NDIA

National Defense Industrial Association
Systems Engineering Division

Moving from predictive planning to empirical planning for Systems Engineering

Evolving Systems Engineering for a Modern Engineering Product Development Environment

March 2024
Version 1.0

National Defense Industrial Association (NDIA)
2101 Wilson Blvd., Suite 700
Arlington, VA 22201
(703) 522-1820
Fax (703) 522-1885
www.ndia.org

Table of Contents

Introduction

Facing a rapidly changing environment including the emergence of near-peer threats, mounting competition on the world stage, and advances in technology, customers and industrial partners recognize the urgent need to adopt modern practices and digital capabilities into the development and delivery of large safety-critical, cyber-physical military systems. Adopting iterative and incremental practices into the cyber-physical realm enables the opportunity to improve delivery of Major Capability Acquisitions (MCA) as defined by DoD 5000.85. One of the challenges MCAs need to overcome is inconsistency in the implementation of such development practices.

In June 2023, the GAO report to congressional committees highlighted the concern that "Programs Are Not Consistently Implementing Practices That Can Help Accelerate Acquisitions" (U.S.GAO 2023). While the adoption of iterative and Agile practices continues to increase in software, GAO reports there is "limited implementation of the Defense Science Board's recommended practices to accelerate software development" and there is not sufficient guidance regarding when programs should perform verification testing which would allow them to safely incorporate innovative technologies to respond to emerging threats.

It is imperative we deliver value faster and more frequently to our user base. We must respond to this need by leveraging modern engineering and digital environments to reduce the ever-growing lead times. These environments offer capabilities and success patterns from multiple bodies of knowledge such as Agile, DevSecOps, Lean, Systems Engineering and Model-Based Systems Engineering (MBSE), cyber-security, and emerging digital capabilities (e.g. digital thread, digital twins, and AI/ML). The practices within these bodies of knowledge provide an effective approach to responsive product development in a risk- and cost-managed environment. The DoD is encouraging and in some instances such as the DoD 5000.87 requiring "government and contractor software teams to use modern iterative software development methodologies" such as Agile, Lean, and DevSecOps (DODI5000.87 2020). Acquisition practices grounded in Agile and DevSecOps methods enable the iterative and incremental delivery of capabilities for weapon systems to respond to the ever-changing threat environment.

Many organizations within the defense community struggle with adapting modern engineering practices. Systems Engineering coupled with Agile and DevSecOps practices is a key enabler to successful execution in an environment where focus is on delivering product while providing ongoing management of the technical baseline and incorporation of new information. This approach provides a consistent understanding of the current and target state of the system across the entire program lifecycle. The responsibilities of the Systems Engineering function do not change in a modern engineering environment, but the techniques used to plan and execute the work must be updated to support the reality of program execution where comprehensive specification up front may no longer be desired - or even possible.

This paper is written for the Department of Defense and the Defense Industrial Base who want to increase the speed of value delivered to users through the adoption of modern engineering and digital practices.

Background

One of the first reports to highlight the concern of long lead time for the delivery of capabilities to the warfighter was presented to the Defense Science Board Task Force in 2009 and was titled, Department of Defense Policies and Procedures for the Acquisition of Information Technology (OUSD 2009). An analysis of 32 major automated information system acquisitions, conducted by the Office of the Assistant Secretary of Defense for Networks and Information Integration (OASD (NII)), calculated that the average time to deliver an initial program capability was 91 months which is 7.5 years. Unfortunately, delivery times have continued to get worse not better according to latest report from Air Force Material Command and is now averaging greater than 15 years. (Hurst 2023).

Actions to address this growing concern continue to evolve.

Modern engineering practices such as Agile and DevOps are needed to manage performance.

In 2014 the Office of the Secretary of Defense / Performance Assessments and Root Cause Analyses (PARCA) established a joint Industry Agile and Earned Value Management (EVM) Collaboration group to address concerns raised by the services that Agile programs within their purview did not have a good answer for how to manage and balance cost, schedule, and technical objectives. This activity resulted in a Department of Defense Agile and EVM Desk Guide for Government program offices and the NDIA Agile and EVM best practices guide for industry program offices (NDIA IPMD 2022) (OSD 2020). While these documents target Agile programs required to implement EVM, there are foundational best practices gleaned from this learning that can be leveraged and applied to the planning and managing of the technical objectives. Specifically, these documents recommend establishing traceability between the Agile technical work, the work breakdown structure (WBS), and the integrated master schedule (IMS) and claiming Agile and iterative progress to inform overall program cost, schedule, and technical status.

The DoD has continued to mature and monitor modern engineering practices.

In November 2020, DoD issued DoD Instruction 5000.89 establishing policy and procedures for Testing and evaluation across five AAF pathways—including the major capability acquisition and MTA pathways—that address cyber-security planning and execution (DODI 5000.89, 2020). DoD leadership has emphasized key practices, such as iterative development. However, most of the 39 programs reported using a modern software development approach to deliver working software for user feedback more slowly than recommended by the industry's Agile practices, which call for rapid, frequent delivery of software and fast feedback cycles. As a result, these programs may lose out on some of the benefits of using a modern approach.

Launch of the Digital Materiel Management Industry Association Consortium.
In 2023, the U.S. Air Force Material Command, along with the Army, defense industry and academia launched the Digital Material Management Industry Association Consortium to collaborate on Digital Materiel Management and Digital Transformation solutions and enable a cooperative framework across engineering, program management, contracting, logistics, financial management, and test and evaluation fields. The intent is to collaborate across the defense community to accelerate integrated capability delivery to the warfighter (DAFDTO 2023).

For additional listing of responses on the need to be more agile and responsive see Appendix A: Additional Background.

Purpose

The purpose of this paper is to recommend improvements to the implementation of current Systems Engineering planning and review practices integral to the DoD Acquisition process.

While not all inclusive, the recommendations proposed in this paper are aligned with modern Systems Engineering practices, the efforts of the Air Force Material Command’s Digital Material Management Industry Association Consortium, NDIA Systems Engineering Division goals, and Digital Building Code for Digital Engineering and Management (DAF 2023).
Our effort is one of many being worked across the DoD acquisition community to address mounting concerns in our ability to act with speed and agility. While in alignment with the efforts, this paper specifically proposes an approach to improve and modernize Systems Engineering requirement management, design, and review activities as part of the acquisition process with emphasis on value delivery. In addition, improvements are proposed that create an opportunity for the contractor and the DoD to review the program's progress more frequently based on iterative reviews enabling the DoD to provide valuable feedback to the contractor based on visible progress using models and demonstrations.
Problem Description

The Department of Defense (DoD) Acquisition Process for major capability acquisitions as illustrated in Figure 1: Major Capability Acquisition, DAU includes multiple phase-gated technical reviews like SRR (System Requirements Review), SFR (System Functional Review), PDR (Preliminary Design Review), CDR (Critical Design Review), and TRR (Test Readiness Review).

These reviews were developed to ensure that the system was at the appropriate level of technical maturity to proceed to the next phase. The current approach for implementing these reviews has faced several challenges and criticisms over recent years. Key issues that have been expressed include extensive lead times, high-cost profile, bureaucratic complexity, inability to adapt to rapid change, scope creep, and overemphasis on documentation and under-emphasis on delivery of value. The phase-gated structure is inadequate to maintain a technical baseline that is expected to evolve over development increments. An alternative approach is proposed and is based on maturing the technical baseline incrementally so that requirements, architecture, design, test approaches, and system documentation remain current and consistent.

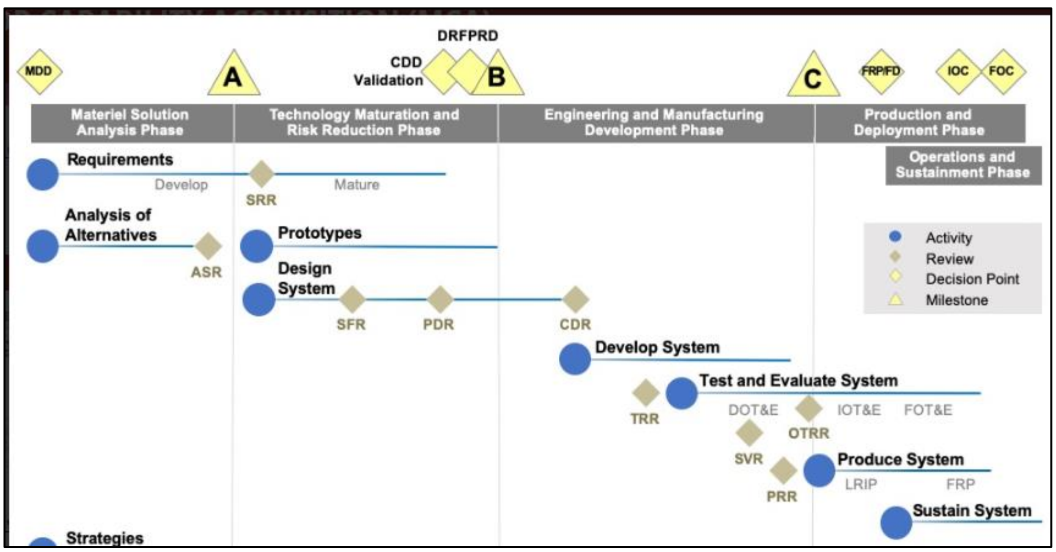


Figure 1: Major Capability Acquisition, DAU

[Major Capability Acquisition (MCA) | Adaptive Acquisition Framework (dau.edu)]

Improvement in the acquisition for software is being realized and has been documented in DoD 5000.87, Software Acquisition Pathway. The approach as illustrated in Figure 2: Lifecycle view of software acquisition, specifically states “Programs will require government and contractor software teams to use modern iterative software development methodologies (e.g., agile or lean), modern tools and techniques (e.g., development, security, and operations (DevSecOps)), and human-centered design processes to iteratively deliver software to meet the users’ priority needs (DODI5000.87 2020). This has been a cornerstone for improving the delivery of software capabilities.

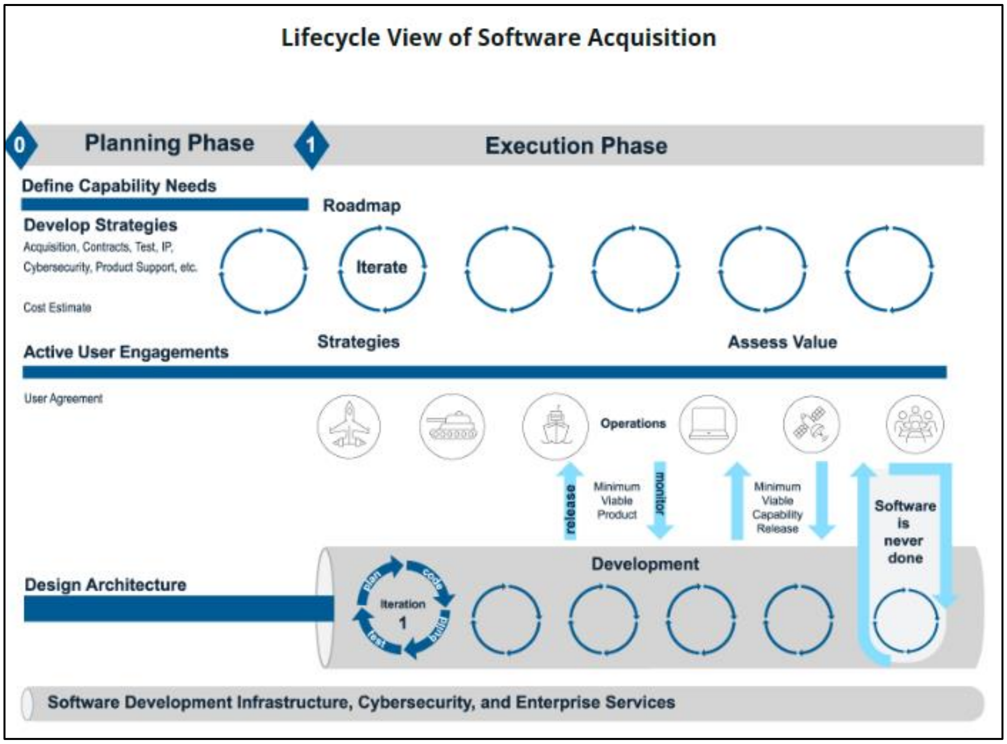


Figure 2: Lifecycle view of software acquisition
(src: Defense Acquisition University (DAU) software acquisition www.dau.edu)

This paper is to offer recommendations to respond to these challenges.

