Systems Engineering Division

2101 Wilson Boulevard, Suite 700, Arlington, VA 22201-3060 • (703) 522-1820 • NDIA.org

IEEE 15288 Meets Lean Agile

Integrating Lean and Agile Principles with Systems Engineering Processes for Modern Defense Acquisition

June 2025
Version 1.0

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 Digital

Robin Yeman, Ph.D.
Leidos, Senior Software Solution Architect

Kelli Houston
Lockheed Martin, Technical Fellow

Dr. Nathaniel Crews
Caltech CTME, Enterprise Agility Coach

Larri Rosser
RTX, Technical Fellow

Jordan Stoner
Lockheed Martin, Agile Coach

Cynthia Ferreira
Scaled Agile, Strategic Advisor

Kennie Garlington
International Council of Systems Engineering, CSEP
Reviewer

Steve Henry
Reviewer

Garry Roedler
Reviewer

John Stough
Reviewer

Executive Summary

In the rapidly evolving landscape of defense and technology, the modernization of systems engineering tools and methodologies is imperative for maintaining a competitive edge and ensuring a responsive advantage. Addressing this imperative requires recognizing the importance of integrating agility into systems engineering processes. This is in alignment with the DoD's Systems Engineering Modernization (SE MOD) efforts, enabling the planning, development and sustainment of high-quality, resilient systems that can meet the challenges of an increasingly complex and dynamic environment. DoD's SE MOD plan leverages established methodologies such as digital engineering, Modular Open Systems Architecture (MOSA), Agile/DevOps, and mission engineering to enhance speed of delivery as depicted in Figure 1.

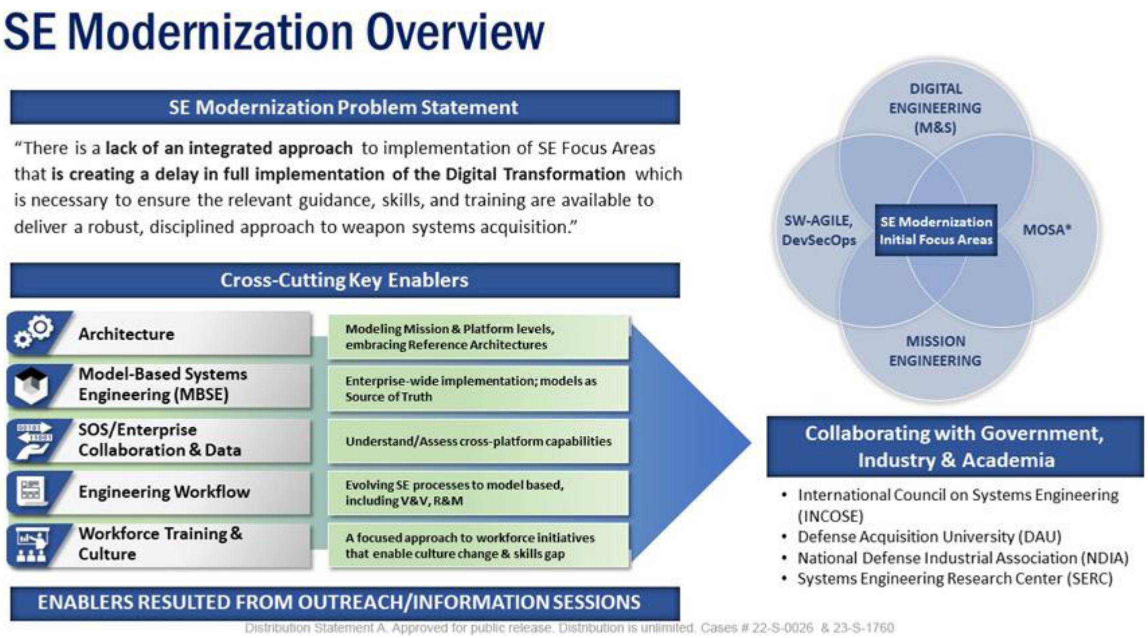

Figure 1: OUSD R&E, Systems Engineering Modernization Overview

The ISO/IEC/IEEE 15288-2023 is the international standard for Systems and Software Engineering - System Life Cycle Processes (referred to as 15288 henceforth) and exists as a cornerstone for systems engineering practices, providing a comprehensive framework for the life cycle processes of man-made systems. Developed and maintained by the International Organization for Standardization (ISO), the International Electrotechnical Commission (IEC), and the Institute of Electrical and Electronics Engineers (IEEE), this standard represents a consensus among global experts on systems engineering processes. While 15288 defines the standard for systems and software engineering and addresses some agile tenets, it does not specifically address how systems engineering processes might integrate Lean Agile principles and practices, which is the focus of this paper.

The Systems Engineering Research Center has also published guidance supporting systems engineering modernization as offered in publications such as the Program Managers Guide to Digital and Agile Systems Engineering Process Transformation (SERC, WRT-1051, 2022). The recommendations presented in this paper build upon the aforementioned standards and best practices, with the goal of ensuring mission readiness and sustainability in an increasingly complex and dynamic environment.

Introduction

The audience for this paper is the national defense community who support and advocate systems engineering modernization with the intent of delivering value that helps ensure the safety and security of our nation and allies.

15288 is a systems engineering standard that is recognized by the Department of Defense as a common process framework for the performance of effective systems engineering through the system life cycle. Standards such as ISO/IEC/IEEE 15288.1 (specific for Defense programs) and ISO/IEC/IEEE 24748-8:2019 further define requirements for systems engineering and technical reviews. "This [15288.1] standard is used by DoD in support of compliance documents, in support of project requirements and other acquisition planning documents such as the government's Systems Engineering Plan (SEP), and as part of a clearly stated and properly scoped acquirer-supplier agreement. This is meant to ensure that government requirements are bid effectively, resourced appropriately, reflected accurately in the proposed Systems Engineering Management Plan (SEMP), project plan and/or schedule and, ultimately, executed in a manner commensurate with effective technical practices" (NDIA Systems Engineering Standards Committee, 2015).

15288 includes four process categories: (1) Agreement processes, (2) Organizational project-enabling processes, (3) Technical management processes, and (4) Technical processes (ISO/IEC/IEEE, 2023). The intent and scope of this paper is to discuss how the 15288 Technical Processes 6.4.1 - 6.4.14, as shown in Figure 2, are viewed in a more iterative, agile, and adaptive context, enabling responsiveness to changing technologies and missions while reducing the cost of that change by leveraging industry success patterns. It is recognized that all four processes are necessary and play a significant role in the engineering of systems. For example, the Technical Processes are managed by the Technical Management Processes. However, the focus of this paper is on the 6.4 Technical Processes.

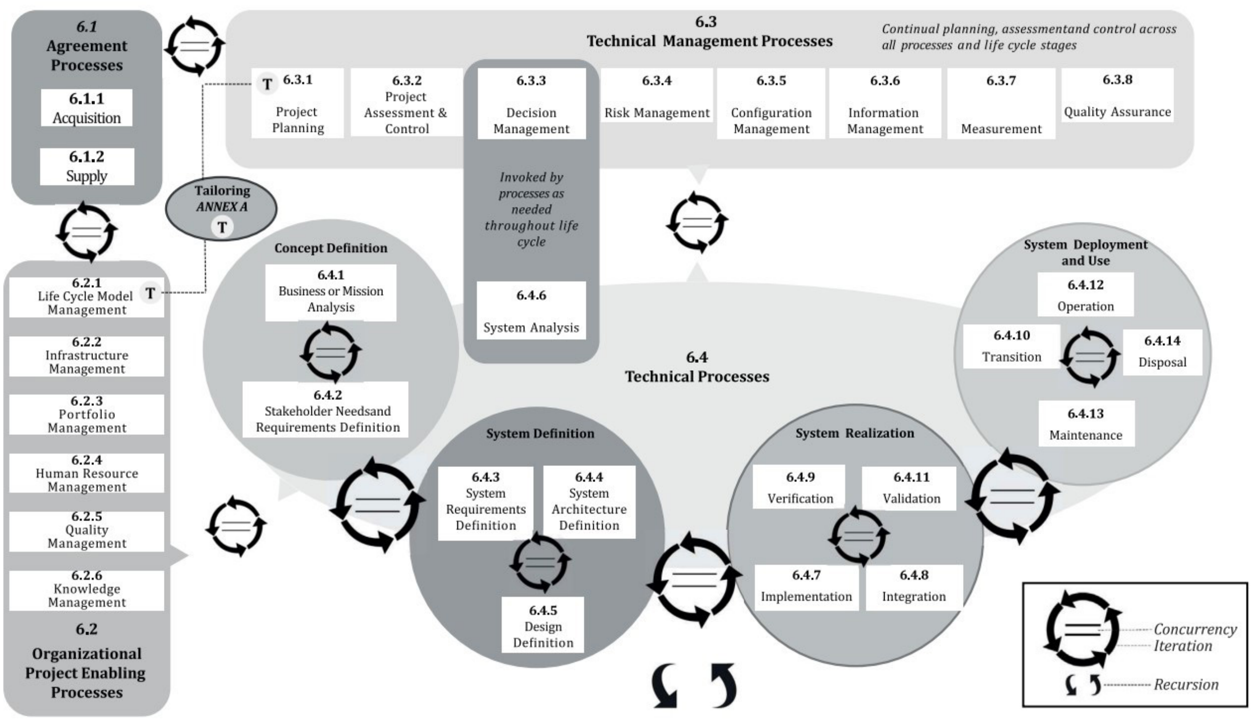

Figure 2: ISO/IEC/IEEE 15288 Processes
Reference: (Reference 15288 figure from Source, Used with permission.)

