WRT-1058
SYSTEMS ENGINEERING MODERNIZATION POLICY, PRACTICE, AND WORKFORCE ROADMAPS

Principal Investigator:
Mr. Thomas McDermott, Stevens Institute of Technology

Co-Principal Investigators:
Dr. Bryan Mesmer, University of Alabama in Huntsville
Dr. Nil Ergin, Penn State University

April 4, 2023

Sponsor: Office of the Under Secretary of Defense for Research & Engineering

SYSTEMS ENGINEERING RESEARCH CENTER

The Networked National Resource to further systems research and its impact on issues of national and global significance

TASK ORDER NO. 0568

Final Technical Report SERC-2023-TR-002

DISCLAIMER

The Systems Engineering Research Center (SERC) is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Office of the Assistant Secretary of Defense for Research and Engineering (ASD(R&E)) under Contract [HQ0034-19-D-0003, TO#0568].

Any views, opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense nor ASD(R&E).

No Warranty.
This Stevens Institute of Technology and Systems Engineering Research Center Material is furnished on an "as-is" basis. Stevens Institute of Technology makes no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material. Stevens Institute of Technology does not make any warranty of any kind with respect to freedom from patent, trademark, or copyright infringement.

This material has been approved for public release and unlimited distribution.

RESEARCH TEAM

The following researchers and students contributed directly to this project and report:

Name	Organization	Labor Category
Tom McDermott	Stevens	Principal Investigator
Bryan Mesmer	Alabama-Huntsville	Co-Principal Investigator
Nil Ergin	Penn State	Co-Principal Investigator
Molly Nadolski	Stevens	Researcher
Adrian Barb	Penn State	Researcher
Robert Paul Lewis	Alabama-Huntsville	Researcher
Kristin Weger	Alabama-Huntsville	Researcher
Amy Guerin	Alabama-Huntsville	Researcher

TABLE OF CONTENTS

LIST OF FIGURES

EXECUTIVE SUMMARY

This report presents the results of the research on research task WRT-1058: Systems Engineering Modernization Policy, Practice, and Workforce Roadmaps. This research task began with a companion research task, WRT-1051: Program Managers Guide to Digital and Agile Systems Engineering Process Transformation. Together, these support a larger set of activities being led by OUSD/RE under the term "Systems Engineering Modernization" (SEMOD). The motivation for SEMOD stems from the need to integrate across independent guidance provided down to the DoD SE and acquisition communities related to Digital Engineering, Modular Open Systems Approach, Mission Engineering, and Software Engineering/Agile/Devops, and across the multiple pathways of the Adaptive Acquisition Framework. The SERC/government research team found *there is a lack of an integrated approach to implementation of SE Focus Areas that is creating a delay in full implementation of the Digital Transformation which is necessary to ensure the relevant guidance, skills, and training are available to deliver a robust, disciplined approach to weapon systems acquisition*.

The SERC has been tasked with multiple research threads in this research:

1. SEMOD Framework: create an integrating framework that incorporate the key activities in each focus area and generate options for program implementation. The initial integration framework was developed on WRT-1051 and used to inform a series of workshops. These workshops resulted in a set of pain points for integration. The updated integration framework, termed "the Supra-System Model," is discussed in Part 1 of this report.

2. SEMOD Roadmaps: develop a set of roadmaps that define research and development activities for long-term implementation of SE Modernization into DoD engineering and acquisition activities. Part 2 of this report updates the SEMOD pain points from WRT-1051 and links these to a set of roadmaps. Part 3 of this report focuses in on the concept of a Government Reference Architecture (GRA) and associated Modular Open Systems Approaches (MOSA) as a means to better manage the government/contractor workflows and authorities over time.

3. SEMOD Policy and Guidance: the Team completed an analysis in WRT-1051 of DoD policy and guidance documents across the SEMOD focus areas and acquisition pathways. Part 4 of this report updates the analysis and provides a sample rewrite of the "Engineering of Defense Systems Guide" that incorporates more of a "how-to" guide to SE Modernization across focus areas and acquisition pathways.

4. SEMOD Information Graph: a goal of SEMOD was to create an information graph linking information from policy and guidance and related lessons learned into an easily accessed digital tool. This research found that there is not a base ontology that links systems engineering and acquisition, definitions are not

consistent, and there is no standard common taxonomy to draw from. Part 5 of this report describes initial research toward building a formal digital ontology linking military doctrine, engineering, and acquisition as a starting point to this goal.

5. SEMOD Lessons Learned: Part 6 of this report provides a set of research activities focused on development and collection of lessons learned in SE Modernization. These lessons learned reflect both current and future needs. All the lessons learned are related to digital transformation and associated digital and model based SE practices. The lessons leaned areas are exemplar reference implementations, adoption, modeling guidance, and data management.

6. Workforce Development Strategies: the project partnered with the Defense Acquisition University to explore workforce development strategies specific to SEMOD. Part 7 discusses this work and opportunities for future effort.

Primary findings of this research include:

- SE Modernization responds to the ongoing digital transformation of DoD acquisition and sustainment activities which have traditionally followed rigorous systems engineering processes. The systems engineering processes remain valid, but the practices need to change to take advantage of the digital transformation. The transformation is guided by the DoD Digital Engineering strategy as an "*an integrated digital approach that uses authoritative sources of system data and models as a continuum across disciplines to support lifecycle activities from concept through disposal*."[1] We derived a primary value statement from digital transformation as "**realized in more seamless and efficient transfer of data and models from underlying performance drivers through models to decisions, as well as ease of drilling back down from decisions to data**" and goal of SE Modernization process as "**create a more agile and responsive acquisition system that can quickly and effectively meet the needs of the warfighter**." To date DoD DE efforts have been more focused on the creation of authoritative sources of data and models than the value achieved by digitizing the underlying transformations and pursuing agile lifecycle process innovation. This is creating slow uptake of modernized systems engineering capabilities and processes in DoD program offices.

- DoD policy and guidance as related to the four focus areas, systems engineering and engineering of defense systems, and the six Adaptive Acquisition pathways[2] is poorly integrated. Current policy and guidance suffer from independent terminology and jargon across each focus area and acquisition pathway. Current policy and guidance provide only limited communication of the intent of the digital transformation. In addition, current policy and guidance remain highly milestone driven, overly focused on new development, and lack focus on update and sustainment - despite DoD calls for more continuous and rapid deployment of capabilities. Finally, the vision in the DoD

[1] DoD Digital Engineering Strategy, 2018.
[2] DoD Instruction 5000.2, Operation of The Adaptive Acquisition Framework, 2020.

Data Strategy of “a data-centric organization that uses data at speed and scale for operational advantage and increased efficiency” is not sufficiently captured into engineering policy and guidance.[3]

- As a result, the systems engineering and related acquisition guidance, as well as much of the systems engineering professional community guidance, continues to operate with a mental model of linear, milestone driven technical and management processes as determined by static, often document based artifacts. The culture is proving difficult to overcome in the DoD and defense industrial base. In this research we developed and have been promoting a new mental model of a systems lifecycle – the “supra-system model” – that is continuously iterated and layered from data to models to decision artifacts. This mental model helped to organize a much more focused set of SEMOD pain points and lessons learned.

- Associated with both the Data Strategy and the Digital Engineering Strategy, programs are finding that cost, complexity and lack of guidance on development of tailored Integrated Development Environments (IDEs) for Systems Engineering remains a primary pain point. WRT-1051 recommended additional development of a set of Exemplar Reference Implementations (ERIs) for these IDEs as tailored to the types of systems and acquisition programs within a DoD program office. This report further recommends the development of a concept of operations and set of use cases as an initial step toward this need.

- This report proposes the need for and actions that should be taken to establish such an exemplar reference architecture.

The organization of this report is intentionally organized into a set of smaller standalone report sections. This was done so that each report could be entered as a standalone artifact into the SE Modernization Body of Knowledge (SEMODBoK).

Relationship to other SERC and AIRC Projects

The Systems Engineering Research Center (SERC) is a federally funded University Affiliated Research Center focused on SE methods, processes and tools and their use in defense acquisition. The Acquisition Innovation Research Center (AIRC) was established along with the SERC to infuse innovation and alternative disciplines from academia to better respond to rapidly changing threats and technological advances. The SEMOD initiative has worked broadly across both SERC and AIRC associated research tasks to draw information related to the SE Modernization needs and goals. These research areas are listed below (with report links if published).

- Transforming Systems Engineering through Model Based Systems Engineering-NAVAIR (https://sercuarc.org/serc-programs-projects/project/26), Transforming Systems Engineering through Model Based Systems Engineering-CCDC (https://sercuarc.org/serc-programs-projects/project/27), and Digital Engineering

[3] DoD Data Strategy, 2020.

Migration Of Evolved Strategic Satcom System Engineering and Technical Management Processes (project WRT-1054)

- Digital Engineering Measures (https://sercuarc.org/serc-programs-projects/project/57).
- Model Curation Innovation and Implementation (https://sercuarc.org/serc-programs-projects/project/88).
- Approaches to Achieve Modularity Benefits in the Acquisition Ecosystem (https://sercuarc.org/serc-programs-projects/project/56).
- Digital Data Management & Analytic Strategy (project WRT-1049).
- Integrated Mission Equipment Architecture Process for Vertical Lift Systems (project ART-016)
- Agile Acquisition: History and Recommendations (project WRT-1049.3).
- Joint Capabilities Integration and Development System (https://acqirc.org/publications/research/joint-capabilities-integration-and-development-system-jcids/).
- Additive Manufacturing and Digital Engineering Strategy Development (https://acqirc.org/publications/research/additive-manufacturing-and-digital-engineering-strategy-development/).
- Data-Driven Capability Portfolio Management Pilot (https://acqirc.org/publications/research/data-driven-capability-portfolio-management-pilot/).
- DE Contracting (project WRT-1057.18g)
- DAU Digital Engineering Simulation (https://sercuarc.org/serc-programs-projects/project/118).
- Mission Engineering Competencies (https://sercuarc.org/serc-programs-projects/project/58).
- Digital Engineering Competency Framework (https://sercuarc.org/serc-programs-projects/project/86).

PART 1: THE SUPRA-SYSTEM MODEL

This report describes a component of the research known as the SE Modernization Integration Framework which became a new lifecycle model depiction we call "The Supra-System Model." This model recognizes the particular role of data and associated digital model transformations in the SE process, which are not explicit in current SE lifecycle model depictions. This report first discusses some of the history of SE and associated lifecycle models in DoD acquisition, then proposes the imperatives for changing current "mental models" of SE and acquisition process.

SE MODERNIZATION INTEGRATION FRAMEWORK

Program managers today are facing a myriad of acquisition process changes centered on the need for more rapid deployment of capabilities, better weapon system portfolio management, and efficiencies created through digital transformation. There is a need for documentation of lessons learned, program best practices, and standard guidance for program Systems Engineering that incorporates a holistic approach inclusive of the four SE Modernization focus areas, the six acquisition pathways, and the digital transformation outlined in the DoD Data Strategy.

In this project we first attempted to derive a framework to integrate across all aspects of future systems engineering by analyzing the text from current SE-related SE standards and the independent DoD guidance from each of these change areas. We found that existing DoD and SE process guidance did not capture the relationships across these areas of interest. We also recognized that systems engineering guidance still retains its historical alignment with defense Major Development and Acquisition Programs (MDAP) and has not become integral with other engineered system approaches such as innovation and prototyping, agile software development, business and service systems, and data-connected systems. We found that we had to step away from history and visualize a new set of mental models to guide the practice of systems engineering in the future. This report starts with a historical view of SE in DoD acquisition activities, discusses the imperatives driving a modernized view, places SE in the digital transformation of all engineering and acquisition, then proposes "The Supra-System Model" as a revised mental model that integrates across all engineering and acquisition activities.

SE HAS LONG BEEN INTEGRAL TO DOD ACQUISITION PROCESSES

Systems engineering principles and methods were adopted by the DoD in the late 1960's/early 1970's as a way to manage technical and programmatic development and risk across the engineering and management components of large complex weapon systems. The DoD published Military Standard 499A, Systems Engineering Management, in 1969. When the first iteration of DoD 5000.01 "The Defense Acquisition System" (DAS) was published in 1971, it defined a systems engineering related set of guidance including consideration for problem/operational needs, alternatives, test and evaluation, and support and update. It also introduced related management activities such as contracting, risk, source selection, and documentation. Mil-Std-499B was introduced in 1992 but was never published, as military standards were cancelled in the early 1990's as part of DoD acquisition reform initiatives. The majority of the concepts in Mil-Std-499B were incorporated into the Defense Acquisition University (DAU) Systems Engineering Fundamentals handbook in 2001. This contained a graphical SE lifecycle process description as well as the now familiar milestone driven acquisition process shown in Figure 1.

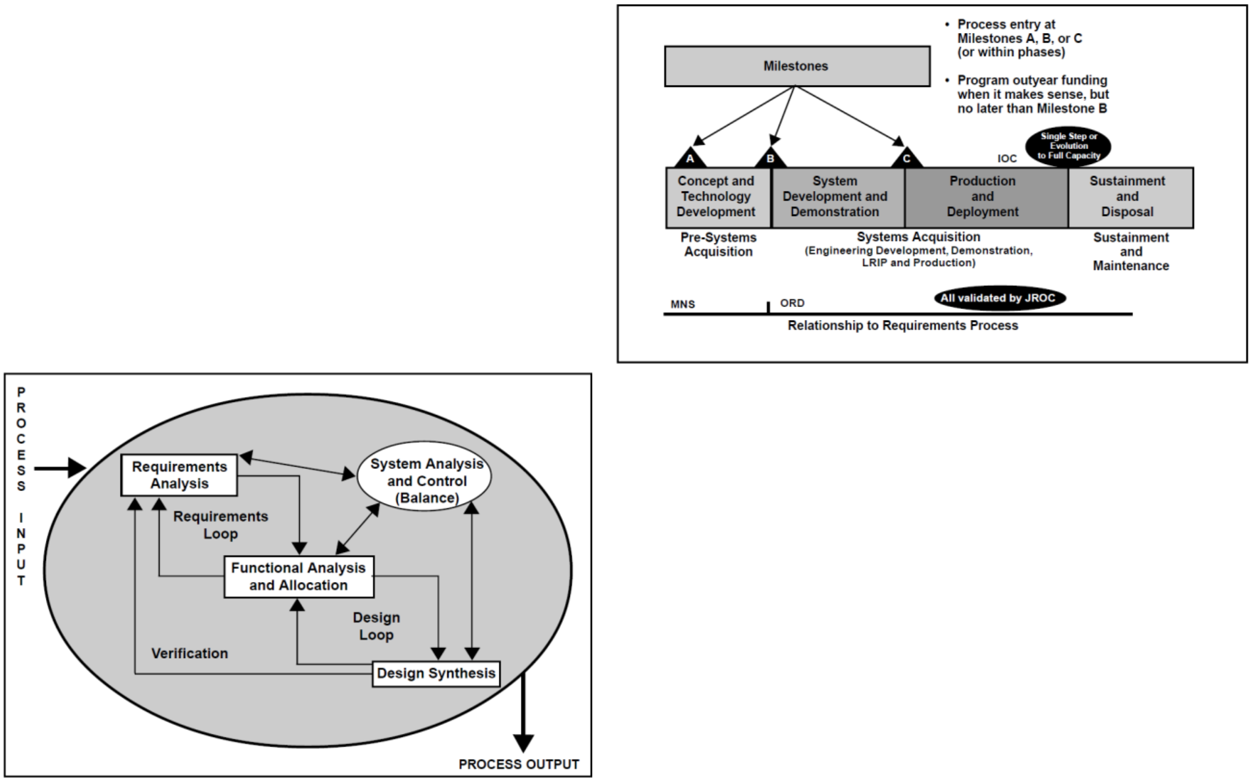


Figure 1. Initial DoD SE Process Model and acquisition process flow (DAU SE Fundamentals 2001).