Recommendations

Collaborating with industry partners, government representatives, and general conference engagements, several best practices are recommended as part of the advancement and modernization of Systems Engineering reviews.

The best practices described in this section are:

- Establish a shared technical vision and solution intent
- Use right-sized planning that is refined over time through short learning cycles
- Perform product-based decomposition implemented through cross-functional teams
- Iteratively and incrementally build, verify, and validate the system
- Invest in the digital ecosystem as an enabler for agility and responsiveness

It is imperative that large-scale, safety-critical, cyber-physical systems ***establish shared technical vision***, ***utilize right-size planning***, ***perform product-based decomposition***, ***incrementally verify and validate the system*** through demonstration of working capabilities that are defined in key artifacts, while ***investing in the digital ecosystem***.

Establish a shared technical vision and solution intent

Establishing a shared technical vision is a critical step in ensuring all stakeholders can move toward a common goal. While it is understood that this practice is already prevalent in our traditional approaches to engineering and solution development, there is some difference in how this is approached in an adaptive environment. A shared technical vision is necessary to establish an early view of the solution intent. The technical vision and solution intent are clearly aligned to address the problem space. As the technical vision and solution intent are shaped and refined, end users (or proxies) are engaged to help further establish and document the essential behaviors of the system. Stakeholders' needs are defined and shaped into clear, measurable objectives that are specific enough to guide the technical work while remaining flexible enough to allow for innovation.

The solution intent is also met with constraints. As with all development efforts, the constraints of the system and foundational requirements such as compliance, safety, security, and interoperability with legacy systems must be understood. The constraints are addressed using an iterative and demonstration process for ongoing validation of the system. With safety critical systems it is imperative that quality is built in to reduce program and safety risks. This iterative approach is also used to guide the architecture and design as well as the incremental standard to assess progress at the integrated system level.
An example of this approach is a fighter aircraft that must produce sufficient speed to provide the required offensive and defensive warfighting capabilities and ensure airspace superiority. How the airplane is implemented to meet these capabilities will be discovered through the iterative, Agile process. The verifiable specification of “sufficient speed” is further refined and iteratively demonstrated to meet the emerging needs of the warfighter. The emphasis on stakeholder engagement and fast feedback loops provides ongoing validation of working, demonstrable capability as the aircraft is developed.

Major capability acquisitions are often large and complex solutions requiring alignment across many teams and suppliers. While the system is defined and developed incrementally, key decision points such as milestones, major program events, and contract deliverables are identified and clearly communicated. These decision points feed into the incremental delivery cycle and are identified jointly between the end-users, government program office, and the contractor. This agreement forms the basis of a product roadmap and is based on system decomposition, minimal viable product demonstrations (MVPs), and risk prioritization.

Modeling of the solution is key. Using models, the technical vision and solution intent continue to evolve and over the course of iterations becomes real capability that can be demonstrated and delivered. Models may include system logical views, functional models, use cases and scenarios, etc. that help understand how the system components and features come together as a whole system. Although the use of models is not a new concept, the implementation and development of these models and an iterative fashion coupled with the idea of living in the model is a quickly evolving practice. Frequent demonstrations as part of the agile and iterative process, focus on the integrated system capabilities that are defined in the model. These models are part of a larger digital ecosystem and inform the digital thread. The digital thread is an integral part of the digital ecosystem and is leveraged to model and demonstrate iterative performance of the system.

The technical vision and solution intent aligns all stakeholders, from individual contributors to program management to users, around a shared understanding of the path forward while enabling collaboration amongst all parties to achieve mission success.

Use Right size planning

> As an Agile program anticipates the development of a theme or epic in the near-term, the program should define the requirements into smaller efforts with more granularity so that the team can properly plan and execute the work.
>
> GAO-20-590G

The misconception that to be Agile, any early definition of the technical system is an anti-pattern since Agile methods allow system implementation to be discovered and emerge; thus, there is no need for system engineering. This misconception has led some teams to jump into implementation without architectural foundations, technical vision, or implementation strategy. Success patterns from the Systems Engineering community have taught us the importance of building architectural enablers that run just ahead of development and the importance of a strategic roadmap that provides a shared understanding amongst all parties of the direction development is headed. Furthermore, experience has shown that programs that do not have an accurate understanding of program status against stakeholder requirements through regular validated learning can lead to significant rework further downstream. Applying multiple horizons of planning provides an approach to address these concerns.

Major weapon systems built within the DoD have significant complexity requiring a multi-tiered planning approach for alignment and integrated system definition. This approach to planning is often referred to as multiple horizons of planning as illustrated in Figure 3 Multiple horizons of planning. Each level of planning has a level of system definition supporting it from operational views that describe the main operational concepts of the system to integrated system views to low level models at the team level as the build out detailed capabilities. Each horizon of planning has a roadmap that provides visibility into the path forward with each level of planning becoming more detailed over time. While multiple levels of decomposition have existed, the primary difference in this approach is that the roadmaps and plans at each level are a time-phased forecast of experiments and learning cycles that are constantly improved and refined over time.

It is recommended that large defense programs create a multi-year roadmap depicting the major capabilities to be delivered; however the level of fidelity of these roadmaps is different that the detailed IMS schedules most programs are used to having. The multi-year roadmap needs to be decomposed into an annual roadmap and then a quarterly incremental plan that is frequently refined.

The annual roadmap depicts the functionalities developed over the quarters while identifying long lead items, supplier integration points, and the demonstration of MVPs.
MVPs (Minimal Viable Product/Next Viable Product) may require several quarters to build but are critical and necessary for buying down program execution risks. An MVP is something that can be demonstrated in a simulated, virtual, or physical environment (with the potential for use by end users). An MVP has enough capability to buy down risk and validate a hypothesis about how the system is supposed to work. MVPs also provide an opportunity for the customer to tie demonstrated system functionality to payment milestones.

The annual roadmap not only identifies MVPs but is further defined by a quarterly increment plan which identifies the features that the agile teams are building and demonstrated at the integrated system level. Detailed feature planning happens just prior to the start of the quarterly increment where it will be developed and tested. Systems Engineering ensures the features are aligned with the major systems capabilities and the architecture is also evolving along with the feature development.

At the sprint/ iteration level, is where teams define their detailed work (often referred to as *stories*) based on priorities, develop a shared understanding of what needs to be built, and plan how they will demonstrate completion of that work. Iteration level work is not planned years out in advance but is done closer to execution of that work as quarterly plans evolve. Each horizon offers the opportunity for user feedback and the opportunity to improve how they work. These are learning opportunities that continuously feed into the next cycle for continuous improvement and modernization of the system.

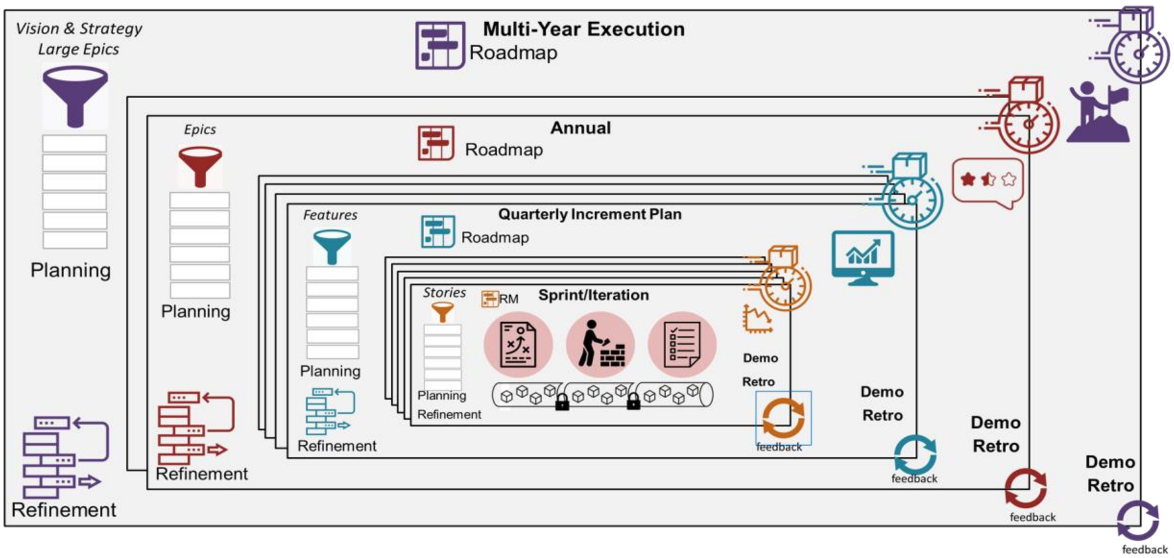


Figure 3 Multiple horizons of planning

Teams begin building iterative designs and development on day one of the program realizing the type of work may vary depending on the maturity of the system, its architecture, and detailed design of the system. Models and designs are incrementally developed and reviewed with regular validation of solution intent through integrated demonstration as illustrated in Figure 4 Iterative and Incremental Delivery.

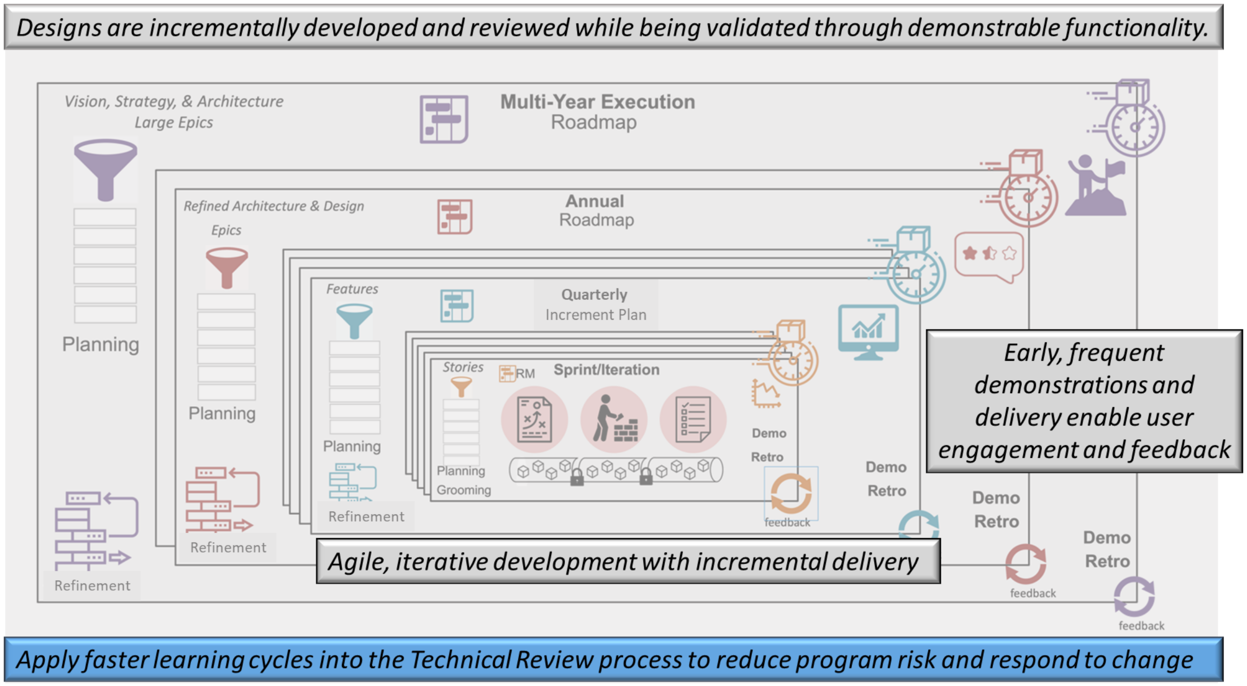


Figure 4 Iterative and Incremental Delivery

Progress is based on these demonstrations of integrated functionality. For cyber-physical systems these early demonstrations are likely performed in simulated or virtual environments and may take advantage of prototypes, 3D printed materials, or other digital capabilities.