When it comes to acquisition, the implementation of Lean Agile principles with 15288 Technical Processes can be applied with any of the Adaptive Acquisition pathways as identified in Figure 3: Adaptive Acquisition Framework Pathways (DAU, 2020). While Lean Agile is most often associated with the Software Acquisition Pathway (5000.87), these principles can be implemented within any of the pathways. For example, Major Capabilities Acquisition (MCA) (5000.85) focuses on major weapon systems which could benefit from a cohesive and agile approach between systems engineering, hardware, and software.

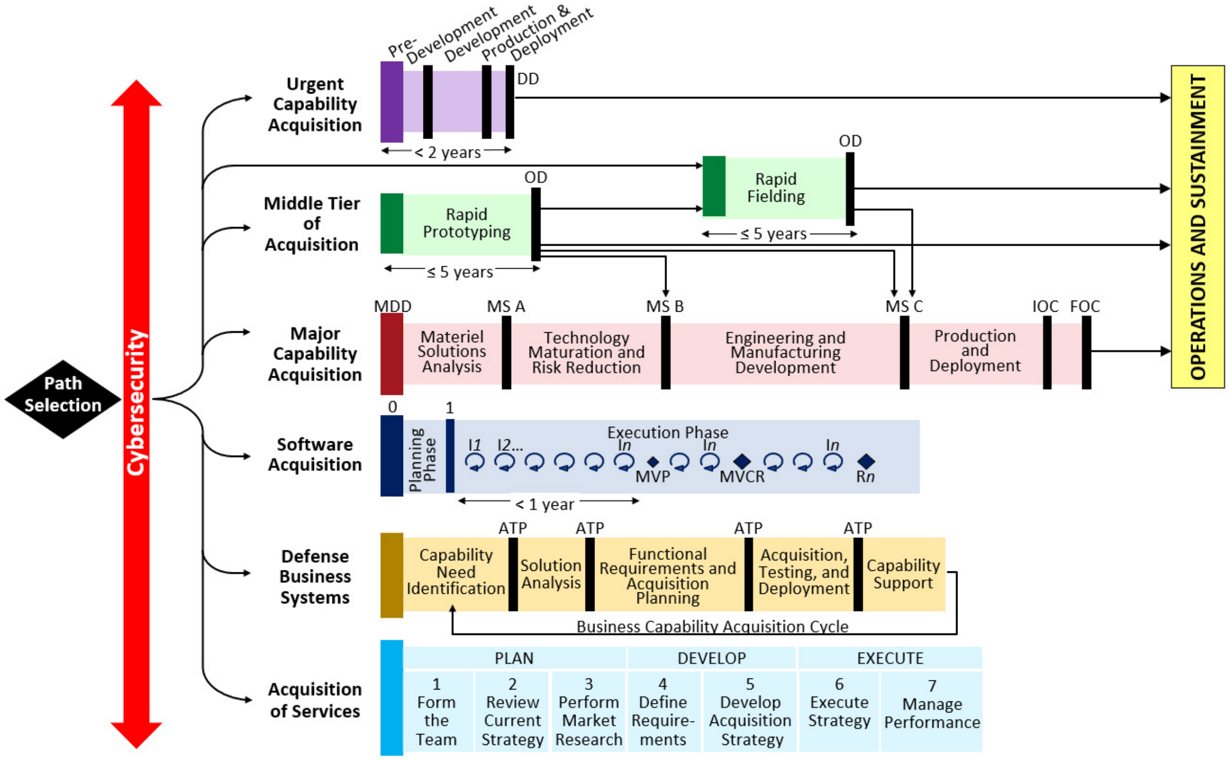

Figure 3: Adaptive Acquisition Framework Pathways, DAU (2020)

The implementation of 15288 and Lean Agile principles provides the following benefits:

- **Improved responsiveness:** Allows systems to adapt faster to changing mission needs and technologies, while retaining the discipline necessary to control risks. Evolving security threats can be addressed quickly.
- **Early and regular demonstration of value:** Demonstrating and delivering of value to the customer early and often, aligning with the DoD's goal of providing capabilities to warfighters more efficiently. This approach provides the opportunity for more frequent value assessments as defined in DODI 5000.97, Digital Engineering.
- **Iterative learning and fast feedback:** Deploy an Agile continuous iterative approach to build, test, and demonstrate integrated system capabilities in shorter cycles resulting in risk reduction throughout the system lifecycle.
- **Waste reduction:** Identifies and eliminates waste in processes through continuous improvement and commitment to quality and excellence.
- **Managed complexity**: Breaking down large systems into smaller, modular components and subsystems, makes them easier to understand, develop, and test. Iterative development reduces cognitive overload and makes the overall system easier to comprehend and evolve.
- **Cross-functional collaboration:** Breaks down silos and increases collaboration between disciplines. Each value stream includes the software, hardware, integration and test, systems engineering, and any other specialty areas needed to develop working functionality while reducing handoffs, delays, and wait time.
- **Cost reduction:** Eliminating waste by streamlining processes to remove non-value-added activities, such as unnecessary documentation or redundant steps, increases efficiency and reduces costs, providing more value to the customer. Implementing continuous feedback loops with stakeholders ensures that the evolving mission analysis meets their needs and expectations, reducing the risk of rework and keeping the analysis relevant and valuable.

An Integrated Perspective

Agile practitioners focused on large complex solutions acknowledge three broad focuses of activity within a project: Inception, Realization, and Transition as shown in Figure 4.

- **Inception:** Where a customer need is articulated, and the design envelope of solution intent is defined.
- **Realization:** Where the solution is iteratively and incrementally designed, implemented, and tested.
- **Transition:** Where the solution is deployed into its operational use.

These phases are depicted as a circular loop to illustrate the recursive and ongoing process of design, build, test, and deliver while leveraging the necessary Technical Management processes throughout.

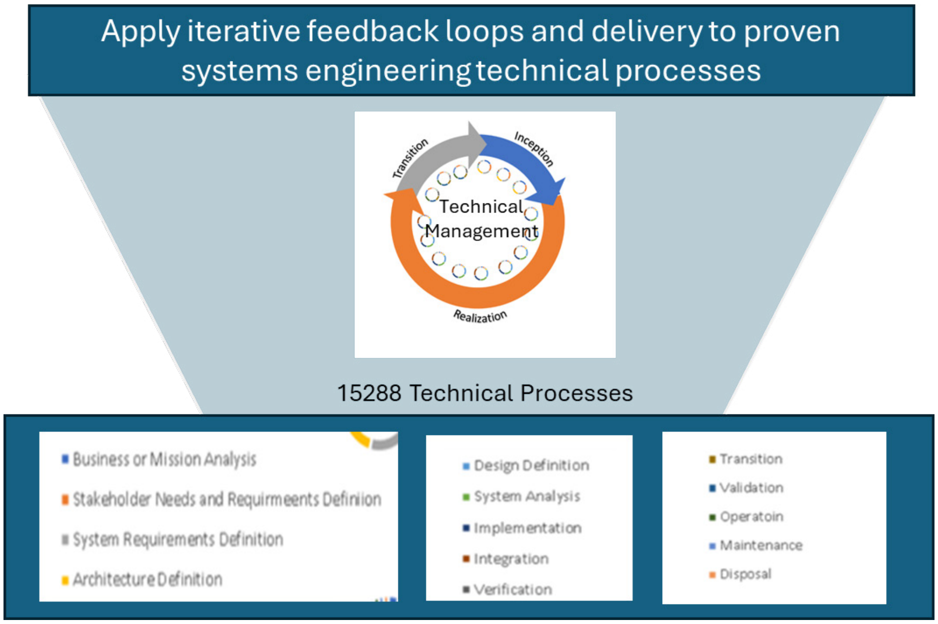

Figure 4: Inception, Realization, Transition

During **Inception,** customer needs are identified and the design framework for the solution is established and agreed upon. Activities focus on creating a clear description of the solution's intent to develop an initial backlog, outline essential needs and constraints, and determine necessary work to begin implementation.