The concept of the V-model was developed simultaneously, but independently, in Germany and in the United States in the late 1980s. It has been used interchangeably to represent 1) the concept of decomposition/synthesis of a systems development into different levels of functional definition, realization, and test (Figure 2); and 2) an SE technical and management process model (Figure 3).

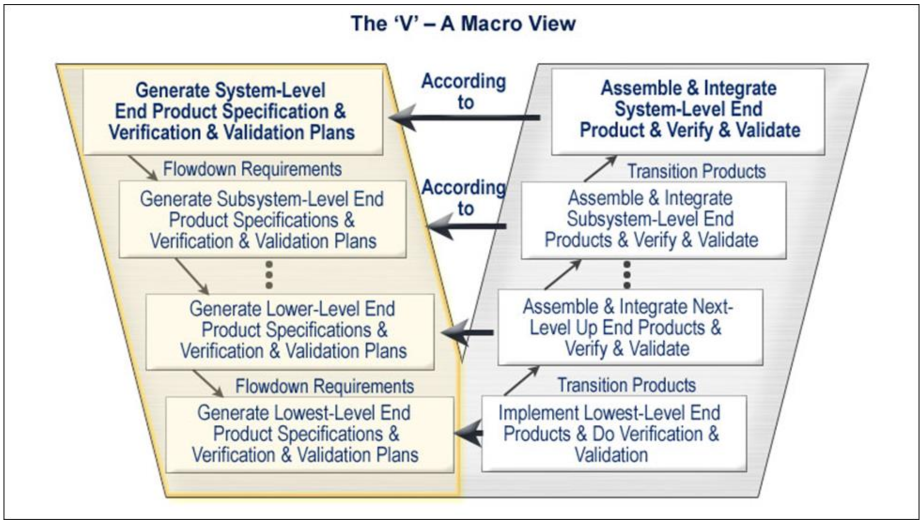


Figure 2. The Vee-model as a functional decomposition/synthesis process (SEBOK)

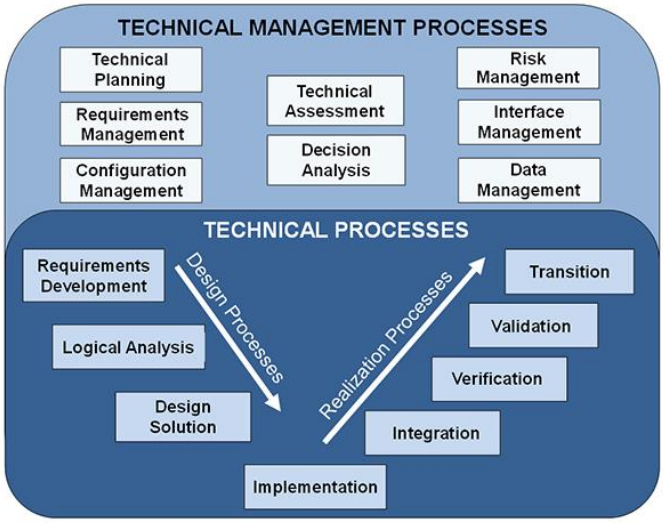


Figure 3. Revised SE Process Models of 2003 in the DAU Acquisition Encyclopedia, using the Vee-model to as a set of technical processes. https://www.dau.edu/acquipedia.

The current DAU documentation of the Vee-model generalizes and combines these two perspectives, as shown in Figure 4. The current acquisition model for MDAPs, now known as Major Capability Acquisitions (MCA) is shown in Figure 5. The current DoD SE Guidebook does not show an equivalent technical review process for the other AAF acquisition pathways.

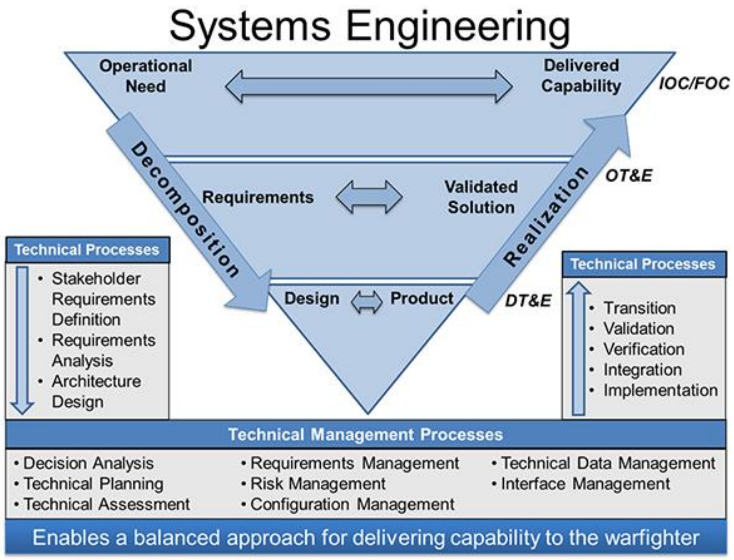


Figure 4. Current DoD SE Processes (DoD SE Guidebook 2022).

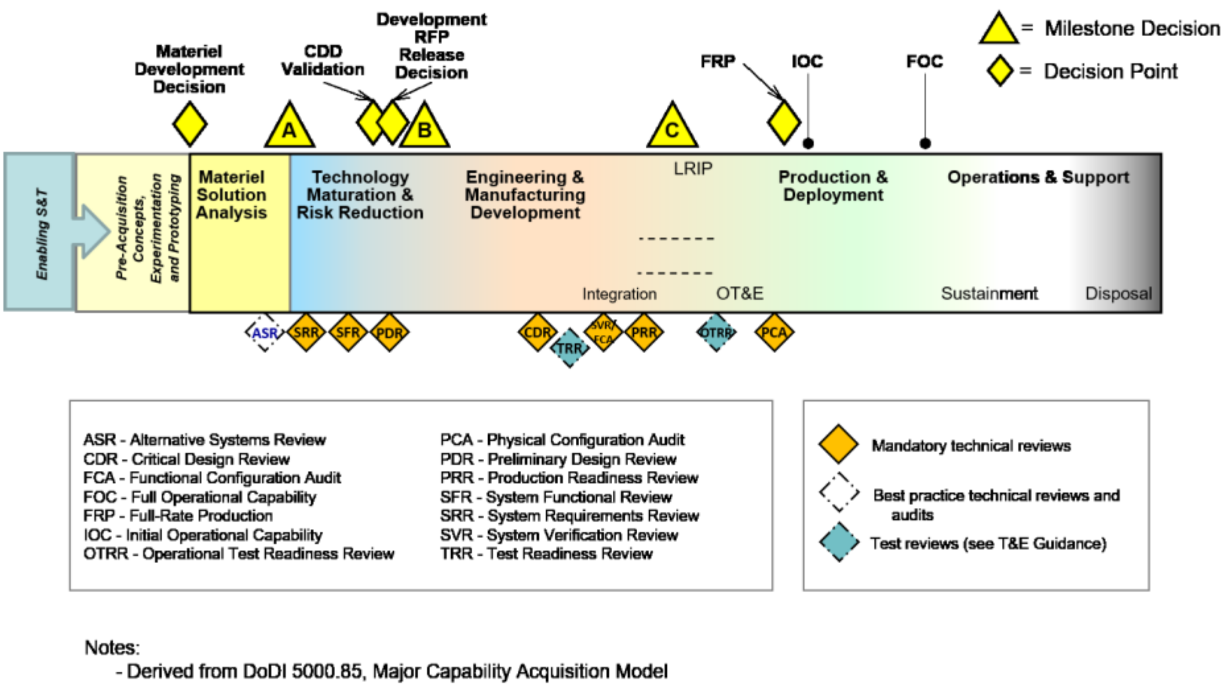


Figure 5. Technical Reviews and Audits for the MCA Life Cycle (DoD SE Guidebook 2022).

The discipline and its use in DoD acquisition has long been associated with realization of physical systems and related equipment, in Major Defense Acquisition Programs (MDAP). These figures are shown to highlight how “mental models” of SE have been codified into DoD acquisition for over 50 years. Meanwhile SE has grown to a much broader discipline, impacting software systems, simulation systems, manufacturing systems, innovation systems, enterprise systems, and other system types used by the DoD. Many of these applications have developed their own lifecycle process models in response to this traditional SE mental model, viewing it as slow and non-responsive to change.

WHY SE MODERNIZATION?

Today many defense capabilities are not only physical; they are software intensive, highly connected, and have extensive automation and user configuration capabilities. Software engineering became a discipline in 1967, manufacturing automation (the third industrial revolution) began in the 1970's, and the World-Wide-Web was invented in 1989. The DoD's Defense Modeling and Simulation Office was opened in the early 1990's and large-scale networked simulation of defense systems followed. All of these have continued to evolve the SE discipline, not as a whole, but as a set of related subdisciplines (systems engineering, software systems engineering, information technology and enterprise architecture, distributed modeling & simulation, and automated manufacturing systems). **It is notable that each of these subdisciplines views lifecycle process and technical review as something that is much more iterative than what is implied by current SE guidance.**

Following successful evolution of the Unified Modeling Language (UML) in the software discipline, the Systems Modeling Language (SysML) was published in 2007 and started the growth in Model-Based Systems Engineering (MBSE) as an improved approach to manage technical and programmatic risk. "Industry 4.0" originated in 2011 and introduced the concept of a "digital twin" as a non-physical product realization. The DoD's Digital Engineering (DE) Strategy was published in 2018, ushering in the vision of a digital era of systems engineering. As the International Council on Systems Engineering noted in their Vision 2035 document: "The future of Systems Engineering is Model Based, leveraging next generation modeling, simulation and visualization environments powered by the global digital transformation, to specify, analyze, design, and verify systems."

Throughout all of this change, the "mainstay" of systems engineering in the DoD, and associated DoD acquisition guidance, has continued to center on physical realization of large-scale monolithic systems and other critical capabilities intended to persist for many years. The need for rigorous definition, analysis and test of these critical systems will always exist, but perhaps the time has come to reintegrate the systems engineering subdisciplines into a common framework that responds to the digital age. Unfortunately, the SE Vision does not state “The future of SE is more iterative and responsive to user needs.” What the SE discipline really needs is to be more agile and responsive, which will be accomplished with more efficient lifecycle processes.

SE MODERNIZATION FOCUS AREAS

The SERC was tasked by the DoD to conceptualize and build an integration framework for SE Modernization as applied to all DoD acquisition life cycles. The DoD published its latest 5000 series guidance, "The Adaptive Acquisition Framework" in 2021. The AAF recognized new development and acquisition pathways for software, IT and business systems, services, and a streamlined "middle tier" acquisition for more mature rapidly fielded systems. This followed a series of legislative directions to the DoD around four focus areas for SE Modernization as defined below:

1. **Digital Engineering (DE)** – Defined in the DoD DE Strategy as "an integrated digital approach that uses authoritative sources of system data and models as a continuum across disciplines to support lifecycle activities from concept through disposal." As directed in DoD policy, "DE will provide for the development, validation, use, curation, and maintenance of technically accurate digital systems, models of systems, subsystems, and their components, at the appropriate level of fidelity to ensure that test activities adequately simulate the environment in which a system will be deployed."

2. **Modular Open Systems Approach (MOSA)** – Defined in DoD policy as "an acquisition and design strategy consisting of a technical architecture that adopts open standards and supports a modular, loosely coupled and highly cohesive system structure." This modular open architecture includes publishing of key interfaces within the system and relevant design disclosure. MOSA introduces the 'build for change, not to last' philosophy from software architecture across all aspects of DoD systems.

3. **Mission Engineering (ME)** – Defined in DoD guidance as "the deliberate planning, analyzing, organizing, and integrating of current and emerging operational and system capabilities to achieve desired mission effects. Mission Engineering is intended to provide engineered mission-based outputs to the requirements process, guide prototypes, provide design options, and inform investment decisions."

4. **Agile Development** – Defined in DoD guidance as "approaches based on iterative development, frequent inspection and adaptation, and incremental deliveries, in which requirements and solutions evolve through collaboration in cross-functional teams and through continuous stakeholder feedback. Agile approaches begin not with detailed requirements, but with a high-level capture of business and technical needs that provides enough information to define the software solution space, while also considering associated quality needs (such as security)."

In addition, the **DoD Data Strategy** (2020) emphasizes data as a strategic asset, collective data stewardship, data collection, enterprise-wide data access and availability, data fit for purpose, and design for compliance. At this point the SE community may be overly focused on "System Models" and underly focused on "System Data" in the Digital Engineering Strategy. As outlined in another SERC project, the Digital Engineering Competency Framework (DECF), data architecture, data standards, data governance, and talent and culture are all essential components of SE Modernization but are new concepts to systems engineers (DECF 2020, DECF 2021).

The four focus areas can be viewed as a layered model with a data strategy at the core, as shown in Figure 6. At the center, as envisioned by the DoD Digital Engineering strategy, is shared and authoritatively managed data. Modernization of systems engineering strives for seamless interoperability and integration of all engineering and management disciplines using authoritative sources of system data and models as the continuum that links the disciplines.