Perform Product-Based Decomposition

In alignment with MIL-STD-881 the system is defined using a product-based approach to support incremental definition and development of modular system elements. In Figure 5 Work breakdown structure examples, there are two alternatives provided for an Automotive Driver Assist System (ADAS). The first example, Functional Decomposition work breakdown structure (WBS), focuses on the functions of an organization and results in hand-offs across the functions with a lack of emphasis on demonstrating integrated system capabilities. While each function may perform well, there is a lack of visibility in the true progress of integrated capabilities. Regular demonstration of integrated capabilities is key for buying down program risk. Therefore, the preference is on the Product-based WBS which focuses on the implementation of functionality and shifts planning and progress evaluation to focus on the completion of elements of the working system. Cross functional teams are product-based teams whose members have the skills necessary to build, test, and demonstrate the features and components of the system.

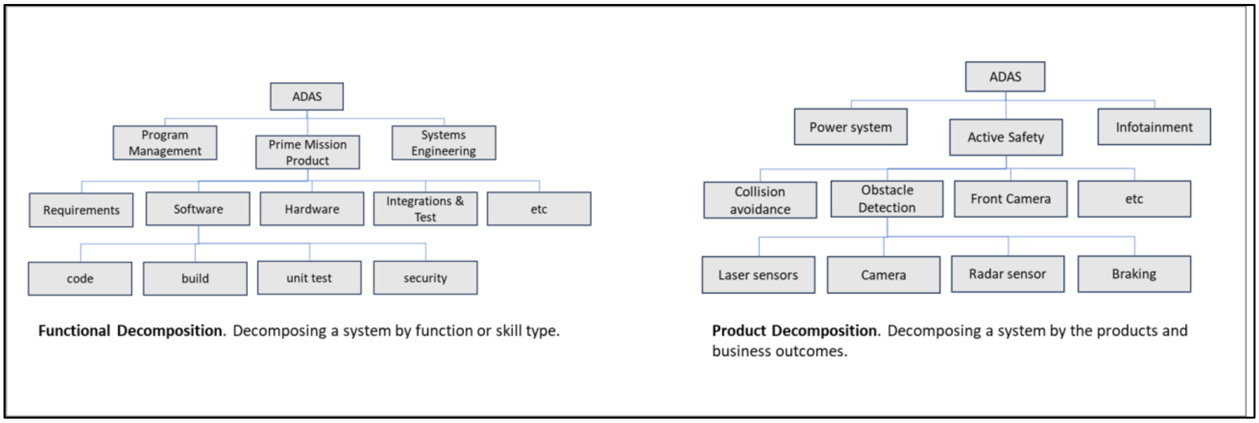


Figure 5 Work breakdown structure examples

Often government and contractors reporting structures are functionally based which often results in functional decompositions. However, the organizational structure is separate and unique from the WBS. For the building of products and systems, a product-based WBS is more appropriate and then build and align cross-functional teams to implement the work. While the teams are aligned around the product, this does not mean the formal reporting structure of the organization needs to change. This model often lends itself to a matrixed organization.

Iteratively and incrementally verify and validate the system

There are multiple approaches to verify and validate the system; however, an Agile approach at the integrated system level provides the lowest risk of rework and the greatest opportunity to optimize feedback.

The Department of Defense (DoD) commonly uses a series of technical reviews, as illustrated in Table 1 Technical Reviews, throughout the acquisition life cycle of a system.

Table 1 Technical Reviews

Review	Purpose	Intent
System Requirements Review (SRR)	Assesses the system requirements captured against the mission needs.	Confirm requirements align with stakeholder expectations
Preliminary Design Review (PDR)	Evaluates the initial design against mission needs to ensure that it meets requirements.	Ensure proposed design and architecture is valid and meets requirements within acceptable risk.
Critical Design Review (CDR)	Evaluates the matured design against mission needs to ensure that it meets requirements.	Ensure matured design and architecture meets requirements within acceptable risk.
Test Readiness Review (TRR)	evaluate the system's readiness for testing by examining if the	Validates the test plan is adequate to minimize risk.

Review	Purpose	Intent
	objectives, methods, resources, and plans are well-defined.	
Operational Readiness Review (ORR)	Evaluates whether the system is ready for operation in the intended context.	Validates that the system can be supported operationally.

Derived from IEEE Standard for Technical Reviews and Audits on Defense Programs (IEEE 15288.2-2014)

Systems Engineering Technical Reviews (SETR) are essential oversight tools that program managers can use to assess the state of the system and program and redirect activity if necessary (DODI 5000.88 Engineering of Defense Systems, 2020). These provide a venue to establish the technical baseline, assess the system's technical maturity, and review and assess technical risks. At each review, the PM, to the extent practicable, uses information from the digital authoritative source of truth and demonstrations to assess key risks, issues, opportunities, and mitigation plans to understand cost, schedule, and performance implications.

These reviews are part of the DoD's structured approach to Systems Engineering with the intent to ensure that a program has the level of technical maturity to proceed to the next development phase at an acceptable risk level. As such, these are typically contractual milestones as well as technical phase gates. Early planning and negotiation of the customer-contractor agreements must identify the system life cycle model and how technical reviews support the development cadence with the ability to adapt to evolving requirements. Leveraging the power of digital environments and integrated system level demonstrations, progress can be measured against what can be demonstrated and informs the overall status toward each milestone. Figure 6 Technical Reviews across the DoD acquisition life cycle illustrates different milestone reviews typically conducted for large DoD programs.

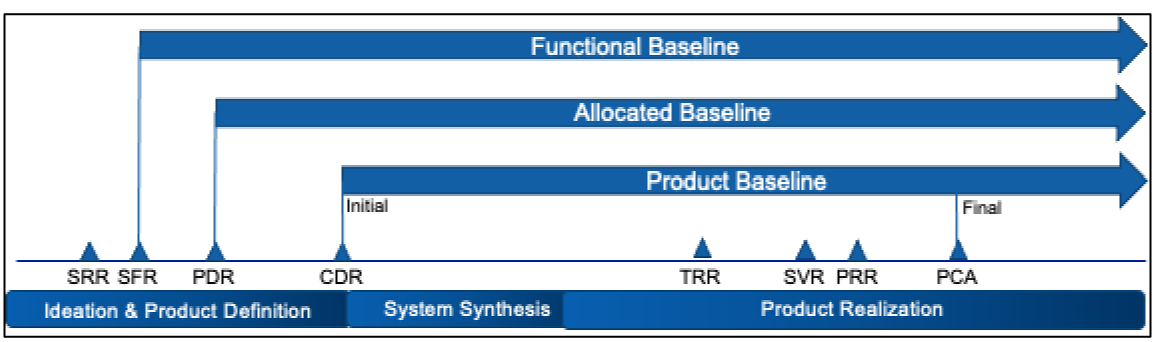


Figure 6 Technical Reviews across the DoD acquisition life cycle

As industry shifts to an environment of greater concurrent engineering, the previous phases of system maturity are distributed across development, requiring an evolution in the timing and content of reviews as depicted in Figure 7 Incremental Technical Reviews manage evolving baselines.

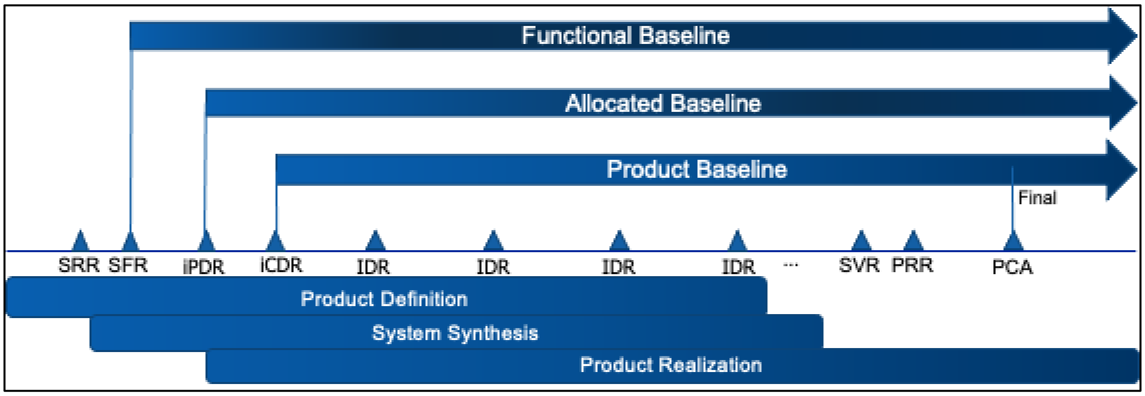


Figure 7 Incremental Technical Reviews manage evolving baselines

In the book *The Fifth Discipline: The Art and Practice of the Learning Organization*, Dr. Peter Senge described feedback loops as a critical part of learning organizations and systems thinking (Senge 2006). Feedback and feedforward loops cause effects within a system to reinforce or counteract each other. When feedback is delayed and infrequent, mounting variation continues to grow which leads to undesirable outcomes and schedule slippage (Kim. G, et al., 2021). Thus, the need for rapid program execution naturally drives the industry to iterative and incremental methods.

When employing these methods, verification and validation are employed through each incremental integration of the system. The developmental regression test suite evolves as the system develops to ensure that each increment is building on a solid foundation. In addition, this approach allows the maturation of acceptance tests such as Developmental/Operational Tests (DT/OT) and Day in the Life (DITL) scenarios so that at every increment, tests are developed to match the current state of the system, supporting Operational Assessment. This allows for feedback on the tests and test approach as well as the system itself. Ongoing compliance testing provides early identification and mitigation of issues to reduce the risk of late discovery and maintain a continuously accreditable posture. Thus, the test posture of the system remains current with the requirements, architecture, design, and developed functionality of the system throughout the development lifecycle. The system is tested to the existing state of development, reducing the risk of late discovery and resultant program impacts.

…the traditional requirements development, Iterative and incremental technical reviews at shorter intervals provide a formalized means of providing feedback on the state of the system and test approaches. Each incremental review satisfies the intent of all the reviews from Table 2 Incremental Delivery Review Content, for the current state of the system, focusing on the functionality developed in the preceding increment (TRR, ORR) and the planned development for the succeeding increment (SRR, SFR, PDR, CDR). These reviews augment the ongoing technical feedback provided during the development process. This workflow requires a different staffing and funding plan for Systems Engineering effort for both stakeholder and developer teams to spread the effort more evenly across the period of performance to accommodate the ongoing maintenance of the technical baseline.

> preliminary design review, and critical design review events may be replaced by incremental design reviews, and, if needed, system-level reviews.
>
> GAO 20-590G (2020)

Table 2 Incremental Delivery Review Content

ΔSRR/SRR/PDR/CDR	ΔTRR/ORR
• Review Mission Objectives, Ground Rules & Assumptions, Functionality, Compliance, Priorities	• Demonstrate System Functionality
• Review Changes to System Models (Architecture, Verification), Risks, Supplier Data, Definition of Done	• Review Test Approach and Results
• Review Planned Work and Integration Points, Performance Measures	• Review Compliance Posture
	• Review Developmental Metrics

In 2014, MITRE released a publication providing consideration on the need and how to tailor DoD acquisition to rapidly delivery capabilities as depicted in Figure 8 Potential Agile Development Model (2014, Modigliani and Chang). Specifically, they recommend programs to repurpose traditional milestone reviews to be more agile and iterative, enabling the development team to address issues sooner. These initial considerations continue to be in line with the ongoing recommendations from industry and some early adopting programs.

