defining these boundaries ensures a consistent and fair comparison of each methodology within the context of satellite development.

Table 7.2: Modeling Assumptions

Category	**Waterfall Model Assumption**	**Agile Model Assumption**
Workflow	Follows NASA defined approach	Iterative and Incremental with Continuous Assurance Plugin (CAP)
Planning	Complete Integrated Master Schedule (IMS) is defined before work starts.	Multiple Horizons Roadmap with Years decomposed into Quarterly Increments into 2-week sprints.
Materials/Components	All required materials and components are available from the start and cause no delays.	All required materials and components are available from the start and cause no delays.
Labor / Skill Availability	Skilled workforce available	Skilled Workforce Available
Integration / Test	Access to test environments	CI/CD Pipeline
Regulatory Compliance and Safety	Validated at the Phase Gates	Automated and continuously validated at each sprint and Increment.
Material Cost	Fixed 5M	Fixed 5M
Labor Cost	$120 per hour	$120 per hour

Work Sizing Assumptions

We leveraged NASA's Cost Estimating Guide [118] to estimate the satellite build costs under both Waterfall and Agile models. Our approach combined analogy cost estimating with an en-

gineering build-up. We created work breakdown structures for each model for the engineering build-up, detailed in Appendices D and E. After comparing our estimates with subject matter experts and adjusting for their experience, we refined them to a rough order of magnitude.

7.4 NASA Development Approach

7.4.1 Description

The NASA Systems Engineering Handbook, initially published as SP-6105 in 1995, provided the foundation for the NASA Waterfall life cycle process [119]. NASA's process for developing air and ground systems follows a linear approach, segmented into distinct project life cycle phases. The life cycle begins with Pre-Phase A: Concept Studies, where ideas and feasible alternatives are generated and evaluated for cost, technical feasibility, and risk. This leads into Phase A: Concept and Technology Development, which refines mission concepts and validates requirements. Phase B focuses on preliminary design and establishing design-dependent requirements and interfaces, while Phase C finalizes the detailed design and prepares for manufacturing. During Phase D, the system is assembled, integrated, and rigorously tested to ensure operational readiness. In Phase E, the system transitions to operations and sustainment, where performance is maintained and necessary upgrades are made. Finally, Phase F addresses decommissioning, data archival, and disposal. For purposes of this paper, we were interested in phases A-D.

Table 7.3: NASA Phases

Phase	Purpose	Inputs	Description	Outputs	Reviews
A	Concept and Technology Development	Mission needs, feasibility studies	Define mission architecture, identify technology gaps	Concept Study Report, preliminary requirements	System Requirements Review (SRR)
B	Preliminary Design & Technology Completion	Concept studies, tech development results	Finalize architecture, complete risk analysis, technology maturation	Preliminary Design Review (PDR), risk reduction results	Preliminary Design Review (PDR)
C	Final Design and Fabrication	PDR results, matured requirements	Conduct detailed design, build and test components	Critical Design Review (CDR), manufactured components	Critical Design Review (CDR)
D	Assembly, Integration, and Test (AIT) & Launch	CDR results, fabricated components	Integrate subsystems, conduct testing, prepare for launch	Fully integrated system, Launch Readiness Review (LRR)	Test Readiness Review (TRR) Flight Readiness Review (FRR), Launch Readiness Review (LRR)

7.4.2 Systems Engineering Technical Reviews (SETRs)

Systems Engineering Technical Reviews (SETRs), illustrated in Table 7.4, are formal technical reviews conducted during the development of a system to ensure that the system meets its requirements and is on track to achieve its intended performance. These reviews are critical checkpoints in the systems engineering process and are used to assess the progress, quality, and technical integrity of the system being developed. SETRs help identify and mitigate risks, ensure the system design is feasible, and verify that it meets its specified requirements.

Objectives of SETRs

- Ensure technical rigor in system development
- Verify system requirements are met
- Identify and mitigate risks early
- Improve stakeholder alignment
- Support decision-making for system maturity

Regulatory and Safety Frameworks Referenced

- System Safety: MIL-STD-882E [120], ISO 26262 [121], IEC 61508 [122], NASA NPR 8715.3 [123]
- Software & Cybersecurity: DO-178C [124], DO-326A [125], NIST 800-53 [126], FedRAMP [127], ITAR [128]
- Electromagnetic & Communication Compliance: MIL-STD-461 [129], FCC regulations [130]
- Environmental & Health Safety: OSHA [131], ANSI [132], EPA [133], REACH [134]
- Aerospace & Space Regulations: FAA Part 450 [135], NASA-STD-8719.14 [136]

Table 7.4: System Engineering Technical Reviews

SETR	Description	Closure Criteria	Regulatory / Safety
System Requirements Review (SRR)	Ensures system requirements are well-defined, complete, and feasible.	All system requirements are clearly defined, traceable, and validated.	Safety constraints, mission assurance standards, environmental regulations, cybersecurity requirements (e.g., NIST 800-53, ITAR, EAR).
Preliminary Design Review (PDR)	evaluates the preliminary system design to confirm compliance requirements,	Preliminary design documentation is complete and demonstrated.	Initial hazard analysis, OSHA safety standards, NASA NPR 8715.3 (for space systems), DO-254 (hardware), DO-178C (software).
Critical Design Review (CDR)	Confirms the final detailed design meets all system requirements and is ready to implement,	Design is finalized, verification methods defined, regulatory assessments complete,	Hazard analysis complete. System-Theoretic Process Analysis, cyber resilience, EMI/EMC compliance (FCC, MIL-STD-461).

SETR	Description	Closure Criteria	Regulatory / Safety
Test Readiness Review (TRR)	Ensures the test plans, procedures, and facilities are complete before verification and validation activities begin	Test procedures are approved, safety certification of test setup, risk assessments are complete	Personnel safety procedures (OSHA, ANSI), hazardous material handling (EPA, REACH), software safety testing (DO-178C, DO-331).
System Verification Review (SVR)	Confirm that the system meets all requirements through test results and analysis.	Verification results demonstrate compliance with all requirements.	Validation of system safety (MIL-STD-882E, IEC 61508), electromagnetic interference (EMI) testing, cybersecurity penetration testing (NIST RMF, FIPS-140).
Flight Readiness Review (FRR)	Confirms that the system meets all requirements and is ready for launch or field use.	Final certifications are complete, all compliance documentation has been submitted, and emergency procedures have been validated.	FAA launch license (14 CFR Part 450), space debris mitigation (NASA-STD-8719.14), ITAR compliance.

SETR	Description	Closure Criteria	Regulatory / Safety
Operational Readiness Review (ORR)	Evaluate whether the system is ready for operational deployment.	system meets operational safety standards, human factors are assessed.	FAA licensing (for space systems), airworthiness certification (if applicable), software assurance compliance (ISO 26262, DO-178C Level A/B).

7.4.3 Model Setup

The NASA Waterfall process was modeled in Innoslate based on NASA's Systems Engineering Handbook (SP-2016-6105). The model illustrated in Figure 7.2 adhered to phased development with sequential stages and formal review gates at each phase.

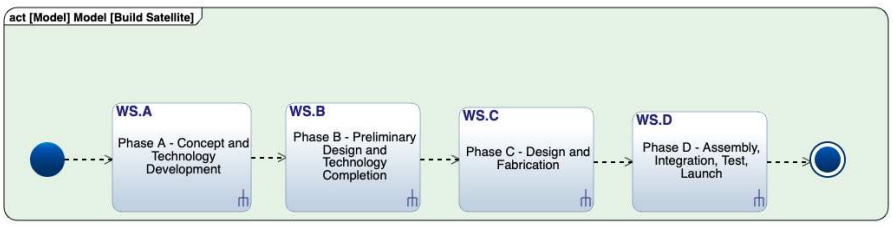


Figure 7.2: Model, Utilizing NASA Systems Engineering Handbook

Phase A: Concept and Technology Development

Phase A, illustrated in 7.3, focuses on defining the mission concept, assessing feasibility, and identifying key technologies required for satellite development. The waterfall model, in general, is

a sequential design process where progress flows steadily downwards through the phases of conception, initiation, analysis, design, construction, testing, production/implementation, and maintenance. However, at the close of Phase A, the customer will only have a series of documents without working capability. The goal of Phase A is to ensure that the project is technically, operationally, and financially viable before proceeding. However, this approach has not seemed to minimize overrun and schedule delays. The GAO has shown that their program cycle times are increasing by an average of 3 years from the planned date [137]. We decomposed the system using Innoslate into small steps and then estimated using a triangular distribution for each step. For example, updating the Concept of Operations is calculated as a minimum of 2 weeks, a maximum of 6 weeks, and 4 weeks as most likely. The approach allows us to get cost and schedule estimates. The modeled system illustrated in Figure 7.3 completed in 1.16 years, which aligns with what GAO reports regarding the time SBIRS Phase A took, which is between 12-18 months [137].

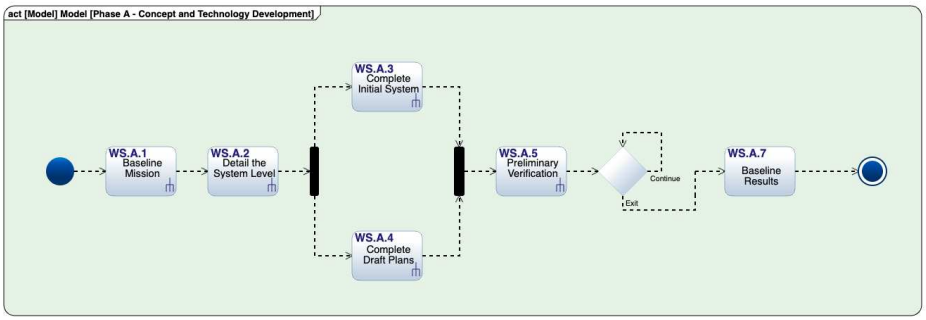


Figure 7.3: NASA Phase A: Concept and Technology Development

Phase B: Preliminary Design and Technology Completion

Phase 2, illustrated in Figure 7.4, the Preliminary Design and Technology Completion phase, is paramount to the success of space missions, serving as a critical bridge between the initial concept and final implementation. The mission design is rigorously refined during this phase, and key technologies are matured to minimize technical risks. Key activities include conducting a

Preliminary Design Review (PDR) to evaluate the system design and maturing critical technologies like advanced propulsion systems or communication arrays. Further activities involve planning system integration and testing, refining cost and schedule estimates, managing risks, and engaging stakeholders. The goal is to ensure the project is on track for successful implementation, within budget, and on schedule, ultimately paving the way for a successful mission.

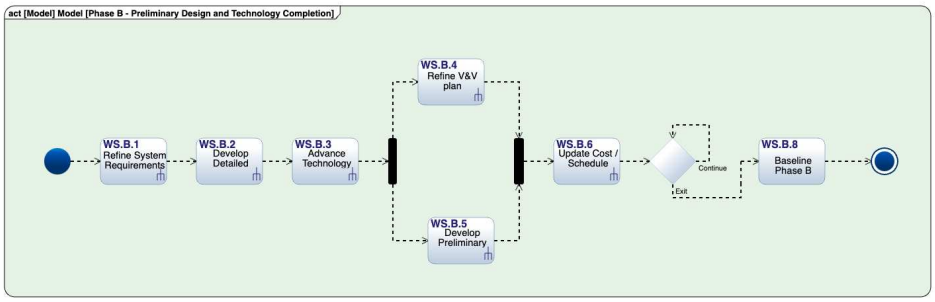


Figure 7.4: NASA Phase B

Phase C: Final Design and Fabrication

NASA's Phase C, illustrated in Figure 7.5, known as the Final Design and Fabrication phase, focuses on completing the detailed design of the system, fabricating and assembling components, and preparing for system integration and testing. Key activities include conducting a Critical Design Review (CDR) to ensure the design meets all mission requirements, finalizing detailed engineering drawings and specifications, and beginning the fabrication and assembly of system components. The project team also develops detailed plans for system integration and testing, continues to manage risks, and engages with stakeholders to keep them informed about the project's progress and any changes to the mission design or objectives.

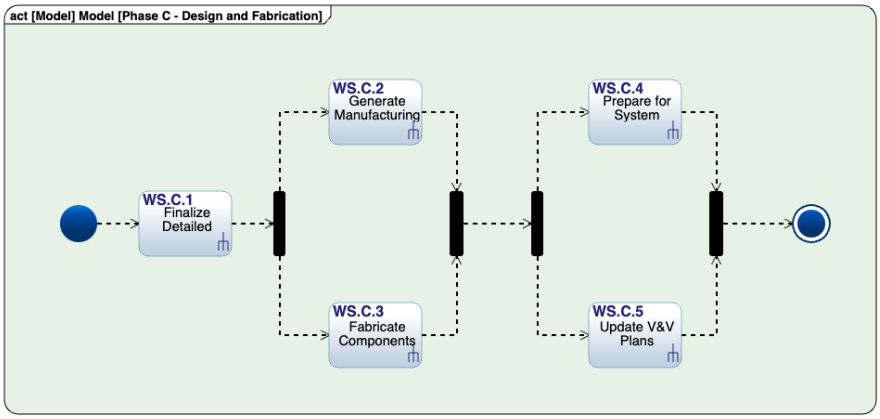


Figure 7.5: NASA Phase C

Phase D: Assembly, Integration, and Testing

Phase D, shown in Figure 7.6, known as the Assembly, Integration, and Testing (AIT) phase, focuses on assembling system components, integrating subsystems, and conducting comprehensive testing to ensure the system meets all mission requirements and is ready for deployment. Key activities include system assembly, integration, and extensive testing to verify performance, reliability, and safety. The project team conducts a Test Readiness Review (TRR) to confirm readiness for testing, prepares the system for operational deployment, and continues to manage risks. Ongoing stakeholder engagement ensures that all parties are informed about the project's progress and any changes associated with mission design or objectives.

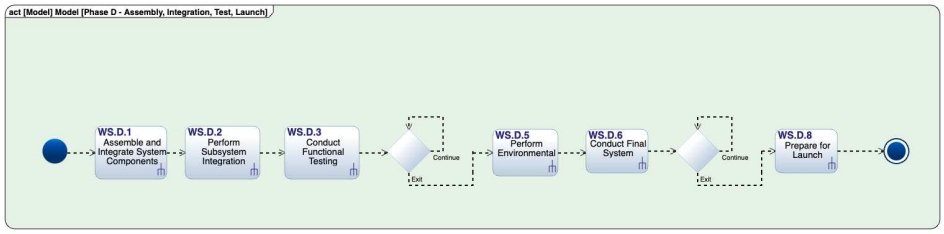


Figure 7.6: NASA Phase D

7.4.4 Analysis and Results

The Monte Carlo simulation in Innoslate provided key insights into the expected duration and labor cost for building a satellite, considering project timelines and resource expenditures variability. The analysis shown in Figure 7.7 revealed that the mean duration to build the Satellite is 5.89 years, with a standard deviation of 1.53 months. This indicates that the average completion time is relatively stable. This stability is due to the assumption that all materials and resources were readily available. This would exhibit much greater variability if supply chain integration were factored in.

In terms of cost, Labor cost was estimated at $7,858,335.14, representing the primary expenditure tracked in the analysis. We assumed the Agile and Waterfall approaches would yield a similar BoM due to the same hardware and components used in both development methodologies. This assumption allowed for a focused comparison of schedule efficiency and labor expenditures between the two methodologies. The findings highlight the expected resource commitment for satellite development, with potential applications in refining project scheduling and cost allocation strategies for future space missions.