Inception activities are not one-and-done but are revisited as the solution matures or as mission needs call for change. 15288 has iterations, recursion, and concurrency of processes as a core tenet with a recognized need for learning cycles, creating a synergistic relationship with the integration of Lean Agile principles. Customer centricity and design thinking practices are foundational for success during this stage. The customer is the heart of the solution

Key 15288 technical processes during Inception include ***Business/ Mission Analysis, Stakeholder Needs and Requirements Definition, System Requirements Definition, and Architecture Definition***. While more detailed design processes occur after ***Inception***, we begin here with emphasis on understanding the mission and stakeholders' needs which is critical when defining a solution centered on delivering customer value. Facilitated engagement with stakeholders is imperative to understanding the problem space and ensuring ongoing feedback is incorporated into the definition process. Analysis and evaluation of options as part of technical decision management is an integral process that begins during Inception and may also find its place during Realization and Transition.

Instead of outlining exhaustive requirements and architecture, the focus is on key requirements and objectives with guardrails for solution development which serve as a foundation for implementation. Specifically, when building cyber-physical systems, defining an architectural roadmap while considering trade-offs and architectural requirements is critical.

Realization begins once the solution intent is understood, enabling the team to begin development. The goal of Realization is to demonstrate vertical slices of functionality that provide value to the user and improves understanding of their needs. Insights about the problem to be solved and evolution of the solution stem from engineering work, demonstrations, discussions, and critiques of working features or user stories.

For new solutions, initial feedback and early validation are demonstrated through a minimal viable product (MVP) or minimal viable capability release (MVCR). For reference, programs following DODI 5000.87, Operation of the Software Acquisition Pathway, "the MVCR delivers initial warfighting capabilities to enhance mission outcomes. The MVCR for applications programs must be deployed to an operational environment within 1 year after the date on which funds are first obligated to acquire or develop new software capability including appropriate operational tests. If the MVP version of the software is determined sufficient to be fielded for operational use, the MVP will become the MVCR" (2020, DoD Software Acquisition Pathway).

Key 15288 technical processes like ***Design Definition, System Analysis, Implementation, Integration, and Verification*** are emphasized during Realization. The team revisits mission/business analysis and stakeholder requirements to refine the solution intent, as needed. Functionality is prioritized by assessing risks, dependencies, and delivery timelines. These activities are foundational for shaping a backlog. Consideration of operational processes, including transition and maintenance, also influences design choices and is discussed with customers and users. A modern systems engineering approach emphasizes the importance of regular, ongoing integration and demonstration of system capabilities, using objective evaluation to evolve the solution intent.

The **Transition Phase** is where the solution is moved into its operational use. During this phase, attention is on the ongoing iterative improvements and incremental delivery of system capabilities. The key 15288 processes of Transition are ***Transition, Validation, Operation, Maintenance, and Disposal,*** with other processes activated as needed for updates. DevSecOps is an integrated practice throughout this, and all three, phases, with attention given to continuous ATO (cATO).

Transition begins early in agile programs, allowing users to evaluate initial valuable functions within weeks or months of startup. These capabilities can either be deployed directly into limited operational use or tested in a relevant evaluation environment. The goal is to integrate functionality into real or simulated use, fostering rapid learning, and maximizing value delivery.

During the transition phase, the emphasis is on delivering business and operational value and validating assumptions made during the design and planning processes. Value delivered, in alignment with the contract, is determined by the end user and is assessed at least annually after the capability is fielded. Specifically, the Software Acquisition Pathway states, "Value assessments will be performed at least annually after the software is fielded to determine if the mission improvements or efficiencies realized from the delivered software are timely and worth the current and future investments from the end user perspective. More frequent value assessments are encouraged if practical (2020, DoD Software Acquisition Pathway)." Facilitating early and ongoing transitions ensures a steady flow of demonstrated value to the customer.

15288 Technical Processes and Lean Agile

15288 is a set of defined processes for defining, building, and delivering a system to achieve business and mission outcomes. These processes can be implemented with either a traditional serial approach or with an iterative, Lean Agile approach. Applying Lean Agile principles is an imperative for modernizing systems engineering, providing the ability to embrace rapidly emerging digital capabilities, responding to changing mission needs, and the need to deal with increasing system complexity while striving to reduce lead time to delivery.

To establish a shared understanding of Lean Agile principles, the following table lists Lean Agile principles and provides a brief description of each. These descriptions are based on Industrial DevOps (Johnson and Yeman, 2023) and the Scaled Agile Framework (SAI, 2025). In addition to a description of each Lean Agile principle, the table highlights the benefits of each principle with systems engineering.

Table 1: Lean Agile Principles Defined

Lean Agile Principles	Description	Systems Engineering Benefits
Adopt Customer and User Centricity	Focus is on understanding customers, their mission, and their operational needs. Time is spent on understanding the problem to be solved, applying design thinking practices, and iteratively developing the solution with fast feedback loops converging toward solution intent. Engaging with the customer early and often is paramount for success.	Ensures solutions address the right problems by aligning system design with customer needs. Early and frequent customer engagement improves solution relevance, reduces rework, and increases stakeholder satisfaction.
Apply Systems Thinking	View the system as an interconnected whole rather than a collection of individual parts. This perspective emphasizes understanding the relationships and interactions between various components and subsystems within the product, as well as their interactions with external systems and environments. When applying systems thinking it is understood that optimizing part of the system does not necessarily optimize the whole system. Avoid decisions that might suboptimize the whole for local optimization.	Optimize the entire system by understanding how components and subsystems interact, reducing the risk of unintended consequences. A holistic approach that enhances system performance, improves interoperability, and ensures the solution meets broader mission and stakeholder needs.
Apply Multiple Horizons of Planning	Balance short-term and long-term planning. Set clear objectives for immediate, intermediate, and long-term goals, ensuring alignment of the roadmap with the overall mission and operational needs of the customer. Regularly review and adjust plans based on feedback and changing circumstances to maintain agility and responsiveness.	Manages uncertainty by aligning short-term actions and iterative development with long-term goals and milestones. Improves adaptability, ensures continuous progress, and enables proactive responses to evolving customer needs and operational environments.

Lean Agile Principles	Description	Systems Engineering Benefits
Take an economic view	Understand the economic, financial, and mission impact across the system life cycle, from portfolio investments in new technologies and capabilities through sustainment and retirement of a system. Understand the cost of delay (i.e., the impact of delivering a mission late), feasibility, and maintainability of the system capabilities. Consider the risks and opportunity assessment, stakeholder/mission value, and the effort to implement the need.	Enables the prioritization of features and decisions that deliver the highest mission value while managing risks and costs. Helps optimize resource allocation and accelerates delivery of critical capabilities, improving mission outcomes.
Organize Around Value	Structure teams and workflows to deliver the highest value to customers and stakeholders. Instead of organizing by function or discipline, form teams around delivering end-to-end value streams.	Ensures alignment with customer needs and accelerates the flow of value across the system lifecycle. Breaks down silos, fostering cross-functional collaboration and aligning teams with delivering system capabilities that directly support customer and mission needs. Accelerates decision-making, enhances system quality, and improves delivery speed.
Assume variability; preserve options	Embrace the inherent uncertainty in complex systems by maintaining flexibility in design and decision-making. Allow for potential solutions to be explored (set-based design) and evaluated, preserving the ability to pivot as new information and feedback are received.	Manages uncertainty by exploring multiple design alternatives, enabling informed decision-making as new information emerges. Reduces the risk of premature commitments, leading to more resilient and adaptable system solutions.
Base milestones on objective evaluation of working systems	Establish clear, measurable milestones that are based on the actual performance and functionality of the system. Ensure progress is evaluated based on tangible outcomes and regular demonstrations of value rather than theoretical plans.	Enables the tracking of progress through tangible outcomes, fostering transparency, accountability and early detection of issues. Improves decision-making by ensuring that system functionality aligns with stakeholder expectations throughout development and operations.
Apply an iterative development and incremental delivery approach with rapid feedback loops	Optimize the flow of value through the system by identifying and eliminating bottlenecks and inefficiencies. Implement incremental and Iterative reviews, development and test. Emphasize continuous integration, automated testing, and continuous deployment along with continuous monitoring and improvement to ensure that value is delivered to the customer as smoothly and quickly as possible.	Helps manage complexity by breaking down large systems into smaller, testable increments, enabling early issue detection and faster course correction. Enhances system quality and alignment with stakeholder needs by continuously validating functionality and performance throughout the systems lifecycle.
Unlock the intrinsic motivation of knowledge workers	Foster a culture that values and empowers knowledge workers by providing them with the autonomy, mastery, and purpose they need to thrive. This involves creating an environment where individuals feel valued, supported, and motivated to contribute their best work.	Enhances creativity, problem-solving, and innovation by empowering team members to take ownership of their work. Improves team morale, boosts productivity, and drives better system outcomes through engaged and motivated contributors.
Decentralize decision-making	Delegates authority to those closest to the work. Leverage the expertise and insight of team members for more responsive and informed decisions. Enable knowledge worker autonomy.	Enables quicker, more informed decisions by leveraging the expertise of team members directly involved in the work. Accelerates problem-solving, increases responsiveness, and improves system performance by fostering greater autonomy and ownership at all levels.