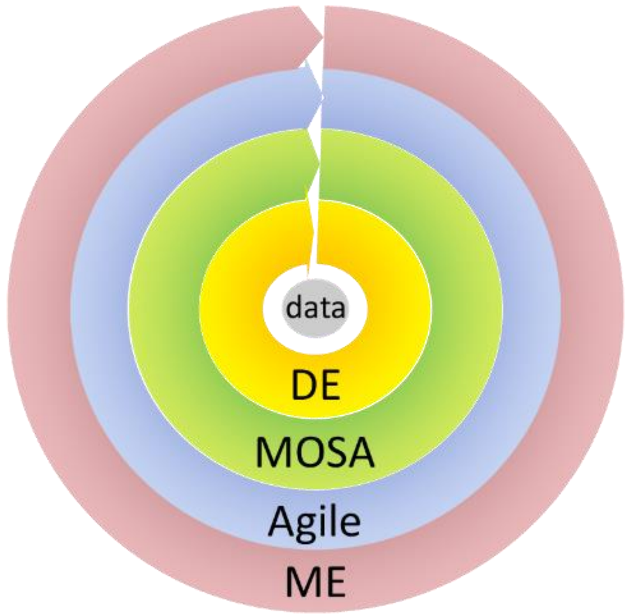


Figure 6. Four focus areas as a layered model.

It should be noted across these focus areas that DE and ME respond to “the future of SE is model-based” while MOSA and Agile respond to “the future of SE is more iterative and responsive to user needs.” These are not at odds, as appropriate use of models provides the foundation for iterative learning. **Fundamentally, modernization of SE lifecycle processes must define how data and models are used to be more iterative and responsive to user needs.** In this project we found that is not the mental model or vision of current policy and guidance related to these focus areas.

DIGITAL TRANSFORMATION OF SYSTEMS ENGINEERING

At the core of SE Modernization is "shared and authoritatively managed data" that can be transformed through various models and tools to create Digital Artifacts. These artifacts are used by various decision makers (in development) and others needing digital access to the design and descriptions of the system across its life cycle. In early years these artifacts were almost always paper documents or drawings, now they are mostly based on digital technologies but far from "seamlessly integrated and interoperable." The cartoon in Figure 7 might best describe the current state of digital artifact development.

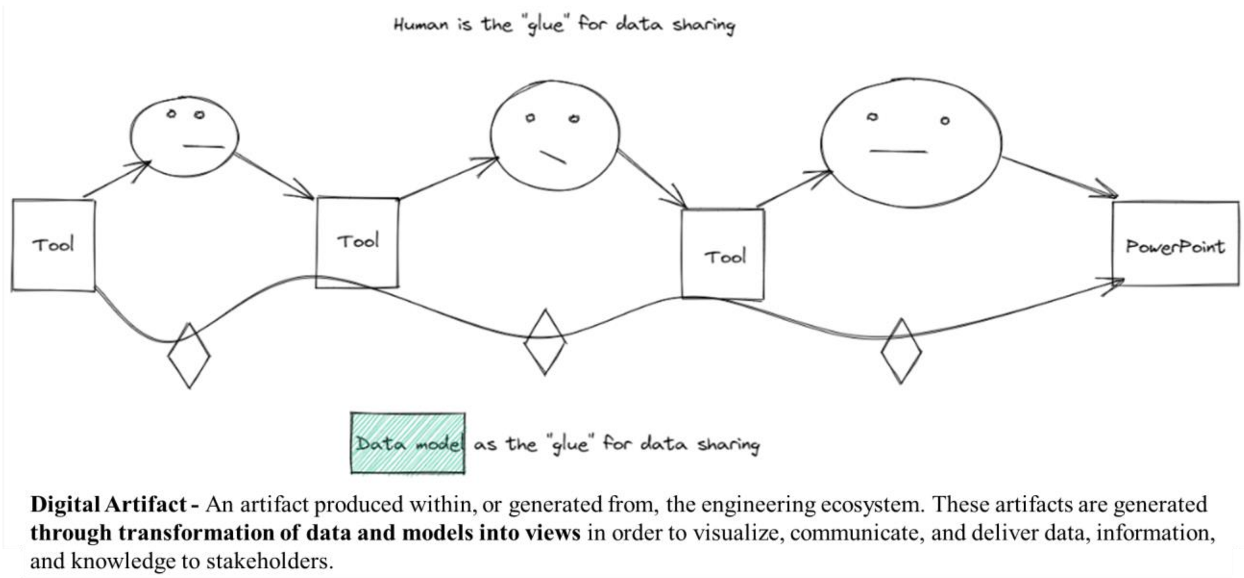


Figure 7. Data Transformation Mental Model.

Systems engineers have long used digital data and various modeling and analysis tools to produce digital artifacts for decision-making. However, the underlying data model has not been "seamlessly shared" and authority for that data has been distributed across independent activities, generally organized by discipline. Much of the "transformation" is still manual interpretation of disparate data and analyses. This manual interpretation limits our ability to be iterative and responsive across disciplines and disciplinary tools. One might describe the current state of systems engineering as seeing the whole while looking through a set of soda straws. We desire a fully integrated, iterative workflow. Today's primary challenge in digital engineering is not so much being "model-based," it is understanding and creating the underlying data model that integrates across requirements, design, and test, and across disciplines and disciplinary processes.

This leads us to the **value statement** for SE Modernization, depicted in Figure 8. SE Modernization Value Depiction.

The value of SE Modernization will be realized in more seamless and efficient transfer of data and models from underlying performance drivers through models to decisions, as well as ease of drilling back down from decisions to data.

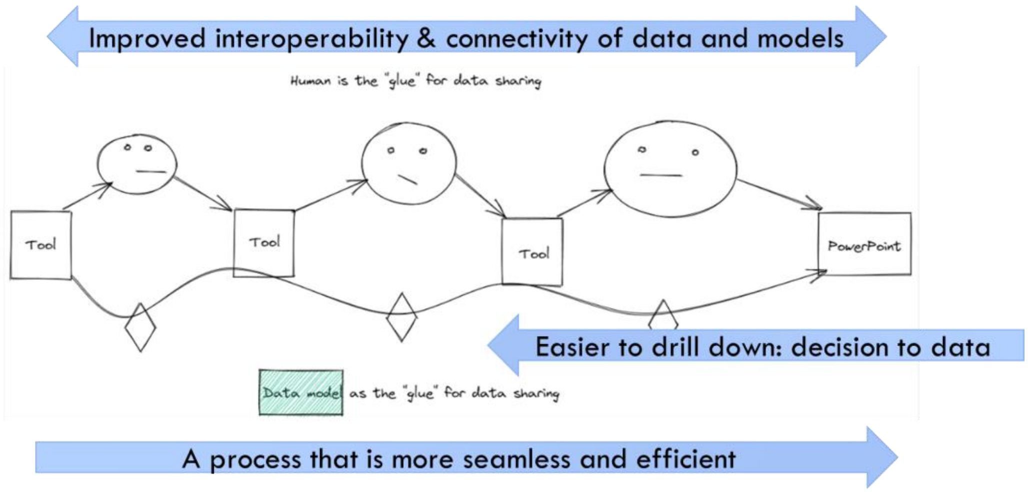


Figure 8. SE Modernization Value Depiction.

Systems engineering and related acquisition processes can be visualized as a set of iterative data transformations from sources of truth that produce artifacts for human consumption – across all stages of a system life cycle.

Figure 9 redraws the widely depicted Define->Realize->Deploy&Use stages of the SE Vee-model in a circular process to represent it as a:

1) set of data transformations at the core;
2) layered across disciplines & tasks;
3) in continuous iterative processes that could be entered from any point.

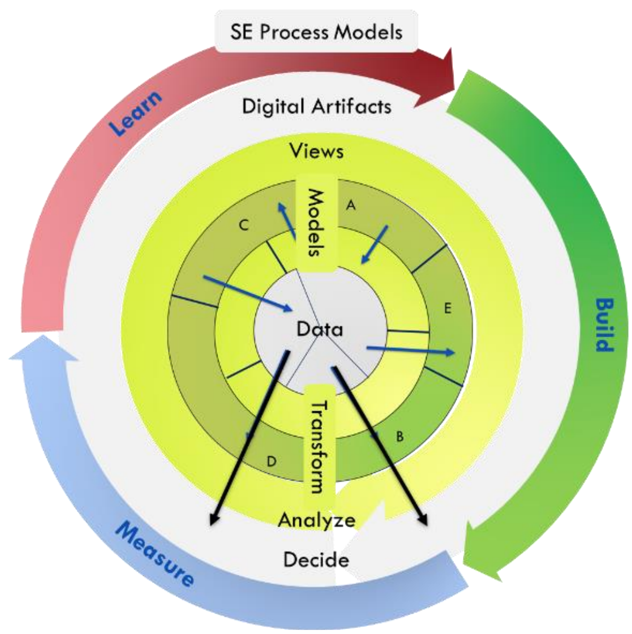


Figure 9. Circular Processes with Data at the Core.

In the figure we generalize define, realize, and deploy as a "Learn->Build->Measure" to be more consistent with current design literature.

In SE technical and management processes, data is transformed through models into views, which support analyses leading to decisions. These transformations have traditionally produced decision artifacts that were disconnected from the underlying data and models, captured in independent static document or presentation forms. Digital artifacts may still be documents or presentable views but should remain digitally connected to the underlying data and models. This process flow reflects "Data Transformed into Models then Analyzed through Views to make Decisions documented in Digital Artifacts." This process flow has been the core of SE technical and management processes within each lifecycle phase since the inception of SE. It just been a largely manual, inefficient process flow.

SE lifecycle processes as defined by ISO/IEC/IEEE 15288 do not define a specific ordering of process areas, but much of the literature and existing mental models imply a process ordering that is started in the learn (define) stages. SE lifecycle processes have been used not just in critical systems where up-front system definition and learning are essential, but also in system innovation, prototyping, and incremental definition activities where build-first is the pathway to learning; and in sustainment life cycles where deployed system measurement and learning should define the next build. This SEMOD circular mental model better recognizes that SE technical and management processes can be applied to any life cycle in any type of system. Figure 10 visualizes the domains of SE in association with the ordering of learn, build, and measure cycles.

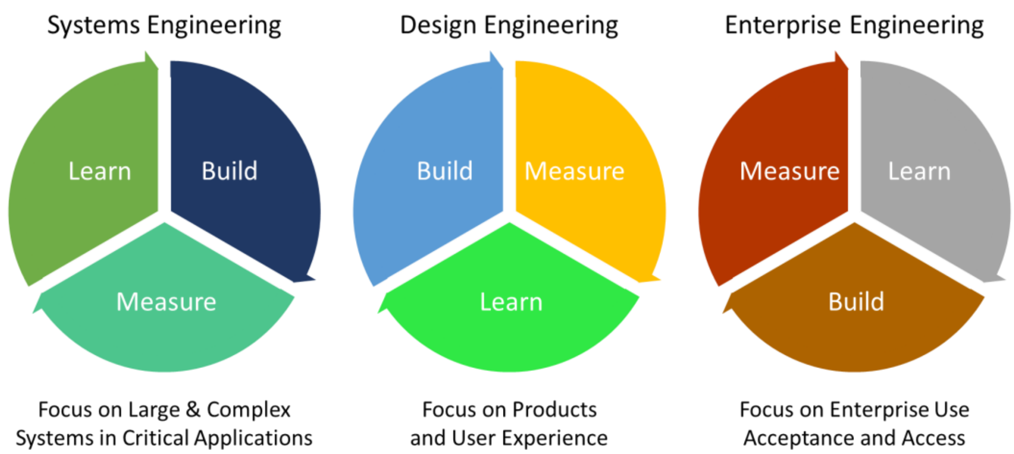


Figure 10. Different lifecycle ordering in different applications of SE.

The DoD published the Adaptive Acquisition Framework (AAF) in 2019 (Figure 11). Between 2019 and 2021, the AAF recognized new development and acquisition pathways for software, IT and business systems, services, and a streamlined "middle tier" acquisition for more mature rapidly fielded systems. In the AAF, the Major Capability Acquisition pathway continues the traditional use of upfront SE rigor and its rigorous Learn->Build->Measure cycle. However, the Urgent Capability, Middle Tier, and Software Acquisition pathways promote an abbreviated definition phase and rapid learning through builds, following SE processes developed in the engineering design and software development fields. The Defense Business Systems and Acquisition of Services pathways are more aligned with the Enterprise Engineering. The challenge of SE Modernization is to maintain appropriate SE rigor and associated process definition in these other pathways. **SE rigor is maintained using the data –>transform–> analyze–>decide flow of Figure 9, not through a specific ordering of SE processes**.

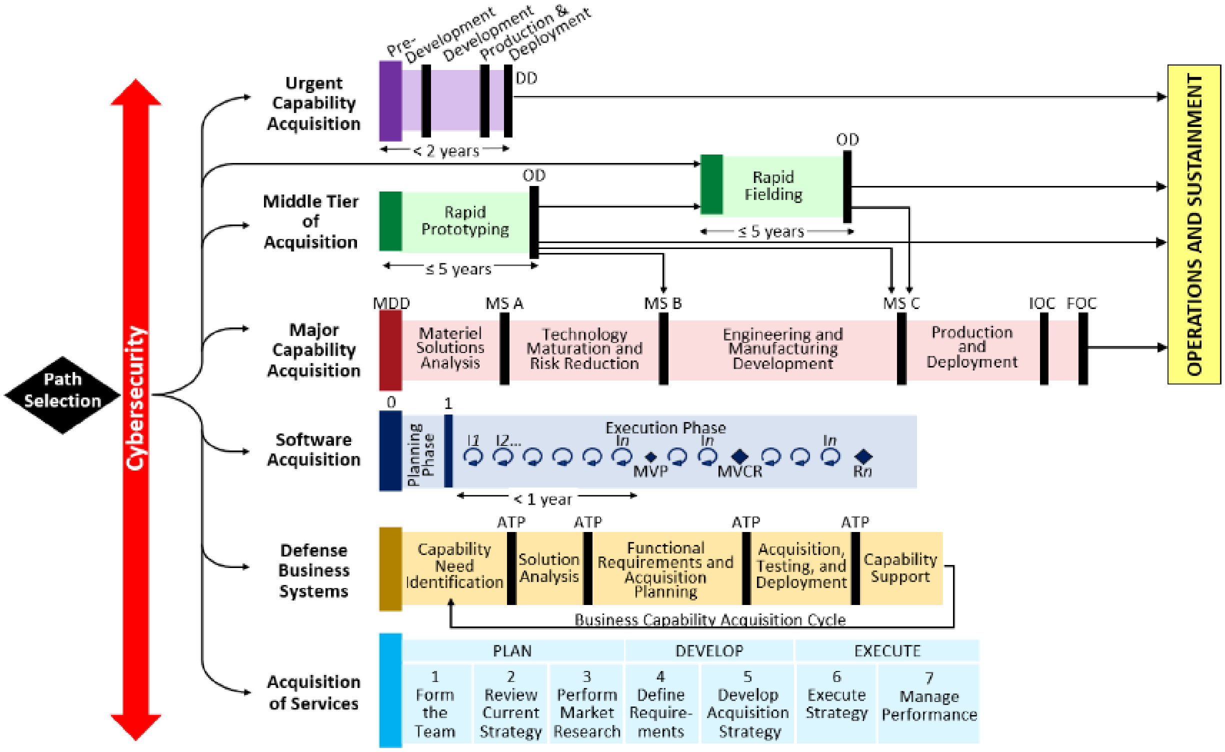


Figure 11. The Adaptive Acquisition Framework (aaf.dau.edu)

The workflow view in Figure 12 shows conceptually how shared and authoritatively managed data is transformed into digital artifacts in different life cycle stages in any pathway. This linear workflow model is familiar and comfortable to system engineers but does not represent the fact that these data transformations into and out of the shared and authoritatively managed federations of data and models actually happen iteratively and continuously across a life cycle. **Increasing responsiveness to the warfighter (or market) does not mean eliminating these critical SE processes, just increasing the number of iterations and shortening the cycle time between them.**

This figure also highlights how the broadly published goal of the DoD Digital Engineering Strategy "Provide an enduring authoritative source of truth" may be misleading to the DoD program management communities. In reality the "source of truth" will be a distributed federation of data and models. The goal should be revised to "Create and govern a set of authoritative data and models in order to share knowledge and resources across the system lifecycle." These data and models might originate in any phase of a systems lifecycle and in any function associated with DoD engineering and acquisition. In fact this will always be the case. "Who owns the data and models" remains a pain point in this transformation.