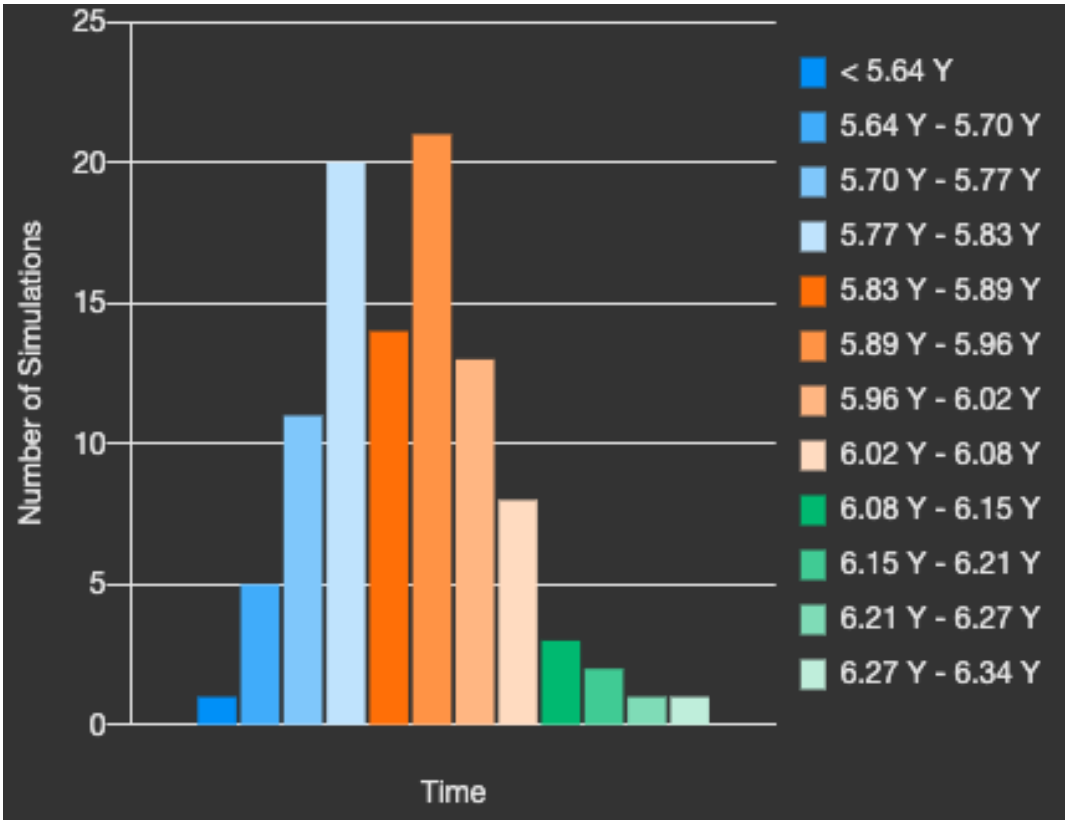


Figure 7.7: Monte-Carlo Analysis of Waterfall Development Process

7.5 Agile Approach

7.5.1 Description

Agile is an iterative and incremental approach to engineering characterized by iterative and incremental development, short feedback loops, continuous integration and verification, and adaptability to change within complex, evolving environments. The traditional Waterfall approach has been the norm for aerospace, but a growing community in Space is transitioning to Agile [138]. Some of these trailblazers include SpaceX [139], Planet Labs [140], Relativity [141]. For this paper, we took inspiration from organizations such as SpaceX [142]. We defined a hypothetical approach, the Continuous Assurance Plugin (CAP), that can support Agile Frameworks by adding specific guidance to support regulatory compliance, safety constraints, and integration complexity.

7.5.2 Continuous Assurance Plugin

The Continuous Assurance Plugin illustrated below in Figure 7.8 begins by leveraging SAFe's core framework. We decomposed the satellite system into a series of Minimum Viable Products (MVPs), Epics, Features, and Stories following core principles of decomposition, abstraction, encapsulation, well-defined interfaces, and independence. This resolved some well-documented challenges in Agile for Hardware: products are difficult to decompose into modules, and systems integration efforts are difficult to break down into small tasks [143].

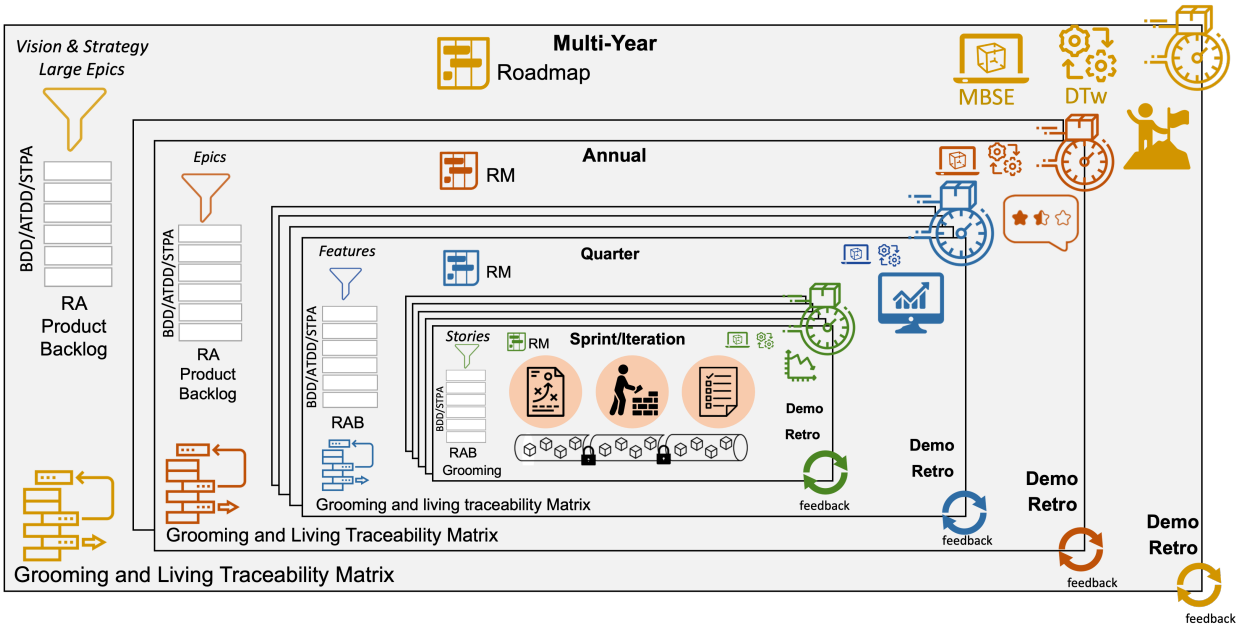


Figure 7.8: Building LS/SC/CP systems with CAP

An MVP is a concept popularized by Eric Ries. He defined MVP as "a product that allows a team to collect the maximum amount of validated learning about customers with the least effort" [144]. We created MVPs associated with each of the satellite subsystems. According to SAFe, Epics are a significant solution initiative that requires an MVP. Due to the size of the fictional case study, our epics and MVPs have a 1 to 1 relationship. Epics decompose into more minor features, which, by definition, need to be less than 12 weeks. Features decompose into stories that need less than a sprint length, typically 2 weeks. Although our Agile approach forgoes Phase-gated Systems

Engineering reviews, we must still meet regulatory compliance and safety assurance requirements through our continuous assurance plugin outlined in Table 7.5. Therefore, we are implementing a continuous assurance approach, contrasting with the traditional waterfall methodology, where assurance is tightly coupled to phase gates.

Table 7.5: CAP Feature / Benefits

Feature	**Description**	**Benefit**
Modular Architecture	Principles of decomposition, abstraction, encapsulation, well-defined interfaces, and independence.	Reduces dependencies across teams, reducing the impact of change.
MBSE	Model everything from requirements and design to verification and validation	Complete transparency of the system, allowing easier communication and reducing integration complexity
Digital Twin	Dynamic is an interactive model that mirrors the system's behavior and performance.	Real-time feedback on the impact when the system is updated. Allow us to explore options safely.
Boundary Objects	Artifacts, documents, terms, or concepts that serve as a point of reference and facilitate communication and understanding between different groups.	Create shared understanding for different teams to collaborate, enabling modularity.

Feature	Description	Benefit
Enabler Stories	Track safety and regulatory tasks in product backlogs (e.g., "As a system, I must comply with MIL-STD-882E for fault tolerance.")	Ensuring that we are building regulatory compliance, safety, and security into the system.
BDD/STPA integration	BDD focus on defining system behavior through user stories and scenarios. At the same time, STPA is a safety analysis technique that identifies potential hazards and ensures that safety constraints are met.	Write safety-focused scenarios that prevent hazards, including edge cases.
ATDD	Defining acceptance criteria and tests for regulatory and safety before development begins.	Ensures capabilities are not accepted unless they comply with functional and safety requirements.
Risk-adjusted backlog	Prioritized backlog that incorporates risk analysis.	Provides transparency into risk exposure in dollars, allowing prioritization of value and safety.
Living Traceability Matrix	Ensures that every requirement is traced to its corresponding design, implementation, and testing artifacts.	, provides transparency that improves change management, supports regulatory compliance, ensures quality assurance, and simplifies audits and reviews.

Feature	Description	Benefit
Test Automation	Place automated tests to continuously verify compliance and safety	Compliance activities are consistently integrated into the development process, reducing the risk of human error and improving overall system quality
CI/CD Pipeline	Use CI/CD pipelines that incorporate HIL and SIL to validate cybersecurity (DO-326A, NIST 800-53) and hardware reliability (ISO 26262)	By integrating SIL and HIL into the CI/CD pipeline, teams can ensure comprehensive testing and validation of the entire system.
Chaos Engineering	CI/CD pipeline that regularly injects failures into the system before they manifest in production.	Enhances the resilience and reliability of systems by intentionally introducing failures and learning from the system's response.
Digital Compliance Checklist	Checklist integrates compliance activities into the Agile workflow.	Provides real-time monitoring, validation, and documentation.
Iterative Reviews	a systematic approach to ensuring that safety and regulatory requirements are continuously met throughout the lifecycle.	Teams can identify and address potential issues early, reducing the risk to the overall safety and reliability of the system.

Feature	Description	Benefit
DoD	Define "Done" criteria to include safety verification and compliance checks for each timebox.	Teams ensure that these critical aspects are addressed consistently and thoroughly throughout the development process

7.5.3 Process versus System of Interest

We demonstrated our Continuous Assurance Plugin for research purposes using a specific system of interest. However, this plugin can be leveraged for any cyber-physical system.

7.5.4 Continuous Assurance Plugin in Action

To navigate the complexities of satellite development while ensuring safety and regulatory compliance, we've implemented a continuous assurance plugin alongside our MVP-driven approach. This allows us to integrate these critical aspects throughout the development lifecycle, moving beyond traditional waterfall phase gates. The following table details nine Minimum Viable Products (MVPs), each contributing to a fully functional satellite, with safety and compliance considerations embedded at every stage.

Table 7.6: Safety and Regulatory Agile Timeline

MVP	**Safety / Compliance Actions**	**Frameworks Validated**
Startup/ Initialization	• Incorporate Safety & Compliance into Risk Adjusted Backlog • Define incremental safety validation workflow • Set up regulatory checklists in Agile tools	• System Safety: MIL-STD-882E [120], NASA NPR 8715.3 [123] • Cybersecurity: NIST 800-53 [126], ITAR compliance tracking [128]
1- Basic Structure & Power System	• Perform Continuous structural risk assessments (MBSE for load/stress) • Automate material compliance tracking (e.g., REACH)	• System Safety: IEC 61508 [122], ISO 26262 [121] • Environmental & Health: REACH [134], OSHA [131], ANSI [132] • Aerospace: NASA-STD-8719.14 [136]

MVP	Safety / Compliance Actions	Frameworks Validated
2 - Command & Data Handling (C&DH)	• Implement early cyber compliance checks (DO-326A, NIST 800-53) • Automate software static analysis	• Software & Cybersecurity: DO-178C [124], DO-326A [125], NIST 800-53 [126], FedRAMP [127]
3 - Attitude Determination & Control	• Integrate real-time fault tolerance testing into Agile test pipelines • Validate software/hardware failure modes in digital twin	• System Safety: MIL-STD-882E [120], IEC 61508 [122] • Cybersecurity: DO-326A [125], ITAR [128]

MVP	Safety / Compliance Actions	Frameworks Validated
4 - Propulsion System	• Perform continuous hazardous material tracking • Automate ITAR compliance for propulsion components	• System Safety: MIL-STD-882E [120] • Environmental & Health: EPA [133], OSHA [131] • Aerospace: FAA Part 450 [135]
5 - Communication System	• Embed EMI/EMC compliance verification within Agile sprints • Automate regulatory spectrum compliance (FCC, ITU)	• Electromagnetic Compliance: MIL-STD-461 [129], FCC [130] regulations • Cybersecurity: NIST 800-53 [126], ITAR [128]

MVP	Safety / Compliance Actions	Frameworks Validated
6 - Thermal System	• Integrate thermal risk modeling into MBSE simulations • Automate compliance with NASA-STD-8719.14	• System Safety: ISO 26262 [121], IEC 61508 [122] • Aerospace: NASA-STD-8719.14 [136]
7 - Payload	• Ensure payload-specific safety testing in sprint test cases • Continuous FAA payload integration compliance tracking	• System Safety: MIL-STD-882E [120], NASA NPR 8715.3 [123] • Aerospace: FAA Part 450 [135], ITAR [128]

MVP	Safety / Compliance Actions	Frameworks Validated
8 - Full System Integration	• Implement incremental safety audits per increment • Continuous traceability of safety requirements via MBSE	• System Safety: MIL-STD-882E [120], IEC 61508 [122] • Cybersecurity: DO-178C [124], DO-326A [125], NIST 800-53 [126]
9 - Launch Ready	• Final safety validations automated in DevSecOps pipeline • Incremental FAA Part 450 launch compliance verified continuously	• Aerospace & Space: FAA Part 450 [135], NASA-STD-8719.14 [136] • Environmental & Health: OSHA [131], EPA [133], ANSI [132]

We address safety and regulatory compliance challenges with our continuous assurance Plugin that integrates continuous safety and regulatory compliance throughout the Minimum Viable Product (MVP) development cycle. This approach decouples safety validation from traditional milestone reviews and embeds incremental safety checks, automated compliance verification, and regulatory traceability within Agile workflows.

7.5.5 Model Setup

We began with the SpaceX approach to decomposing the Satellite into a modular set of capabilities [142]. Once we decomposed the system, we outlined a series of MVPs and NVPs to deliver the system. "Minimum Viable Product" (MVP) and "Next Viable Product" (NVP) refer to a concept in product development grounded in the principles of the Lean Startup methodology, which emphasizes rapid iteration, customer feedback, and adaptive planning [145]. We leverage Planet Labs' approach to rapidly create and integrate a prototype and then evolve it with software updates to deliver satellites with relatively short design cycles. Before delivery for launch, the focus is on developing and validating a minimum viable product (MVP) [140]. The top-level model is shown in Figure 7.9

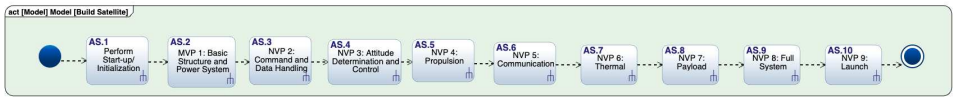


Figure 7.9: Agile Satellite Approach

Start-up and Initialization

Modeled in Figure 7.10, our Agile approach's Startup and Initialization timebox focuses on establishing the foundational digital environment, complete with a digital thread that spans the entire development process. This digital-first strategy is gaining traction, as illustrated by Istari's $19 million contract to digitally certify Lockheed Martin's X-Plane [146], and the digital system building approaches used in Formula 1 [147]. Key inputs include mission and system requirements, regulatory and safety constraints, performance parameters, development tools, and stakeholder involvement. The process begins with analyzing mission objectives and refining system requirements into actionable backlog items. Key performance and compliance factors are reviewed, and initial SysML and 3D modeling help define system structure and behavior. A roadmap outlines

incremental Minimum Viable Products (MVPs) for phased development. Business rhythms are established to ensure synchronization, an Agile performance measurement baseline is defined, and development teams are structured. A product backlog is created, incorporating ATDD acceptance criteria for each feature. Development and test environments are established, including Continuous Integration/Continuous Delivery (CI/CD) pipelines and automated testing frameworks. Critically, during this increment, we integrate safety and compliance directly into our workflow by incorporating safety and compliance into the risk-adjusted backlog, defining an incremental safety validation workflow, and setting up regulatory checklists within our Agile tools. We ensure we meet MIL-STD-882E, NASA NPR 8715.3, NIST 800-53, and ITAR compliance tracking. Outputs of this increment include a risk-adjusted backlog, incorporating quantitative risk analysis [148], an Agile performance measurement baseline covering budget, scope, and schedule [149], a roadmap defining MVPs [150], draft management and technical plans, and a finalized organizational structure. The Monte Carlo analysis for this portion of the project had a Mean of 4.05 Months with a standard deviation of 11.38 days.