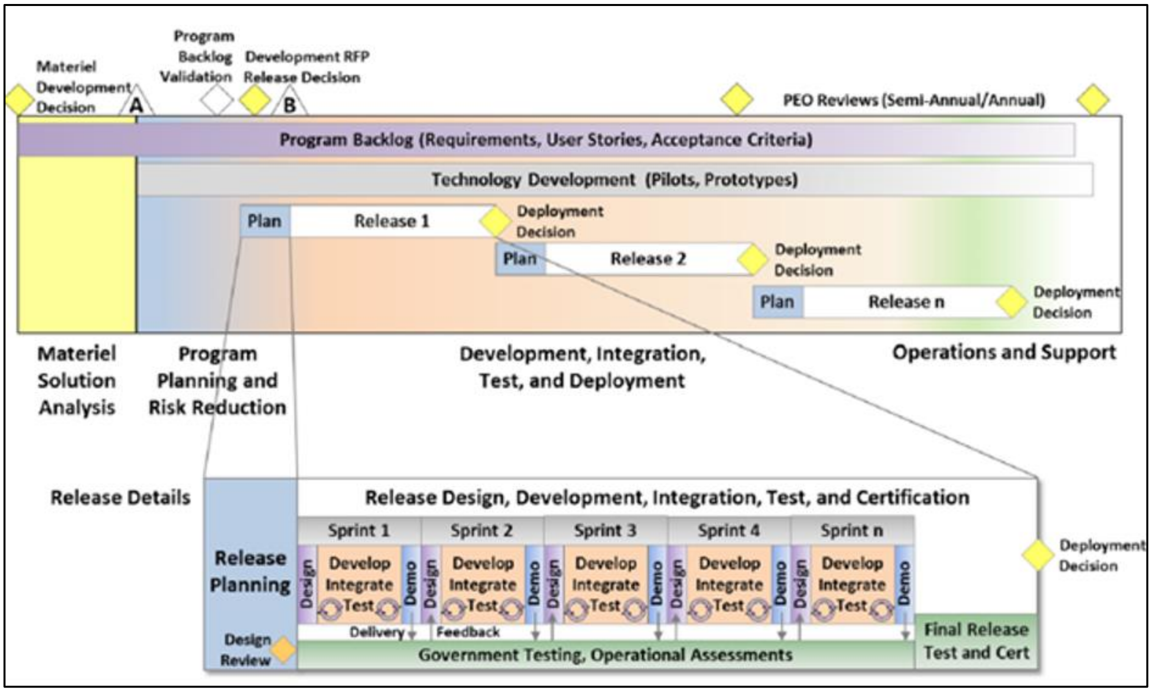


Figure 8 Potential Agile Development Model (2014, Modigliani and Chang)

The objective of shorter reviews is to focus on regular, minor capability releases and involve users in system specification and development to ensure operational value. Current recommendations build on these concepts to incorporate the need for even more rapid development and the additional capabilities of modern engineering product development environments.

Agile, as depicted in Figure 9 Agile Lifecycle, is an empirical lifecycle. Requirements, designs, and solutions evolve through collaboration between self-organizing cross-functional teams with regular user feedback during integrated system demonstrations. For complex cyber-physical systems demonstrations may be conducted in virtual environments initially and as the system evolves it is also demonstrated in the physical space. Agile promotes the ability to adapt in a rapidly changing landscape through its iterative cycle. This enables increased responsiveness to changing mission needs and is applicable across acquisition phases.

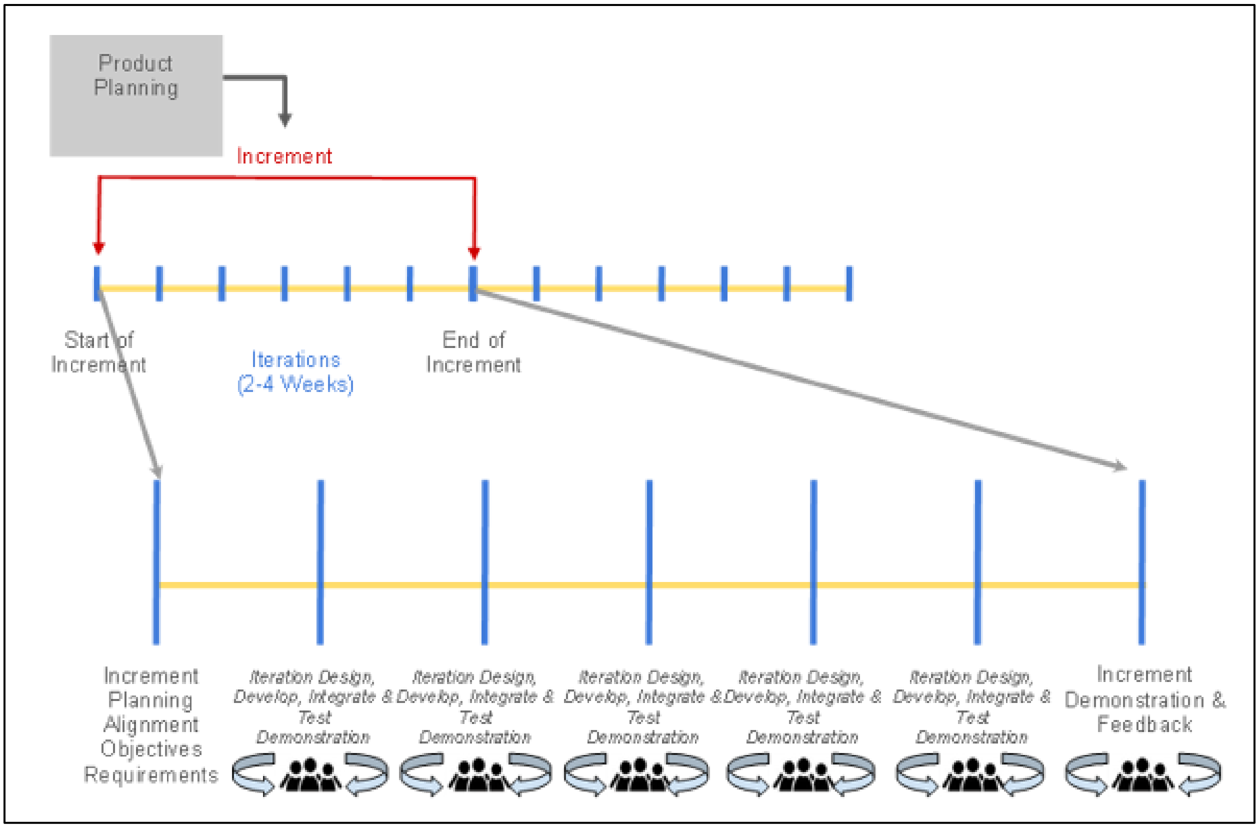


Figure 9 Agile Lifecycle

This approach reduces risk through early and frequent integration and demonstrations at the systems level. It provides program managers with better visibility into progress by seeing what is working at the integrated system level.

Conducting Agile Incremental delivery reviews

To conduct incremental delivery as outlined in Figure 10 Incremental Delivery, it is important that the program spend time in adequate Agile product planning and then building the development roadmap. Initial product planning consists of several activities such as understanding the current architecture of the system, capturing a shared vision, the mission and solution intent, formation of product teams, and product decomposition that is used to shape feature priorities.

Shifting from product planning into execution, teams plan a series of event-based technical reviews through a series of increments, reviewing delivered capability, and making any necessary adjustments. Increments are often aligned to quarters although some programs opt for a shorter duration. Each increment planning begins by doing a walkthrough of the architecture definition, model or interface changes, and ensures alignment across the teams.

Capability and technical reviews are incremental and iterative, built on previous reviews, and conducted in the customer's presence. A Cross-Functional Team (CFT) comprising skills in systems, software, hardware, test, and specialty disciplines come together for the development, testing, and demonstration of integrated features and to review incremental progress with the customer. Teams are living in the model which is updated alongside feature development and ensures traceability between the models, requirements, and new system functionality. This means CDRLs are no longer paper documents that are often outdated by the time they reach the customer but instead the information is in model and stays up to date. This modernized approach emphasizes the need for investment in the digital ecosystem.

This approach *reduces program risk* exposure and enables course adjustments in small cycles. Changes in small cycles are more cost effective than making changes late in the game where schedule slippage and cost overruns are often the result.

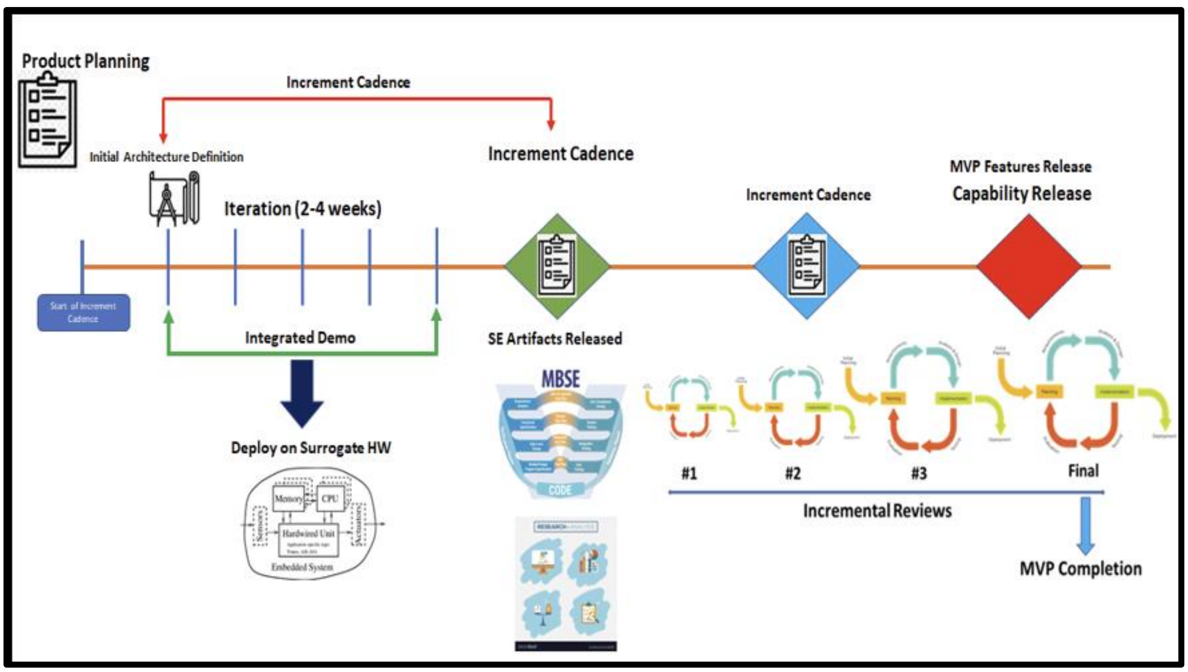


Figure 10 Incremental Delivery

Product Planning - Product Planning is a continuous control activity that encompasses the entire product goals of the program and establishes the Product Backlog and Product Roadmap. **Increment Cadence** – Standard time box for program planning and implementation, typically quarterly, that culminates in a demonstrable portion of the solution (could be demonstrable in a simulation environment, early hardware or final system).

Capability Release - A Capability Release is typically based on customer agreement that specifies what the product must do in context to the release plan. Could be for a control group if there is enough feedback that can be obtained.
Deployment – Any release that is not customer driven but instead is team driven. Something that is “deployed” has not been released to a customer but is something that is demonstrable so the development team can learn and get feedback from stakeholders.

The timing of these incremental reviews may vary but quarterly is the most frequently requested interval by the DoD as it aligns with a natural business rhythm already established. Each incremental review is conducted in collaboration with the customer and includes a verifiable portion of the completed product. The customer is contextual, for example in a DoD cyber-physical program, it is the service that engages with the contractor within the DoD such as U.S. Army, U.S. Navy, or U.S. Air Force.