Lean Agile Principles	Description	Systems Engineering Benefits
Make data-driven decisions	Utilize data and analytics to inform decision-making at all levels of the organization. Collect, analyze, and act on data to drive continuous improvement and ensure decisions are based on objective evidence rather than intuition or assumptions. Capture traceability across the digital thread to provide insight into the current state of the system and to provide real-time data for improved decision-making.	Improves accuracy and effectiveness by relying on objective evidence and data rather than assumptions. By leveraging real-time data and traceability across the digital thread, engineers can monitor system performance, identify issues early, and drive continuous improvement, ensuring the system meets stakeholder needs throughout its lifecycle.
Architect for Speed and Change	Focus on designing and implementing systems and processes that enable rapid adaptation and continuous improvement, particularly in the context of cyber-physical systems. This includes modular architecture considerations, clearly defined interfaces, and a modular open systems approach.	Enables the creation of flexible, adaptable systems that can quickly respond to evolving requirements and technological advancements. Using modular architecture and clear interfaces reduces integration time, facilitates continuous improvement, and supports faster, more efficient updates to fielded cyber-physical systems.

In the following sections, we discuss how these Lean Agile principles and the 15288 technical processes can work together to drive value delivery. For ease of identification, Lean Agile principles are italicized and bolded throughout this section.

Business or Mission Analysis Process

A primary outcome of Business or Mission Analysis is defining "the problem or opportunity space," which refers to understanding customer and mission needs and what problem needs to be solved. Clearly defining the mission, business, or operational problem to be solved or the opportunity is a first step for achieving success. Business and mission outcomes are defined that align with warfighter needs and customer expectations. Applying mission engineering during this process ensures that the right things that are deemed necessary (i.e., technologies, systems, SoS, or processes) to achieve intended mission outcomes are identified (DoD, Mission Engineering Guide, 2023). The DoD Mission Engineering Guide states reinforces mission understanding recognizing that "from the beginning, it's important to have a clear understanding of what goal or decision will be informed as this will drive subsequent choices throughout the process." Emphasis is placed on customer and stakeholder collaboration and understanding the business and mission needs which is revisited throughout the systems life cycle.

Customer centricity is critical in this process as it involves understanding existing threats and opportunities aligned with what the customer wants. The customer is not always the "end user" (e.g., pilot, warfighter, or astronaut), but understanding their perspective in contract and acquisition is crucial. Understanding mission needs and priorities is paramount in defining a solution that ties to desired outcomes. Identifying the most critical capabilities and mission threads, solution providers adopt ***an iterative development and incremental delivery approach with fast feedback loops*** focusing on the essential capabilities first. This approach helps establish clear milestones where specific capabilities and system characteristics can be evaluated on a regular cadence, allowing teams to gather feedback from the customer and users. This approach combined with a Modular Open Systems Approach ***(Architect for Change and Speed)*** offers the opportunity to make necessary adjustments to fulfill mission needs and to revisit new and emerging missions.

Stakeholder Needs and Requirements Definition Process

Stakeholder Needs and Requirements Definition process is vital for translating stakeholder needs into actionable system requirements. ***Adopt a customer/user-centric mindset*** and engage with stakeholders through open communication, collaborative workshops, and feedback sessions from the outset, effectively defining stakeholder requirements, identifying any constraints and other non-functional requirements, and prioritizing needs. This process is a key step in ensuring customer/user alignment, defining the problem space, and converging toward solution intent.

Design thinking techniques such as empathy mapping or journey mapping are often employed to capture what a user group might be experiencing or feeling which helps build better systems with the user in mind. High level ConOps and operational value streams help communicate and visualize multiple views of the solution intent. Developing these scenarios exercises ***systems thinking***, as multiple stakeholder interactions throughout the system life cycle are considered. Scenarios are revisited on a regular cadence and updated iteratively based on a timeframe that fits best for stakeholders. Mission threads, use cases and value streams are key inputs into defining the digital threads used to connect authoritative data and orchestrate digital models and information across a system's life cycle. As Stakeholder Needs are understood, capabilities and requirements play a key role in structuring both the technical solution and the organization that will deliver the solution, directly supporting the principle of organizing around value.

Collaborating with stakeholders on a regular basis as part of the ***iterative and incremental development and delivery process*** provides the opportunity to address changing priorities and to include new stakeholders into the process. Regular collaboration and engagement with stakeholders can yield more positive relationships and greater customer intimacy.

Systems Requirements Definition Process

Requirements Definition is a systematic process that captures and analyzes stakeholder needs to establish clear and measurable system requirements, translating stakeholder views into a technical framework that meets operational, and business needs and ensures traceability from needs to requirements. Use cases and scenarios form the foundation of the digital thread, providing traceable links across the system lifecycle and serving as the basis for test case development.

Effective ***iterative and incremental development*** ensures requirements are traceable and aligned with system objectives while addressing key constraints such as performance, quality, safety, and regulatory compliance.

In an ***iterative and incremental development process***, the Systems Requirements Definition Process is performed continuously, refining and elaborating requirements over time. Initial high-level requirements are defined based on stakeholder needs, with more detailed requirements emerging through ongoing analysis, ***feedback***, and system demonstrations. Early iterations focus on building minimal viable products (MVPs) that demonstrate and deliver the highest stakeholder value, enabling rapid feedback and validation. Each iteration will test and update requirements, enabling early detection of gaps, reducing risk, and ensuring alignment with evolving stakeholder priorities.

This approach supports adaptive ***data-driven decision-making*** and promotes a clearer understanding of system needs as the solution evolves. Careful management of this process is required to ensure scope does not evolve beyond contractual agreements. ***Incremental development and reviews*** enable ongoing management of the technical baseline.

In a model-based enterprise, the Systems Requirements Definition Process is performed using models to represent and analyze system requirements, ensuring that they are clearly defined, traceable, and consistent. These models serve as a central repository for capturing and evolving requirements throughout the system lifecycle, enabling stakeholders to visualize and validate requirements early and ***continuously***. As requirements evolve, the models are updated to reflect changes, providing a real-time, accurate representation of the system's needs and ensuring traceability to design, test, and validation activities.

"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 the model and stays up to date. This modernized approach emphasizes the need for investment in the digital ecosystem" (NDIA SED ADAPT, 2024).

Example. Models and the supporting infrastructure are critical in a team's ability to define and test how to best meet requirements, especially when building cyber-physical systems. As programs define the requirements, the use of models can be used to iterate and gain fast feedback on various ways to meet the requirement. An example of this is Alliant Techsystems Inc (since acquired by Northrop Grumman) which had the requirement to optimize the structural design of the Launch Abort Motor for the Orion Multi-Purpose Crew Vehicle, now known as the Orion Spacecraft. The Launch Abort Motor is a critical component designed to lift the space crew module off the primary launch vehicle in the event an emergency should occur during takeoff. In agreement with Lockheed Martin, the prime contractor, they turned their focus on the initial design of the steel manifold, which weighed just under 2,000 lbs., and how they might reduce the weight of the manifold while meeting the functional and quality requirements. The team leveraged fast iterations enabled by a digital ecosystem to innovate and test many options before settling on a new titanium manifold that met the new target weight of 1,300 lbs. or less (Phipps, Young and Christensen, 2011). This demonstrates how iterative development and fast feedback loops can be used to improve a system's quality.