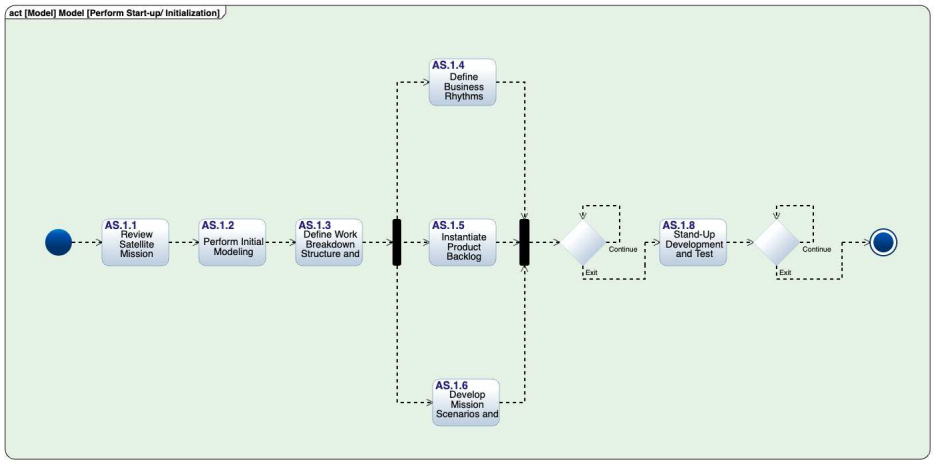


Figure 7.10: Start-up and Initialization

MVP 1 Basic Structure and Power System

MVP 1, as shown in Figure 7.11, establishes the foundational framework and power system required for the Satellite: basic structure and power. This increment begins with backlog grooming and Program Increment (PI) planning, ensuring that acceptance criteria are well-defined. The roadmap, design specifications, and Interface Control Documents (ICDs) guide the development. Concurrently, teams gather materials, including the primary and secondary structures, solar arrays, batteries, and a power distribution unit (PDU). The assembly process involves constructing the Satellite's frame, installing solar panels and batteries, and integrating the PDU to regulate power distribution. In parallel with the assembly and testing, we perform continuous structural risk assessments using Model-Based Systems Engineering (MBSE) for load and stress analysis and automate material compliance tracking (e.g., REACH). These actions ensure adherence to critical safety and regulatory standards, including System Safety (IEC 61508, ISO 26262), Environmental & Health (REACH, OSHA, ANSI), and Aerospace (NASA-STD-8719.14). Testing focuses on validating structural integrity, power generation, and energy storage, ensuring all components function as expected before progressing to the next MVP.

Completion of MVP 1 results in a digitally validated structural and power system [151], with test reports confirming performance and risk-adjusted backlog updates informing the next development iteration. The demonstration showcases the assembled structure, operational solar arrays, and digitally demonstrated functional power distribution. The Monte-Carlo Analysis for this increment showed a Mean of 2.3 Months with a standard deviation of 8 days. This would take much longer if we had not assumed we had procured materials.

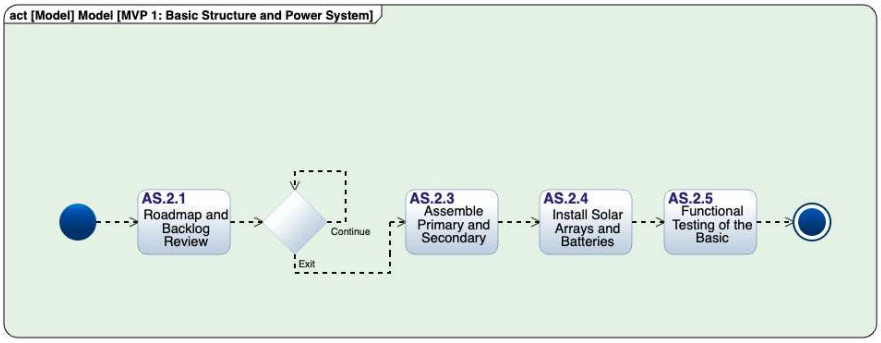


Figure 7.11: Structure and Basic Power

NVP 2 Command and Data Handling

Command and Data Handling (C&DH), modeled in Figure 7.12, focuses on integrating the Satellite's central processing and data management system, ensuring it can receive, process, and execute commands while handling telemetry and onboard data storage. This increment of development includes critical safety and compliance actions: implementing early cyber compliance checks (DO-326A, NIST 800-53) and automating software static analysis. The development starts with backlog grooming, PI planning, and refining acceptance criteria. The key components include the Onboard Computer (OBC), Data Storage Unit, Telemetry Interface, and redundant processing modules, all integrated and tested within our NASA-verified digital environment [152]. The process involves assembling and connecting the OBC, configuring data storage, linking telemetry interfaces, and deploying the initial software stack to validate system functionality. Testing ensures command execution, data processing, and real-time system health monitoring, confirming that the C&DH system meets mission requirements before progressing to the next MVP. This work is conducted to meet the following standards: DO-178C, DO-326A, NIST 800-53, and FedRAMP.

The successful completion of NVP 2 results in an integrated and validated C&DH system, providing a functional command execution and data handling framework. This system is foundational

for controlling all subsequent subsystems, including Attitude Determination and Control (ADC), Propulsion, and Communication, ensuring the Satellite can effectively manage operations and respond to mission commands. The output includes a risk-adjusted backlog, an operational OBC, verified telemetry reporting, and test reports confirming system reliability by following an iterative Agile approach similar to Liubimov's approach for CubeSat [153]. Monte Carlo Analysis for this increment had a mean of 2.22 months with a standard deviation of 8 days.

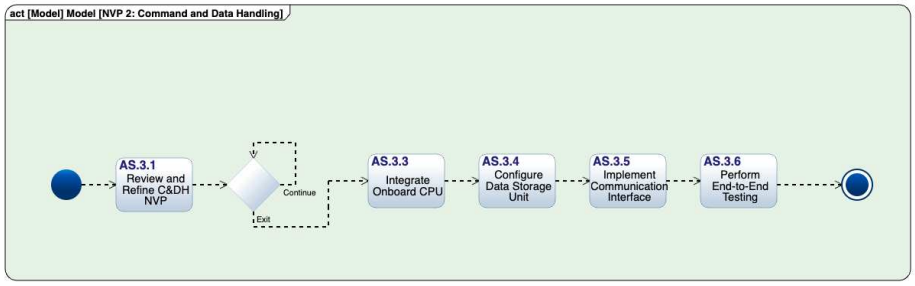


Figure 7.12: Command and Data Handling

NVP 3 Attitude Determination and Control (ADCS)

As illustrated in Figure 7.13, the Attitude Determination and Control System (ADCS) enables the Satellite to determine and adjust its orientation in Space. This increment begins with backlog grooming and Program Increment (PI) planning. Key steps include integrating and configuring ADCS sensors, implementing attitude determination algorithms, and testing system responsiveness under Hardware-in-the-Loop (HIL) and Software-in-the-Loop (SIL) simulations. Testing ensures the system accurately determines orientation, executes attitude corrections, and maintains stability under simulated mission conditions.

In parallel with the ADCS integration and testing, we integrate real-time fault tolerance testing into Agile test pipelines and validate software/hardware failure modes in the digital twin. These

actions ensure adherence to critical safety and regulatory standards, including System Safety (MIL-STD-882E, IEC 61508) and Cybersecurity (DO-326A, ITAR).

Successful completion of NVP 3 results in a fully operational ADCS, with validated attitude accuracy, control responsiveness, and integration with the onboard computer. Key outputs include calibration reports, Reaction Control System (RCS) performance logs, end-to-end integration test reports, and updated Interface Control Documents (ICDs). The RCS is a system of thrusters used to control the attitude and position of the Satellite. These validations ensure the ADCS can support precision pointing for payload operations, stable communication alignment, and controlled maneuvers in future MVPs. MVP 3 sets the foundation for integrating propulsion, communications, and mission-specific payload operations by establishing a stable and autonomous orientation control system. The Monte Carlo analysis shows a Mean of 2.81 months with a standard deviation of 9 days.

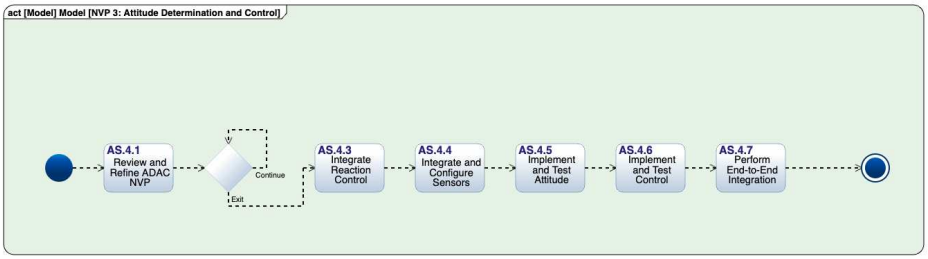


Figure 7.13: Attitude Determination and Control

NVP 4 Propulsion System

The process begins with backlog grooming and PI planning, ensuring alignment with previous NVPs such as Attitude Determination and Control (ADCS) and Command & Data Handling (C&DH). The integration phase includes installing the propulsion unit, fuel tanks, valves, and sensors, and implementing thruster control algorithms to regulate fuel flow and thrust activation. Testing employs Hardware-in-the-Loop (HIL) and Software-in-the-Loop (SIL) simulations, assessing

system responsiveness under simulated orbital conditions to validate fuel system functionality, thruster performance, and maneuver execution before final integration. (HIL simulations test the hardware and software together, while SIL simulations focus on testing software components.) Monte Carlo analysis for this MVP indicated a mean completion time of 3.28 months with a standard deviation of 11 days. This work is conducted to meet the following standards: System Safety: MIL-STD-882E; Environmental & Health: EPA, OSHA; and Aerospace: FAA Part 450.

The successful completion of NVP 4 ensures validated thruster performance, fuel flow control, and essential maneuvering capability, enabling the Satellite to conduct orbital corrections and maintain stability. Key outputs confirm propulsion functionality within expected mission parameters, including integration and test reports, updated Interface Control Documents (ICDs), and end-to-end system validation results. This increment lays the foundation for higher-level operations, such as payload positioning, communication adjustments, and station-keeping, while resolving anomalies and refining system parameters for future NVPs.

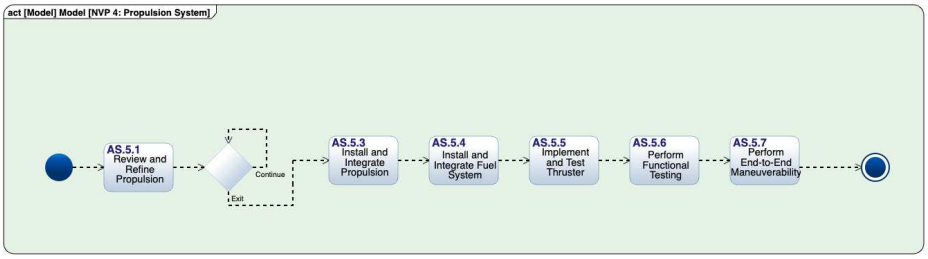


Figure 7.14: Propulsion

NVP 5 Communication System

The Communication subsystem, illustrated in Figure 7.15, focuses on integrating and validating the Satellite's ability to transmit and receive data reliably, a critical function for maintaining mission control and data integrity. This increment includes vital safety and regulatory actions: embedding EMI/EMC compliance verification within Agile sprints and automating regulatory spec-

trum compliance (FCC, ITU). This increment involves installing and testing transmitters, receivers, amplifiers, and high/low-gain antennas, ensuring seamless integration with the Command and Data Handling (C&DH) system. The system's communication control algorithms are deployed and validated through Hardware-in-the-Loop (HIL) and Software-in-the-Loop (SIL) setups, simulating real-world orbital conditions. (HIL simulations test the hardware and software together, while SIL simulations focus on testing software components.) Functional testing ensures data transmission rates, telemetry downlink, and ground station communication operate within expected parameters before full system integration. RF performance metrics are vital to ensure the signal strength and quality are within acceptable ranges for reliable communication. Monte Carlo analysis for this MVP indicated a mean completion time of 2.8 months with a standard deviation of 9 days. This work is conducted to meet the following standards: Electromagnetic Compliance: MIL-STD-461, FCC regulations; Cybersecurity: NIST 800-53, ITAR.

The successful completion of MVP 5 results in a validated communication system, enabling secure and efficient data exchange between the Satellite and the ground station. Output includes integration test reports, RF performance metrics, updated Interface Control Documents (ICDs), and resolved anomaly logs. This MVP ensures that telemetry, remote command execution, and payload data transmission function as required, laying the groundwork for full operational deployment. With a robust and tested communication link, the Satellite is prepared for advanced mission operations, including real-time system monitoring and data collection, supporting the final integration and launch readiness phases.

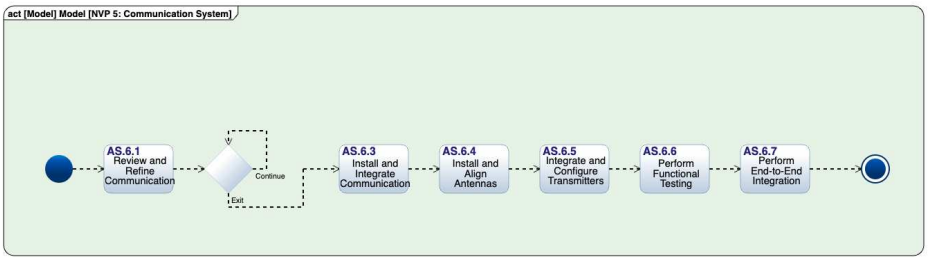


Figure 7.15: Communication

NVP 6 Thermal

The Thermal Control System, modeled in Figure 7.16, ensures that the Satellite can maintain stable operating temperatures in extreme orbital conditions, a critical function for preserving the integrity and performance of all onboard systems. This MVP includes vital safety and regulatory actions: integrating thermal risk modeling into MBSE simulations and automating compliance with NASA-STD-8719.14. This subsystem integrates radiators, heaters, Multi-Layer Insulation (MLI), and temperature sensors, ensuring thermal regulation across all subsystems. The process begins with Program Increment (PI) planning and backlog refinement, followed by the installation of thermal hardware and validation through thermal vacuum (TVAC) chamber testing and simulations. (TVAC testing simulates Space's vacuum and extreme temperature conditions to ensure the thermal system can perform as expected.) The thermal control algorithms are implemented and tested under simulated operational scenarios to verify heat dissipation, insulation efficiency, and active temperature regulation. Functional and end-to-end integration tests confirm that radiators manage excess heat, heaters prevent cold-related failures, and MLI stabilizes subsystem temperatures, ensuring compliance with mission requirements. MLI is vital to minimize heat transfer through radiation, the primary form of heat transfer in Space. Monte Carlo analysis for this MVP indicated a mean completion time of 2.7 months with a standard deviation of 9 days. This work is conducted to meet the following standards: System Safety: ISO 26262, IEC 61508; Aerospace: NASA-STD-8719.14.

The successful completion of nVP 6 results in a validated thermal system, with test reports confirming temperature stability, heater responsiveness, and subsystem integration with the power and structural systems. Key output includes updated Interface Control Documents (ICDs), integration test reports, and an adjusted backlog reflecting lessons learned. This MVP establishes a reliable thermal management framework, protecting critical satellite components and enabling sustained operation in Space. A robust and tested thermal system prepares the Satellite for mission operations and long-duration performance in extreme environments.