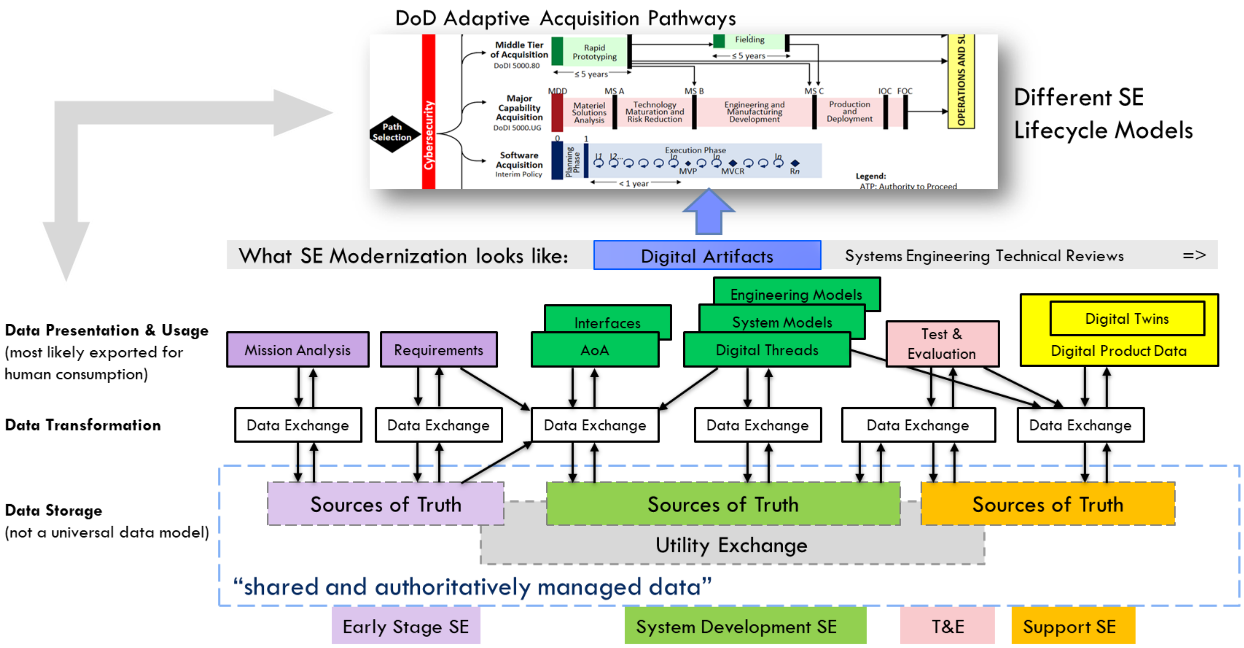


Figure 12. Data Transformation into the Life Cycle.

This figure is particularly relevant to SE modernization, as "Data Management" is not currently defined as a disciplinary process in SE standards or DoD engineering policy, and data is engineered with different processes and disciplines than other traditional SE related disciplines. However, data engineering and data modeling do follow SE processes.

This leads us to the "roots" of the **integration framework** for SE Modernization, which must address how shared and authoritatively managed data and models are defined, built, deployed, and used in DoD systems:

New SE lifecycle processes must evolve that address shared and authoritatively managed sets of digital data and models associated with the full lifecycle of the system itself, not just a single acquisition program lifecycle.

We found in our interviews and workshops on this project, the terms data, digital models, digital artifacts, digital threads, and virtual systems or "digital twins", all have different definitions and driving forces behind their lifecycles. They are not being viewed in an integrated set of lifecycle and process models. In this research we developed a more integrative view of an SE lifecycle model which we call "The Supra-System Model." This mental model was created to be a discussion tool to distinguish historical SE lifecycle and process models from the needs today for a modernized approach.

THE SUPRA-SYSTEM MODEL: EVOLVING THE SE MODERNIZATION FRAMEWORK

This discussion begins with background from an abbreviated literature review. Thullier and Wippler in their chapter "Finding the Right Problem" from the book Complex

Systems and Systems of Systems Engineering caution us to always consider three lifecycles associated with any system, each with interdependencies and relative positions in the evolution of a system (Thullier and Wippler 2013):

- "*the system lifecycle*: the "experiences" of the system itself;
- *the program lifecycle* of the system: the rhythm of the project during study, development, production, etc. of the system;
- *the engineering cycle*: the processes and activities involved in engineering the system."

Historical SE literature tends to portray these different lifecycles as simultaneous, and combined into a disciplinary framework known as the SE Lifecycle. This may have been appropriate when most SE activities were focused on large scale physical systems, but with wider application of SE they have become more distinct and separated in their purpose.

It is important to note that there are two established definitions of the term "Lifecycle" (Merriam-Webster):

1. "the series of stages in form and functional activity through which an organism passes between successive recurrences of a specified primary stage" (multi-generational)

2. "a series of stages through which something (such as an individual, culture, or manufactured product) passes during its lifetime." (single generational)

Systems Engineering and the "Systems Lifecyle" as defined by ISO/IEC/IEEE 15288:2015 and the Project Management Institutes (PMI) project lifecycle tend to follow the second definition. Design Engineering, Software Engineering, and Enterprise Engineering models tend to match the first definition better.

ISO/IEC/IEEE 15288:2105 also clearly defines itself as a set of process descriptions for describing the lifecycle of systems created by humans, from an engineering and organizational or project viewpoint. In other words, the Vee-Model representation of SE standards and DoD Acquisition guidance reflects a single pass through engineering and program lifecycle activities but the "experiences of the system itself" will progress through a number of such engineering and program lifecycles.

Thullier and Wippler note that in the lifecycle of the system itself the "experience of the system" must be evaluated in periods and across levels of temporal invariance. In their description the lifecycle a system progresses from idea to a virtual existence (models, documents, software, and today many digital artifacts) to a physical existence. SE technical and management process divides these into stages. SE processes recognize "within each level [of abstraction], we may distinguish periods of time which we may observe the integrity of the structure and behavior of the system [as invariant]" and use these periods to enable interdisciplinary and collaborative activities, referred to as phase gates or decision points. Virtual or intangible artifacts by their nature can cycle through more rapid periods of change than physical artifacts (Thullier and Wippler 2013).

Thullier and Wippler also note that program lifecycle phases “are aligned (or mixed in) with key steps (or stages) of the system lifecycle. This allows us to fix program phases on integrated, coherent, and stable states of the system in question, and thus to make important decisions at precise moments in the life of the system.” They further note that the engineering cycle is “the process that consists of moving from need…to an optimized solution – i.e. the best compromise integrating all constraints (cost/ time/performance) for the entirety of the phases and situations involved in the system lifecycle...This should not, however, be taken to mean that these processes must be carried out in a sequential manner” (Thullier and Wippler 2013). In other words, the idea that the system lifecycle, the program lifecycle, and engineering lifecycles can be combined together is a fallacy. There are “periods of temporal invariance” where we can view these lifecycles together in order to make important decisions, otherwise they should be considered as independent. **Trying to force them to remain in lockstep limits our ability to be iterative and responsive.**

Hossein et al. (2019) in “A Historical Perspective on Development of Systems Engineering Discipline: A Review and Analysis” categorize across a large body of literature from 1929 to 2018 four themes that define SE discipline:

1. SE as a management process,
2. SE as a technical process,
3. SE as an interdisciplinary and collaborative approach, and
4. SE as a holistic or systems-oriented (as opposed to general) problem solving approach.

We should not assume that each of these are the same thing, but that they may individually drive the disciplines that seek to engineer systems.

Systems engineering lifecycles and processes are not new, they have just evolved in different ways since first envisioned in the 1960’s. Stanley Shinners in the 1967 book “Techniques of Systems Engineering” first introduced the concept of SE as the methodological approach to define, realize, and deploy a system inherent in today’s SE lifecycle processes. Shinners defined the general techniques of SE that exist today: understand the problem, consider alternative solutions, choose the most optimum design, synthesize the system, test the system, compare test results with requirements and objectives, and update the system characteristics and data. This early process flow represents the basis for SE as the “technical and management driven systems oriented problem-solving process” that permeates much of the SE literature today (Shinners 1967). It also is the basis of software DevOps practice. This “systems engineering rigor” should not be changed but must be applied to all systems, both virtual and physical, in any program management lifecycle.

Arthur David Hall in A Methodology for Systems Engineering (1962) stated that SE must consider “For a given system, the environment is the set of all objects outside the system: (1) a change in whose attributes affect the system and (2) whose attributes are changed by the behavior of the system" Thus we cannot bound the system away from its external context but must consider the experience of the system to be affected both by the technical and management processes that evolve the system and the external situations that seek to adapt the system.

THE SUPRA-SYSTEM MODEL: THE SE MODERNIZATION MENTAL MODEL

Thus, there are four individual lifecycles that may affect the "experience of the system" and must be distinguished if we want an SE process model that reflects any acquisition pathway with the SE rigor we have been accustomed to in historical SE and acquisition lifecycle process models. One is the lifecycle of the system itself and potentially of the offspring it produces (both aspects of the lifecycle definition). Two others are the engineering and program or project lifecycles, which conduct processes internal to the life of the system. Finally, is what we call the "supra-system" lifecycle which reflects the direct experiences of the system itself in its operational context as related to the closest other systems it interacts with. A supra-system is defined is a larger system that integrates or contains other systems.

In addition to recognizing that each of the four lifecycle/process models may be individually relevant, the roots of our integration framework require that each of these lifecycle processes must evolve that address shared and authoritatively managed sets of digital data and models associated with the full lifecycle of the system itself, not just a single acquisition program lifecycle. Some of this data is contextual data in the supra-system. The established DoD views that combine management processes/lifecycle and technical processes/lifecycle do not fit well into the circular data-oriented mental model: technical (engineering) iterations and management (program) iterations have very different decision processes and respond to different types of data. Furthermore, SE as a holistic or systems-oriented problem solving approach that reflects both the system and the supra-system. These are visualized together in Figure 13.

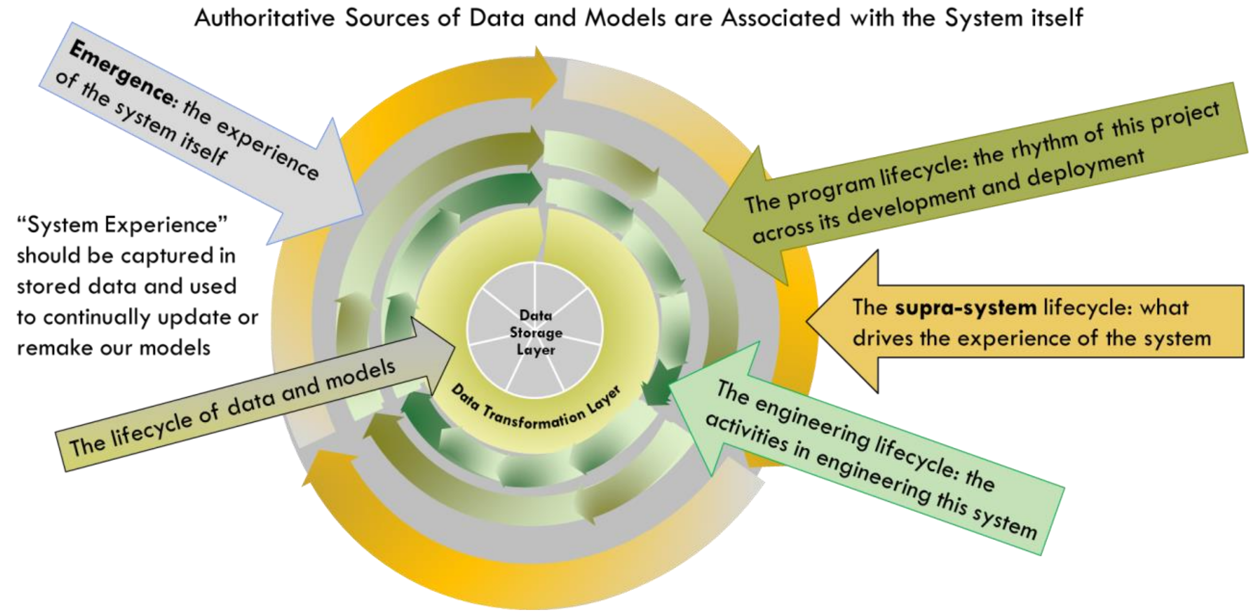


Figure 13. Multiple lifecycles of interest centered on data and models.

To reflect fully the model of Figure 13, the team shifted to developing a new conceptual view of the full SE Modernization Lifecycle, shown in Figure 14. This view is an attempt to capture everything associated with DoD engineering and acquisition in one mental model. It must be tailored and redrawn based on differing types of development, delivery, and support processes. This view is complex, but with study it becomes insightful in several ways. First, it illustrates systems engineering as a cyclic approach, rather than a linear one. Although almost all literature attempting to standardize on a lifecycle model will say that activities are ongoing and should continue through the lifecycle, the circular illustration drives this point home more visually and directly.

Second, this integration framework makes the digital transformation clear using a layered model with data storage and transformation at the core, models as the data transformation layer, and systems engineering process areas as the outer layers. Third, it organizes the colors of the outer ring and related SE process in the "Build/Measure/Learn" context, capturing the underlying goal of continuous iterative development. Finally, it recognizes that data and models may come from any experience of a system, including pre-Material Development Decision (MDD), post Operational Test and deployment and support.