When using an Agile approach to incremental milestone reviews, illustrated in Figure 11 Incremental Reviews*,* stakeholders agree to conduct the reviews as a set of many smaller reviews more frequently with a smaller amount of work than is reviewed in traditional waterfall review. At the end of each review, the customer approves the work completed before the teams move on to the next increment of work. Regular validation ensures the building of the right thing and progress is based on successful demonstrations against defined acceptance criteria.

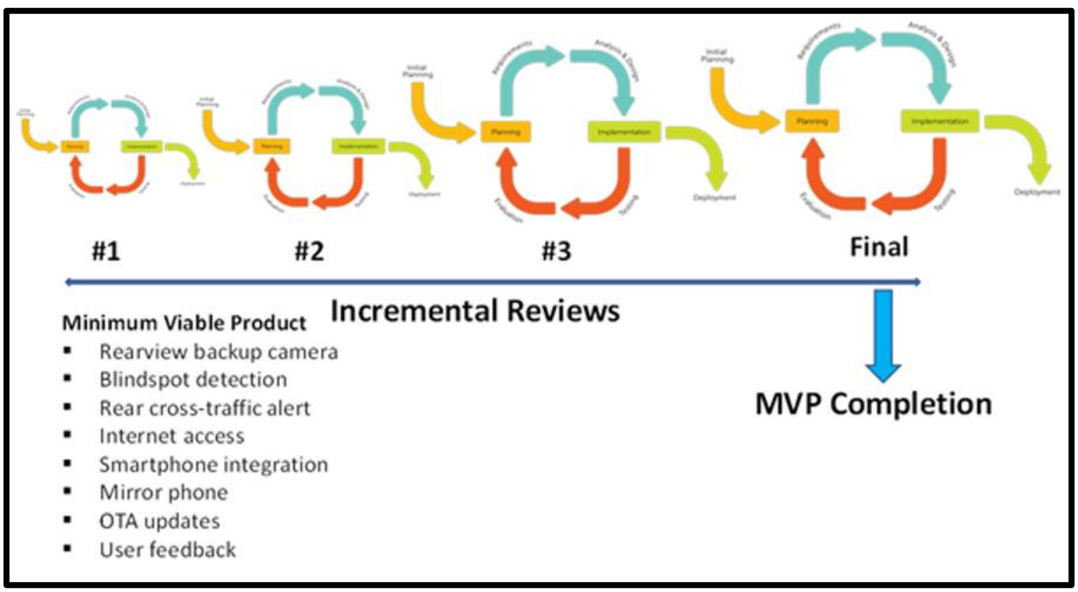


Figure 11 Incremental Reviews

Implementing an iterative, incremental approach allows technical teams to make decisions based on more up-to-date pieces of information and demonstrated objectives of something working. The feedback from stakeholders based on observation enables the development team to take advantage of lessons learned, find issues early, and reduce rework through late discoveries. The system continues to evolve based on validated learning and demonstration. This framework provides for a disciplined approach to maturing the technical baseline on an ongoing basis incorporating incremental specification, development, and evaluation. Bi-directional feedback ensures that stakeholders and developers continually update both specified system capabilities and the shared understanding of what is possible within all identified constraints of system development. Each incremental review establishes the shared acceptance of any changes to the technical baseline.

Making Tradeoffs (Time, $ and Technical)
When building systems there are several tradeoffs to consider that often stem from competing priorities and the constraints of time, cost, quality, scope, and resources. A healthy feedback loop between the customer and the cross-functional team reduces the difficulty in making these trades. Agile program execution must balance the intentionally planned and emergent work necessary to deliver valuable functionality to the customer with the capacity and capabilities of the development teams. At the system level, a Product Manager role serves as the bridge between customer and contractual needs and the requirements and priorities governing the work to be completed by the development teams. Ongoing feedback and a focus on continuous improvement allows the team to provide an optimal approach to development with a smooth flow of product delivery. At the same time, the product backlog provides a prioritized list of capabilities to maximize value delivery and address risks early within team capacity. This ensures a steady and sustainable pace of product development and evaluation that both customer and development teams can support.

Ensuring Reviews are based on demonstrations of working capabilities.

The acquisition agencies of the Department of Defense are currently awaiting the appropriate readiness to support the shift towards frequent, iterative reviews with demonstrations to improve speed of delivery. The question remains: *how the DoD can efficiently transition to reviews that contain demonstrations of working capabilities*.

With an incremental and iterative design approach, it becomes necessary to recast requirements into a set of outcome-based capabilities that incrementally deliver the solution. Demonstrating system capabilities in a gradually increasing fidelity environment from test to staging becomes the primary item of inspection during the incremental reviews. The benefit of this approach is that it delivers the validated system capability to the user sooner. The user feedback from the demonstration can be rapidly incorporated into the next implementation of the system. Each cycle offers validated learning that improves the next cycle of development.

With each iterative review there is the potential for the customer and development team to work together to update the product roadmap. This approach allows the team to refine the overall solution based on what they have learned, rather than following a fixed plan. Once rapid manufacturing, which is the manufacturing processes that serve the fast and flexible production of prototypes and series parts (e.g., 3D printing). When automated testing is finished, the cost of missing design and requirement reviews can be replaced by incremental development of system capabilities.

Invest in the digital ecosystem

The emergence and maturity of the digital ecosystem offers those building cyber-physical systems with capabilities never experienced before. As a result of the digital transformation efforts across the defense community, faster systems feedback with regular experimentation and learning of what works and what doesn't, is made possible digitally versus having to wait until all the hardware is ready as in the past. The digital ecosystem is a concept where virtual and physical models/components are integrated, matured, and maintained to fully represent the system being developed/modified throughout the systems' life cycle, including operations and sustainment. The goal is to move from static paper artifacts to a living digital model. Early in development teams can begin with demonstrating capability to users through models and simulations using tools. The digital ecosystem enables technical agility.

As the system matures, demonstrations move from cyber or digitally only into cyber-physical demonstrations building on increasing levels of fidelity to include digital twins and digital threads. This provides a shared understanding and visibility of the system across all stakeholders. A key attribute to an effective digital ecosystem is one that can "round-trip" between design and implementation so that lessons learned through implementation can be maintained in the ecosystem. The regular cadence of incremental development and evaluation provides for the ongoing synchronization of digital artifacts with the state of the system as developed. This ensures that the system and its digital representation remain aligned. The digital ecosystem provides both a tool and processing environment (digital thread) that supports development practices and the ability to model, demonstrate, and test a digital representation of the system (digital twin). This enables a clear tie between requirements and their implementation as well as early integration and evaluation using digital models. Over the product life cycle, the digital ecosystem provides for efficient change impact analysis, anomaly investigation, and performance analysis.

Recommendations Applied

To demonstrate how these recommendations might be applied, a vignette of the development of the Advanced Driver Assist System (ADAS) is used in the following sections. This system demonstrates how to apply Agile and modern engineering practices at the systems level for a cyber-physical system. This type of system shares key properties that are found in many DoD type systems and as such it can be used as a generic example and applied across multiple domains. The vignette follows the recommendations of creating a shared technical vision, utilizing right sizing planning, performing product-based decomposition, iteratively and incrementally validating the system incorporating technical reviews, and investing in the digital ecosystem to enable technical agility and resiliency of complex systems. Greater exploration of these practices is captured in a follow-on paper, “Example Cyber-Physical Agile Implementation” and provides further detail on the capability decomposition, technical decisions, rationale, and product roadmap development needed to realize the ADAS product.

Shared Technical Vision

The ADAS is a cyber-physical system, as shown in Figure 12 ADAS vehicle, that fuses sensors, actuators, and effectors to provide the user with information needed to accomplish a mission. Their primary objectives are to create capabilities that keep the driver, passengers, and pedestrians safe, offer In-Vehicle Infotainment, and provide User Convenience features, and provide the user with diagnostics and other vehicle information. Through the incremental development of this capability, the relative value of system features, the best configuration of sensor (type and quantity), and performance parameters will evolve throughout the implementation.

Vision: Develop an advanced driver assistance capability to provide a user with enhanced safety, in-vehicle infotainment, and user feedback.

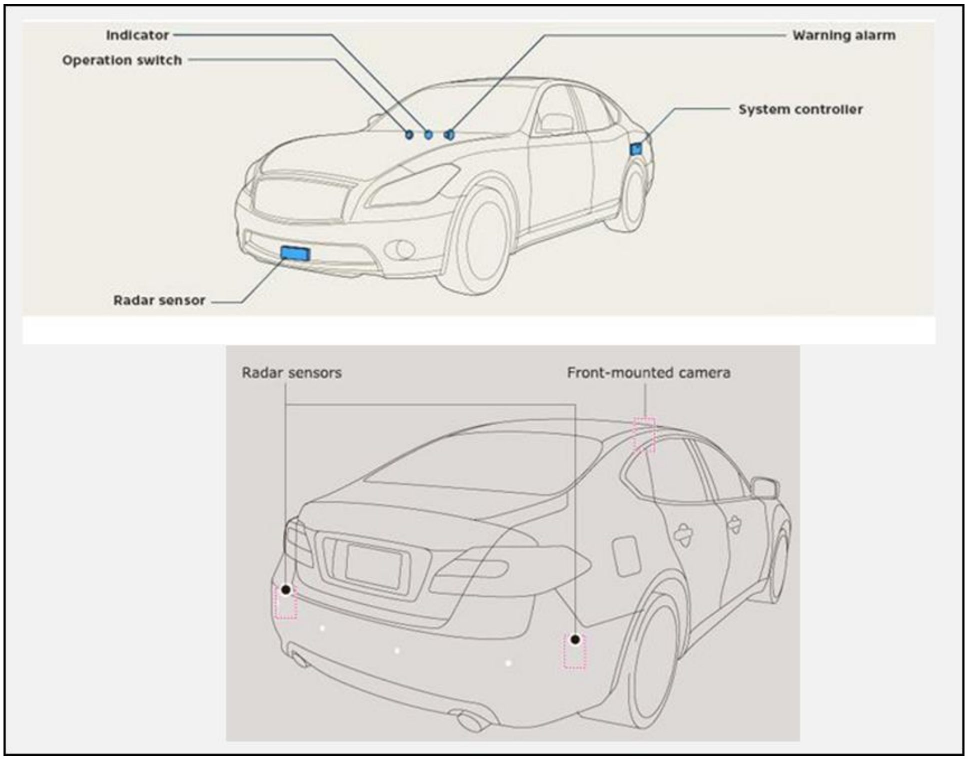


Figure 12 ADAS vehicle

Right size planning

Right size planning applies the multiple horizons in which each horizon occurs on a regular cadence with synchronization of the system. Each planning horizon is planned to a different level of granularity ensuring scope is understood while providing the ability to adapt to changing priorities.

With the ADAS, a vision has been established and the cadence for the multiple horizons have been defined. In this scenario, increments align with quarters with 6 two-week iterations. There is an existing digital twin of the ADAS including the key automobile interfaces to provide the engineering team with the ability to demonstrate the new advanced driver assistance capability first in the virtual environment and then in the physical system. The teams are beginning to identify the areas of the architecture that are impacted and documenting the requirements in the form of automated tests that are connected to the digital twin.

During this phase of the planning the system architects with supporting engineers from all the core disciplines define the architectural approach to developing the ADAS Solution. This team establishes core architectural decisions, identifies the functionality required of the various system modules, key interfaces and core behavioral threads that the system needs to implement in order to provide a technical foundation that is able to implement the ADAS solution. This effort concludes with an initial definition of system functions needed to deliver the key system capabilities - Enhanced Safety, In-Vehicle Infotainment and User Feedback.

Developing an incremental and iterative plan requires framing the solution to allow the initial core features to be delivered rapidly so feedback can be generated to help the team guide the roadmap implementation. The system capabilities and features are organized into a prioritized set of experiments that can be incrementally released. The Product Manager supported by the Product Owners prioritize the features and capabilities based on what is most critical to the customer - Enhanced Safety, In-Vehicle Infotainment or User Feedback. Once prioritize these features and capabilities are expressed as an MVP and a series of NVPs that realize the complete system. This information, along with major milestones (trade shows, consumer purchasing seasons, investment stages and key demonstration events) are then used to build the product roadmap.