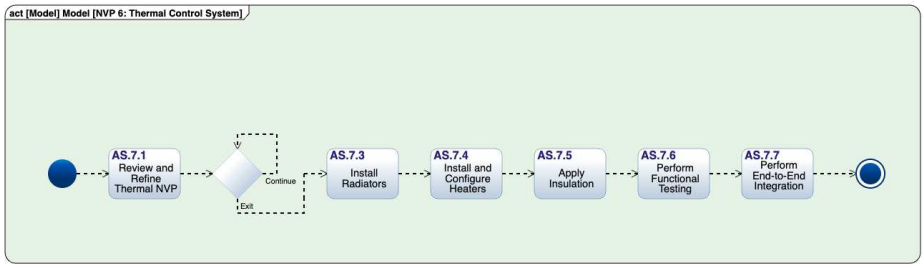


Figure 7.16: Thermal

NVP 7 Payload System

The Payload System, illustrated in Figure 7.17, focuses on integrating and validating the scientific instruments and data processing capabilities essential for the Satellite's mission, specifically designed to [mention specific scientific objectives]. This increment includes critical safety and regulatory compliance activities: ensuring payload-specific safety testing in sprint test cases and continuous FAA payload integration compliance tracking. This increment ensures seamless integration with the Command and Data Handling (C&DH), Power, and Communication Systems, enabling efficient data collection, processing, and transmission. The process begins with planning for the Program Increment (PI), refining backlog priorities, and defining key milestones. The scientific instruments, power and data harnesses, and payload control software are installed and tested using Hardware-in-the-Loop (HIL) setups, functional test benches, and simulated opera-

tional scenarios. (HIL setups allow for testing hardware and software components in a simulated environment, ensuring they function together as expected.) Functional testing validates instrument accuracy, data acquisition, and real-time processing, ensuring stable payload operations before full system integration. The Monte Carlo analysis for this MVP indicated a mean completion time of 2.83 months with a standard deviation of 9 days. This work is conducted to meet the following standards: System Safety: MIL-STD-882E, NASA NPR 8715.3; Aerospace: FAA Part 450, ITAR.

The successful completion of NVP 7 results in a validated payload system with proven data collection, processing, and communication capabilities. Key output includes integration test reports, updated Interface Control Documents (ICDs), and functional verification results, confirming power efficiency, onboard computer integration, and ground station connectivity. This NVP ensures the Satellite is fully equipped for its mission by establishing a robust payload management and data transmission framework. With all payload components successfully tested and integrated, the Satellite is prepared for final system validation and launch preparation in the next phase.

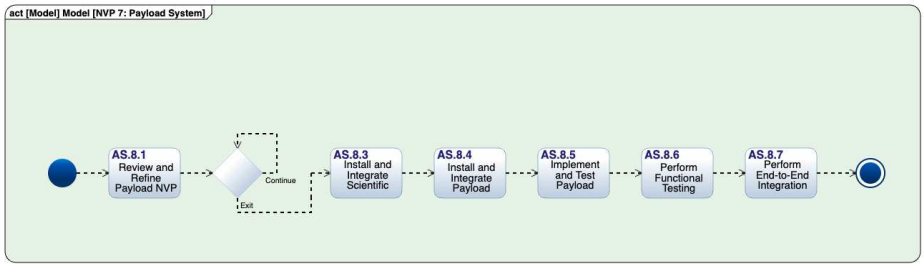


Figure 7.17: Payload

NVP 8 Full System Integration

The Full System Integration step, illustrated in Figure 7.18, ensures that all previously developed subsystems—including structure, power, command and data handling (C&DH), attitude determination and control (ADCS), propulsion, communication, thermal, and payload—are successfully assembled into a fully functional satellite, a pivotal achievement for mission success.

This NVP includes critical safety and regulatory actions, such as implementing incremental safety audits per increment and continuous traceability of safety requirements via MBSE and digital twin. This increment begins with Program Increment (PI) planning, refining integration steps, and validating that all Interface Control Documents (ICDs), mission objectives, and testing procedures are in place. The integration process involves assembling mechanical, electrical, and data systems, ensuring seamless subsystem interaction. The payload control software is deployed and tested to verify command execution, telemetry monitoring, and data processing, while power and data harnesses are connected to ensure full operational capability.

The final integration test reports updated ICDs, and mission validation reports comprehensively assess system performance. These ICDs are vital for documenting and controlling the interfaces between the many subsystems of the Satellite. Comprehensive functional and environmental testing is conducted to validate the Satellite's performance under real-world conditions. Thermal Vacuum (TVAC) tests simulate space conditions, ensuring the thermal control system functions as expected. Vibration and acoustic tests ensure structural integrity for launch, verifying that the Satellite can withstand the stresses of liftoff. Hardware-in-the-loop (HIL) and Software-in-the-Loop (SIL) setups are used for mission simulations, verifying end-to-end mission execution from launch to operational scenarios. (HIL tests combine hardware and software components, while SIL tests focus on software components.) A ground station emulator validates the Satellite's ability to receive and execute ground commands, perform orbital maneuvers, and process payload data. Monte Carlo analysis for this MVP indicated a mean completion time of 3.1 months with a standard deviation of 8 days. This work is conducted to meet the following standards: System Safety: MIL-STD-882E, IEC 61508; Cybersecurity: DO-178C, DO-326A, NIST 800-53.

With full-system functionality verified, this MVP confirms that the Satellite is mission-ready and compliant with all regulatory requirements. The successful integration and testing of all components ensure the Satellite can withstand launch stresses, operate reliably in orbit, and achieve mission objectives. This milestone prepares the Satellite for final launch readiness assessments, marking the transition from development to deployment.

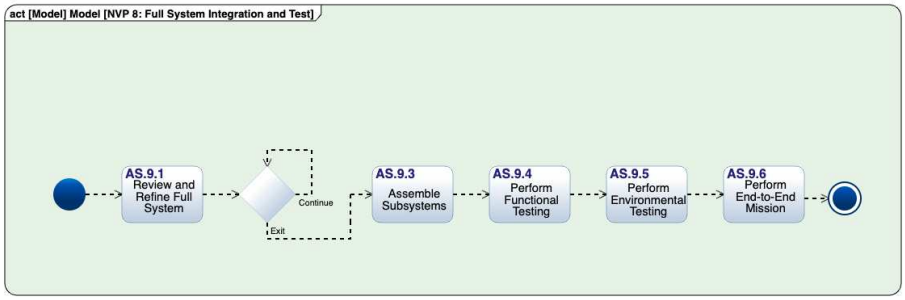


Figure 7.18: Full System Integration

NVP 9 Launch

The final MVP, shown in Figure 7.19, Launch Readiness ensures the Satellite is fully prepared for launch, validating mechanical, electrical, and software integration with the launch vehicle and ground control systems, a crucial step for mission success. This NVP includes critical safety and regulatory actions: final safety validations are automated in the DevSecOps pipeline, and incremental FAA Part 450 launch compliance is verified continuously. This increment involves final pre-launch inspections, system validation, and compliance certification, ensuring the Satellite can withstand launch conditions and establish a stable connection with ground control. The Launch Readiness Checklist, mission software, and telemetry systems are tested in a simulated launch control environment, verifying that the Satellite can receive and execute commands post-deployment. (Simulating a launch control environment allows for verification of all procedures and software in a controlled setting.) The ground control interface is validated, ensuring seamless data transmission between the Satellite and ground stations. Monte Carlo analysis for this NVP indicated a mean completion time of 2.8 months with a standard deviation of 9 days. This work is conducted to meet the following standards: Aerospace & Space: FAA Part 450, NASA-STD-8719.14; Environmental & Health: OSHA [126], EPA, ANSI.

With successful final system checks, integration with the launch vehicle, and regulatory approval, this MVP confirms that the Satellite is flight-ready and has no unresolved technical issues. Key outputs include final inspection reports, launch readiness certification, and validated telemetry systems. These telemetry systems are essential for monitoring the Satellite's health after launch. This milestone marks the transition from development to operational deployment, ensuring the Satellite is cleared for launch and prepared for its mission in orbit.

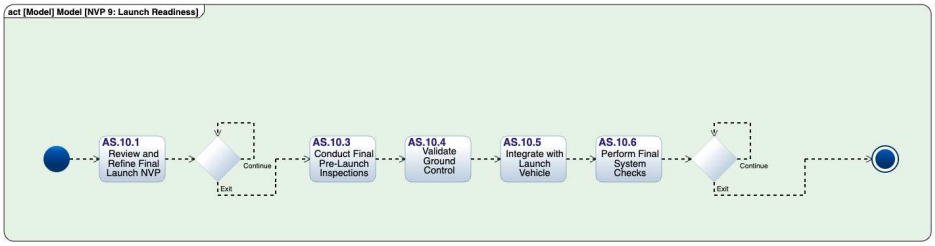


Figure 7.19: Launch

7.5.6 Analysis and Results

The Monte Carlo simulation illustrated in Figure 7.20 for the Agile satellite development approach yielded a mean duration of 2.4 years with a standard deviation of 1 month. In terms of Labor cost, we calculated $2,636,244.12. This result indicates a highly predictable development timeline, with Agile allowing for faster delivery compared to the Waterfall approach's mean duration of 5.89 years. The relatively low standard deviation further reinforces that Agile's incremental development cycles, iterative feedback loops, and continuous integration practices help maintain schedule stability, even in complex system builds.

Importantly, this analysis assumed full availability of materials and resources, meaning that delays related to procurement, supply chain disruptions, or resource shortages were not factored into the simulation. This assumption contributed to the high predictability of Agile's results, minimizing variability in the projected timeline. In real-world conditions, Agile's adaptability to changing

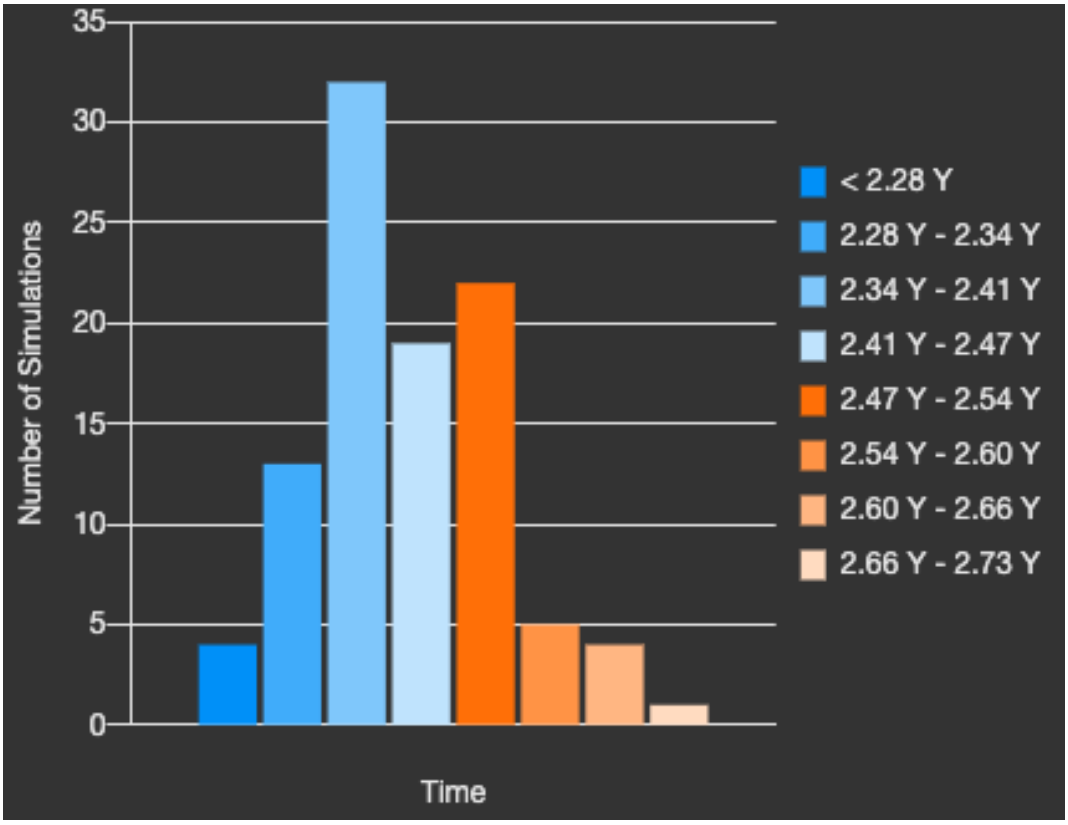


Figure 7.20: Monte-Carlo Analysis of Agile Development Cycle

requirements and resource fluctuations may provide an advantage over Waterfall, which tends to experience more schedule slips when unexpected constraints arise. The findings demonstrate that, under optimal conditions, Agile can deliver a satellite in less than half the time of a traditional approach while maintaining low schedule uncertainty, making it a viable methodology for accelerating space system development.

7.6 Discussion

This study compares Agile and Waterfall methodologies for satellite system development, evaluating their impact on timeline efficiency, risk mitigation, and regulatory compliance. The Monte Carlo analysis demonstrated that Agile significantly reduces development time, with a mean duration of 2.4 years compared to Waterfall's 5.89 years, while maintaining a lower standard deviation. These findings suggest that Agile's iterative cycles, continuous integration, and incremental vali-

dation contribute to a more predictable and efficient development process. Our results align with the results Ciric found in their paper regarding Agile Project Management (APM) [154].

7.6.1 Agile's Impact on Development Efficiency

With its iterative approach, Agile development significantly shortens development timelines by fostering early and frequent testing, thereby minimizing late-stage rework—a clear advantage over the waterfall model's delayed validation. We implemented a Continuous Assurance Plugin to bolster agility in safety-critical and regulated domains, seamlessly integrating ***people***, ***process***, and ***technology***. This framework prioritizes the inclusion of regulatory compliance and safety expertise within development teams, early engagement of auditors for automated test development, and continuous collaboration with subject matter experts during reviews. Process enhancements include embedding compliance and safety user stories into the risk-adjusted product backlog, which is constantly managed to address emerging risks proactively, conducting hazard analysis via STPA, and employing continuous validation checklists [155] [156]. Technology is leveraged through advanced verification and validation using MBSE, digital twin simulations, robust automated testing, and integrated toolsets. We emphasize quantifiable metrics, such as reduced compliance defects and improved safety rates, and cultivate a culture of shared responsibility and continuous improvement. By integrating a risk-adjusted product backlog, we ensure that risk management is a dynamic and integral part of the development process, allowing teams to respond swiftly to potential issues and maintain project agility while upholding stringent safety and compliance standards.

7.6.2 Challenges in Applying Agile to Safety-Critical Systems

Despite its advantages, Agile's implementation in a safety-critical domain such as regulatory compliance, safety assurance, integration complexity, traceability & documentation, and Organizational communication barriers.

Regulatory Compliance

Space systems must comply with stringent industry standards, including MIL-STD-461, NASA-STD-8719.14, FAA Part 450, NIST 800-53, and ITAR regulations. Traditional Waterfall models inherently align with these compliance requirements through predefined verification stages. Conversely, Agile's iterative approach requires decoupling regulatory compliance checks from stage gates and moving to right-size documentation [157], performing incremental compliance checks with checklists [158].In addition, we can model compliance using Model-Based Systems Engineering (MBSE), simulating the impact using a digital twin [159]. For instance, digital twin simulations can assess compliance impact by virtually testing system responses to various regulatory scenarios. Compliance with these regulations is crucial for ensuring space systems.

Safety Assurance

Failures in safety-critical systems necessitate rigorous methodologies that ensure consistent safety evaluations across design and operational stages. The integration of a continuous assurance Plugin that employs Behavior Driven Development (BDD) and Acceptance Test Driven Development (ATDD) can complement the Systems-Theoretic Process Analysis (STPA) framework to enhance safety verification cite wang2018. STPA is particularly effective in identifying potential failure modes, as it views safety violations as a result of unsafe interactions among components rather than merely from component failures [160].In addition to STPA, implementing Model-Based Systems Engineering (MBSE) systematically organizes safety system designs within agile frameworks. MBSE enhances the iterative development approach by documenting safety requirements, facilitating clear communication among stakeholders, and integrating safety checks into the user story definition of done [161]. We utilize the risk-adjusted product backlog to prioritize safety concerns continuously. In conclusion, the continuous assurance plugin that integrates BDD, ATDD, and STPA, enhanced by MBSE, and managed in a risk-adjusted backlog presents a robust approach for managing safety in complex systems.