Requirements should focus on "what" the system must achieve without implying implementation, ***allowing knowledge workers autonomy*** in determining "how." Agile practices such as feature and story mapping can assist in understanding and facilitating iterative exploration of needed functionality demonstrated in the models, enabling quick feedback loops and improved validation. Requirements definition works well with defining the features and stories. Features and stories convey what user or system functionality is needed and includes demonstrable acceptance criteria, clearly articulating what done looks like.

Systems Architecture Definition Process

Defining the systems architecture includes capturing the fundamental concepts or properties of a system, its elements and their relationships, and the governing principles for the realization and evolution of this system

In traditional life cycle models, the system architecture needed to be fully defined before any development could begin. A modernized approach still recognizes the need to have a defined architecture, especially for cyber-physical systems, as it serves as the blueprint for development. "No matter the system you are building, it is important to have an intentional road map of the required elements when architecting your system, because there are so many trade-offs that need to be considered, including change, usability, availability, observability agility, manufacturability, reusability, security, and scalability" (Johnson and Yeman, 2023).

A well-defined, modular architecture that has been ***architected for change and speed*** is crucial for effectively implementing and evolving solutions while meeting non-functional requirements like resilience and performance. In such an architecture, common standards and open frameworks underpin the system structure, decomposing it into capabilities that can evolve over time without disruption and enabling ***incremental delivery***. Developing modular architectures based on open standards (e.g., MOSA) supports interoperability and integration of technologies from diverse sources, enabling innovation, flexibility, scalability, and affordability, while reducing time, cost and risk. A modular open systems approach is required by law and according to the MOSA TriServices memo released in December 2024, "Department of Defense (DoD) Armed Forces face rapidly evolving threats across the world. The dynamic and rapid change of adversary capabilities observed in current conflicts necessitates a critical warfighting capacity to integrate advanced capabilities to counter and maintain a warfighting advantage. To meet this threat, Modular Open Systems Approach (MOSA) shall be implemented and promulgated among the Military Services to facilitate rapid transition and sharing of advanced warfighting capability to keep pace with the dynamic warfighting threat" (Tri Services Memo, 2024). In a Lean Agile operating model, the teams are ***organized around value***, where that value is reflected in the solution delivered. Thus, an effective pattern for organizing around value is to organize the teams around key value streams and the system architecture. When that architecture is modular, consisting of highly cohesive and loosely coupled modules, the resulting organization will be highly cohesive and loosely coupled teams with minimal dependencies that enable value to flow without interruption.

Design Definition Process

Design Definition involves translating system requirements into clear and actionable design specifications that meet both architecture and stakeholder needs. Employing an ***iterative and incremental approach,*** known as evolutionary design, provides the opportunity for the design to evolve alongside the implementation. Rather than a comprehensive upfront design, agile practices establish a flexible foundation with "just enough" design to start, deferring less critical decisions until more information is available. This approach allows for flexibility in how requirements are met as teams ***assume variability and preserve options.*** By applying set-based design, multiple options can be explored and iteratively tested, leading to the most optimal solution. This allows testing and validation of assumptions before zeroing in on a final solution.

Key decisions focus on ***taking an economic view*** that significantly affects the solution's affordability, viability, and feasibility along with understanding the cost of delay. This allows for ***continuous*** adaptation based on changing requirements and stakeholder feedback. With physical systems there are constraints related to size, weight, power, and cost (SWaP-C). These constraints can be explored and optimized iteratively using digital tools, 3D printing, AR/VR, and other models (Johnson, et al, 2022).

As with an agile architecture which is ***architected for change and speed,*** the key to agile design is the decomposition of the system into modular components based on open standards, as required by DoD MOSA standard. Modular components are loosely coupled and highly cohesive, enabling easier modifications and early functionality delivery. Clearly defined open interfaces enable components to evolve independently while maintaining overall reliability. These Lean Agile principles are aligned with the TriServices memo released in 2024 December and the need for modular designs and open standards. The memo specifically directs that "all DoD acquisition officers commit to all five MOSA pillars: (1) employing a modular design, (2) designating modular interfaces, (3) leveraging consensus-based open standards, (4) establishing enabling environments, and (5) certifying conformance (DoD Armed Forces, 2024). Throughout the technical processes, these pillars continue to be defined and validated to the extent possible.

Organizing around value enables the team structure and technical architecture to evolve in a way that facilitates flow, maximizing value for the customer. This approach provides the necessary information about the system to ensure implementation aligns with architectural goals. The design, like the overall system architecture, provides the inspiration for a team structure that is designed to flow value. In agile environments some teams are organized around modular components, reducing team interdependencies and enabling value to flow without interruption. Complex systems often have a mix of complicated sub-system and component teams, stream-aligned feature teams, supporting infrastructure or platform teams and other enabling teams as defined in Team Topologies (Skelton and Pais, 2019). For example, given a CubeSat, teams might be defined in the following way.

Team	Description	Example
Stream-aligned team	Aligned to a flow of work where the team delivers value as independently as possible	Payload Team
Complicated sub-system team	Team with specialized knowledge needed to build advanced components	Guidance, Navigation, and Control
Enabling Team	Specialists in a given domain area	Cyber security
Platform team	Provides internal platforms, tools, etc	Continuous Delivery pipeline

Reference: Johnson and Yeman (2023)

Agile design balances flexibility with stability; a solid architectural foundation supports necessary adjustments. Prioritizing economically sound design choices and reusing existing components, minimizes costs and risks and ***takes an economic view*** while delivering effective solutions that meet stakeholders' evolving needs.

Systems Analysis Process

The Systems Analysis process lays a foundation for informed technical decisions, conflict resolution, and evaluating alternatives. It validates and refines system requirements and designs to align with stakeholder needs ***(Customer/user centricity)*** and performance goals. Through simulations, modeling, and testing, teams can detect and address potential issues early, improving the system's reliability, efficiency, and effectiveness. The measurement process adopted by the teams include frequent demonstrations and assessments with clear criteria to help ensure robust evaluations and ***data-driven decisions*** throughout the system development life cycle.

Taking an economic view balances fidelity with cost considerations, determining the necessary level of analysis while managing constraints. Integrating just-in-time elaboration and ***iterative*** refinement, focuses efforts on high-value areas, optimizing resources and addressing uncertainties early.

Systems thinking enhances this process by providing a holistic view of system behaviors and performance measures, ensuring improvements maximize overall effectiveness. Collaborative engagement with stakeholders and cross-functional teams aligns analysis activities with evolving needs, ultimately delivering high-quality solutions that meet operational goals.

Digital environments significantly enhance systems analysis processes and technical agility by providing integrated platforms for modeling, simulation, and data analysis. These tools support real-time collaboration, early issue detection, and alignment with stakeholder needs. Leveraging digital tools improves traceability (between models and features, iterative and continuous technical reviews and audits, certification, testing plans, procedures etc), decision-making, and frequent integration across the system lifecycle. Digital environments (models, simulation, digital twins) yield the ability to test new functionality where the cost of change is lower. Leveraging artificial intelligence and machine learning, systems engineers can identify patterns from large amounts of data which can assist in validating or refining system behaviors and optimize performance (Johnson and Yeman, 2024).

This approach not only enables regular demonstration of value to the customer on cadence, ***basing milestones on objective evaluation of working systems***, but also provides the opportunity to ***take an economic view*** and ensure that system analysis and design activities are cost-effective and deliver high-value insights while prioritizing system features that offer the greatest return on investment.

For cyber-physical systems, it is important to leverage digital environments for rapid design and ***feedback loops*** where changes are more cost-efficient. An example of rapid development in a digital ecosystem is SKYBORG, "an autonomous aircraft teaming architecture that enable[d] the Air Force to posture, produce and sustain mission sorties at sufficient tempo to produce and sustain combat mass in contested environments. This initiative leverage[d] digital models to enable rapid design, test, and validation cycles with warfighter feedback with the intent of delivering faster and at lower costs. (AFRL, 2025).

Implementation Process

Implementation is where system elements are created to satisfy the requirements, architecture, and design. Implementation requires managing constraints such as technical feasibility, resource availability, budget limitations, and regulatory compliance to avoid delays and quality issues.