The use of Agile Cross-Functional Teams within the organizational breakdown structure looks different than traditional methods. Organizing teams around the physical architecture of the system under development is key as it minimizes the handoffs required to keep the teams coordinated. **Error! Reference source not found.**, shows a notional organizational structure k eeping these principles in mind.

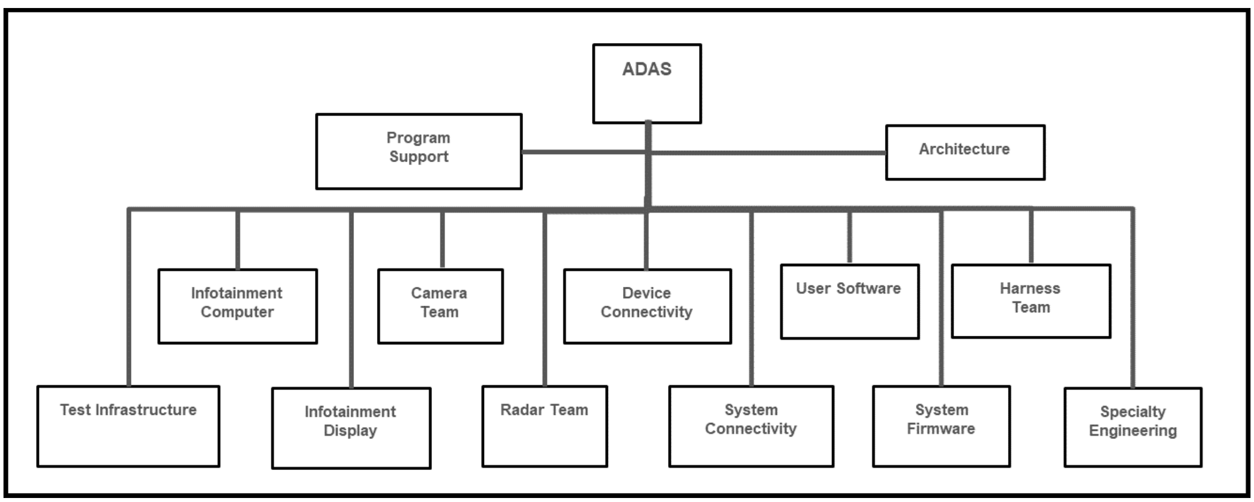


Figure 13 Organizational Breakdown Structure to implement ADAS

Product based decomposition

In this scenario, the vehicle exists in production and has a product-based work breakdown structure to work from illustrated below in Figure 14 Vehicle WBS. We have cross-functional teams that have all the skills needed to upgrade the advanced driver assistance features.

Automobiles are complex safety-critical cyber-physical systems and must adhere to strict safety standards such as ISO 26262 therefore our teams adhere to a Modular Open Systems Approach to the architecture which allows the advanced driver assistance team to make significant changes without negatively impacting automotive safety integrity levels (ASILs), which are used to classify the necessary safety measures to achieve an acceptable risk level.

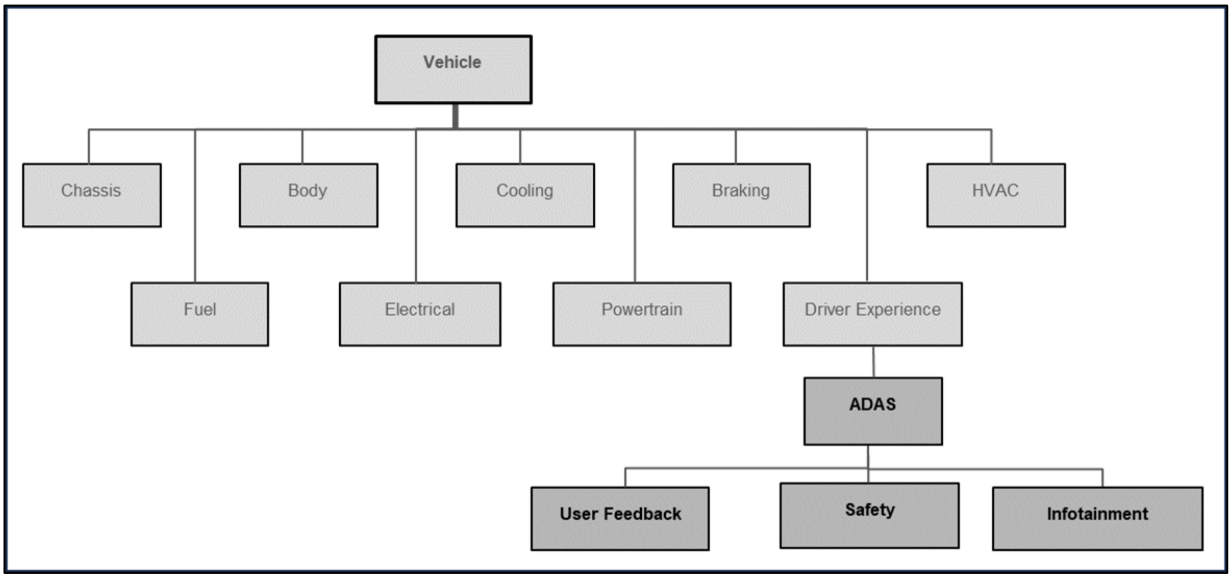


Figure 14 Vehicle WBS

The features of the ADAS safety capability are illustrated in Table 3 Features of the ADAS Safety Capability

Table 3 Features of the ADAS Safety Capability

System Feature		Description
1.	Electronic Stability Control	Improve a vehicle's stability by detecting and reducing loss of traction (skidding).
2.	Rearview Backup Camera	Enables drivers see the area behind their vehicle when they are reversing.
3.	Blind Spot detection	Alerts drivers to vehicles in their blind spots during lane changes through visual and audio cues.
4.	Pedestrian detection	Alerts drivers of pedestrians near their vehicle through visual and audio cues.
5.	Forward collision warning	Alerts drivers of impending forward collisions through visual and audio clues.
6.	Rear cross-traffic warning	Warns the driver of oncoming traffic from the sides when reversing out of a parking spot through visual and audio cues.

7.	Lane departure warning	Warns the driver when the vehicle begins to move out of its lane and, in some cases, can automatically steer the car back into the lane.
8.	Lane keep assist	Alerts the driver when they are drifting out of lane, if the driver does not respond, lane assist gently steers the automobile back into lane.
9.	Active head restraints	Automatically restrains drivers head in the event of a real collision.
10.	Automatic high beams	When the front camera senses headlights of on-coming traffic, the high beams are switched to low.
11.	Parking assist	A set of technologies that partially or fully complete parking for drivers.

The features of the In-Vehicle infotainment capability are illustrated below in Table 4 Features of the ADAS In-Vehicle Infotainment capability.

Table 4 Features of the ADAS In-Vehicle Infotainment capability

	System Feature	Description
1.	Weather alerts	Provides real-time information about local weather conditions leveraging cellular data services.
2.	Traffic alerts	Provides real-time information about local traffic conditions leveraging cellular data services.
3.	Driving directions	Provides GPS-based navigation to the driver selected location.
4.	Smartphone integration	Provides integration to the driver's cellular device through the infotainment system.
5.	Mirror Phone	Enables the vehicle to show a replica of smart-phone screen.

The features of the ADAS User Feedback capability are illustrated below in Table 5 ADAS In-Vehicle Infotainment Features.

Table 5 ADAS In-Vehicle Infotainment Features

	System Feature	Description
1.	Home Internet Connectivity	Enables over the air updates and provides for store and forward user analytics reporting.
2.	User Analytics	Provides the development team with user product engagement information.
3.	Mobile Connectivity	Provides the ADAS system connectivity with mobile phone networks (3g / LTE / 5g). Provides feedback for all installed systems.
4.	Over the air updates	Provides the ability to update ADAS features through the internet.
5.	Premium Mobile Connectivity	Allows the user to subscribe for enhanced data bandwidth via Mobile Connectivity for streaming bandwidth while not connected to the Home Internet.

To better understand the value of the capabilities being delivered by this improvement, the Product Manager works closely with users to determine the features that will provide the most value. This is done by establishing objective key results for each ADAS capability. The organization of ADAS features will be guided by key questions that the product team will want to answer as the product matures. An example of one of these key questions might be “What is more important to our user community - Infotainment while they drive or Enhanced Safety.” Given the need to have answers for these key questions the Product Manager will prioritize the features based on cost to implement (in terms of time, money and effort), feature value to the end-user, and ability to answer key questions. Prioritization is also influenced by implementation dependencies (e.g., You can’t use software if you don’t have an infotainment computer etc.).

A prioritized set of features combined with key milestones, funding constraints and market driven dates are used to organize the features into a roadmap consisting of those features needed to create a minimum viable product (MVP), and subsequent next viable products (NVPs). Roadmaps are intended to be light-weight, flexible and easy to update so that the ADAS team can leverage crucial insights gained from the fielding of early MVPs to answer key questions and further guide the order in which features are developed and delivered to the users of the system.

Armed with all the data, the Product Manager works the cross-functional teams to update the product backlog and prioritize the features to update the advanced driver assistance capability. The prioritized backlog is illustrated in Table 6 ADAS Product Backlog. The teams support the Product Owner in grouping the feature updates into four different releases labeled MVP (for Minimum Viable Product), NVP-1 (Next Viable Product 1), NVP-2 (Next Viable Product 2) and MVP to NVP-1. The highest valued features are delivered first to the users.

Table 6 ADAS Product Backlog

	Feature	Priority	Rel	Business Value
1.	Rearview backup camera	1	MVP	Operator Safety, Demonstrate System Architecture
2.	Blind spot detection	2	MVP	Operator Safety
3.	Rear Cross-Traffic warning	3	MVP	Computer, Rear Camera, Touchpad Screen
4.	Mobile Connectivity	4	MVP	User Convenience - provided by 5g connection
5.	Smartphone Integration	5	MVP	Forward, Left, Right, and Rear Sensors, Computer,
6.	Mirror Phone	6	MVP	User Convenience - provided by Bluetooth
7.	Over-the-Air Updates	7	MVP	Enabler for interim releases between MVP and NVP-1

	Feature	Priority	Rel	Business Value
8.	User Analytics	8	MVP	Business Value - Understand the value of each feature - used to guide future roadmap decisions
9.	Driving Directions	9	MVP to NVP-1	User Convenience
10.	Traffic alerts	10	MVP to NVP-1	User Convenience
11.	Weather alerts	11	MVP to NVP-1	User Convenience
12.	Updated UX	12	MVP to NVP-1	User Convenience
13.	Premium Mobile Connectivity	13	MVP to NVP-1	User Convenience
14.	Lane Keeping	14	NVP-1	Integrate Actuators into the System
15.	Lane Departure Warning	15	NVP-1	Side Collision Safety
16.	Home Internet Connectivity	16	NVP-1	Decrease the cost of data collection
17.	Pedestrian Detection	17	NVP-2	Prevent Accidents
18.	Forward Collision Warning	18	NVP-2	Increase Safety
19.	Active Head Restraints	19	NVP-2	Protect Passengers
20.	Automatic High beams	20	NVP-2	Increase Driver Situational Awareness
21.	Parking Assist	21	NVP-2	User Convenience
22.	Electronic Stability Control	22	NVP-2	Improve Handling

NOTE: MVP - Minimum Viable Product and NVP - Next Viable Product

This table is assembled into a product roadmap which then forms the basis for the development for the project plan. Figure 15 Product Roadmap, shows a notional roadmap based on the prioritized backlog defined in Table 6. The roadmap creates a visual display, showing key milestones for ATP and MVP-1 (in black), capabilities (in dark gray) and the supporting features (in light gray). Some capabilities are spanning across MVP-1 into the NVP.