Integration Complexity

Our Continuous Assurance plugin supports the challenge of integration complexity by supporting Agile with MBSE and Digital Twins. This synergistic approach fosters early integration and validation, which are pivotal in managing the inherent complexities associated with these systems. MBSE provides a structured and formalized method for capturing CPS's requirements, architecture, and design, thus establishing a well-documented framework that supports iterative development. The integration of MBSE and Digital twins to support Agile was demonstrated by Vodyaho with the Smart City case study, which managed transport and flows in St. Petersburg (Russia) [162]. Digital Twins complement MBSE by creating real-time virtual representations of physical systems, allowing continuous integration and validation throughout development. They enable hardware-software co-simulation, predictive analytics, and real-world scenario testing without waiting for full system deployment. Integrating Agile, MBSE, and Digital Twins reduces delivery times and lowers risk exposure [163].

Traceability and Documentation

Model-Based Systems Engineering (MBSE) within our Continuous Assurance plugin enhances traceability and documentation processes. A common perception is that Agile teams often neglect documentation, presenting a barrier to effective traceability. However, integrating Agile methodologies with MBSE can address these challenges while ensuring the documentation is appropriately scaled and valuable. MBSE leverages models to facilitate various systems engineering activities, including requirements capture, system functionalities identification, and verification tasks, significantly improving traceability (updating requirements as changes occur) compared to traditional document-based methods [164] [112]. For Agile teams, embracing these methods allows for a more adaptable documentation process that aligns with rapid development cycles while maintaining compliance with traceability requirements cite bussemaker2022.

Organizational Culture Barriers

The fictional case study did not demonstrate unique considerations regarding overcoming organizational and cultural barriers. This contrast between industry mindsets creates inherent challenges when introducing agile methodologies. Safety-critical industries prioritize risk minimization and predictability, often adopting a 'fail-safe' rather than 'fail-fast' mentality. In contrast, agile methodologies emphasize cross-functional, self-organizing teams. However, traditional structures in safety-critical sectors typically separate engineering, safety, regulatory compliance, and testing into separate silos. Furthermore, the heterogeneous teams familiar with cyber-physical systems tend to increase resistance to change. Heterogeneity, defined in this context as the diversity of team backgrounds and perspectives, significantly impacts communication, collaboration, and overall teamwork dynamics, as noted by Grotto [165]. These implications highlight the challenges of integrating agile practices into environments where rigid, siloed structures have historically prevailed. Socio-technical systems (STS) theory may effectively resolve the difficulties of incorporating agile methodologies into safety-critical industries. STS theory emphasizes the interplay between social and technical factors in organizational systems, recognizing that both aspects must be considered for optimal performance. Therefore, by utilizing STS theory, safety-critical industries can better integrate agile methodologies."

7.7 Conclusion

This study demonstrates that Agile can significantly reduce satellite development time while maintaining predictable scheduling and adherence to regulatory requirements. However, its application in safety-critical space systems requires specific adaptations, including incremental safety audits, continuous compliance tracking, and advanced risk modeling. While Agile's benefits are evident, its limitations in full-system integration and long-term mission assurance highlight the potential value of a hybrid development model. Future research should investigate Agile's impact on mission reliability, cost, and scalability in real-world space system deployments.

Chapter 8

Conclusion, Contribution, Future

8.1 Conclusion

This dissertation systematically addressed three critical research questions related to applying Agile methodologies to large-scale, safety-critical, cyber-physical (LS/SC/CP) systems. First, this study comprehensively evaluated existing Agile Scaling Frameworks, highlighting their comparative strengths, weaknesses, and suitability for the complexities inherent in LS/SC/CP environments. The analysis revealed that while many frameworks offer beneficial practices, significant adaptations were essential to meet stringent regulatory, safety, and system integration requirements.

Second, the research employed a robust mixed-methods approach, incorporating extensive survey data and detailed qualitative insights from interviews to assess the current state of Agile adoption in LS/SC/CP domains. This approach uncovered widespread application yet notable challenges, such as regulatory compliance, safety assurance, integration complexity, documentation/traceability, and organizational/team dynamics. The findings also highlighted specific adaptations organizations employ, including embedding compliance and safety practices directly into Agile workflows and enhancing collaboration through continuous stakeholder engagement.

Finally, this study quantitatively and qualitatively assessed the impact of Agile methodologies compared to traditional Waterfall approaches. We developed a fictional case study where we modeled building a mid-size Leo satellite using the NASA Waterfall approach and the Agile approach with our continuous assurance plugin. Rigorous modeling and simulation demonstrated that Agile significantly improved development timelines with increased adaptability. Implementing proposed adaptations, notably the Continuous Assurance Plugin, effectively mitigated identified challenges, enabling organizations to maintain agility while upholding rigorous safety and compliance standards. This research provides actionable insights and clear pathways for enhancing Agile practices within complex, regulated LS/SC/CP system development environments.

8.1.1 Sociotechnical Considerations

This research focused on the technical implementation of Agile in LS/SC/CP systems. However, the application of Agile involves more than technology; it also encompasses social elements. Sociotechnical systems theory (STT) posits that organizational systems are best understood and improved by considering both the 'social' (people, culture, relationships) and 'technical' (technology, processes) aspects as interdependent, complex elements [166]. The social dynamics, including power relationships, culture, trust, and communication, significantly impact the success of Agile methodologies. To ensure the success of this approach, we need to delve deeper into these social dynamics within the context of LS/SC/CP systems.

8.2 Research Contribution

8.2.1 Industry Contributions

Rigorous Comparative Analysis Agile Scaling Frameworks Conducted rigorous comparative analysis of the top 10 Agile scaling frameworks, assessing their suitability for large-scale, safety-critical cyber-physical systems and providing actionable insights for industry practitioners.

Empirical Insights on the current state of Agile applied to LS/SC/CP Systems This mixed-methods research study used surveys and interviews to investigate how Agile methodologies are implemented in Large-Scale, Safety-Critical, Cyber-Physical (LS/SC/CP) systems. Combining these methods allowed the researchers to gather quantitative and qualitative data, providing a more comprehensive understanding of the topic. This approach helped fill knowledge gaps and reveal patterns that had not been systematically documented.

8.2.2 Systems Engineering Contribution

Model-based comparison of two development lifecycles Developed comprehensive comparative models using Innoslate to empirically validate the significant impact of Agile methodologies over traditional Waterfall approaches in terms of reduced development timelines, improved cost-efficiency, and enhanced system adaptability through detailed Monte Carlo analysis.

Continuous Assurance Plugin A continuous assurance plugin was developed to decouple these critical checks from traditional waterfall systems engineering technical reviews (SETRs). This framework enhances efficiency by directly embedding regulatory compliance and safety expertise within development teams. Key process enhancements include embedding compliance and safety user stories into the development cycle, proactive risk management, hazard analysis via Systems-Theoretic Process Analysis (STPA) integrated with BDD, and continuous validation checklists. Technologically, the framework leverages Model-Based Systems Engineering (MBSE), digital twin simulations, and automated testing. These technologies facilitate real-time analysis and validation, enhancing the speed and accuracy of compliance and safety assessments.

Adaption Strategies and Best Practices for Agile in Safety-Critical Domains The research identified and categorized adaptations for regulatory compliance, safety assurance, and complexity management in Agile development for LS/SC/CP systems, introduced new strategies blending Agile with traditional safety practices to enhance agility without compromising safety, and provided actionable recommendations.

Advancement of Agile and Systems Engineering Integration By decoupling safety and regulatory compliance activities from traditional phase gates, the project enabled the incremental application of Agile principles in complex, regulated systems engineering environments, enhancing understanding and bridging the gap between Agile software methods and system-level adoption.

8.2.3 Impacts

- An evidence-based assessment of Agile Scaling Frameworks for LS/SC/CP.
- A structured evaluation of the current state of Agile in the industry via surveys and interviews.
- A validated process model comparing Agile vs. Waterfall in satellite development.
- Actionable recommendations to industry, government, and academia for adapting Agile methodologies in safety-critical domains.

This work advances theory and practice in adopting Agile for large-scale, safety-critical, cyber-physical systems. It is a foundation for future studies, policy-making, and industry implementations.

8.3 Limitations

While this research aims to provide comprehensive insights into implementing Agile principles and practices in LS/SC/CP systems, several limitations must be acknowledged.

Firstly, the variability in the implementation and maturity of Agile practices across diverse organizations and industries presents a significant limitation. The collected data may not fully capture the breadth of Agile approaches, potentially introducing biases into the findings. Additionally, the study's reliance on self-reported data from practitioners introduces subjectivity, which can affect the accuracy of the results. Furthermore, the complexity and specificity of LS/SC/CP projects limit the ability to generalize the conclusions to all cyber-physical systems or industries.

Secondly, methodological constraints impact the research. The comparative analysis of Agile and Waterfall methodologies, based on a hypothetical project, does not account for all contextual factors influencing project outcomes. Variations in organizational culture, system architecture, and external factors such as regulatory requirements can significantly affect effectiveness. Moreover, this research does not account for isolating the effects of Agile practices from other concurrent organizational changes.

Finally, the proposed guidelines and frameworks for implementing Agile in LS/SC/CP systems are based on current best practices and empirical evidence. As new methodologies and technologies emerge, these guidelines may require updates and revisions to maintain relevance and applicability. This dynamic nature of the field necessitates ongoing refinement of the research findings.

While the research provides valuable insights, models, and empirical evidence, the limitations highlight areas where further work is needed to validate findings, expand applicability, and refine Agile methodologies for safety-critical domains. Recognizing these constraints helps frame the contributions realistically and set the stage for future advancements.

8.4 Future Work

Future research should include empirical validation through pilot programs across multiple domains to quantify Agile's impact on key performance metrics such as delivery speed, defect rates, cost, and system integration complexity. Specifically, conducting comparative case studies between Agile with the continuous assurance recommendations and traditional Waterfall approaches across different large-scale, safety-critical, cyber-physical (LS/SC/CP) systems would provide concrete data on Agile's effectiveness in regulated environments.

Additionally, there is an opportunity to explore the incorporation of AI-driven assistance within Agile workflows. AI could enhance Agile practices by automating documentation, assisting in requirement traceability, automating model development, predicting system risks, optimizing risk-adjusted backlog prioritization, and improving real-time decision-making. Future studies should investigate how machine learning models and AI-enhanced DevSecOps pipelines can reduce cycle times, improve system adaptability, and enhance overall system resilience in highly complex environments. This research will address key questions: How can AI be effectively integrated into Agile workflows to improve efficiency and reduce risks? What are the limitations of current AI-driven Agile tools, and how can they be overcome?

Future research should expand to encompass the sociotechnical dimensions of Agile transformation. Sociotechnical Systems Theory (STT) emphasizes the inseparability and mutual shaping of the social and technical subsystems within organizations. Investigate how social elements such as team trust, leadership behaviors, power dynamics, cross-disciplinary communication, and organizational culture influence Agile outcomes. In particular, research could explore how these dynamics differ across domains (e.g., space exploration, medical diagnostics, transportation infrastructure) and levels of system criticality (e.g., life-support systems, patient monitoring).

Bibliography

[1] Phil Zimmerman, Tracee Gilbert, and Frank Salvatore. Digital engineering transformation across the department of defense. *The Journal of Defense Modeling and Simulation*, 16(4):325–338, 2019.

[2] James Womak, Daniel T Jones, and Daniel Roos. The machine that changed the world. *New York: Rawson Associates*, 1990.

[3] Michael Herman. Introducing enterprise scrum for business agility: Scale scrum from single teams to whole organizations, 2017.

[4] Craig Larman and Bas Vodde. *Large-Scale Scrum: More with LeSS*. Addison-Wesley Professional, 2016.

[5] Jeff Sutherland and Scrum Inc. *The Scrum@Scale Guide*, 2023. Accessed: 2025-03-29.

[6] Jochen Krebs. *Agile portfolio management*. Microsoft Press, 2008.

[7] Dean Leffingwell. *SAFe 4.5 Reference Guide: Scaled Agile Framework for Lean Enterprises*. Addison-Wesley Professional, 2019.

[8] Scott Ambler and Mark Lines. Introduction to disciplined agile delivery. Project Management Institute, 2020.

[9] Henrik Kniberg and Anders Ivarsson. Scaling agile@ spotify, 2012.

[10] Ken Schwaber. The nexus guide, 2015.

[11] Kevin Thompson. Recipes for agile governance in the enterprise, 2013.

[12] Kim Dikert, Maria Paasivaara, and Casper Lassenius. Challenges and success factors for large-scale agile transformation: A systematic literature review. *Journal of Systems and Software*, 119:87–108, 2016.

[13] J. C. Knight. Safety critical systems: challenges and directions. In *Proceedings of the 24th International Conference on Software Engineering*, pages 547–550, May 2002.

[14] Edward A. Lee and Sanjit A. Seshia. *Introduction to Embedded Systems: A Cyber-Physical Systems Approach*. MIT Press, 2nd edition, 2017.

[15] Lise Tordrup Heeager and Peter Axel Nielsen. A conceptual model of agile software development in a safety-critical context: A systematic literature review. *Information and Software Technology*, 103:22–39, 2018.

[16] Miren Illarramendi, Leire Etxeberria, Xabier Elkorobarrutia, and Goiuria Sagardui. Increasing dependability in safety critical cpss using reflective statecharts. In *Computer Safety, Reliability, and Security: SAFECOMP 2017 Workshops, ASSURE, DECSoS, SASSUR, TELERISE, and TIPS, Trento, Italy, September 12, 2017, Proceedings 36*, pages 114–126. Springer, 2017.

[17] Fakhrina Fahma, Wahyudi Sutopo, Eko Pujiyanto, and Muhammad Nizam. Research trends on smart connected products in the industry 4.0: A systematic literatur review. In *E3S Web of Conferences*, volume 465, page 02007. EDP Sciences, 2023.

[18] Elvis Hozdić and Peter Butala. Concept of socio-cyber-physical work systems for industry 4.0. *Tehnički vjesnik*, 27(2):399–410, 2020.

[19] Elias G Carayannis and Joanna Morawska-Jancelewicz. The futures of europe: Society 5.0 and industry 5.0 as driving forces of future universities. *Journal of the Knowledge Economy*, 13(4):3445–3471, 2022.

[20] Michael Riesener, Christian Dölle, Alexander Keuper, Marc Fruntke, and Guenther Schuh. Quantification of complexity in cyber-physical systems based on key figures. *Procedia CIRP*, 100:445–450, 2021. 31st CIRP Design Conference 2021 (CIRP Design 2021).

[21] Ondrej Burkacky, Johannes Deichmann, Philipp Pfingstag, and Julia Werra. Semiconductor shortage: How the automotive industry can succeed. *McKinsey & Company*, 2022.

[22] Winston W Royce. Managing the development of large software systems.[online] http://www. cs. umd. edu/class/spring2003/cmsc838p. *Process/waterfall. pdf*, 1970.

[23] N. B. Ruparelia. Software development lifecycle models. *ACM SIGSOFT Software Engineering Notes*, 35(3):8–13, 2010.

[24] Barry W. Boehm and Kevin J. Sullivan. Software economics: a roadmap. In *Proceedings of the Conference on The Future of Software Engineering*, pages 319–343, 2000.

[25] Rooh Ullah Jan, Muhammad Usman, Muhammad Faisal Abrar, Najeeb Ullah, Muhammad Asshad, and Sikandar Ali. Scaling agile adoption motivators from management perspective: An analytical hierarchy process approach. *Scientific Programming*, 2021.

[26] Scott W. Ambler and Mark Lines. *Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise*. IBM Press, 2012.

[27] Tor Stålhane, Geir Kjetil Hanssen, and Tor Myklebust. The application of iso 26262: 2011 and agile practices in a research project. In *2012 Seventh International Conference on Availability, Reliability and Security*, pages 567–572. IEEE, 2012.

[28] Tasuku Ishigooka, Habib Saissi, Thorsten Piper, Stefan Winter, and Neeraj Suri. Safety verification utilizing model-based development for safety critical cyber-physical systems. *Journal of Information Processing*, 25:797–810, 2017.

[29] Scarlet Rahy and Julian M Bass. Managing non-functional requirements in agile software development. *IET software*, 16(1):60–72, 2022.

[30] Jan-Philipp Steghöfer, Eric Knauss, Jennifer Horkoff, and Rebekka Wohlrab. Challenges of scaled agile for safety-critical systems. In *International Conference on Product-Focused Software Process Improvement*, pages 350–366. Springer, 2019.