Applying multiple horizons of planning is the key to balancing these constraints in real time, providing flexible predictability. Longer-range strategic planning (program roadmap) ensures the teams are aligned on the final destination and have a high-level understanding of how the desired scope will be delivered on time and under budget. It is critical for understanding long-lead items and performing critical path analysis. Shorter-term rolling wave planning uses the current context/environment, including team capacity, to detail-plan near-term activities, providing the team the means to refine the plan based on the current reality. Agile planning is also ***incremental,*** meaning that the details emerge, and empirical, meaning it is influenced by learning/***feedback.***

An ***economic view*** ensures that implementation decisions are financially viable while keeping in mind the impact of the cost of delay. While there may be financial impacts, there are also schedule or time implications that need to be accounted for. This elevates the need for delivering mission capabilities at the speed of relevance.

Detailed designs are realized by physical components or software modules through activities like coding, hardware fabrication, and systems integration. These activities must be executed with a focus on efficiency, quality, and adaptability of evolving requirements. Value creation occurs by ensuring alignment with stakeholder expectations ***(customer/user centricity)*** and efficient resource use. Disciplined and ongoing testing and automation helps mitigate risks and enhances overall reliability and performance, enabling ***data-based decisions.*** The adoption of modern development practices such as model-based systems engineering, digital engineering, and Industrial DevSecOps practices across the development and deployment of cyber-physical systems, enhances collaboration, reduces rework, and accelerates the delivery of mission-critical capabilities. Figure 5 demonstrates the iterative cycle and feedback loops for capability implementation.

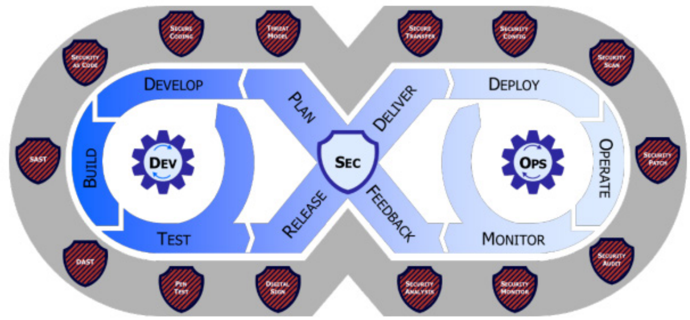

Figure: 5 DevSecOps Distinct Lifecycle Phases and Philosophies, 2021

Retrieved from [DoD Enterprise DevSecOps Strategy Guide]

Systems thinking facilitates seamless execution by considering dependencies and defining a critical path; especially for complex cyber-physical systems and where alignment with suppliers is critical. Leverage value stream mapping to improve and streamline the process and remove outdated processes that do not directly contribute to value creation. Streamlining processes and automating repetitive tasks can reduce cycle time and improve system quality. Use metrics to measure the improvements and **make data-driven decisions.**

Iterative development and Incremental delivery with short feedback loops ensures early functionality validation. Focusing on value and fostering collaboration among stakeholders, suppliers, and development teams, are essential for enhancing efficiency and ensuring the final integrated solution meets stakeholder needs within budget and technical constraints. Effort is on delivering frequently to the maximum extent possible. Iterative development and incremental delivery for software and hardware systems is in alignment with DODI 5000.87 which states, "The embedded software path provides for the rapid development, deployment, and insertion of upgrades and improvements to software embedded in weapon systems and other military-unique hardware systems. The system in which the software is embedded could be acquired via other acquisition pathways" (e.g., major capability acquisition) (OUSD, 2020).

Integration Process

The Integration process encompasses planning, preparation, and progressive aggregation of system elements, ensuring that interfaces are identified, activated, and verified for proper interoperation. Additionally, it should include systematic interface check-outs with enabling systems to confirm functional interaction and compliance with system requirements.

Systems thinking is important during integration because it considers the interdependencies between subsystems, which reduces risks and enhances interoperability.

In an ***iterative and incremental delivery*** model, integration starts early and occurs continuously, with clear points for evaluating functioning components against stakeholder requirements, enabling the team to ***base milestones on objective evaluation of working systems.*** These principles reduce program risk through early discoveries and enable agility as the team can pivot as necessary to address integration issues. By focusing on critical functionalities first and employing methods like continuous integration (CI) and automated testing, teams can detect issues early and minimize rework. This results in improved efficiencies and reduces costs as issues found late in the development life cycle are more costly to fix.

To help demonstrate the alignment of the 15288 Technical Processes with the iterative, flow nature of Lean Agile, we have taken the technical processes and highlighted where they appear within the DevSecOps phases as highlighted in Figure 6. This helps visualize how the Technical processes are recurring processes and not a serial approach to defining and building. and delivering value to the user community. The goal is applying systems engineering discipline with speed and agility to enable delivery to users at the speed of relevance.

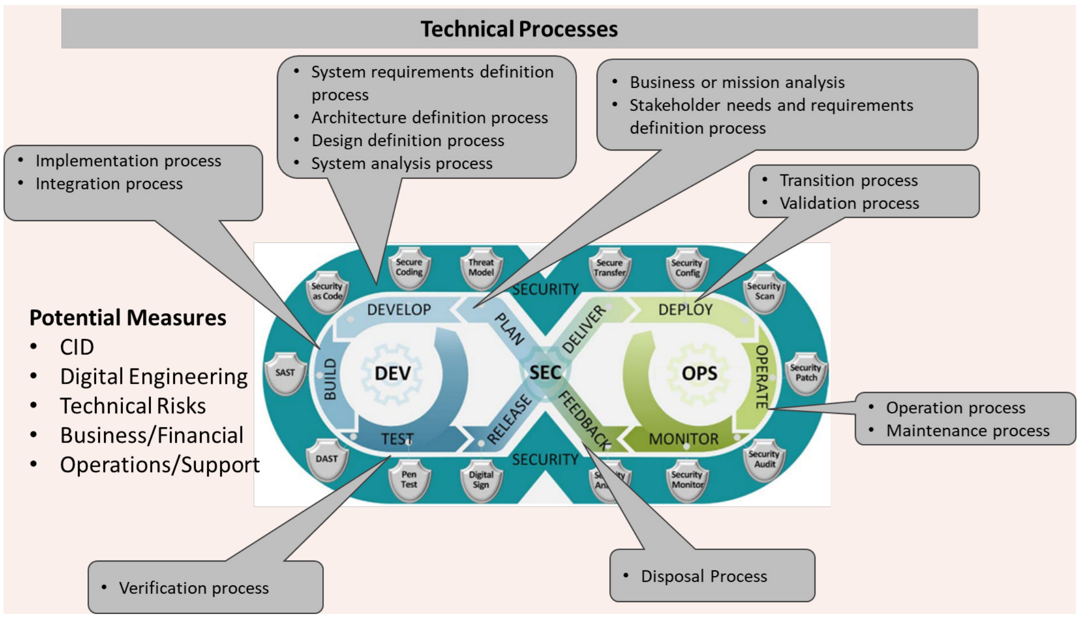

Figure 6: DevSecOps and 15288 Technical Processes
Reference: Henry, S. (2023).

With the emergence and continual advancement of digital tools such as 3D printing capabilities, digital shadows/models, digital twins, simulations, and AR/VR, new features can be regularly integrated and tested, even for cyber-physical systems leveraging the digital environments. These digital capabilities are planned and prioritized and become part of the road map to ensure there is budget and ample time to develop and implement them. Artificial intelligence can be used to quickly evaluate options and support engineers' ***data-driven*** decision-making process. Leveraging the variety of digital tools and environments which have become more prevalent, enables hardware to be iteratively designed as it moves from a digital model into a physical solution. With digital capabilities it is now possible to conduct integrated product demonstrations of software, hardware, and mechatronics, ensuring ***milestones are based on objective evaluation of working systems.*** These integrated solutions provide objective evidence of progress by looking at the state of the product versus task completion. This provides stakeholders the opportunity to ***iteratively*** validate the solution in progress while reducing rework which is often due to late discoveries. "Integration shifts the question from "What is the state of hardware?" and "What is the state of software?" to "What is the state of our integrated solution?" (Johnson, Yemen, et al, 2022). This provides programs the ability to measure progress not against task completion but rather against demonstration of progress toward meeting mission and business objectives.

Example. An example of these practices can be seen in the U.S. Army's Future Vertical Lift (FVL) program. With the rapid pace of change in technologies and the need to respond to changing threats and missions, the FVL program leverages agile practices, MOSA, and digital transformation capabilities. The government choices on architecture and MOSA were strongly supported in the U.S. Government Accountability Office (GAO) MOSA review and the GAO protest report for the program.