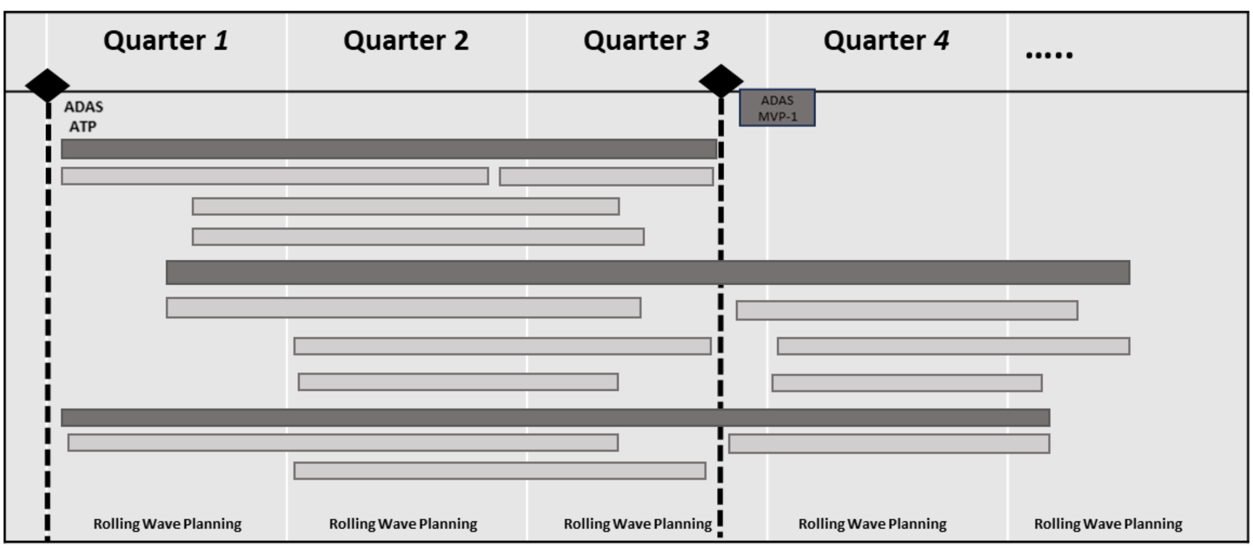
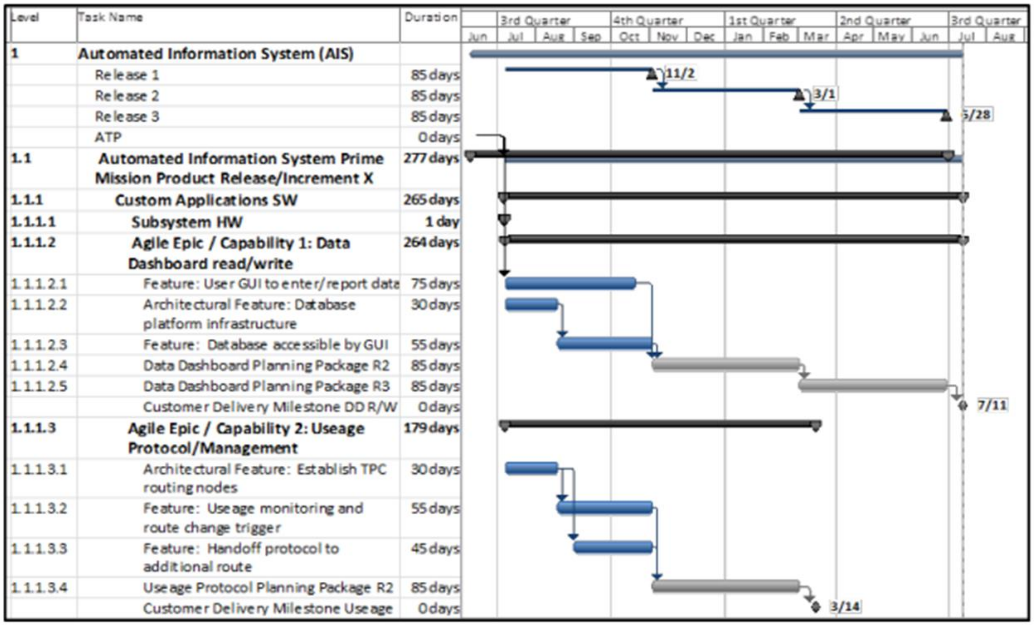


Figure 15 Product Roadmap

The roadmap is then estimated, and resource loaded to generate the program schedule as shown in Figure 16 Program Schedule.



Example of an IMS subset, based on the WBS example in Table 2-1.[6]

Figure 16 Program Schedule

(Reference: © 2022 NDIA IPMD An Industry Practice Guide for Integrating Agile and Earned Value Management on Programs)

It is important that it is easy to update both the roadmap and the schedule as they will be revised during each planning event to incorporate lessons learned, user feedback and updated market conditions.

Iterative and incremental verification and validation

The CFTs perform a design, implement, test, and deploy approach for each feature, making up the roadmap's MVP. In the above example, the Rearview Backup Camera team will complete their allotted stories and prepare for conducting a review with the customer. Similarly, other CFTs complete the Blind Spot Detection, Rear Cross-Traffic Alert, etc., as planned during the planning and road mapping phase. Each Team will now be ready for their respective reviews.

Key to successful execution is that verification and validation test approaches mature in step with the capabilities of the system. As features are completed, so are their matching acceptance and operational utility tests, building up to a test suite for each MVP that then aggregates into the system level final acceptance tests to ensure a fieldable product. This reduces the risk of latent defects or non-compliances for each MVP so that each successive MVP builds on a stable and tested baseline.

Getting Started

The implementation of the approach outlined in this document is not a one-size fits all. How the practices are implemented is based on the specific needs and life-cycle stage of the product and the DoD acquisition process. The following are recommendations to help get started and are not all-inclusive. These recommendations based on feedback from industry and the NDIA Systems Engineering ADAPT community.

- Use RFP language to encourage an iterative and incremental approach for specification, development, and evaluation with demonstrations leveraging digital capabilities.
- Plan for collaboration between government customer and contractor to establish milestones and major program events tied to regular demonstration of functionality, including the ability to leverage virtual and modeled systems as well as physical environments.
- Ensure technical reviews incorporate digital artifacts and that these artifacts persist to drive design and development throughout the program lifecycle.
- Plan for early and ongoing evaluation of models and simulated systems to facilitate common customer-contractor understanding of scope, solution intent, and feature development. Ensure the customer team (including any external compliance representation) is planned and funded to support this level of engagement.
- Ensure the customer-contractor team establishes the workflow for maintaining shared understanding of the technical baseline through persistent models or other repository.

Conclusion

This white paper addresses several issues facing the Department of Defense as identified by the NDIA cross-government and industry collaboration over the past several years. In preliminary form, these ideas were presented at the NDIA Systems and Mission engineering conference in November 2022 and again shared in 2023 for feedback on the concepts from a wider community. Industry and government collaboration over the 12 months have matured the recommendations presented in this paper.

The following summarizes key take-aways to improve and modernize Systems Engineering design and review activities as part of the acquisition process with emphasis on value delivery.

1) **Concern**: One concern raised during our collaboration on this white paper is how the iterative system behavioral threads get verified and how to obtain early user feedback for improved validation. This is a challenge since the physical portion of cyber-physical system lags the software needed for a digital ecosystem consisting of Digital Threads and Digital

Twining of the software implementation on a complex cyber-physical system development program.

Recommendation: Model Based Development and Model Based Systems Engineering leverage the digital ecosystem and allows for visualizing the outcome and obtaining early customer feedback throughout the development and operations lifecycle of the system. Hardware can be introduced into verification with architectural models through increasing digital fidelity leading up to hardware-in-the-loop testing as physical components are available for integration.

2) **Concern**: The need for Systems Engineering on these complex programs is not only necessary but increasingly important in ensuring models and the digital ecosystem continue to evolve. Keeping the entire team up to date throughout the incremental implementation of a system is crucial. Large upfront technical reviews must be validated along the way.

   Recommendation: An evolving architecture with regular reviews and feedback is key to program performance. System engineering activities help prioritize early and ongoing system objectives and architecture and play a significant role in aiding the initial planning and technical definition of the system. In addition, it is necessary to ensure that the overall architecture and product vision are updated and maintained based on early and regular feedback from users. This approach benefits both the customer and the contractor by preventing miscommunication and contract scope creep. An Agile methodology for milestone reviews is a win-win solution.

3) **Concern**: Traditional milestone reviews such as SRR, SFR, PDR, CDR fail to provide timely feedback from the customer to the contractor. The existing approach does not address the need for validated learning cycles for complex systems that emerge and continuously improve.

   Recommendation: To address this issue and take advantage of Agile' s benefits of speed and adaptability, in the interim, consider conducting milestone reviews on smaller quanta of work that is completed in shorter durations, such as program increments or MVPs and NVPs. The vision is to leverage existing Agile iterative reviews to achieve the necessary objectives required by traditional reviews on each of the increments being implemented.

4) **Concern**: Technical Reviews, verification approach and integration are necessary for major weapon systems and capability acquisition; however, the current traditional approach does not promote modern practices for agility and validated learning through short feedback cycles. Schedule slippage often the result.

Recommendation: Technical Reviews ensure that agreements between teams are being met. A best practice is that each interface has a 2-sided test that the system needs to satisfy. As long as a module is able to pass the interface test then the overall needs of system integration will be met. Ideally, this progress is demonstrated quarterly and continue to evolve.

5) **Concern**: The current approach for technical reviews limits the ability to adapt and evolve iteratively and incrementally. A more intentional approach for incremental development can help buy-down program risk and provides the opportunity for increased user engagement.

   Recommendation: Technical reviews are performed in an incremental fashion. The initial capability of a component meeting an interface does not need to fully satisfy the interface, but rather must partially meet some fragment of the capability. This approach allows for incremental technical reviews that show gradually maturing capability until the entire capability is met. The system continues to evolve based on validated learning and demonstration to the end-user. The continued evolution allows for improved validation of the system and ensures the system is meeting mission needs.

6) **Concern**: Increasing complexity of systems and need for shorter lead times requires a different approach to developing systems. In the words of General Duke Z. Richardson, Commander, Air Force Material Command, "There is no time for antiquated serial processes, inadequate teaming, or lifecycle and functional stovepipes. The Air Force relies on Air Force Materiel Command (AFMC) to deliver war-winning capabilities. AFMC must shatter existing paradigms and adopt enterprise Digital Materiel Management (DMM) capabilities to radically accelerate our fielding, sustainment, and modernization. Models must replace documents. Structured data must replace disparate information. Digital collaboration must break down decision stovepipes."

   Recommendation: Employ tailorable collaborative, concurrent, and iterative digital engineering approaches across the system life cycle for increased enterprise agility. For enhanced collaboration, the following are considerations adapted from IEEE Organizational and Collaborative Activities include (Reference IEEE 15288, para 5.3.3 Organizational and Collaborative activities):

   - System life cycle activities are performed in a collaborative and iterative manner involving stakeholders and subject matter experts concurrently, as practical,
   - Define a collaborative framework within the organization or between organizations to facilitate the involvement and range of stakeholders and experts. A collaborative

framework includes shared methods, tools, and other resources, and establishes an environment for better continuous communication and technical reviews to maintain shared vision and values.

- Collaborative engineering is an element of iterative and incremental development approaches to help ensure timely feedback and communication across the stakeholders and continuous assessment of the technical progress and program risks.

The intent is for the guidance and recommendations captured in this paper to offer a foundation for further collaboration across the NDIA and defense community as we improve in delivering value faster to those who work diligently every day to ensure the safety and security of our nation and allies.

References

DAF (2023). Digital Building Code for Digital Engineering and Management. Department of the Air Force. Retrieved March 8, 2024 from Digitial-Building-Code-2022-Memo-Signature-Final.pdf (dafdto.com).

DAFDTO (2023). DMM Industry Association Consortium Sharing Site. Retrieved March 10, 2024 from DMM Industry Association Consortium sharing site - DTO Digital Guide (dafdto.com).

DOD (2009). DOD Policies and Procedures for the Acquisition of Information Technology. Retrieved from DSB-DoD-Policies-on-Acquiring-Information-Tech-2009.pdf (armywarcollege.edu).