[31] Yang Wang and Stefan Wagner. Towards applying a safety analysis and verification method based on stpa to agile software development. In *Proceedings of the International Workshop on Continuous Software Evolution and Delivery*, pages 5–11, 2016.

[32] Bob Walrave, Sharon Dolmans, Kim E van Oorschot, Arno LP Nuijten, Mark Keil, and Stefan van Hellemond. Dysfunctional agile–stage-gate hybrid development: Keeping up appearances. *International Journal of Innovation and Technology Management*, 19(03):2240004, 2022.

[33] Paweł Weichbroth. A case study on implementing agile techniques and practices: Rationale, benefits, barriers and business implications for hardware development. *Applied Sciences*, 12(8457):1–23, 2022.

[34] Hirotaka Takeuchi and Ikujiro Nonaka. The new new product development game. *Harvard business review*, 64(1):137–146, 1986.

[35] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. The agile manifesto. Agile Alliance, 2001.

[36] Petri Kettunen and Maarit Laanti. Future software organizations–agile goals and roles. *European Journal of Futures Research*, 5(1):16, 2017.

[37] Department of Defense. Cereal breakfast food, ready-to-eat, fortified. Technical Report MIL-C-43205G, Department of Defense, United States of America, 1986.

[38] Lucas T Khoza and Carl Marnewick. Waterfall and agile information system project success rates-a south african perspective. *South African Computer Journal*, 32(1):43–73, 2020.

[39] L Rahmania, Moh Safii, C Jayanti, Vertic Eridani Budi Darmawan, and Yuh-Wen Chen. The development of swseum (semantic web museum) in mpu purwa museum malang. In *Pro-*

ceedings of the 1st International Conference on Literature Innovation in Chinese Language, 2022.

[40] Mary Poppendieck, Thomas David Poppendieck, and Tom Poppendieck. *Implementing lean software development: from concept to cash*. Pearson Education, 2007.

[41] David J Anderson and Andy Carmichael. *Essential kanban condensed*. Blue Hole Press, 2016.

[42] Kent Beck. Embracing change with extreme programming. *Computer*, 32(10):70–77, 1999.

[43] Juha Koskela and Pekka Abrahamsson. On-site customer in an xp project: empirical results from a case study. In *Software Process Improvement: 11th European Conference, EuroSPI 2004, Trondheim, Norway, November 10-12, 2004. Proceedings 11*, pages 1–11. Springer, 2004.

[44] N. Carroll, F. O. Bjørnson, T. Dingsøyr, K. R. Rolland, and K. Conboy. Operationalizing agile methods: examining coherence in large-scale agile transformations. In *Agile Processes in Software Engineering and Extreme Programming – Workshops*, pages 75–83, 2020.

[45] Andreas Grundler and Markus Westner. Scaling agile frameworks vs. traditional project portfolio management: Comparison and analysis. In *Proceedings of the International Conferences on Internet Technologies & Society (ITS 2019), Hongkong*, pages 51–62, 2019.

[46] Gadi Aleksandrowicz and et. all Arbel. *Designing Reliable Cyber-Physical Systems*. Springer, 2017.

[47] Nis Ovesen. The challenges of becoming agile: Implementing and conducting scrum in integrated product development. 2012.

[48] Annette Isabel Böhmer, Philipp Hugger, and Udo Lindemann. Scrum within hardware development insights of the application of scrum for the development of a passive exoskeleton.

In *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 790–798, 2017.

[49] Md. Zahidul Islam, Yuzhang Lin, Vinod M. Vokkarane, and Venkatesh Venkataramanan. Cyber-physical cascading failure and resilience of power grid: A comprehensive review. *Frontiers in Energy Research*, 2023.

[50] Natalia Rykhtikova. Main directions of the development of the risk management system in corporations. In *MATEC Web of Conferences*, volume 212, page 07012. EDP Sciences, 2018.

[51] Abhishek Saini and Ruchi Sehrawat. An intelligent and efficient cnn-aes framework for image block encryption with a multi-key approach. *Engineering Research Express*, 7(1):015206, 2025.

[52] Andrei Carniel, Juliana De Melo Bezerra, and Celso Massaki Hirata. An ontology-based approach to aid stpa analysis. *IEEE Access*, 11:12677–12697, 2023.

[53] Shufeng Chen, Siddartha Khastgir, Islam Babaev, and Paul Jennings. Identifying accident causes of driver-vehicle interactions using system theoretic process analysis (stpa). In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3247–3253. IEEE, 2020.

[54] EG Chukwurah and S Aderemi. Harmonizing teams and regulations: strategies for data protection compliance in us technology companies. *Computer Science & IT Research Journal*, 5(4):824–838, 2024.

[55] Jeff A Estefan and Tim Weilkiens. Mbse methodologies. In *Handbook of model-based systems engineering*, pages 47–85. Springer, 2023.

[56] Cacia Ploeg, Kimberly Lai, and Alison Olechowski. Prioritization of best practices in the implementation of model-based systems engineering. In *INCOSE International Symposium*, volume 32, pages 961–975. Wiley Online Library, 2022.

[57] Kimberly Lai, Thomas Robert, David Shindman, and Alison Olechowski. Integrating safety analysis into model-based systems engineering for aircraft systems: A literature review and methodology proposal. In *INCOSE International Symposium*, volume 31, pages 988–1003. Wiley Online Library, 2021.

[58] Roy K Tsui, John M Borky, and Thomas H Bradley. Applying model-based systems architecture processes (mbsap) methodology for diversified mbse projects with efficient systems of systems accomplishments. In *INCOSE International Symposium*, volume 30, pages 1568–1580. Wiley Online Library, 2020.

[59] Maryam H Gracias and Erika E Gallegos. Transitioning perspectives: Agile and waterfall perceptions in the integration of model-based systems engineering (mbse) within aerospace and defense industries. *The ITEA Journal of Test and Evaluation*, 45(4), 2024.

[60] Robin Yeman and Suzette Johnson. Advancing industrial devops: Harnessing digital twins and ai for future-ready engineering. In *AIAA SCITECH 2025 Forum*, page 0284, 2025.

[61] Michael Grieves and John Vickers. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. *Transdisciplinary perspectives on complex systems: New findings and approaches*, pages 85–113, 2017.

[62] Robin Yeman and Suzette Johnson. The application of industrial devops using digital twins. In *Enterprise Technology Leadership Journal: Spring 2024*. IT Revolution, 2024.

[63] Mary A Bone, Mark R Blackburn, Donna H Rhodes, David N Cohen, and Jaime A Guerrero. Transforming systems engineering through digital engineering. *The Journal of Defense Modeling and Simulation*, 16(4):339–355, 2019.

[64] Hoon Yeub Jeong, Soo-Chan An, Yeonsoo Lim, Min Ji Jeong, Namhun Kim, and Young Chul Jun. 3d and 4d printing of multistable structures. *Applied Sciences*, 10(20):7254, 2020.

[65] Yu Hua Dai and Xi Wang. Design and verification of a metal 3d printing device based on contact resistance heating. *Solid State Phenomena*, 298:64–68, 2019.

[66] Jannik Reichwein, Sven Vogel, Stefan Schork, and Eckhard Kirchner. On the applicability of agile development methods to design for additive manufacturing. *Procedia CIRP*, 91:653–658, 2020.

[67] Haroon Sheikh, Corien Prins, and Erik Schrijvers. Artificial intelligence: definition and background. In *Mission AI: The new system technology*, pages 15–41. Springer, 2023.

[68] Daniel Byrne, Vincent Hargaden, and Nikolaos Papakostas. Application of generative ai technologies to engineering design. *Procedia CIRP*, 132:147–152, 2025.

[69] Barbara Kitchenham et al. Systematic literature reviews in software engineering–a systematic literature review. *Information and Software Technology*, 51(1):7–15, 2009.

[70] V. Rantala, K. Könnölä, S. Suomi, M. Isomäki, and T. Lehtonen. Agile embedded system development versus european space standards. *International Journal of Information Systems and Social Change*, 8(1):1–23, 2017.

[71] Digital.ai. The challenges and solutions of scaling agile across the enterprise: Insights from the 17th state of agile report, 2023.

[72] Kieran Conboy and Noel Carroll. Implementing large-scale agile frameworks: challenges and recommendations. *IEEE software*, 36(2):44–50, 2019.

[73] Rui M Lima, José Dinis-Carvalho, Thiago A Souza, Elisa Vieira, and Bruno Gonçalves. Implementation of lean in health care environments: an update of systematic reviews. *International Journal of Lean Six Sigma*, 12(2):399–431, 2021.

[74] Zbyslaw Dobrowolski, Łukasz Sułkowski, and Peter Adamisin. Innovative ecosystem: the role of lean management auditing. *Marketing i menedžment innovacij*, 13(3):9–20, 2022.

[75] Katie Anderson. *Learning to Lead, Leading to Learn: Lessons From Toyota Leader Isao Yoshino on a Lifetime of Cont*. Integrand Press, 2022.

[76] Mike A. Beedle. *Enterprise Scrum: An Adaptive Method for Project Success*. Addison-Wesley Professional, 2014.

[77] Jing. A brief history of agile: Mike beedle - tragic loss to scrum community. ZenTao Blog, 2022. Accessed: 2024-08-18.

[78] Craig Larman. *Agile and iterative development: a manager's guide*. Addison-Wesley Professional, 2004.

[79] Ivar Jacobson, Jeff Sutherland, Brian Kerr, and Barbora Buhnova. Better scrum through essence. *Software: Practice and Experience*, 52(6):1531–1540, 2022.

[80] Jeff Sutherland and JJ Sutherland. *Scrum: the art of doing twice the work in half the time*. Crown Currency, 2014.

[81] Linda Kester, Erik Jan Hultink, and Abbie Griffin. An empirical investigation of the antecedents and outcomes of npd portfolio success. *Journal of Product Innovation Management*, 31(6):1199–1213, 2014.

[82] Dean Leffingwell. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley Professional, Boulder, Colorado, 2010.

[83] Elkin Doney Suárez-Gómez and Carlos Arturo Hoyos-Vallejo. Scalable agile frameworks in large enterprise project portfolio management. *IEEE Access*, 11:98666–98684, 2023.

[84] Abheeshta Putta, Ömer Uludağ, Shun-Long Hong, Maria Paasivaara, and Casper Lassenius. Why do organizations adopt agile scaling frameworks? a survey of practitioners. In *Proceedings of the 15th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 1–12. ACM, 2021.

[85] Martin Kalenda, Petr Hyna, and Bruno Rossi. Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, 30(10):e1954, 2018.

[86] F. Almeida and E. Espinheira. Large-scale agile frameworks: a comparative review. *Journal of Applied Sciences, Management and Engineering Technology*, 2(1):18–20, 2021.

[87] Simon Bourk and Patricia Kong. An introduction to the nexus™ framework. *Scrum. org whitepapers, Haziran*, pages 3–4, 2016.

[88] Ersin Ersoy, Engin Çallı, Batuhan Erdoğan, Selami Bağrıyanık, and Hasan Sözer. A longitudinal case study on nexus transformation: Impact on productivity, quality, and motivation. *Journal of Software: Evolution and Process*, 36(5):e2615, 2024.

[89] Florin Dumitriu, G Meșniţă, Dumitru Oprea, and LD Radu. Challenges of scaling agile at organization level. In *Proceedings of the 18th International Conference on Informatics in Economy. Education, Research and Business Technologies*, pages 339–344, 2019.

[90] R. J. Yeman and Y. K. Malaiya. A systematic literature review of the application of agile applied to large-scale, safety-critical, cyber-physical systems. In *2024 IEEE International Conference on Data and Software Engineering (ICoDSE)*, pages 13–18, 2024.

[91] Joseph D Miller. *Automotive system safety: Critical considerations for engineering and effective management*. John Wiley & Sons, 2019.

[92] Tor Myklebust and Geir Kjetil Hanssen. Risk management in agile software development for safety-critical systems. *Journal of Software: Evolution and Process*, 26(5):478–495, 2014.

[93] Kai Petersen and Claes Wohlin. Documentation and agile practices in industry: A survey on their usage and impact. *Journal of Systems and Software*, 83(3):502–511, 2010.

[94] O. A. Popoola, H. E. Adama, C. D. Okeke, and A. E. Akinoso. Conceptualizing agile development in digital transformations: theoretical foundations and practical applications. *Engineering Science & Technology Journal*, 5(4):1524–1541, 2024.

[95] M. Laanti, O. Salo, and P. Abrahamsson. Agile methods rapidly replacing traditional methods at nokia: a survey of opinions on agile transformation. *Information and Software Technology*, 53(3):276–290, 2011.

[96] J. B. Barlow, J. S. Giboney, M. Keith, D. M. Wilson, R. M. Schuetzler, P. B. Lowry, and A. Vance. Overview and guidance on agile development in large organizations. *SSRN Electronic Journal*, pages 31–33, 2011.

[97] Xiaofeng Wang, Kieran Conboy, and Minna Pikkarainen. Assimilation of agile practices in use. *Information Systems Journal*, 22(6):435–455, 2012.

[98] Brendan Julian, James Noble, and Craig Anslow. Agile practices in practice: towards a theory of agile adoption and process evolution. In *International Conference on Agile Software Development*, pages 3–18. Springer, 2019.

[99] Ömer Uludağ, Matheus Hauder, Martin Kleehaus, Christina Schimpfle, and Florian Matthes. Supporting large-scale agile development with domain-driven design. In *Agile Processes in Software Engineering and Extreme Programming: 19th International Conference, XP 2018, Porto, Portugal, May 21–25, 2018, Proceedings 19*, pages 232–247. Springer, 2018.

[100] H. M. Alzoubi and Y. Ramakrishna. Investigating the mediating role of information sharing strategy on agile supply chain. *Uncertain Supply Chain Management*, pages 273–284, 2020.

[101] D. Gerster, C. Dremel, W. Brenner, and P. Kelker. How enterprises adopt agile structures: a multiple-case study. In *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2019.

[102] J. M. Bass. Artefacts and agile method tailoring in large-scale offshore software development programmes. *Information and Software Technology*, 75:1–16, 2016.

[103] A. Salameh and J. M. Bass. Spotify tailoring for b2b product development. In *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 61–65, 2019.

[104] T. Dingsøyr, N. B. Moe, T. E. Fægri, and E. A. Seim. Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, 23(1):490–520, 2017.

[105] Ying Kei Tse, Minhao Zhang, Pervaiz Akhtar, and Jill MacBryde. Embracing supply chain agility: an investigation in the electronics industry. *Supply Chain Management: An International Journal*, 21(1):140–156, 2016.

[106] Maria Paasivaara, Benjamin Behm, Casper Lassenius, and Minna Hallikainen. Large-scale agile transformation at ericsson: a case study. *Empirical Software Engineering*, 23:2550–2596, 2018.

[107] Tomas Gustavsson. Institutional logics in large-scale agile software development transformations. In *Agile Processes in Software Engineering and Extreme Programming–Workshops: XP 2021 Workshops, Virtual Event, June 14–18, 2021, Revised Selected Papers 22*, pages 12–19. Springer, 2021.

[108] A. Atzberger and K. Paetzold. Current challenges of agile hardware development: What are still the pain points nowadays? In *Proceedings of the Design Society: International Conference on Engineering Design*, volume 1, pages 2209–2218, 2019.

[109] R. Kumar, K. Singh, and S. K. Jain. An empirical investigation and prioritization of barriers toward implementation of agile manufacturing in the manufacturing industry. *The TQM Journal*, 33(1):183–203, 2020.

[110] Sven Theobald and Anna Schmitt. Dependencies of agile teams–an analysis of the scaled agile framework. In *International Conference on Agile Software Development*, pages 219–226. Springer, 2020.

[111] M. K. Buniya, I. Othman, R. Y. Sunindijo, A. A. Karakhan, A. F. Kineber, and S. Durdyev. Contributions of safety critical success factors and safety program elements to overall project success. *International Journal of Occupational Safety and Ergonomics*, 29(1):129–140, 2022.

[112] Moe Huss, Daniel R Herber, and John M Borky. An agile model-based software engineering approach illustrated through the development of a health technology system. *Software*, 2(2):234–257, 2023.