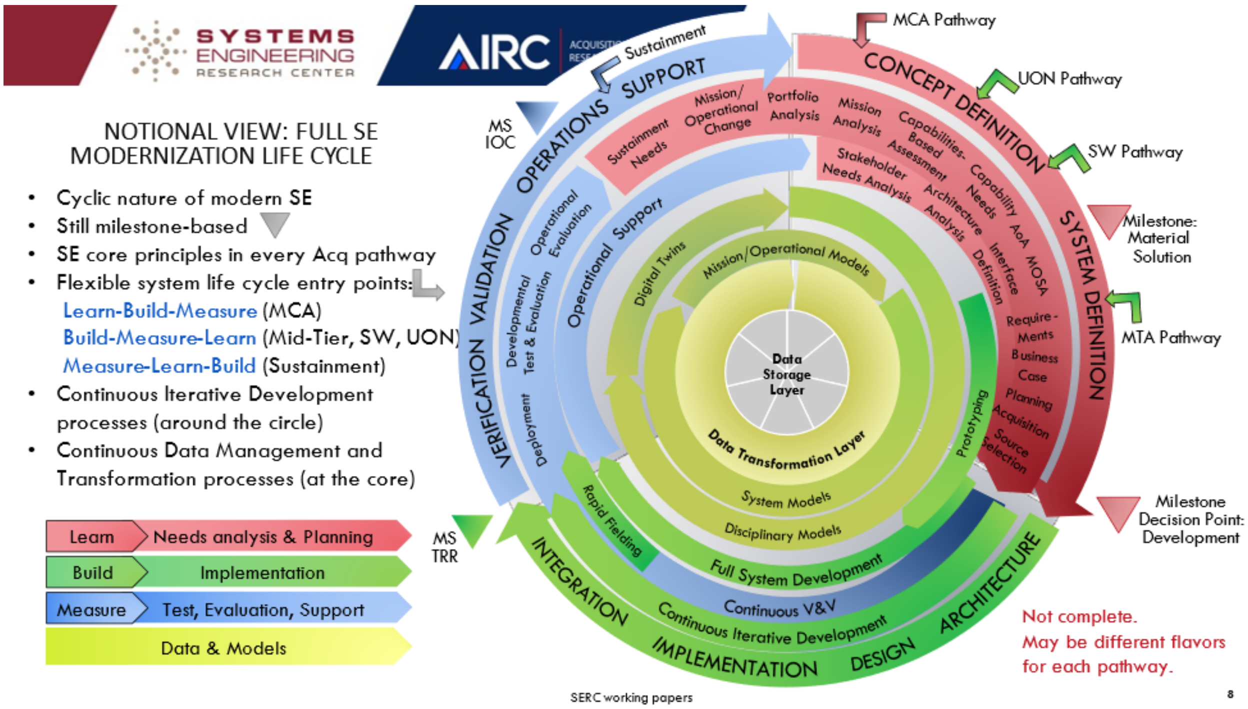


Figure 14. Full Integration Framework.

The integration framework depicted here incorporates traditional DoD acquisition milestones (triangles). However, it highlights them in the context of the multi-faceted work going on and where they fall within the broader context. It highlights the different DoD acquisition pathways and associate SE process instantiations. These fundamentally begin at different points in the system life cycle but should still follow a rigorous SE process model.

SUMMARY

This research found:

"The value of SE Modernization will be realized in more seamless and efficient transfer of data and models from underlying performance drivers through models to decisions, as well as ease of drilling back down from decisions to data."

And:

"New SE lifecycle processes must evolve that address shared and authoritatively managed sets of digital data and models associated with the full lifecycle of the system itself, not just a single acquisition program lifecycle."

In addition, newer systems engineering subdisciplines like software systems engineering, information technology and enterprise architecture, distributed modeling & simulation, and automated manufacturing systems "view lifecycle process and technical review as something that is much more iterative than what is implied by current SE guidance." This research found that the mission of SE Modernization, contrary to much of the published "future of SE" literature, should focus less on models and more on "increasing responsiveness," by promoting lifecycle processes that "increase the number of iterations and shorten the cycle time between them."

This research found that data management has become a critical systems engineering process area for today's systems and enterprises, and this needs to be added to SE lifecycle process standards and associated education and training.

Finally, the research found that current literature does not distinguish between system, program, engineering, and "supra-system" lifecycle processes and current acquisition guidance often "forces them to remain in lockstep which limits our ability to be iterative and responsive." Acquisition processes should provide more flexibility to the systems engineering community in how they use SE lifecycle processes, and acquisition pathways in the AAF should provide more guidance on the use of SE in each pathway.

References

DAU Acquisition Encyclopedia, https://www.dau.edu/acquipedia

DAU Systems Engineering Fundamentals, DAU Press, January 2001.

Office of the Deputy Director for Engineering, Office of the Under Secretary of Defense for Research and Engineering, Systems Engineering Guidebook, February 2011, https://ac.cto.mil/wp-content/uploads/2022/02/Systems-Eng-Guidebook_Feb2022-Cleared-slp.pdf

N. Geier and K. Alexander, "Systems Engineering Modernization Initiative," NDIA Systems and Mission Engineering Conference, November 2021.

A. D. Hall, A Methodology for Systems Engineering. Van Nostrand, 1962.

N. Hossain, R. Jaradat, M. Hamilton, C. Keating, and S. Goerger, "A Historical Perspective on Development of Systems Engineering Discipline: A Review and Analysis," Journal of Systems Science and Systems Engineering, December 2019, pp. 1–35.

ISO/IEC/IEEE 15288:2015, Systems and software engineering — System life cycle processes.

T. McDermott and W. Benjamin, SERC Final Technical Report SERC-2022-TR-009, Program Managers Guide to Digital and Agile Systems Engineering Process Transformation, September 2022.

Mil-Std-499 (USAF), 17 July 1969.

Mil-Std-499A (USAF), 1 May 1974.

P. Thullier and J. Wippler, “Finding the Right Problem,” in D. Luzeaux and J. Rualt, eds., Large Scale Complex Systems and Systems of Systems, ISTE, 2011.

Systems Engineering Body of Knowledge, www.sebokwiki.org.

PART 2: SEMOD PAIN POINTS AND ROADMAPS

This report presents a list of and discussion of the SE Modernization Pain Points and a set of short term and long term roadmaps for further development. The pain points were developed on SERC Project WRT-1051 and have been updated slightly based on a workshop completed February 28, 2023.

The integration of all of the research on this project was used to create a comprehensive set of roadmaps for future SE Modernization development. Per the sponsor request, these roadmaps were produced in the format of other SERC research roadmaps which can be accessed at https://sercuarc.org/research-roadmaps/.

SE MODERNIZATION – PAIN POINTS

Throughout the project, the team conducted outreach to government functional area leads, system program offices, science and technology organizations, professional societies, and other commercial entities who could discuss their SE modernization experiences. These discussions led to a comprehensive set of pain points that were developed to inform future SE modernization roadmaps. This section presents the final pain points analysis.

OUTREACH ACTIVITIES

The research team conducted four formal workshops with government, industry and academia to gain insights. The workshops included:

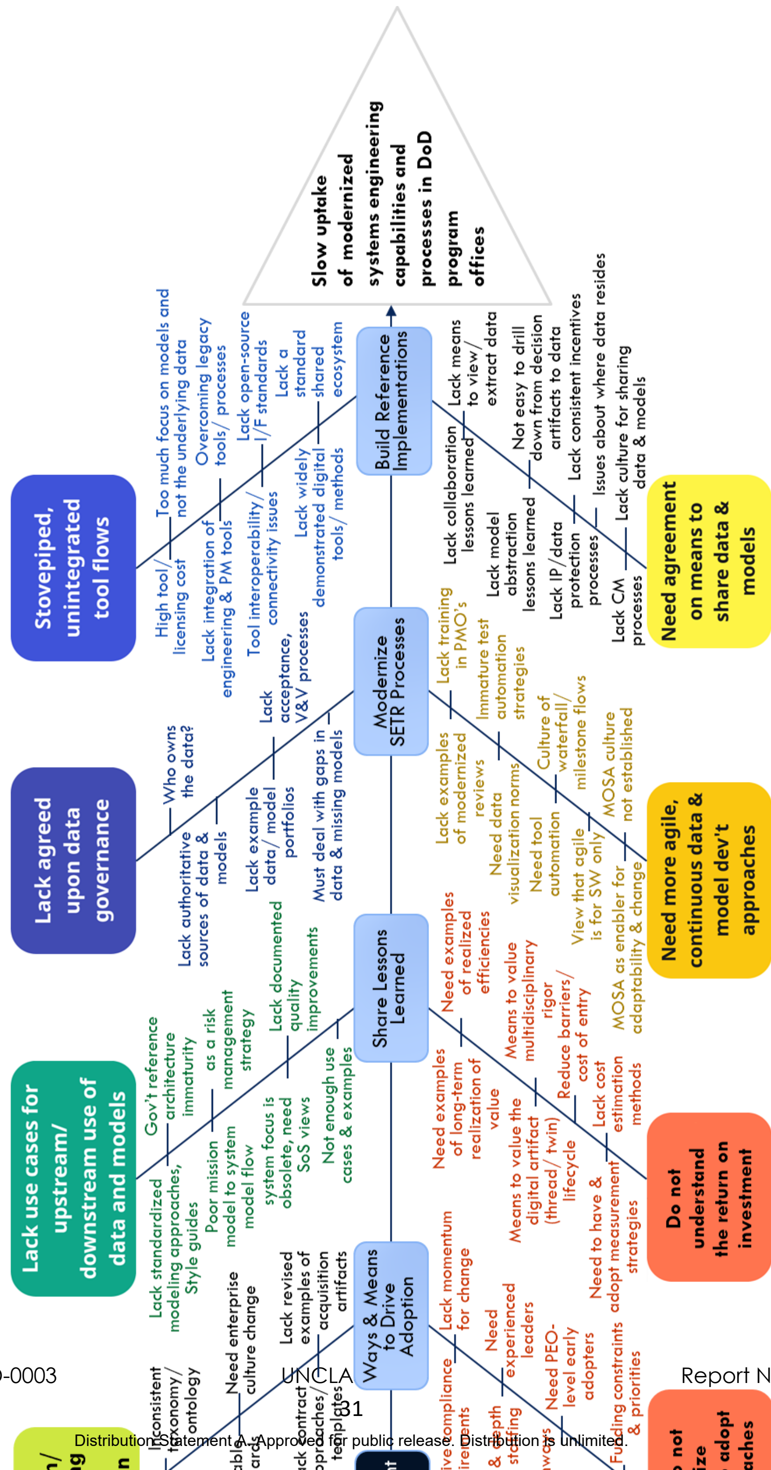
1. Translating Digital Engineering into Pragmatic Impact (November 2021)
2. SE Modernization Strategy (January and June 2021) – conducted jointly with the International Council on Systems Engineering.
3. Digital Artifact Workshop (February 2022) – conducted jointly with DAU.
4. SE Modernization Pain Points Workshop (February 2023) – conducted with government sponsors. This was a final update to review the pain points at the completion of the lessons learned analysis.

Across the project, the team also had a number of individual discussions with DoD functional area leads and system program offices in interviews that were led by the sponsor. These activities generated a number of statements that were used to inform a comprehensive set of SE Modernization pain points.

PAIN POINTS AGGREGATION

The pain points are organized into an Ishikawa (fishbone) diagram in order to provide a categorization framework and a rough ordering of pain points and needs. The full diagram is shown in Figure 15.

Figure 15. SE Mod Pain Points.



The detailed pain points in each causal path are not easily readable in the figure and will be explained further. The overall organization of the diagram represents as an input our SE Modernization value statement:

- **Seamless and efficient digital flows from data to decision artifacts and from decision artifacts back to data.**

And as an output the primary problem these pain points would address:

- **The slow implementation of modernized systems engineering processes in DoD Program Offices.**

The organization of the diagram represents four primary recommendation areas driven each by two primary aggregated pain points. These are simply categorized below:

- **Build Reference Implementations**
  - Stovepiped and unintegrated tool flows.
  - Need agreement on means to share models and data.

- **Modernize Systems Engineering Technical Review (SETR) Processes**
  - Lack agreed upon data governance.
  - Need more agile, continuous data and model development approaches.

- **Share Lessons Learned**
  - Lack use cases for upstream/downstream use of data and models.
  - Do not understand the return on investment.

- **Ways and Means to Drive Adoption (specific to DoD acquisition)**
  - Lack acquisition/engineering process integration.
  - Policies do not incentivize programs to adopt new approaches.

The detailed pain points for each recommendation area are summarized below. They have been updated and also include a set of stated stakeholder needs.

Build Reference Implementations

1. The DoD should build and share representative reference implementations that support seamless and efficient digital flows across engineering, program management, and acquisition processes. The tool, data, and model infrastructure will necessarily be tailored to the needs of the systems development. Digital tools and data/model infrastructure must exist in a combination of both acquirer and supplier tool and data/model infrastructures when contracting. Over time, standard use cases and method/tool patterns will emerge.
   a. Current tools and tool workflows are stovepiped and unintegrated. An outcome of digital systems engineering is to improve ability to collaborate across disciplines and disciplinary tools.

i. High tool/licensing cost, need enterprise level agreements and standards
ii. Today the focus is on modeling tools, need a much broader data management focus and set of processes
iii. Tools lack standard integration of engineering and program management data
iv. Tools need to support seamless and efficient ways to integrate and connect data & models
v. The community has been at this for a while, need efficient ways to transition from legacy tools/processes to the latest more capable tools
vi. Government acquirer and supplier tools and methods need to be built into a standard shared ecosystem across programs
vii. Developed and demonstrated approaches need to be widely shared across programs

Stakeholder needs:

viii. need resources: high labor and tool costs make this unaffordable for many programs
ix. need integration of engineering and program management related data
x. need better tool support to connect data and models
xi. need better ability to transition from legacy tools and models to newer tool options and versions
xii. need better acquirer/provider sharing processes in both directions

b. Need agreement across programs and across organizations on the means to share data and models and related SE practices

i. The community has not yet developed a culture for sharing data and models
ii. There is a lack of lessons learned and standard approaches to address where the data and models will reside in the digital infrastruture
iii. Effective configuration management processes need to be developed, along with intellectual property and data protection mechanisms
iv. Program managers as acquirers and their suppliers lack the incentives (voluntary or contracted) and means for sharing data & models
v. "Seamless and efficient" means ability to easily drill down from review artifacts to models to data, today's tools and methods lack the ability to easily view/extract data at different levels
vi. Need lessons learned and best practices on the appropriate fidelity of models for different decision processes
vii. Need lessons learned and best practices on how to collaborate around models and data

Stakeholder needs:

viii. need to build a culture of collaborating/sharing
ix. need effective configuration management processes for data and models
x. need to develop intellectual property management processes

xi. need to develop and apply contractual incentives to share data and models between acquirer and supplier
xii. need tool suites that provide means to view/extract data at different levels
xiii. need tools/guides that support model fidelity design for interoperability
xiv. need modeling style guidance standards/examples for different use cases
xv. need to standardize on acquirer/supplier data and model storage and access approaches