DOD (2017). NATIONAL DEFENSE AUTHORIZATION ACT FOR FISCAL YEAR 2018. Retrieved 12/17/2023 from 2018 National Defense Authorization Act.

DOD (2020). DOD Instruction 5000.87 Operation of the Software Acquisition Pathway. Retrieved 12/17/2023 from DoDI 5000.87, "Operation of the Software Acquisition Pathway," October 2, 2020 (whs.mil).

DOD (2020) DOD Instruction 5000.88 Engineering of Defense Systems. Retrieved 12/17/2023 from DoDI 5000.88, "Engineering of Defense Systems," November 18, 2020 (whs.mil).

DOD (2020). DOD INSTRUCTION 5000.89 TEST AND EVALUATION. Retrieved 12/17/2023 from DoD Instruction 5000.89 - Test and Evaluation.

DOD Defense Standardization Program. Modular Open Systems Approach. Retrieved 12/17/2023 from DSP :: MOSA (dla.mil).

DOD OUSD (2022). DOD Systems Enigneering Guidebook. Retrieved 12/17/2023 from Systems Engineering Guidebook (cto.mil).

IEEE Standard for Technical Reviews and Audits on Defense Programs. (2015). IEEE Std 15288.2-2014, 1–168. https://doi.org/10.1109/IEEESTD.2015.7105321

Kim, G., Humble, J., Debois, P., Willis, J., & Forsgren, N. (2021). The DevOps Handbook, Second Edition. IT Revolution.

Modigliani, P., & Chang, S. (2014). Defense Agile Acquisition Guide: Tailoring DoD IT Acquisition Program Structures and Processes to Rapidly Deliver Capabilities. Www.mitre.org. https://www.mitre.org/news-insights/publication/defense-agile-acquisition-guide-tailoring-dod-it-acquisition-program

Public Law 115-91 Article VIII “ACQUISITION POLICY, ACQUISITION MANAGEMENT, AND RELATED MATTERS” (Dec 12, 2017). House.gov. https://uscode.house.gov/statutes/pl/115/91.pdf

NDIA IPMD. "An Industry Practice Guide for Integrating Agile and Earned Value Management on Programs" Version 1, dated December 9, 2022.

OSD (2020) Agile and Earned Value Management: A Program Manager's Desk Guide. Retrieved 12/17/2023 from AAP Agile and EVM PM Desk Guide Update Approved for Nov 2020_FINAL.pdf (osd.mil).

Senge, Peter M. (2006). The Fifth Discipline: The Art & Practice of the Learning Organization, 2nd Ed. Random House Business.

U.S.GAO (2020). Agile Assessment Guide: Best Practices for Agile Adoption and Implementatioin. GAO-20-590G. Retrieved 12/17/2023 from Agile Assessment Guide: Best Practices for Agile Adoption and Implementation | U.S. GAO.

U.S.GAO (2022) Weapon Systems Annual Assessment: Challenges to Fielding Capabilities Faster Persist. GAO-22-105230. Retrieved 12/17/2023 from https://www.gao.gov/assets/gao-22-105230.pdf.

U.S.GAO (2023) Weapon Systems Annual Assessment: Programs Are Not Consistently Implementing Practices That Can Help Accelerate Acquisitions. GAO-23-106059. Retrieved 12/17/2023 from Weapon Systems Annual Assessment: Programs Are Not Consistently Implementing Practices That Can Help Accelerate Acquisitions | U.S. GAO

Appendix A: Additional Background

In 2014 the Office of the Secretary of Defense / Performance Assessments and Root Cause Analyses (PARCA) established a joint Industry Agile and Earned Value Management (EVM) Collaboration group to address concerns raised by the Services that Agile programs within their purview did not have a good answer for how to manage and balance cost, schedule and technical objectives. This activity resulted from a Department of Defense Agile and EVM Desk Guide for Government program offices and an NDIA Agile and EVM best practices guide for industry program offices. Although these documents target programs that are required to implement EVM, the best practices can be leveraged to provide guidance around the planning and managing of the technical objectives within an Agile environment to the system engineering community. Specifically, these documents recommend establishing traceability between the Agile technical work, the work breakdown structure (WBS), and the integrated master schedule (IMS) and claiming Agile progress to inform overall program cost, schedule, and technical status.

a. NDIA guide "An Industry Practice Guide for Integrating Agile and Earned Value Management on Programs" Version 1, dated December 9, 2022.
b. OSD guide "Agile and Earned Value Management: A Program Manager's Desk Guide," dated November 11, 2020.

The 2018 National Defense Authorization Act contained the following sections related to Software and Agile acquisitions. Reference (Public Law 115-91 "TITLE VIII—ACQUISITION POLICY, ACQUISITION MANAGEMENT, AND RELATED MATTERS" Subtitle H - Provisions Relating to Software Acquisition.
Sec. 871. Noncommercial computer software acquisition considerations.
Sec. 872. Defense Innovation Board analysis of software acquisition regulations
Sec. 873. Pilot program to use agile or iterative development methods to tailor major software-intensive warfighting systems and defense business systems.
Sec. 874. Software development pilot program using agile best practices.
Report to Congress on Software Development Activity Completion Section 874 of the National Defense Authorization Act for Fiscal Year 2018 (P.L. 115-91), dated October 03, 2019, summarizes the result of the pilot programs and offers five recommendations.

"...Pilots were encouraged to streamline their processes to support Agile delivery. Specific areas of relief were provided in the areas of earned value management (EVM) or EVM-like reporting; development of an Integrated Master Schedule (IMS); use of traditional life cycle methodologies; and additional relief from upfront detailed planning and requirements artifacts and processes."

"… Pilots were requested to specific roles, specifically a Program Manager (PM) with programmatic decision-making authority; and a Product Owner (PO), Engineering Lead, and Design Lead."

"...Develop a plan containing the overall vision; a rapid merit-based contracting procedure; a continuous engagement approach, to include frequent and iterative user feedback and validation; and incorporation of commercial best practices related to modern application development, Testing, integration, monitoring, and deployment.

"...requested that pilots implement award processes that take no longer than three months. from three months … frequent and iterative user engagement and validation; delivery of a functional prototype or Minimum Viable Product (MVP) within three months from the award; and follow-on delivery of iterative development cycles no longer than four weeks apart."

"... guidance requested the inclusion of a modern backlog tracking tool as well as Agile development metrics to track the pace of work, completeness of scope of testing activities, and delivery progress relative to the product roadmap and goals for each iteration."

Although progress is being made based on this guidance and other initiatives of the Department of Defense, there remain concerns within the Department of Defense (DoD) about adopting agile and delivering capabilities more frequently. See GAO Report: https://www.gao.gov/assets/gao-22-105230.pdf. [x]

"…In November 2020, DoD issued DoD Instruction 5000.89 establishing policy and procedures for Testing and evaluation across five AAF pathways—including the major capability acquisition and MTA pathways—that address cyber-security planning and execution…. DoD leadership has emphasized key practices, such as iterative development. However, most of the 39 programs reported using a modern software development approach to deliver working software for user feedback more slowly than recommended by the industry's Agile practices, which call for rapid, frequent delivery of software and fast feedback cycles (see figure). As a result, these programs may lose out on some of the benefits of using a modern approach."
All these efforts indicate opportunities to improve and the need to act urgently. We must evolve our current ways of working, embrace modern practices and digital capabilities that enable faster delivery cycles as we act with speed and agility.

Appendix B: Glossary

Agile Approach: End user(s) team with developers to make instant decisions on user functionality. High level requirements are initially prioritized and developed quickly by small teams to get a working product quickly to the customer. Multiple, rapidly executed Increments are developed, and capabilities are released to the customer as soon as possible. Prototypes may be used as a starting place and utilize a modular, open-systems approach. (Reference https://www.dau.edu/glossary)

Agile Cross-Functional Teams: A group of people with different functional expertise working toward a common goal. Typically, Agile Cross-Functional teams are composed of 5 to 11 people who have individual specialties and the ability to collaborate to solve the problem at hand. Ideally team members would have a T-Shaped skill set with expertise (The leg of the T) and the ability to work across functional disciplines (the bar of the T). The advantage of cross-functional teams is that they break down organizational silos, minimize handoffs, and foster collaboration and problem solving.

CI/CD Pipeline: Continuous integration and continuous delivery/deployment through the automation of the software development and release process.

Cyber-Physical Systems (CPS): An "engineered systems that are built from, and depend upon, the seamless integration of computation and physical components. Advances in CPS will enable capability, adaptability, scalability, resiliency, safety, security, and usability that will expand the horizons of these critical systems" (Reference: US National Science Foundation)

DevOps: An organizational engineering culture and practice that aims at unifying system development and operations. The main characteristic of DevOps is to automate, monitor, and apply at all phases of the engineering lifecycle: plan, develop, build, test, release, deliver, deploy, operate, and monitor. In DevOps, testing is shifted left through automated testing – this is a key differentiator since functional capabilities are tested and built simultaneously.

DevSecOps Pipeline: A collection of DevSecOps tools, upon which the DevSecOps process workflows can be created and executed. DevSecOps tools are a tailored series of software products configured to integrate end-to-end software definition, design, development, test, delivery, and potentially deployment in a highly automated and secure way.

Digital Twin: A digital twin is a virtual representation of real-world entities and processes, synchronized at a specified frequency and fidelity. Digital twins exist to replicate configuration, performance, or history of a system. Two primary sub-categories of digital twin are digital instance and digital prototype.

Digital Thread: A data-driven architecture that enables the use of digital tools and representations for design, evaluation, and life cycle management.

DoDAF: DoD Architectural Framework.

End User: A user of the system. This may or may not be the paying customer, which in government context the paying customer may be the agency acquiring the capability.

Integrated Master Schedule (IMS). An integrated, networked schedule containing all of the detailed activities necessary to accomplish project objectives. When coupled with the Integrated Master Plan (IMP) or similar milestone hierarchy, it provides the time spans needed to complete the accomplishments and criteria of the IMP or major program events. The IMS is typically used to produce the various levels of schedules for the project (summary master, intermediate, and detailed). (Reference: NDIA IPMD EVMS Agile Guide, 2022).

Lean Six Sigma: A set of tools used to optimize processes by eliminating waste and reducing variation.

Modular Open Systems Approach (MOSA). As defined by the DoD Defense Standardization program, MOSA uses a system architecture that allows severable major system components at the appropriate level to be incrementally added, removed, or replaced throughout the life cycle of a major system platform to afford opportunities for enhanced competition and innovation.

MVP: Minimum Viable Product – Need an authoritative definition

NVP: Next Viable Product (n) - Need an authoritative definition

PEO: Program Executive Officer

Release: A grouping of capabilities or features that can be used for demonstration or evaluation. A release may be internal for integration, testing, or demonstration, or external to system test or as user delivery. A release may be based on a time block or on product maturity.

Work Breakdown Structure (WBS). A product-oriented structure that depicts the subdivision of effort required to accomplish project objectives. It is an organized method to break down a product into sub-products and at the lowest level, the tasks to be accomplished. It is used for planning, budgeting, work authorization, performance measurement, tracking, and reporting purposes. (Reference NDIA IPMD EVMS Agile Guide, 2022).

List of Figures

Authors and Contributors

Thank you to the authors and reviewers from NDIA Systems Engineering division, the NDIA ADAPT (Agile) working group, and other members of NDIA who contributed to this publication. Their diverse perspectives, expertise, and insight defined success patterns in applying lean-agile principles to the delivery of value.

Dr. Suzette Johnson, Northrop Grumman, NG Fellow, Lean Agile
Gordon Kranz, Enlightened Integrated Program Management, LLC, President
Ken Kubo, Northrop Grumman, Director/Engineering Technologist
Sundar Thyagarajan, L3Harris, Sr. Scientist, Systems Engineer
Robin Yeman, Carnegie Mellon SEI Space Domain Lead
Paul Zajac, Lockheed Martin, Technical Fellow