[113] A. Dasgupta, M. M. Islam, O. F. Nahid, and R. Rahmatullah. Engineering management perspectives on safety culture in chemical and petrochemical plants: a systematic review. *ACADEMIC JOURNAL ON SCIENCE, TECHNOLOGY, ENGINEERING & MATHEMATICS EDUCATION*, 1(01):36–52, 2024.

[114] Tesmanian. Spacex launched 45 starlink satellites on a falcon 9 rocket, 2023.

[115] Cristina T Chaplain. Space acquisitions: Dod continues to face challenges of delayed delivery of critical space capabilities and fragmented leadership, statement of cristina t. chaplain, director, acquisition and sourcing management, testimony before the subcommittee on strategic forced, committee on armed services, us senate. In *United States. Government Accountability Office*. United States. Government Accountability Office, 2017.

[116] United States Government Accountability Office. Defense acquisitions: Dod needs to improve its implementation of agile software development. In *United States. Government Accountability Office*, September 2023.

[117] Warren K Vaneman. Enhancing model-based systems engineering with the lifecycle modeling language. In *2016 Annual IEEE Systems Conference (SysCon)*, pages 1–7. IEEE, 2016.

[118] NASA. Nasa cost estimating handbook version 4.0, appendix c: Cost estimating methodologies, 2015. Accessed: 2025-03-29.

[119] Steven R Hirshorn, Linda D Voss, and Linda K Bromley. *NASA Systems engineering handbook*, 2017.

[120] DD Defence. Mil std 882-e-standard practice for system safety. *Washington, USA: USA*, 2012.

[121] International Organization for Standardization. *ISO 26262: Road vehicles – Functional safety*. International Organization for Standardization, 2018.

[122] Simon Brown. *Overview of IEC 61508. Design of electrical/electronic/programmable electronic safety-related systems*, volume 11. IET, 2000.

[123] National Aeronautics and Space Administration. NPR 8715.3: NASA General Safety Program Requirements, 2020.

[124] Radio Technical Commission for Aeronautics (RTCA). *DO-178C: Software Considerations in Airborne Systems and Equipment Certification*. RTCA, 2011.

[125] Radio Technical Commission for Aeronautics (RTCA). *DO-326A: Airworthiness Security Process Specification*. RTCA, 2014.

[126] National Institute of Standards and Technology. NIST Special Publication 800-53: Security and Privacy Controls for Federal Information Systems and Organizations. Technical report, NIST, 2020.

[127] General Services Administration. Federal Risk and Authorization Management Program (FedRAMP), 2021.

[128] U.S. Department of State. International Traffic in Arms Regulations (ITAR), 2021.

[129] Interface Standard. Mil-std-461g: Requirements for the control of electromagnetic interference characteristics of subsystems and equipment. *Department of Defense, USA*, 2015.

[130] Federal Communications Commission. FCC Regulations on Electromagnetic Compatibility, 2021.

[131] Occupational Safety and Health Administration. OSHA Standards and Regulations, 2022.

[132] American National Standards Institute. *ANSI Safety Standards*. ANSI, 2021.

[133] U.S. Environmental Protection Agency. Environmental Regulations and Compliance Standards, 2022.

[134] European Chemicals Agency. Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH), 2022.

[135] Federal Aviation Administration. 14 CFR Part 450: Launch and Reentry Licensing Requirements, 2021.

[136] TW Wilcutt. Process for limiting orbital debris. *NASA, Washington, DC, USA, Tech. Rep. NASA-STD-8719.14 A*, 2021.

[137] U.S. Government Accountability Office. Next generation overhead persistent infrared: Cost and schedule challenges. Technical Report GAO-24-106831, U.S. Government Accountability Office, 2024.

[138] J. Eduardo Ferreira Ribeiro, João Gabriel Silva, and Ademar Aguiar. Weaving agility in safety-critical software development for aerospace: From concerns to opportunities. *IEEE Access*, 2024.

[139] Matthew Peterson and Gregory Mocko. Case study in agile for hardware: Aerospace systems. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 88407, page V006T06A004. American Society of Mechanical Engineers, 2024.

[140] Kenneth Donahue, Kiruthika Devaraj, James Mason, and Meric Ozturk. Planet's agile software development for spacecraft. In *Proceedings of Small Satellite conference*, 2024.

[141] Bernardo Araujo. How relativity space is able to 3d print a rocket in 60 days? agile is the answer., 2019. Accessed: 2025-03-07.

[142] Ryan de Freitas Bart. Is hardware agile worth it?-analyzing the spacex development process. In *AIAA SCITECH 2024 Forum*, page 2054, 2024.

[143] Jake Drutchas and Steven Eppinger. Guidance on application of agile in combined hardware and software development projects. In *Proceedings of the Design Society*, volume 2, pages 151–160, 2022.

[144] Eric Ries. Minimum viable product: a guide. *Startup lessons learned*, 3(1), 2009.

[145] Regan Stevenson, Devin Burnell, and Greg Fisher. The minimum viable product (mvp): theory and practice. *Journal of Management*, 50(8):3202–3231, 2024.

[146] Istari Digital. Istari digital unveils x-plane to become world's first digitally-certified aircraft, August 19, 2024.

[147] M Gholami Mayani, M Svendsen, and SI Oedegaard. Drilling digital twin success stories the last 10 years. In *SPE Norway Subsurface Conference*, page D011S007R001. SPE, 2018.

[148] Susan Parente. Agile quantitative risk analysis. *PM World Journal*, 7, 2018.

[149] Glen B Alleman, Thomas J Coonce, and Rick A Price. Building a credible performance measurement baseline. *The Measureable News*, 1(4), 2014.

[150] Stefan Trieflinger, Jürgen Münch, Jan Schneider, Emre Bogazköy, Patrick Eißler, Bastian Roling, and Dominic Lang. Product roadmapping processes for an uncertain market environment: A grey literature review. In *Lean and Agile Software Development: 5th International Conference, LASD 2021, Virtual Event, January 23, 2021, Proceedings 5*, pages 111–129. Springer, 2021.

[151] Claudio Mirabella, Michele Tuccillo, and Pierluigi Della Vecchia. A model-based systems engineering digital certification framework for general aviation aircraft. *Aerotecnica Missili & Spazio*, pages 1–16, 2024.

[152] Terry R Hill, Patricia Nicoli, Gregory J Pierce, Kurt Woodham, and Frank Gati. Digital transformation of the nasa engineering domain. In *2024 IEEE Aerospace Conference*, pages 1–15. IEEE, 2024.

[153] Oleksandr Liubimov, Ihor Turkin, Vladimir Pavlikov, and Lina Volobuyeva. Agile software development lifecycle and containerization technology for cubesat command and data handling module implementation. *Computation*, 11(9):182, 2023.

[154] Danijela Ciric, Bojan Lalic, Danijela Gracanin, Nemanja Tasic, Milan Delic, and Nenad Medic. Agile vs. traditional approach in project management: Strategies, challenges and reasons to introduce agile. *Procedia Manufacturing*, 39:1407–1414, 2019.

[155] Mohammad Hadi Zahedi, Alireza Rabiei Kashanaki, and Elham Farahani. Risk management framework in agile software development methodology. *International Journal of Electrical & Computer Engineering (2088-8708)*, 13(4), 2023.

[156] Marcel Vieira, Jean CR Hauck, and Santiago Matalonga. How explicit risk management is being integrated into agile methods: results from a systematic literature mapping. In *Proceedings of the XIX Brazilian Symposium on Software Quality*, pages 1–10, 2020.

[157] Joaquim Manuel Silva Cardoso Rodrigues, J Eduardo Ferreira Ribeiro, and Ademar Aguiar. Improving documentation agility in safety-critical software systems development for aerospace. In *2022 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 222–229. IEEE, 2022.

[158] Mounia Zaydi, Yassine Maleh, Hayat Zaydi, Youness Khourdifi, Bouchaib Nassereddine, and Zohra Bakouri. Agile security and compliance integration. *Agile Security in the Digital Era: Challenges and Cybersecurity Trends*, page 68, 2024.

[159] Imane Bouhali, Vincent Idasiak, Jacques Martinez, Faida Mhenni, Jean-Yves Choley, Luca Palladino, and Frederic Kratz. A collaboration framework using digital twin for dynamic simulation and requirements verification based on mbse and the mic concept. In *2024 IEEE International Systems Conference (SysCon)*, pages 1–8. IEEE, 2024.

[160] Hyungju Kim, Mary Ann Lundteigen, Andreas Hafver, and Frank Børre Pedersen. Utilization of risk priority number to systems-theoretic process analysis: A practical solution to manage a large number of unsafe control actions and loss scenarios. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 235(1):92–107, 2021.

[161] Alexander Ahlbrecht, Wanja Zaeske, and Umut Durak. Model-based stpa: towards agile safety-guided design with formalization. In *2022 IEEE International Symposium on Systems Engineering (ISSE)*, 2022.

[162] Alexander Vodyaho, Nataly Zhukova, Alexey Subbotin, and Fahem Anaam. Towards dynamic model-based agile architecting of cyber-physical systems. *Sensors*, 22(8):3078, 2022.

[163] Rene Honcak and Ana Wooley. An mbse approach for virtual verification & validation of systems with digital twins. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, pages 390–400, 2024.

[164] Luca Boggero, Pier Davide Ciampa, and Björn Nagel. An mbse architectural framework for the agile definition of system stakeholders, needs and requirements. In *AIAA Aviation 2021 Forum*, page 3076, 2021.

[165] Angela R Grotto and Jeanine K Andreassi. Mix it up? the influence of team composition on employee perceptions of stressors in a post-merger environment. *The Journal of Applied Behavioral Science*, 58(3):442–476, 2022.

[166] ERIC Trist and FRED Emery. Sociotechnical systems theory. In *Organizational Behavior 2*, pages 169–194. Routledge, 2015.

Appendix A

Survey Instrument

Table A.1: Survey Questions

	Question	Potential Answers
1.	What is your domain of work?	Medical Aerospace Automotive Energy Manufacturing Other - specify
2.	What is your current job title?	C-Suite Executive Program Manager Product Owner Engineer Other - specify
3.	How many years of experience do you have in your field?	0-5 years 6-10 years 11-15 years 16-20 years 21+ years

	Question	Potential Answers
4.	To what extent are you applying Agile to Large-Scale, Safety-Critical, Cyber-Physical Systems?	Not at all Rarely Sometimes Often Always
5.	Are you using any specific Agile frameworks in your large-scale, safety-critical, cyber-physical systems projects?	Scaled Agile Framework (SAFe) Large Scale Scrum (LeSS) Scrum@Scale Disciplined Agile (DA) Scrum Kanban Other-specify
6.	How integral is Agile to your project management practices?	Range 0-100%
7.	Why is Agile being considered for large-scale, safety-critical, cyber-physical systems in your organization? Order by priority	Adaptability Speed Cost Quality Manage Complexity Collaboration

	Question	Potential Answers
8.	What challenges have you encountered applying Agile to large-scale, safety-critical, cyber-physical systems?	Regulatory compliance Safety Resistance to change Integration with existing processes Hardware Constraints Other - specify
9.	Which augmentations do you leverage when applying Agile to large-scale, safety-critical, cyber-physical systems?	Safety Stories Adding Specialized Artifacts Modeling/Simulation/Digital Twins Additional Program Management Hazard Analysis Other - specify
10.	Do you think there are gaps in existing Agile frameworks to safely apply Agile to large-scale, safety-critical, cyber-physical systems that must be addressed?	Range 0-100%
11.	If you answered "Yes" to the previous question, please specify the gaps you have identified:	Lack of safety assurance practices Insufficient support for regulatory Lack of support for digital approaches Lack of guidance for HW/SW Coord Other - specify

	Question	Potential Answers
12.	Agile enhances the flexibility and responsiveness of large-scale, safety-critical, cyber-physical systems projects.	Strongly Disagree Disagree Neutral Agree Strongly Agree
13.	Applying Agile in large-scale, safety-critical, cyber-physical systems improves collaboration among team members.	Strongly Disagree Disagree Neutral Agree Strongly Agree
14.	Agile Methods accelerate the delivery of project milestones in large-scale, safety-critical, cyber-physical systems.	Strongly Disagree Disagree Neutral Agree Strongly Agree
15.	Agile in large-scale, safety-critical, cyber-physical systems projects enables better management of complex requirements.	Strongly Disagree Disagree Neutral Agree Strongly Agree

	Question	Potential Answers
16.	Adaptations to Agile practices are necessary for its successful implementation in large-scale, safety-critical, cyber-physical systems.	Strongly Disagree Disagree Neutral Agree Strongly Agree
17.	Regulatory compliance is maintained through Agile practices in large-scale, safety-critical, cyber-physical systems projects.	Strongly Disagree Disagree Neutral Agree Strongly Agree
18.	Agile is too risky to use when building large-scale, safety-critical, cyber-physical systems.	Strongly Disagree Disagree Neutral Agree Strongly Agree

Appendix B

Interview Instrument

Table B.1: Interview Questions

		Question	Follow up
1.	Role	Can you briefly describe your role and experience?	How has your role influenced your approach?
2.	Industry	What industry or domain do you work in? (e.g., Medical, Aerospace, Automotive, Energy, Manufacturing)?	How does the nature of your industry impact the way you implement Agile methodologies?
3.	Title	What is your current job title?	How has your experience level shaped your understanding or adaptation of Agile practices?
4.	Experience	How many years of experience do you have in this domain?	How does your position influence decision-making and the adoption of Agile practices?
5.	Extent	To what extent are you applying Agile to large-scale, safety-critical, cyber-physical systems?	What factors contributed to the level of Agile application in your projects?
6.	Frameworks	Are there any specific Agile frameworks you use?	How do these frameworks align with the needs of safety-critical projects? Do you feel any adaptations are necessary?

		Question	Follow up
7.	Challenges	What challenges have you encountered applying Agile to large-scale, safety-critical systems?	Can you explain how these challenges affected project timelines or outcomes?
8.	Adaptations	What adaptations have you made when applying Agile to these systems?	How have these adaptations impacted your processes?
9.	Tailor	In what ways have you had to adapt Agile practices better to suit your projects' safety and compliance needs?	Can you describe a situation where a standard Agile practice did not work and had to be modified for your specific environment?
10.	Waterfall	Have you found a need to balance traditional Waterfall practices with Agile in your projects?	What criteria do you use to determine when Agile is appropriate versus when traditional methods should be used?
11.	Flexibility / Collaboration	Do you believe that Agile enhances flexibility and collaboration within your teams? If so, how?	Could you provide examples of improved team dynamics or collaboration resulting from Agile?
12.	Milestones	How has Agile impacted your milestones?	In what ways has Agile impacted both short-term and long-term project goals?

		Question	Follow up
13.	Risks	Are there risks associated with using Agile in safety-critical environments? If so, how do you mitigate these risks?	What would you recommend for organizations considering Agile for similar environments?
14.	Gaps	What gaps do you see in current Agile frameworks for large-scale, safety-critical, cyber-physical systems?	How do you think these gaps can be addressed? Are there specific areas where more detailed guidance is needed?
15.	Evolution	Looking forward, how do you think Agile practices will evolve in your industry?	
16.	Technology	With the growing importance of AI, machine learning, and digital twins?	