Modernize Systems Engineering Technical Review Processes

2. There is a lack of agreed upon governance for data and models across programs, organizations, disciplines, and lifecycle phases. These have traditionally been exchanged in static artifacts (many digitized) at phase completion points like technical reviews, configuration audits, and transition points between major activities. Future digital systems engineering strategies have reimagined these as living digital threads and digital twin that live alongside the realized and deployed physical systems across the full life of the virtual system. The processes to collaborate across disciplines and organizations fully within digital model-based environments are not yet mature.
    a. DoD needs to modernize their SE technical review (SETR) and collaboration processes to focus on use of data and models instead of static presentation artifacts
        i. Who owns the data? Need standard structural and process approaches
        ii. Programs lack existing authoritative sources of data & models to build from
        iii. Programs lack examples of data/model portfolios and experience in managing them
        iv. Programs lack mature processes and methods for accepting and validating data/models consistent with modern continuous development and integration methods
        v. Programs need ways to identify and manage what data/models are needed when, and experience/risk processes to manage the gaps in data and models
        vi. Programs lack updated approaches to contract for data and models in a way that encourages collaborative use

    Stakeholder needs:

        vii. need better structural and process approaches for government data/model ownership, including between government functions
        viii. need to develop and mature libraries of data and models
        ix. need good examples of data/model portfolios in program offices
        x. need processes for acceptance and validation of authoritative data and models
        xi. need better decision processes for establishing program data/model needs

    xii. need better processes to determine gaps and risks to define data/model requirements

b. PMs need to develop more agile and continuous data & model development processes
    i. A modular open systems approach (MOSA) is the enabler for both the data/model infrastructure and the product data lifecycle, this must be recognized as a necessary step to adaptability and change as built into the Engineering culture
    ii. The prevailing view of agile as a software development approach must be overcome, and used to change the prevailing view of development as a set of waterfall milestones
    iii. Both programs and tool infrastructures lack standards and norms for visualizing digital data and models in reviews
    iv. Programs lack examples of modernized technical and management reviews
    v. Program offices lack training on how to execute modernized SE processes
    vi. Efficiency will come from automation, need tool automation and especially model-based evaluation and test strategies

Stakeholder needs:

    vii. need modular open systems approaches for data/model infrastructures
    viii. need modular open systems approaches for tool infrastructures
    ix. need to broadly develop a culture for continuous iterative development
    x. need digital information exchange standards for tech/program reviews
    xi. need visualization standards for tech/program reviews
    xii. need better training on model development, model governance, and model review
    xiii. need to realize more automation from the tools
    xiv. need automated evaluation and test strategies and tools for models and simulations

Share Lessons Learned

3. The DoD needs to organize and share lessons learned across all components. There are still relatively few defense system program offices that are implementing digital systems engineering and there appears to be little reuse of approaches from program to program, service to service at an enterprise level. Industry enterprise level approaches are more mature but still remain unique to program. The details of these implementations and lessons learned from them are not being widely shared.
    a. lack use cases for upstream and downstream use of data and models
        i. System program offices lack standardized approaches in practice for defining and using models and related data to specify and manage their developed and acquired systems
        ii. These would standardize on government reference architectures for both SE infrastructure and portfolios of systems – there is a lack of mature examples

iii. Models and data should be viewed as a risk management strategy – need a documented process and a program management focus
iv. The integration of mission/SoS models and system models is immature, program offices need SoS level views as stand-alone system models cannot reflect changes in context/use over time
v. System program offices lack documented examples of SE Mod as a quality improvement process
vi. There are not enough use cases and examples of SE Mod benefits

Stakeholder needs:

vii. need standard digitalized versions of engineering and acquisition processes and methods
viii. need reference approaches for data/model standardization and sharing
ix. need Program Manager guidance for using models as a risk mitigation strategy
x. need integration examples that span mission, enterprise and system architectures
xi. need examples of digital and model-based SE in Quality Assurance processes
xii. need more use cases showing the benefits of these transformations

b. Do not yet understand the benefits of and return on investment for SE Modernization
   i. High cost of tools and adoption strategies are a barrier to entry for many program offices who are not given funding/schedule relief for this transformation
   ii. Programs need revised cost estimation models that reflect efficiency of SE modernization components
   iii. Programs need to have and to adopt measurement strategies and specifications for SE in general and modernized SE
   iv. Programs need a means to value the multidisciplinary rigor and integration that comes with SE Mod
   v. Programs need means to value the life cycle benefits and use of sustained digital artifacts
   vi. Programs need examples of program realized efficiencies, and need long-term examples of the realized value of SE modernization

Stakeholder needs:

vii. need dedicated resources to support the implementation of new methods and tools
viii. need schedule consideration and program planning that includes methods development and training
ix. need revised/more complete cost estimation tools that reflect data collection and models
x. need standardized measurement strategies/approaches for DE and SE
xi. need examples of program realized efficiencies
xii. need means to quantify the value of the interdisciplinary rigor gained from DE

xiii. need means to quantify the value to operational evaluation from using DE processes
xiv. need means to quantify the value to production from using DE processes
xv. need means to quantify the value to sustainment from using DE processes

Ways and Means to Drive Adoption

4. The DoD needs ways and means to drive adoption into Program Offices and other related government functions. This is a large transformation effort and current guidance and policy does not easily translate to government and acquisition functions. Government activities such as mission engineering, requirements development, science and technology, technology development and prototyping, test and evaluation, and operations and maintenance must all contribute to development, use, and sharing of data and models. Acquisition activities such as budgeting, contracting, data rights and intellectual property, information security, planning, and others must adapt, particularly to the collaborative workflows inherent to digital system engineering, and make use of the engineering information digitally available which impacts their activities.
   a. Lack acquisition/engineering process integration
      i. There is not an effective terminology that integrates across different acquisition pathways and different areas of policy and guidance, causing confusing and lack of focus
      ii. Digital transformation is an enterprise level cultural change and the top-down/bottoms-up learning needed is just underway
      iii. Most programs involve legacy systems and these program offices are unable to/unwilling to integrate new SE practices into legacy systems improvements
      iv. Standard contract approaches and templates for defining & procuring artifacts in the digital ecosystem are not yet available
      v. Need program office consumable visualization standards for dashboards that aid management
      vi. There are not enough examples of acquisition artifacts available from early adopters

   Stakeholder needs:

      vii. need standardized terminology/ontology across all acquisition and engineering functions
      viii. need organizational change management/cultural change cases
      ix. need examples of legacy system adoption of DE and MBSE
      x. need standard contracting approaches and templates for collaborating around models
      xi. need more examples of acquisition artifacts resulting from data and models
      xii. need program office standard progress visualization approaches to model based acquisition

b. Policies do not incentivize programs to adopt new approaches
    i. Current guidance is stovepiped and inconsistent across acquisition pathways and engineering/management processes, maturing slowly
    ii. The DoD lacks an enterprise strategy to fund DE infrastructure
    iii. Some programs are early adopters, but digital transformation is not yet at the portfolio level
    iv. The DoD needs experienced individuals who can lead adoption of modernized SE practices, as well as breadth and depth of staffing to implement those practices
    v. Effective compliance measures are needed to measure adoption and build momentum for change

Stakeholder needs:

vi. need enterprise-wide infrastructure funding approaches that improve affordability
vii. need portfolio approaches and examples of data/model sharing across programs
viii. need experienced adoption leaders to manage cultural change in program offices
ix. need breadth and depth of staffing, and specialized training linked to roles
x. need compliance measures & QA standards for shared data and models

These pain points are offered up as a list for further development. In next steps the government should take the initiative to agree upon and formalize each pain point (as was done with the Digital Engineering pain points) then develop plans and measurement approaches to track each item.

As part of this research, the pain points were used to inform our SE Modernization roadmaps. Each roadmap description in the next section lists the relevant pain points.

SE Modernization Research Roadmaps

The culmination of the integration framework, pain points, and other research led to a set of digital SE modernization roadmaps to inform future developmental activities in this area. These roadmaps are not detailed in time, but generally represent a 5-year timeframe of activities to advance the SE Modernization initiative. The full roadmap is shown in Figure 16.

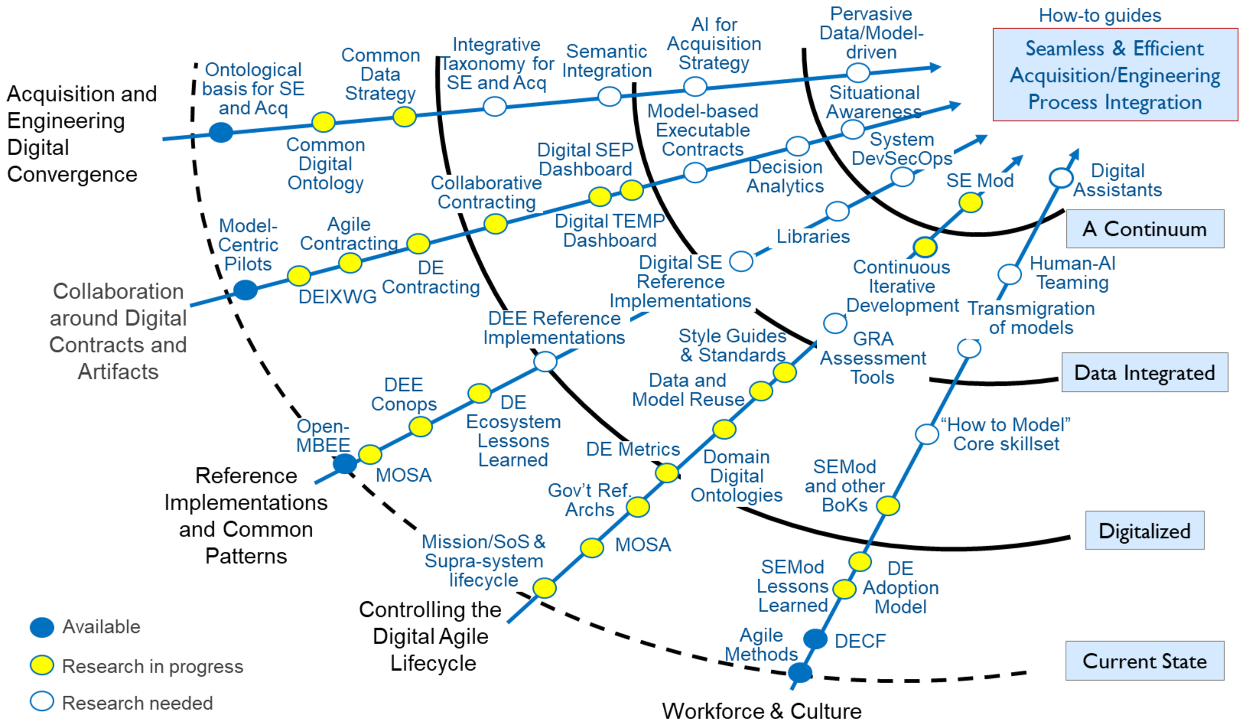


Figure 16. SE Modernization full roadmap.

The roadmap has a set of verticals (arrows) leading to a visionary outcome or set of outcomes, and each circle on these verticals represent a capability that is either currently existing (blue), in development now (yellow), or needs to be started (white). The color codes are our assessment of the current capability. The arcs in the diagram represent "capability frontiers" starting from the Current State, moving through a frontier where SE and Acquisition are fully Digitalized, then a frontier where and SE and Acquisition processes are fully Data-driven and integrated with underlying data artifacts, and finally to where SE and Acquisition look more like A Continuum of capability development and deployment activities instead of standalone programs with large time gaps. The convergence at the end of the vertical and frontiers is "Seamless and Efficient Acquisition/Engineering Process Integration" ideally represented in sets of policy issuances and related "How-to Guides" that aid program offices and other military acquisition/engineering functions in their transformation.

The following sections discuss each vertical in the roadmap with linkage to the SE Modernization pain points. Each vertical description includes the pain points, a bullet form description of each capability, and a discussion of next steps in terms of developmental activities.

Acquisition and Engineering Digital Convergence

Figure 17. Acquisition and Engineering Digital Convergense Roadmap.

This vertical responds in general to all of the pain points associated with the lack of seamless acquisition/engineering process integration, in particular the lack of an effective terminology that integrates across acquisition areas of change. In our research we found that the policy and guidance that links acquisition and engineering lacks clarity in language and there is no standard lexicon that defines this linkage across all of the pathways of the Adaptive Acquisition Framework (AAF). One can view the defense acquisition process in total as a command and control process, integrating across military needs and uses (doctrinal), acquisition practice, and engineering activities. There is currently no underlying ontology to drive data convergence, and no integrative taxonomy that spans military doctrine, acquisition, and engineering that can connect program data and methods.

The SE Modernization policy analysis found "*The common modernization driver in all of these (SE Mod) focus areas is seamless and efficient transfer of data and models from underlying performance drivers through models to decisions, as well as ease of drilling back down from decisions to data. This does not mean everything must be connected (that is unlikely to ever happen) but that the process to move up and down the data transformation space is efficient and produces better quality. With this mental model of improved access and flow, a common integration framework can be pursued. Without it, stove-piping of people, processes and tools across lifecycle stages will continue to occur. The purpose of SE Modernization is thus to support more seamless and efficient digital integration of data and models across all program management, engineering, and acquisition process areas as well as deployment and use of military systems. We found this intent to be generally lacking in the current policy and guidance*."

In addition, the policy analysis found "*there is an inconsistent level of descriptive detail across documents by focus area that creates confusion. There is also varying sets of terminology and jargon used in different policies and guides that makes integration difficult*." As a result, the analysis recommended an ontology effort being conducted to identify the more specific recommendations for language consistency across policy areas. This led to the ontology research on the project described separately.