Verification Process

Verification determines whether system requirements are met by demonstrating, inspecting, testing, and analyzing the system or its components to ensure they perform as intended and meet stakeholder expectations. It confirms alignment with specifications, regulatory compliance, and readiness for subsequent phases like validation and deployment. In compliance with the Tri-Services memo on MOSA for DOD Weapon Systems, programs "must include the appropriate verification method and timing within the acquisition program's development to ensure MOSA conformance" (DoD Memo, 2024). In an agile context, the successful achievement of verification milestones is that the system meets defined requirements and is based on ***objective evaluation of working systems.***

Conducting verification early and continuously in an iterative, incremental lifecycle helps identify and correct defects quickly, reducing rework and mitigating risks, enabling teams to make data-driven decisions.

For safety-critical systems, ongoing compliance and conformance testing is vital to detecting issues early and preventing late-stage discoveries. Verification must balance constraints such as resources, time, budget, security, and regulatory compliance. Automated testing, simulated environments, and ***iterative*** analysis improve efficiency and confirm both functional and non-functional requirements. This systematic approach supports Lean-Agile principles, maximizing value delivery while minimizing cost and ensuring high-quality systems are delivered on time.

Transition Process

The Transition Process ensures a seamless handover from development to operations, enabling the system to function as intended and deliver value to stakeholders. The system must meet operational requirements and objectives, ensuring reliability in the intended environment, and includes sufficient documentation and training for release and long-term sustainability. Comprehensive testing, integration, and validation confirm that the system performs as expected and satisfies stakeholder needs.

The Transition process integrates the practices of Industrial DevSecOps (the application of Lean Agile and DevOps to cyber-physical systems) for the full development and sustainability of cyber-physical systems (CPS) to facilitate ***rapid development, deployment, and feedback loops.*** The emphasis is on "building-in" high security and safety features with a fully automated risk management process aiming to reduce the traditional lengthy authorization process to a "continuous ATO (Authority to Operate). Leveraging a shift-left mindset, the needs of the transition process are integrated earlier in the technical processes. Agile's CI/CD pipeline facilitates the quick and secure deployment of modular components (a system ***architected for change and speed***) and new features, enabling organizations to release updates or add new modules without disrupting the entire system" (DoD Enterprise DevSecOps Guide, 2021). When building and releasing capabilities for cyber-physical systems there are two specific scenarios that come into play during the transition processes. One scenario is specific to software-only enhancements after the CPS has already been launched: for example, a satellite receiving new software capability enhancements to improve performance or security. Software enhancements can take full advantage of the DevSecOps pipeline into transition and release. Another scenario might be for a new CPS that has not been released, the transition phase is more complicated as it includes both software and hardware components and extensive testing; for example, the initial launch of the satellite. Lean Agile principles apply in both scenarios.

Effective Transition processes minimize downtime, reduce operational risks, and enhance system readiness. Furthermore, addressing constraints such as scheduling limitations, resource availability, interoperability with existing systems, regulatory compliance, and user acceptance ensures that the system operates reliably in its operational environment. Utilizing emerging technologies like Machine Learning Ops (MLOps) and Artificial Intelligence Ops (AIOps) can enhance system monitoring and analysis, providing (near) real-time feedback on system performance and identifying areas for improvement. This systematic approach fosters ***continuous improvement by enabling feedback loops and iterative enhancements*** based on operational performance, metrics, and real-time data when possible.

Validation Process

Validation evaluates the system or system element to ensure it fulfills stakeholder needs in its intended environment. It assesses whether the system performs its functions correctly, reliably, and effectively, considering real-world scenarios and user interactions. Validation confirms the system's suitability for its intended use, ensuring it delivers desired outcomes and aligns with functional requirements, user needs, operational objectives, and regulatory standards. By incorporating real-world scenarios ***(systems thinking)*** and user feedback ***(customer-centricity, iterative development and incremental delivery)***, objective evidence is gathered to assess if the system performs as expected. Clarifying what validation actions can be done early and often is key to ensuring ***early feedback***, shortened time to value and economical solution development and delivery. This systematic approach builds stakeholder confidence, minimizes risks, and ensures compliance with regulations while delivering a validated system that meets stakeholder needs efficiently.

Operations Process

Operations is where a system is used in its intended environment to fulfill its specified functions and capabilities. It encompasses activities such as monitoring, maintenance, support, user training, configuration management, and implementing upgrades to sustain functionality and optimize performance. The goal is to ensure the system operates efficiently, reliably, and securely while delivering continuous value to users and stakeholders ***(user/customer centricity).***

Exercised focus is on minimizing downtime, addressing user needs, and enhancing system performance through proactive maintenance, monitoring, and troubleshooting. ***Feedback loops from users and stakeholders enable iterative improvements*** and updates, ensuring the system adapts to evolving needs and continues to deliver value throughout its lifecycle.

Operational data and metrics provide valuable insights into system performance. Real-time data from the physical system is collected and fed into the digital models and digital twins. Leveraging digital twins with artificial intelligence and machine learning can lead to improved performance, predictive maintenance, ***data-driven decision-making,*** and operational agility. Operational agility "provides a ***fast feedback loop*** for operations to quickly assess alternatives and respond and identify ways to be competitive and agile" (Johnson and Yeman, 2024).

Maintenance Process

Maintenance is the systematic process of sustaining a system's operational capability throughout its lifecycle. This includes monitoring performance, correcting faults, updating components, and adapting to changing environments and user needs. The primary goal is to maximize system availability, reliability, and performance while minimizing downtime, ensuring that the system remains functional, secure, and compliant over time.

Systems that have been ***architected for change and speed*** enable successful maintenance by incorporating modular, flexible designs that simplify updates, repairs, and enhancements throughout the system life cycle. Modular architectures with well-defined interfaces allow individual components to be modified or replaced without impacting the entire system, reducing downtime and minimizing disruption. This approach enhances system longevity, reduces maintenance costs, and improves the system's ability to deliver ongoing value to stakeholders.

Systems designed with scalability and adaptability in mind can more easily accommodate evolving requirements, emerging technologies, and changing operational needs. Automated testing, version control, and configuration management practices further support efficient maintenance by enabling rapid verification of changes and ensuring system integrity.

Effective maintenance practices, such as preventive checks and system optimizations, are essential for long-term performance and sustainability. Leveraging emerging technologies such as machine learning improves predictive maintenance and predictive diagnostics of cyber-physical systems once deployed. Digital twins with artificial intelligence can also provide value in the testing of new or enhanced capabilities after the system has been launched. Testing in a virtual/digital environment can improve safety and security before releasing to the production environment, which is part of the Transition process (shift-left testing is a key aspect of an ***iterative and incremental development and delivery process)***. Leveraging these technologies requires investment, which requires planning for that investment early in the acquisition process. This approach promotes ***continuous improvement and long-term sustainability.***

Logistics stakeholders are engaged early in the design and assessment of modular components to ensure maintainability, interoperability, and operational efficiency are prioritized from the outset. Their inputs are integrated into system evaluations to optimize modular design for lifecycle support. Structured reviews, field data collection, and direct engagement with logistics personnel provide feedback on real-world performance, enabling iterative improvements. Formal feedback loops within logistics planning processes, performance metrics, and qualitative insights from focus groups and interviews help continuously refine the modular design to better align with logistics requirements and challenges.

Disposal Process

Disposal is where a system or system element is decommissioned or disposed of at the end of its operational life cycle. It involves safely removing the system or system element from service, dismantling its components, and managing residual materials, all while minimizing environmental impact and adhering to regulatory requirements. Proper disposal mitigates risks and ensures compliance with legal and ethical standards.

Disposal involves optimizing resource recovery and minimizing environmental effects. Strategic plans identify suitable disposal methods and assess risks. By conducting risk assessments and compliance audits, ensure that disposal activities meet regulatory standards and protect against liabilities. Effective disposal also facilitates the recovery and recycling of valuable materials, promoting resource conservation and cost savings. This approach not only supports sustainability goals but also builds stakeholder confidence through a commitment to ethical and environmentally responsible practices.

Lean Agile principles improve the Disposal process by integrating early planning for the disposal in the ***iterative and incremental development and delivery plan.*** Iterative evaluation of disposal options adapts to evolving needs and regulations ***(assume variability, preserve options).*** Automation, reusable components, and prioritized stakeholder needs reduce waste, cost, and risk (take an economic view). This approach enhances efficiency, sustainability, and alignment with system lifecycle goals.