Appendix C

Satellite Specifications

Table C.1: Satellite Specifications

	Specification	**Specification Values**
1.	Orbit	400km-1200km
2.	Launch Mass	250kg
3.	Available Payload Mass	up to 130kg
4.	Max Solar Array Power	1kW
5.	Redundancy	Dual-string
6.	Power System	66V system power 28V, 12V, 9V rails available for payload
7.	Communication Data Rate	S-band: 125 Kbps uplink 2 Mbps downlink X-Band: 650 Mbps downlink
8.	Propulsion	2150s hall effect
9.	Thrust	1.1mN
10.	Dimensions without Solar Panels	82cm x 58cm x 39cm
11.	Pointing Accuracy	10 to 50 arcseconds higher accuracy available

Appendix D

Waterfall Satellite WBS

Table D.1: Waterfall WBS

WBS ID	Name
WS	Build Satellite
WS.A	Phase A - Concept and Technology Development
WS.A.1.1	Refine Mission Objectives
WS.A.1.2	Update concept of operations
WS.A.1.3	Feasibility of assessment of Concept
WS.A.1.4	Validate solution feasible
WS.A.1.5	Baseline Mission Concept
WS.A.2	Detail the System Level Requirements
WS.A.2.1	Decompose Mission objectives to functional requirements
WS.A.2.2	Define Performance Requirements
WS.A.2.3	Allocate requirements to subsystems
WS.A.2.4	Identify Environmental and Interface Requirements
WS.A.2.5	Ensure Compliance with Standards and Regulations
WS.A.2.6	Validate/Verify System Requirements with stakeholders
WS.A.2.7	Document and Baseline Requirements
WS.A.3	Complete Initial System Architecture
WS.A.3.1	Review Satellite System Objectives and Scope
WS.A.3.2	Identify and Map Key Subsystems
WS.A.3.3	Define Component Interactions / Interfaces
WS.A.3.4	Architecture Feasibility Analysis

WBS ID	Name
WS.A.3.5	Architecture review / approved by stakeholders
WS.A.3.6	Baseline Preliminary Architecture
WS.A.4	Complete Draft Plans
WS.A.4.1	Develop Technical / Management Plans
WS.A.4.2	Develop WBS
WS.A.4.3	Develop Schedule and Resource allocation
WS.A.4.4	Develop Cost
WS.A.4.5	Review Plans with Stakeholders
WS.A.4.6	Baseline Approved Draft Plans
WS.A.5	Preliminary Verification and Validation (V & V) Planning
WS.A.5.1	Define V & V objectives and Methods
WS.A.5.2	Develop Test Plan and needed facilities
WS.A.5.3	Define Test Scenarios
WS.A.5.4	Link Test Scenarios to requirements
WS.A.5.5	Validate V & V Plan
WS.A.5.6	Baseline V & V Plan
WS.A.6	SRR/SDR
WS.A.7	Baseline Results
WS.B	Phase B - Preliminary Design and Technology Completion
WS.B.1	Refine System Requirements
WS.B.1.1	Review Preliminary Requirements
WS.B.1.2	Decompose High-Level Requirements
WS.B.1.3	Validate Feasibility
WS.B.1.4	Link Requirements to Verification Methods
WS.B.1.5	Validate Preliminary Requirements with Stakeholders

WBS ID	Name
WS.B.1.6	Baseline Phase B Requirements
WS.B.2	Develop Detailed Design
WS.B.2.1	Allocate Requirements to subsystems and components
WS.B.2.2	Refine Subsystems and Interfaces
WS.B.2.3	Incorporate Trade Study Decisions
WS.B.2.4	Conduct Simulation and Analysis
WS.B.2.5	Validate Detail Design with Stakeholders
WS.B.2.6	Baseline Detailed Design
WS.B.3	Advance Technology development
WS.B.3.1	Critical technology investigation
WS.B.3.2	Prototypes and breadboards
WS.B.3.3	Conduct Laboratory and Environmental Testing
WS.B.3.4	Validate TRL Levels
WS.B.3.5	Integrate technologies into subsystem Design
WS.B.3.6	Baseline Advanced Design
WS.B.4	Refine V & V plan
WS.B.4.1	Review Preliminary V & V Plan
WS.B.4.2	Link Requirements to Detailed V & V Methods
WS.B.4.3	Refine Test Scenarios and Environments
WS.B.4.4	Develop V & V Procedures
WS.B.4.5	Validate the V & V Plan
WS.B.4.6	Document and Finalize the Refined V & V Plan
WS.B.5	Develop Preliminary Operations Plan
WS.B.5.1	Define Mission Phases
WS.B.5.2	Identify Operational Tasks

WBS ID	Name
WS.B.5.3	Develop Nominal and Off-Nominal Procedures
WS.B.5.4	Identify Required Resources and Define Roles/ Responsibilities
WS.B.5.5	Conduct Operational Feasibility Analysis
WS.B.5.6	Validate the Ops Plan
WS.B.5.7	Baseline Preliminary Operations Plan
WS.B.6	Update Cost / Schedule Baseline
WS.B.6.1	Review Preliminary Estimates
WS.B.6.2	Integrate Updated Design Details
WS.B.6.3	Validate Resource Availability
WS.B.6.4	Conduct Feasibility Analysis
WS.B.6.5	Validate updated Cost / Schedule
WS.B.6.6	Baseline Updated Cost / Schedule
WS.B.7	Conduct PDR
WS.B.8	Baseline Phase B
WS.C	Phase C - Design and Fabrication
WS.C.1	Finalize Detailed Design
WS.C.1.1	Refine Subsystem Design / Resolve PDR Deficiency
WS.C.1.2	Perform Detailed Analyses
WS.C.1.3	Update Interface Definitions
WS.C.1.4	Design Optimization
WS.C.1.5	Finalize designs
WS.C.1.6	Conduct Critical Design Review
WS.C.1.7	Baseline Finalized Design
WS.C.2	Generate Manufacturing and Assembly Plans

WBS ID	Name
WS.C.2.1	Develop Detailed Processes for Fabricating and Assembling Components
WS.C.2.2	Define Material Specifications, Tolerances, and Quality Checks
WS.C.2.3	Define Suppliers / Procurement Needs
WS.C.2.4	Specify Assembly Sequences
WS.C.2.5	Define and Document Test and Inspection
WS.C.3	Fabricate Components
WS.C.3.1	Procure materials and components
WS.C.3.2	Manufacture components
WS.C.3.3	Perform quality inspection / Resolve Defects
WS.C.3.4	Conduct acceptance tests
WS.C.3.5	Baseline / Document Results
WS.C.4	Prepare for System Integration and Verification
WS.C.4.1	Verify Compatibility of All Subsystems
WS.C.4.2	Assemble Test Configurations for Critical Subsystems
WS.C.4.3	Develop Integration Schedules and Test Sequences
WS.C.4.4	Validate Tools, Facilities, and Procedures for Integration
WS.C.4.5	Conduct Dry Runs or Mock Assembly as Needed
WS.C.5	Update V & V Plans
WS.C.5.1	Incorporate new design details and test configurations
WS.C.5.2	Refine test cases and success criteria
WS.C.5.3	Align V & V activities with finalized design and risks
WS.C.5.4	Define test environments and resource requirements
WS.D	Phase D - Assembly, Integration, Test, Launch
WS.D.1	Assemble and Integrate System Components

WBS ID	Name
WS.D.1.1	Set up the assembly environment
WS.D.1.2	Assemble the primary and secondary structures
WS.D.1.3	Install subsystems sequentially
WS.D.1.4	Secure subsystems and perform mechanical fit checks
WS.D.1.5	Verify alignment and interfaces with ICDs
WS.D.2	Perform Subsystem Integration
WS.D.2.1	Integrate electrical and data connections
WS.D.2.2	Power-on subsystems incrementally
WS.D.2.3	Test software functionality and compatibility
WS.D.2.4	Resolve interface or communication issues
WS.D.2.5	Validate integrated subsystem performance
WS.D.3	Conduct Functional Testing
WS.D.3.1	Perform end-to-end functional testing of satellite systems
WS.D.3.2	Verify data transmission and communication
WS.D.3.3	Test payload functionality
WS.D.3.4	Record test results and address any anomalies
WS.D.4	Conduct Test Readiness (TRR)
WS.D.5	Perform Environmental Testing
WS.D.5.1	Perform Thermal Vacuum (TVAC) Testing
WS.D.5.2	Perform Vibration Testing
WS.D.5.3	Perform EMI/EMC Testing
WS.D.5.4	Acoustic and Shock Testing
WS.D.6	Conduct Final System Validation
WS.D.6.1	Conduct final end-to-end testing under simulated operational conditions

WBS ID	Name
WS.D.6.2	Perform Software-in-the-Loop (SIL) and Hardware-in-the-Loop (HIL) Test
WS.D.6.3	Validate satellite performance against mission success criteria
WS.D.6.4	Generate final validation reports and resolve any last issues
WS.D.7	Conduct Operational Readiness (ORR)
WS.D.8	Prepare for Launch
WS.D.8.1	Conduct Final Inspections
WS.D.8.2	Package the Satellite for Transport to the Launch Site
WS.D.8.3	Coordinate with the Launch Provider for Satellite Integration
WS.D.8.4	Perform Pre-Launch Readiness Checks

Appendix E

Agile Satellite WBS

Table E.1: Agile WBS

WBS ID	Name
AS	Build Satellite
AS.1	Perform Start-up/ Initialization
AS.1.1	Review Satellite Mission Requirements
AS.1.2	Perform Initial Modeling
AS.1.3	Define Work Breakdown Structure and MVP Roadmap
AS.1.4	Define Business Rhythms
AS.1.5	Instantiate Product Backlog
AS.1.6	Develop Mission Scenarios and Acceptance Test Cases
AS.1.7	Conduct PI Planning for Startup / Initialization
AS.1.8	Stand-Up Development/Modeling/Test/DTw Environments
AS.1.9	Validate with "Hello World"
AS.2	MVP 1: Basic Structure and Power System
AS.2.1	Roadmap and Backlog Review/Groom for structure
AS.2.2	Perform PI Planning for Basic Structure and Power
AS.2.3	Assemble Primary and Secondary Structures
AS.2.4	Install Solar Arrays and Batteries
AS.2.5	Functional Testing of the Basic Structure and Power System
AS.3	NVP 2: Command and Data Handling
AS.3.1	Roadmap and Backlog Review/Groom for C & DH NVP
AS.3.2	PI Planning with Risk-Adjusted Backlog for C & DH

WBS ID	Name
AS.3.3	Integrate Onboard CPU
AS.3.4	Configure Data Storage Unit
AS.3.5	Implement Communication Interface
AS.3.6	Perform End-to-End Testing and Demo to stakeholders
AS.4	NVP 3: Attitude Determination and Control
AS.4.1	Roadmap and Backlog Review/Groom for ADAC NVP
AS.4.2	Perform PI Planning with Risk-Adjusted Backlog for ADAC
AS.4.3	Integrate Reaction Control System
AS.4.4	Integrate and Configure Sensors
AS.4.5	Implement and Test Attitude Determination Algorithm
AS.4.6	Implement and Test Control System Responsiveness
AS.4.7	Perform End-to-End Integration and Testing Demo to stakeholders
AS.5	NVP 4: Propulsion System
AS.5.1	Roadmap and Backlog Review/Groom for Propulsion NVP
AS.5.2	Perform PI Planning with Risk-Adjusted Backlog for Propulsion
AS.5.3	Install and Integrate Propulsion Unit
AS.5.4	Install and Integrate Fuel System
AS.5.5	Implement and Test Thruster Control Software
AS.5.6	Perform Functional Testing
AS.5.7	Perform End-to-End Maneuverability Testing Demo to stakeholders
AS.6	NVP 5: Communication System

WBS ID	Name
AS.6.1	Roadmap and Backlog Review/Groom for Communication NVP
AS.6.2	Perform PI Planning with Risk-Adjusted Backlog for Communication
AS.6.3	Install and Integrate Communication
AS.6.4	Install and Align Antennas
AS.6.5	Integrate and Configure Transmitters and Receivers
AS.6.6	Perform Functional Testing
AS.6.7	Perform End-to-End Integration and Testing Demo to stakeholders
AS.7	NVP 6: Thermal Control System
AS.7.1	Roadmap and Backlog Review/Groom for Thermal NVP
AS.7.2	Perform PI Planning with Risk-Adjusted Backlog for Thermal
AS.7.3	Install Radiators
AS.7.4	Install and Configure Heaters
AS.7.5	Apply Insulation
AS.7.6	Perform Functional Testing
AS.7.7	Perform End-to-End Integration and Testing Demo to stakeholders
AS.8	NVP 7: Payload System
AS.8.1	Roadmap and Backlog Review/Groom for Payload NVP
AS.8.2	Perform PI Planning with Risk-Adjusted Backlog for Payload
AS.8.3	Install and Integrate Scientific Instruments
AS.8.4	Install and Integrate Payload
AS.8.5	Implement and Test Payload Control Software

WBS ID	Name
AS.8.6	Perform Functional Testing
AS.8.7	Perform End-to-End Integration and Testing Demo to stakeholders
AS.9	NVP 8: Full System Integration and Test
AS.9.1	Roadmap and Backlog Review/Groom for Full System Integration NVP
AS.9.2	Perform PI Planning with Risk-Adjusted Backlog for Full System Integration
AS.9.3	Assemble Subsystems
AS.9.4	Perform Functional Testing
AS.9.5	Perform Environmental Testing
AS.9.6	Perform End-to-End Mission Validation Demo to stakeholders
AS.10	NVP 9: Launch Readiness
AS.10.1	Review and Refine Final Launch NVP
AS.10.2	Perform PI Planning with Risk-Adjusted Backlog for Launch Readiness
AS.10.3	Conduct Final Pre-Launch Inspections
AS.10.4	Validate Ground Control Interface
AS.10.5	Integrate with Launch Vehicle
AS.10.6	Perform Final System Checks
AS.10.7	Confirm Launch Readiness

Appendix F

Acronyms

Acronym	Definition
ACDS	Attitude Determination and Control
AEHF	Advanced Extremely High Frequency
ATDD	Acceptance Test Driven Development
AI	Artificial Intelligence
AIT	Assembly, Integration, and Test
APM	Agile Portfolio Management
BoM	Bill of Materials
BDD	Behavior Driven Development
C2	Command and Control
cATO	Continuous Authority to Operate
CD&H	Command and Data Handling
CDR	Critical Design Review
CI/CD	Continuous Integration and Continuous Delivery
CP	Cyber-Physical
CPS	Cyber-Physical Systems
CoP	Communities of Practice
DA	Disciplined Agility
DE	Digital Engineering
DoD	Definition of Done
DT	Digital Thread
DTw	Digital Twin
EMC	Electromagnetic Compatibility

Acronym	Definition
EMI	Electromagnetic Interference
EAT	Executive Action Team
ECSS	European Cooperation for Space Standardization
FAA	Federal Aviation Administration
FCC	Federal Communications Commission
FMEA	Failure Mode Effects Analysis
FRR	Flight Readiness Review
FTA	Fault Tree Analysis
GAO	Government Accountability Office
GDPR	General Data Rights Protection
GPS	Global Position Systems
HIL	Hardware In the Loop
HCPS	Human-Cyber-Physical Systems
ICD	Interface Control Document
IRB	Institutional Review Board
IoT	Internet of Things
ITAR	International Traffic in Arms Regulations
ITU	International Telecommunication Union
KBPS	Kilobits Per Second
KG	Kilograms
LeSS	Large Scale Scrum
LEO	Low Earth Orbit
LML	Lifecycle modeling language
LRR	Launch Readiness Review
LS	Large Scale
LS/SC/CP	Large-Scale, Safety-Critical, Cyber-Physical

Acronym	Definition
MBE	Model Based Engineering
MBPS	Megabits per second
MBSE	Model Based Systems Engineering
MLI	Multi-layer Insulation
MVP	Minimum Viable Product
NIT	Nexus Integration Team
NVP	Next Viable Product
OBS	Onboard Computer
OSHA	Occupational Safety and Health Administration
PDR	Preliminary Design Review
PDU	Power Distribution Unit
PMBok	Program Management Book of Knowledge
PI	Program Increment
PO	Product Owner
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
RAGE	Recipes for Agile Governance
RAMS	Reliability, Availability, Maintainability, and Safety
RAMMS	Risk Assessment and Mitigation Management System
RF	Radio Frequency
R-Scrum	Regulated Scrum
SAFe	Scaled Agile Framework
SBIRS	Space Based Infrared System
Safe-Scrum	Safe Scrum
SC	Safety Critical
SCA3DA	Safety-Critical Agile Adoption Assessment

Acronym	Definition
SDR	Software Defined Radios
SETR	Systems Engineering Technical Review
SIL	Software in the Loop
SLR	Systematic Literature Review
SM	Scrum Master
SME	Subject Matter Expert
SoS	Scrum of Scrums
SRR	System Requirements Review
STPA	System-Theoretic Process Analysis
SysML	Systems Modeling Language
SVR	System Verification Review
TLR	Technical Readiness Level
TRR	Test Readiness Review
TT&C	Telemetry, Tracking, and Control
TVAC	Thermal Vacuum Chamber
UAF	Unified Architecture Framework
XP	eXtreme Programming