The driver for this roadmap is to enable more use of machine learning and other artificial intelligence technologies to integrate data across engineering and acquisition courses of action defined by missions and capabilities. In the long-term engineering and acquisition execution processes should become fully paperless and data/model-driven to massively reduce cycle times and increase program success. The common technical basis for this convergence is the use of semantic web technologies and their underlying digital ontologies, which must be developed for both the engineering and acquisition domains (there is not a published domain ontology for either). This drives our recommendation of a set of capability development activities noted in the following list:

- **Ontological basis for SE and Acquisition**: Upper-level ontologies such as Basic Formal Ontology (BFO) provide a formal structure to integrate ontologies. Mid-level ontologies such as Common Core Ontology (CCO) and Navy Strategic Systems Program (SSP) integrate taxonomies of generic classes and relations across all domains of interest and supports domain level ontology development. Research on this project demonstrated that acquisition and engineering domain ontologies can be created and linked to these existing published ontologies. This research provides a completed starting point.
- **Common Digital Ontology**: A suite of digital engineering and acquisition domain ontologies needs to be developed and published that facilitate convergence of acquisition and engineering groups, and integration of life cycle activities (acquisition vs. operation and service).
- **Common Data Strategy**: Strategies for data governance, data engineering, and data analytics need to be defined to drive business intelligence and analysis for decision making in DoD engineering, operations, acquisition, and program management. An ongoing AIRC project has defined a draft Innovative Data-Enabled Acquisition Strategy (IDEAS) framework that promotes "*the use of quality pervasive digital information, models, data, and analysis to empower cultural changes and innovation by improving acquisition workforce decisions, policies, functions, and processes to produce better and more timely outcomes and value for the warfighter*." This work is linking digital acquisition and digital engineering and should be continued.
- **Integrative Taxonomy for SE and Acquisition**: The ontology efforts with respect to SE Modernization will provide a digital foundation to resolve a common taxonomy across systems engineering guidelines, acquisition related guidelines, program management guidelines and operational doctrine. This is necessary to digitally integrate all sources of knowledge for engineering and acquisition domains consistent with military doctrine.
- **Semantic Integration**: This is needed to transform sources of knowledge into knowledge representations that can be further used for domain level inferencing, comparison, and gap analysis across DoD operational, acquisition, and engineering domains, and to use these sources to design Courses of Action (CoAs) for engineering and acquisition execution. CoA's are the core of military operational planning, and should become the core of acquisition strategy. For example, in the commercial domain AI-based CoA tools are now emerging in business planning activities such as customer relationship management, procurement management, and supply chain management.
- **AI for Acquisition Strategy**: In the longer-term machine learning and agent-based modeling approaches can be employed to produce and wargame predictive CoA strategies for agile acquisition. The DoD should explore research in this area.
- **Pervasive Data- and Model-driven**: In the fulfillment of this roadmap. engineering and acquisition execution processes should become fully paperless and data/model-driven to maximize efficiency and flow, massively reduce cycle times, and increase program success.

Defense acquisition and engineering today come together in a common framework under the DoD 5000 series policies and guides for material development and acquisition. However, the disciplines are not well integrated. Development of an underlying digital data imperative to link these disciplines together through common data analysis and visualization tools and digital course of action guides will provide a foundation for convergence. The combination of SERC and AIRC research is already working in this direction, and foundational research on ontologies and taxonomies, data strategies, and analytical tools should be continued. Research that recasts and links these activities using operational command and control should be explored.

COLLABORATION AROUND DIGITAL CONTRACTS

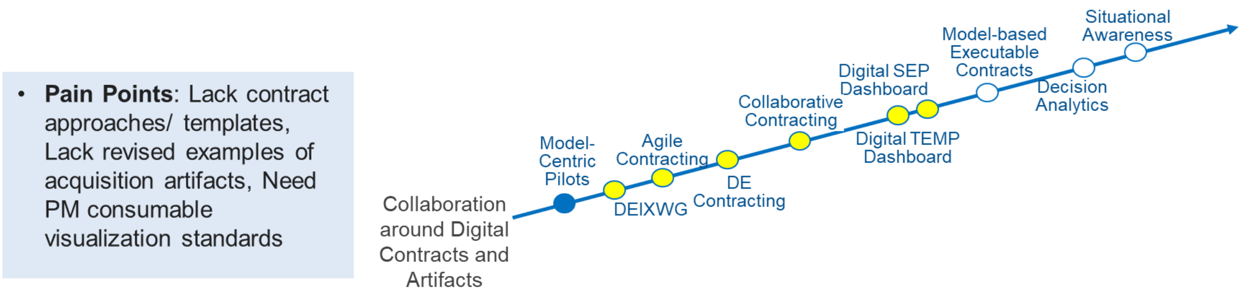


Figure 18. Collaboration Around Digital Contracts Roadmap.

This vertical also responds to all of the pain points associated with the lack of seamless acquisition/engineering process integration, in particular the lack of digital contract approaches/ templates, lack of revised examples of acquisition artifacts, and the need for better program manager consumable visualization standards. Responding to recommendations in the phase 1 research task WRT-1051 report, AIRC was separately funded to pursue research on Contracting for Digital Engineering. This effort is looking at DE contracting workflows and how government/government and government/contractor collaborate around models in program workflows. Digital model-based process that produce static document-based artifacts outside of the digital workflow create inefficiency and waste. Long-term SERC "Model-Centric Engineering" research with NAVAIR and Space Command addressed the question "can we do everything in the model" and integrated not just system modeling but also contracting, workflow, reviews, and approvals into the DE tool suite. Initial SERC/AIRC research efforts are envisioning conversion of Systems Engineering Plans (SEP) and Test and Evaluation Master Plans (TEMP) to digital artifact driven formats as these are the top-level SE required plans. (Refer to research tasks WRT-1043 and WRT-1071). These future tools should be considered as model-based interactive dashboards with near real-time program status, not digital static artifacts. In the longer term the DoD should strive for fully digital automated contracting, analytical tools, and program visualization capabilities that encourage collaboration around data and models.

The driver for this roadmap is to enable more use of machine learning, software orchestration platforms, advanced data analytics, and new visualization tools to gain full situational awareness across engineering and acquisition courses of action. In the long-term, engineering and acquisition execution processes should move away from static artifacts and highly aggregated milestones toward near-real time dashboards that apply advanced data analytics and continuous situational awareness of program operations. This also requires convergence around the use of semantic web technologies and digital ontologies, but this vertical is oriented around advancements in digital workflows and data analytics. This drives our recommendation of a set of capability development activities noted in the following list:

- **Model-Centric Pilots**: The SERC has completed several demonstration pilot programs exploring the art of the possible to achieve a representative set of SE and Acquisition combined activities 100% "in the model." These so far have been oriented toward systems engineering technical processes but could be extended to additional acquisition activities as defined in a full contracting activity.
- **Digital Engineering Information Exchange Working Group (DEIXWG)**: This is a DoD sponsored community activity with INCOSE to develop a set of "common views" for executing digital, model-based engineering and technical reviews. This also demonstrates the art of the possible in the ability to standardize key program activities "in the models." The DoD must continue to develop and promote these pilot activities, but must close the loop around lessons learned and implementation guidance to create more standardization across programs.
- **Agile, DE, and Collaborative Contracting**: There is a need for more flexible workflow-based contracting approaches for collaboration around data and models. Existing efforts to define and standardize contracting language are still following the current standards of static artifacts exchanged at major contract milestones, and only adding data and models to the static deliverables lists. Deliverables and contract points need to be driven by the flow of the engineering lifecycle, not the program lifecycle, although these should be linked. The DE Contracting research task will provide initial recommendations for these three areas, but further work will need to be done to develop and standardize DE contracting in line with agile workflows and collaborative processes.
- **SEP Dashboard**: All acquisition pathways and programs of any size should maintain and follow a Systems Engineering Plan (SEP) that defines and controls the engineering and management lifecycle activities. This remains an SE best practice independent of approval authority. This research envisions a digital version of the SEP that provides an interactive dashboard for a program office to plan, monitor, and control the program development process, built from modern data analytic and planning tools.
- **TEMP Dashboard**: All acquisition pathways and programs of any size should maintain a Test & Evaluation Master Plan (TEMP) that defines and controls the joint government/ contractor responsibilities and planning for test, verification, and validation. This research envisions a digital version of the TEMP that provides an interactive dashboard for a program office to plan, monitor, and control the systems integration, developmental test, and operational evaluation processes, built from modern data analytic and planning tools.
- **Model-Based Executable Contracts**: There is a need to bridge the gap between current legal language and digital data exchange using declarative (outcome-based)

transaction models, and software orchestration (dynamic workflows for multiple task automation). Model-based executable contracts are software programs that are stored and executed using blockchains to manage the transactions. Software orchestration automates the configuration and management of these programs. Outcome-based transaction models could move the completion of a static milestone and deliverable to a linkage between a performance milestone and evidence in a digital model (for example "100% design release" is satisfied when the product line management tool plan versus release metric reaches 100% as seen in the SEP dashboard). Commercial best practices to automate these contractable transaction points are evolving. Research in this area could better link engineering, program lifecycle, and technical management progress in automated, digital, machine learnable processes. Why is this important? Ideally, we could better plan and contract for the evolution of the system, not just the engineering and program management tasks.

- **Program Decision Analytic Tools**: Common digital ontologies and data strategies will enable development of new digital decision analysis tools using emerging artificial intelligence and visualization technologies to improve acquisition decision making. Linking deliverable progress to discrete tasks in an integrated master schedule to track actual system development progress is extremely inefficient. In modern software/DevOps environments all the software development and program metrics are integrated and tracked within a single tool suite. In other words management metrics are directed linked to the software code. Interdisciplinary digital systems models can ideally extend this concept to full program level tracking.
- **Program Situational Awareness**: The long-term goal is digitally connected visualization dashboards that achieve full near real time situational awareness and measures of performance across all engineering, technical, and management activities, even in extremely large projects.

REFERENCE IMPLEMENTATIONS AND COMMON PATTERNS

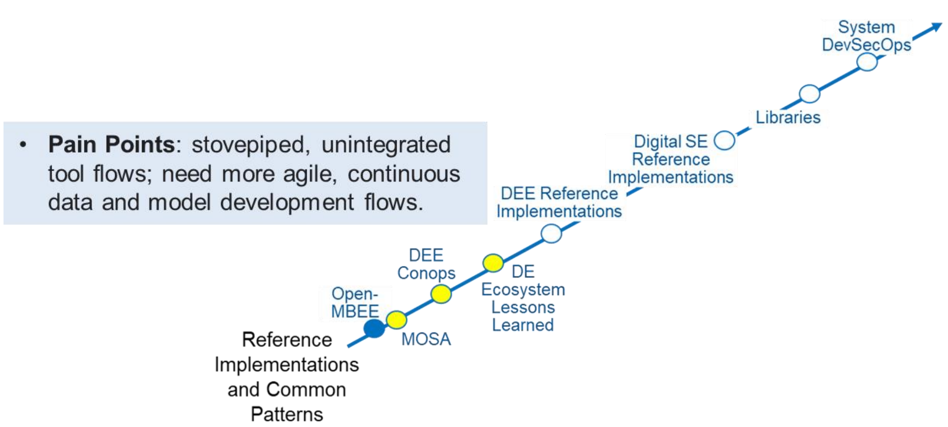


Figure 19. Reference Implementations and Common Patterns Roadmap.

This vertical responds in general to all of the pain points associated with current stovepiped and unintegrated tool flows and the need for more agile and continuous data and model development approaches. The Jet Propulsion Lab's Open-source Model-Based Engineering Environment (OpenMBEE) was the first toolset to support management of all data and all systems and disciplinary models in a systems modeling framework. The SERC Model-Centric Engineering research demonstrated that model management and visualization tools like Open-MBEE could be used to orchestrate both engineering and program workflows completely in the digital toolset. Today, the exemplar tool patterns used by most DoD programs that are supporting data and model-driven acquisition remain fragmented and non-standardized. This leads a program office to develop around the simplest tool infrastructures and defeats the core value of SE Modernization: seamless and efficient digital flows.

Modernized program offices need seamless and efficient digital engineering and acquisition ecosystems – information technology systems that support digital development workflows and products. While digital program management ecosystems can likely be standardized at the enterprise level, the digital engineering ecosystems will naturally vary by the types of systems being developed, produced, and maintained, and the disciplines needed in the process (a software only ecosystem will be simpler than a manufactured physical product ecosystem; a prototype might use a simpler ecosystem than a full weapon system lifecycle management ecosystem). This drives our recommendation of a set of capability development activities noted in the following list:

- **MOSA:** The government has mandated a modular open systems approach (MOSA) for its weapon systems. MOSA principles must also set the core business and technical approach for DE Ecosystem (DEE) architectures.
- **DE Ecosystem Conops:** As a next step, OUSD(R&E) should develop agreed upon concepts of operations and use cases for creating program/enterprise DE ecosystems and development of a joint federated model for procurement & assistance in development.
- **DE Ecosystem Lessons Learned:** The community needs an organized and categorized Body of Knowledge collecting lessons learned from government and industry on DE ecosystem patterns.; as well as an continual effort to translate lessons learned into action.
- **DE Reference Implementations:** As a further step, OUSD(R&E) should evaluate and promote community endorsed patterns and reference implementations for program/enterprise DE ecosystems based on differing system/SoS types. The DoD has done similar work in their software factory environments such as the Air Force Cloud One software hosting environment and toolset options. This can be structurally reused and extended to all engineering and management tools, but development still would need to address interoperable tool pipelines, not just tool hosting.
- **Digital SE Reference Implementations:** Longer-term we need community endorsed patterns for data/model development applications and associated procedural modeling techniques that determine how we model things. Systems modeling tools today all differ in the procedural methodologies they encode in their tools, as well as their underlying meta-models. Different enterprises are developing different modeling style guides, and these are incompatible. This may be primarily a

sharing of lessons learned and artifacts, but the DoD should encourage and perhaps host this sharing. In the longer term standardized training related to not just tools but also modeling methods is needed.

- **Libraries:** The tools and models should evolve over time where system lower levels of abstraction become standard libraries and designing a system model becomes more pattern-based. In the Software, Microelectronics Design, and Computer Aided Design communities library reuse has become normal. At the systems level much more research and development is needed.
- **System DevSecOps:** Ultimately this vertical should support a DevSecOps model for system capability deployment, where DE ecosystem and acquisition program management tooling supports continuous integration and deployment of warfighter capabilities from any program and any organization.

Controlling the Digital Agile Lifecycle

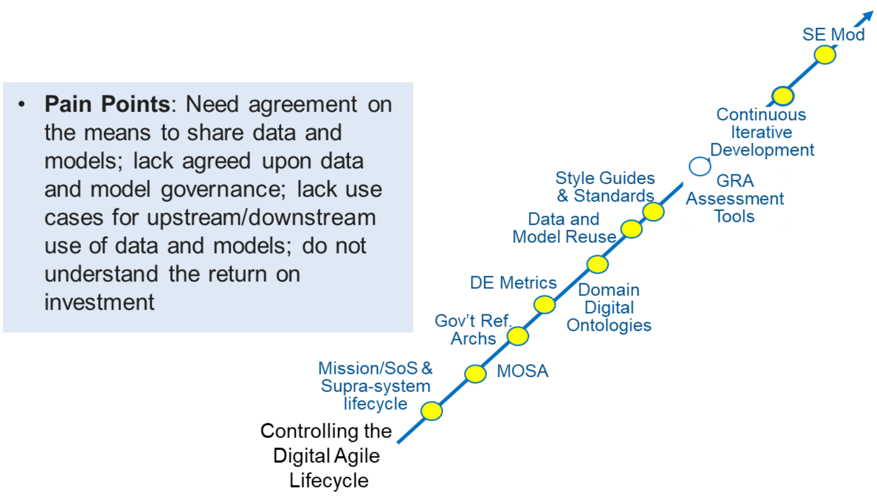


Figure 20. Controlling the Digital Agile Lifecycle Roadmap.