Conclusion

Applying Lean Agile principles to the 15288 technical processes helps address the growing complexity, rapid technological advancements, and demand for faster capability delivery in modern systems engineering. While this paper does not exhaustively cover every application of Lean Agile to systems engineering technical processes, it aims to inspire new possibilities. Organizations have embraced Lean Agile for software development—the next step is extending these principles to systems engineering to create more adaptive, efficient processes.

The following are a few key takeaways that summarize the integration of Lean Agile principles with the 15288 technical processes:

- **Customer-Centric Systems Engineering**: Prioritize customer and mission needs across the entire system lifecycle, using feedback from rapid, integrated learning cycles to deliver value at the right time.
- **Iterative Development and Incremental Delivery:** Build iteratively for faster feedback improved quality, and early risk mitigation. Deliver incrementally and release at the speed of need.
- **Modular and Flexible Architectures** (architect for change and speed): Design systems for change by leveraging modular architectures that support adaptability, reduce complexity, and enable continuous integration and verification. A well-defined, modular architecture and MOSA are crucial for effectively implementing and evolving solutions while meeting non-functional requirements like resilience and performance.
- **Cross-Functional Collaboration**: Foster collaboration across engineering, development, testing, and operations teams to align activities, accelerate decision-making, and improve quality.
- **Optimizing Flow**: Apply Lean principles to remove bottlenecks, automate repetitive tasks, and prioritize high-value work to improve the speed and predictability of system delivery.
- **Economic Decision-Making**: Make trade-offs that balance technical performance, cost, schedule, and risk to maximize value throughout the system life cycle.
- **Continuous Verification and Validation**: Embed verification and validation activities into the development process to detect defects early, reduce rework, and ensure the system meets stakeholder expectations. Use Model-Based Systems Engineering (MBSE), a digital infrastructure, and digital twins to support early and continuous Verification and Validation (V&V).
- **Learning Culture**: Embrace a mindset of experimentation, feedback, and continuous improvement to enhance system performance and adaptability over time.

Lean Agile principles with 15288 Technical processes can enhance the delivery of high-quality, resilient systems capable of meeting the demands of dynamic, mission-critical environments, particularly in defense and aerospace.

We agree that modernization is more than merely updating systems engineering processes; it is about "advancing the engineering practice" and fostering cross-organizational collaboration to optimize defense capabilities, processes, and technical expertise to accelerate value delivery (OUSD R&E, 2022).

References

Office of the Under Secretary of Defense, Research and Engineering (OASD R&E). (2022). Systems Engineering and Architecture. Department of Defense. Retrieved from Systems Engineering – DoD Research & Engineering, OUSD(R&E).

Office of the Under Secretary of Defense for Research and Engineering. (2023, June). SERC WRT-1051-2023 . Department of Defense. https://www.cto.mil/wp-content/uploads/2023/06/SERC-WRT-1051-2023.pdf

NDIA Systems Engineering Standards Committee. (2015). Guidance for Utilizing Systems Engineering Standards (IEEE 15288.1 and IEEE 15288.2) on Contracts for Defense Projects. NDIA. Arlington, VA. Retrieved February 2025 from https://www.ndia.org/-/media/sites/ndia/meetings-and-events/divisions/systems-engineering/studies-and-publications/guidance-for-utilizing-se-standards-ieee-july-2015.pdf?download=1

ISO/IEC/IEEE FDIS 15288 Systems and software engineering — System life cycle processes "ISO/IEC/IEEE International Standard - Systems and software engineering--System life cycle processes," in ISO/IEC/IEEE 15288:2023(E) , vol., no., pp.1-128, 16 May 2023, doi: 10.1109/IEEESTD.2023.10123367. keywords: {IEEE Standards; ISO Standards; IEC Standards; Software engineering; Systems engineering and theory; Product lifecycle management},

Defense Acquisition University. (2020). Adaptive Acquisition Framework Pathways. Retrieved from Adaptive Acquisition Framework | Adaptive Acquisition Framework

U.S. Department of Defense. OUSD R&E. (2020). DoDI 5000.88: Engineering of Defense Systems. Retrieved from https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500088p.PDF

International Organization for Standardization (ISO), International Electrotechnical Commission (IEC), & Institute of Electrical and Electronics Engineers (IEEE). (2023). Systems and software engineering -- System life cycle processes. Standard number: ISO/IEC/IEEE 15288:2023

U.S. Department of Defense. (2020). DoDI 5000.87: Operation of the Software Acquisition Pathway. Retrieved from DoDI 5000.87, "Operation of the Software Acquisition Pathway," October 2, 2020

U.S. Department of Defense. OUSD R&E.(2023). DoD Mission Engineering Guide. Washington DC. https://ac.cto.mil/wp-content/uploads/2023/11/MEG_2_Oct2023.pdf

NDIA Systems Engineering ADAPT. (2024). Moving from predictive planning to empirical planning for Systems Engineering. NDIA. Arlington, VA.

Phipps, Young and Christensen. (2011). Structural Optimization Helps Launch Space Payloads. Altair. Retrieved from https://altair.com/docs/default-source/resource-library/c2r2011-structural-opt.pdf?sfvrsn=33f8a9fc_3.

Johnson, S., & Yeman, R. (2023). Industrial DevOps. IT Revolution: Portland, OR.

10 USC 4401: Requirement for modular open system approach in major defense acquisition programs; definitions

NDIA Systems Engineering Division. (2023). MOSA Implementation Considerations, Information Needs and Metrics. NDIA. Arlington, VA.

Johnson, Yeman, Koehnemann, Shupack, Yasar, Grinnell, Brey, Farley, and Corman. (2022). Overcoming Barriers to Industrial DevOps: Working with the Hardware-Engineering Community. The DevOps Enterprise Journal 4, no. 2 (Fall 2022).

Skelton, M., & Pais, M. (2019). Team Topologies: Organizing for fast flow of value. IT Revolution: Portland, OR.

Johnson, S., & Yeman, R. (2023). Industrial DevOps. IT Revolution: Portland, OR.

Johnson,S. and Yeman, R. (2024).The Application of Industrial DevOps Using Digital Twins. IT Revolution: Portland, OR.

U.S. Army Research Laboratory. (2023). SKYBORG—Air Force Research Laboratory. Retrieved from https://www.afrl.af.mil/

AFRL. (2025). Skyborg. Retrieved from SKYBORG – Air Force Research Laboratory.

U.S. Department of Defense. (2020). DoDI 5000.87: Operation of the Software Acquisition Pathway. Retrieved from DoDI 5000.87, "Operation of the Software Acquisition Pathway," October 2, 2020

Johnson, S., Yeman, R., Koehnemann, H., Shupack, J., Yasar, H., Grinnell, R., Brey,D., Farley, S., and Corman, J. (2022). Overcoming Barriers to Industrial DevOps: Working with the Hardware-Engineering Community. IT Revolution: Portland, OR.

DOD Armed Forces. (2024). Memorandum for Service Acquisition Executives and Program Executive Officers. Retrieved from Tri-Service-Memo-Signed-17Dec2024.pdf

Department of Defense (DoD). (2021). DoD Enterprise DevSecOps Guide. U.S. Department of Defense. Retrieved from https://dodcio.defense.gov/Portals/0/Documents/Library/DoD_Enterprise_DevSecOps_Strategy_Guide.pdf

Johnson,S. and Yeman, R. (2024).The Application of Industrial DevOps Using Digital Twins. IT Revolution: Portland, OR.

Henry, S. (2023, Oct. 16-19). Program MOSA Transformation. NDIA Systems & Mission Engineering conference. Norfolk, VA. USA.

NATIONAL DEFENSE INDUSTRIAL ASSOCIATION
AFFILIATED ORGANIZATIONS

The National Defense Industrial Association is the trusted leader in defense and national security associations. As a 501(c)(3) corporate and individual membership association, NDIA engages thoughtful and innovative leaders to exchange ideas, information, and capabilities that lead to the development of the best policies, practices, products, and technologies to ensure the safety and security of our nation. NDIA's membership embodies the full spectrum of corporate, government, academic, and individual stakeholders who form a vigorous, responsive, and collaborative community in support of defense and national security. For more than 100 years, NDIA and its predecessor organizations have been at the heart of the mission by dedicating their time, expertise, and energy to ensuring our warfighters have the best training, equipment, and support. For more information, visit **NDIA.org**