This vertical responds to many of the pain points in the build reference implementations, modernize SETR processes, and share lessons learned goals of SE Modernization. The government needs to develop or acquire and maintain the digital artifacts it needs to control the whole digital and agile lifecycles of its systems. This is the full supra-system lifecycle model that extends beyond any individual program or systems engineering lifecycle. We also list MOSA in this vertical because it provides the Title 10 acquisition authority for the government to retain and manage and make available all the long-term business and technical aspects of their mission, enterprise, and system architectures. A central concept of this vertical is architecture. Architectures are models and the evolution of digital systems engineering and architecting will provide greater fidelity of this control as the tools and methods evolve over time. Government reference architectures[4] (not just requirements) should become the acquisition baseline in each acquisition pathway, allowing the government more ability to manage adaptation and change as threats change and technology evolves.

This vertical is the core of SE Modernization. In the long-term, SE Modernization promotes more seamless and efficient system iteration through data and model reuse and continuous iterative deployment practices. This drives our recommendation of a set of capability development activities noted in the following list:

- **The Supra-System Model**: this thread begins with the re-envisioned acquisition/SE lifecycle model discussed separately in the research. The government needs to view SE as the integration framework for all technical and management lifecycles and processes, not just a material development. This includes mission architectures, system of system (SoS) architectures, and system architectures to the level that the government needs to control the full system "experience" across its lifecycle.
- **MOSA**: MOSA is the government's business and technical approach to manage adaptability and affordability of defense systems over time, managed at the portfolio and architecture level. It must not be viewed as just an interface control and intellectual property management tool, but as technical and management process to define and control government developed versus contracted aspects of a full system architecture over time. MOSA is a mechanism to control and manage an architecture, the focus should be on the methods, processes, tools, and skills associated with architectural design.
- **Government Reference Architectures**: GRA's are government developed, owned, and maintained authoritative sources of data and models that guide system design, development, production, and sustainment in an acquisition program. These exist at enterprise or portfolio levels. Figure 21 provides a good overview of the supra-system lifecycle activities associated with a GRA (Martin 2022). It should be noted that continuing to segregate policy and training under ME, MOSA, DE, and modeling and simulation functional areas bypasses the opportunity to grow system architecture functions, roles, and skills in the DoD. The DoD must invest in both the digital methods and related skills associated with reference architecture development.

[4] *Reference Architecture: an authoritative source of data and models that guides and constrains the instantiations of multiple architectures and solutions. The goal is to provide templates for solutions in a particular domain that stress commonality.

*Government Reference Architectures: guide system design, development, production, and sustainment in an acquisition program. A GRA should exist at the mission level and system of systems/family of systems level but can provide standardized approaches at any level of a system.

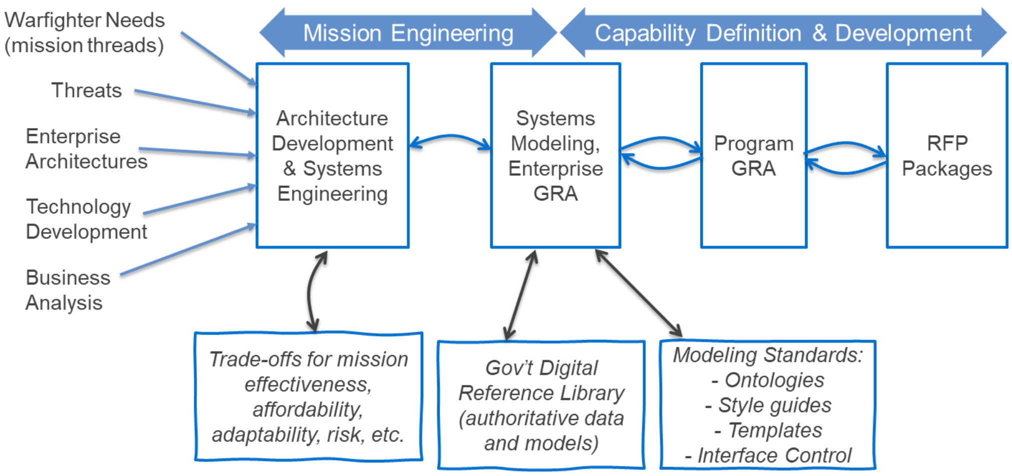


Figure 21. Government Reference Architecture in the Acquisition Process (adapted from Martin).

- **Digital Engineering Metrics**: Digital artifacts provide a more direct path to measuring and improving efficiency and quality of the defense systems development process to improve system deployment, cost, and schedule outcomes, as they allow us to directly measure the software artifacts that make up the digital thread and digital twins. Previous SERC research supported development and publication of the initial DE Measurement Framework (PSM 2022). Programs need to begin collecting data to inform longer-term enterprise measures and DE measurement metrics and reports need to become formal program office requirements. Also, efforts that continue to update this framework and collect lessons learned should be continued.
- **Domain Digital Ontologies**: A GRA will be expressed in a reference library which will become the digital graph of domain-specific models and relationships between entities in a mission, SoS, or system. With document-based systems engineering and acquisition exact language based relationships are not important, in digital modeling these domain level ontologies have become critical. Domain digital ontologies and their development will be necessary for constructing the data models that underly authoritative sources. Programs and portfolios must maintain these, and must train and employ people to manage them.
- **Data and Model Reuse**: A longer term outcome development of government maintained and provided libraries of mission/SoS/system data, models, and reference architecture templates will be reuse. Reuse will reduce program to program data ambiguity, improve learning, and increase overall speed of acquisition. Federations of reusable software models and other components is both a cultural change and a research program that needs its own well-funded set of programs.
- **Style Guides and Standards for Systems Models**: Systems modeling tools and the SysML language are relatively new compared to other software languages. There is a huge need to develop and share guides that produce consistency in system modeling methods and design as well as tools to improve interoperability and reuse across programs, portfolios and services. With SE Modernization community can wait for this to happen or we can put in place programs to encourage standardization.

- **GRA Assessment Tool**: What data is needed to say a GRA is acceptable, what are the criteria that the data and models need to meet? There are many standard reference models but little characterization of reference model standards. A specific research project should be conducted to advance the foundations, discipline, and practice of reference modeling.
- **Continuous Iterative Development**: CID is both an architecting and development process approach to manage risk by separately architecting platforms and capabilities and more frequently deploying and validating capabilities. It is now the core of software/DevOps approaches but very different from the traditional MDAP related DoD SE approaches. The DoD needs to continue to define, promote, develop, and support CID across both architecture and more continuous development processes.
- **Systems Engineering Modernization (SEMod)**: The outcome of this project in the long-term: evolution of SE lifecycle processes and digital tools to improve the efficiency and quality of defense systems development.

WORKFORCE AND CULTURE

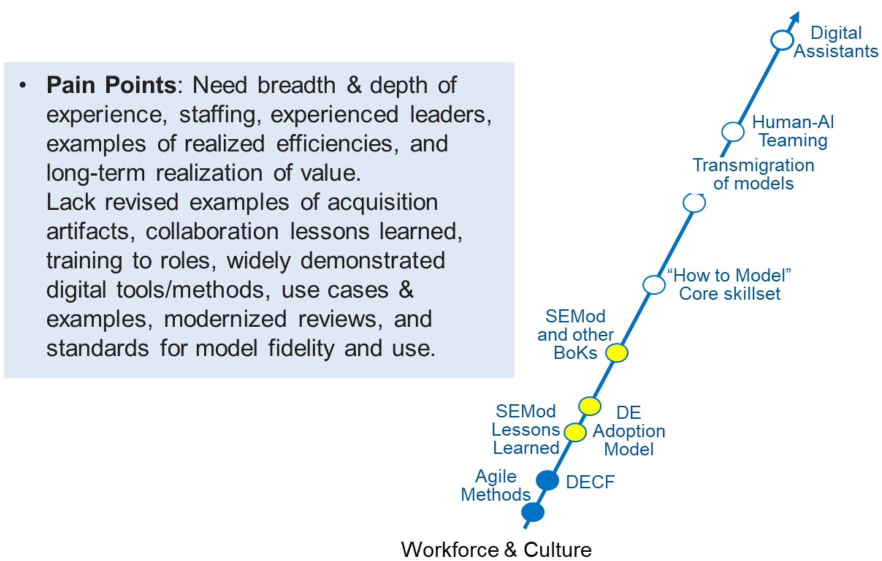


Figure 22. Workforce and Culture Roadmap.

Pain points related to workforce development and cultural adoption of modernized SE appeared in every "rib" of our pain points fishbone diagram. In this project we were teams with the DAU on their SE Modernization project, which consists of a set of training materials and topical webinars. Overall, these materials still largely separate content for each focus area, and need to be more integrative over time.

There are several unifying themes from the SE Modernization that need to drive workforce development activities in the future. These include moving all the software components of SE (data, models, software products) to established team and enterprise agile methods and tools, fully developing the digital competencies of an SE, and making the foundations of modeling and simulation essential skills of any government technical and management professional. In addition the capture of evolutionary lessons learned and a related body of knowledge as defined at the start of this initiative should continue. The final theme is automation, as efficiency will come from not just connectivity but from task automation. This drives our recommendation of a set of capability development activities noted in the following list:

- **Agile Methods**: DoD should promote integration and adoption of agile methods across all engineering and acquisition activities, not just software. There are enough published examples of agile transformation in hardware-intensive industries (Tesla, Saab) to build from, and many defense contractors have already made the transition. Any government processes that disrupt “flow” (in agile terms) should be retired or at most remain with rationale.
- **Digital Engineering Competency Framework (DECF)**: The SERC completed an extension of SE competencies into the digital engineering realm with the DECF. DAU is using the framework to guide their digital engineering courseware in development. Training by doing (within the DE tool ecosystem) is necessary to experience the benefits and create the culture. The SE community needs to address these competencies generally in all education and training to continue to grow the workforce.
- **Digital Engineering Adoption Model**: The SE Modernization project produced an initial framework to organize the benefits, enablers, and change management strategies and lessons learned for DE adoption. The DoD should develop a more quantitative approach to their digital and model-based systems engineering maturity assessments using this initial framework and adjust it over time. The DoD should further how-to guidance and sharing of lessons learned to programs on their adoption journeys.
- **SEMod Lessons Learned**: The SE Modernization project curated an initial set of key lessons learned from government and industry reflecting their digital transformation journeys across engineering, program and technical management, and acquisition, as well as a searchable framework for capturing these. Continued sharing of lessons learned is critical to government/contractor SE modernization.
- **SEModBOK**: This project prototyped a body of knowledge with the goal to reflect the integration of both fundamental knowledge and how-to guidance for doing modernized digital engineering and acquisition practice. The prototype SEModBOK contained only high level guidance, based on this research we now have additional artifacts and lessons learned. This reference library will continue to be useful as the initiative proceeds.
- **“How to Model” Core Skillset**: An aspirational goal in this roadmap is that everyone in acquisition and engineering gains core skillsets to represent data and solve problems in digital models. In the adoption research we identified three separate roles each of need training in modeling fundamentals and practice: reviewers (all who need to know how to use models to make decisions); developers (people building and maintaining the models); and architects (senior engineers assisting with the content of the models). The foundational concepts underlying development and use of models are well understood and can be trained to anyone. The DoD should provide the focus on this training area. Additionally, material should

be prepared and used to help leadership and decision makers transition from a pre-AAF method of acquisition, to a digitalized method, with automation. Help the leadership be comfortable with the change.

- **Transmigration of models** (transmigration: from one state of existence to another): This is an appropriate term to define a state where the people in any one acquisition role are comfortable using models from any other role (to some level of abstraction). This implies both advances in educational methods that train "how to model" and in modeling tools that can adjust views between abstraction levels based on reviewer needs. Three specific needs from the pain points analysis are visualization standards for tech/program reviews, tool suites that provide means to view/extract data at different levels, and model fidelity and modeling style guidance for different use cases.
- **Human-AI Teaming**: The companion SERC roadmap on AI and Autonomy identifies a number of detailed research and technical areas for future SE linked to Human-AI Teaming and Digital Cognitive Assistants for augmenting engineering and acquisition tasks that include information intensive activities or have inefficient workflows. Workforce development in this area should also concentrate on design of appropriate tasking and human-computer interface strategies for digital assistants in partnership with humans. The output of research in this area would be **Digital Cognitive Assistants**: AI assistants that help to identify areas to focus on, data analysis results, etc. SE Modernization workforce programs should not be just training humans, but also analyzing opportunities to improve workforce productivity, create communication improvements, redefine training needs, etc. to take advantage of the digital transformation.

RECOMMENDATIONS

At the convergence of the roadmaps is "How-to Guides for Seamless and Efficient Acquisition/Engineering Process Integration." The DoD must create a more agile and responsive acquisition system. To achieve this it must move with the rest of the business world to digitally transform itself and eliminate unnecessary and wasteful manual tasks. The challenge for the Office of the Undersecretary of Defense is to lead this transformation – the transformation will not succeed if every service and program office is left to independently undergo its own transformation.

REFERENCES

S. Martin, Introduction to Government Reference Architecture, https://www.youtube.com/watch?v=Zxpzl5F42fo.

T. McDermott and W. Benjamin, SERC Final Technical Report SERC-2022-TR-009, Program Managers Guide to Digital and Agile Systems Engineering Process Transformation, September 2022. https://sercuarc.org/serc-programs-projects/project/123.

Practical Software and System Measurement (PSM), Digital Engineering Measurement Framework version 1.1, 2022, https://psmsc.com/DEMeasurement.asp.

SERC Research Roadmaps, https://sercuarc.org/research-roadmaps/.

PART 3: REFERENCE IMPLEMENTATIONS AND COMMON PATTERNS

This section is updated from the WRT-1051 report section titled Recommended Reference Implementation. The collected pain points and roadmap continue to reflect a set of needs and recommendations to improve the “barriers to entry” facing government and industry SE Modernization around their digital data management and tool infrastructures. Collected stakeholder needs include:

- need to build a culture of collaborating/sharing
- need effective configuration management processes for data and models
- need to develop intellectual property management processes
- need to develop and apply contractual incentives to share data and models between acquirer and supplier
- need tool suites that provide means to view/extract data at different levels
- need tools/guides that support model fidelity design for interoperability
- need modeling style guidance standards/examples for different use cases
- need to standardize on acquirer/supplier data and model storage and access approaches
- need resources: high labor and tool costs make this unaffordable for many programs
- need integration of engineering and program management related data
- need better tool support to connect data and models
- need better ability to transition from legacy tools and models to newer tool options and versions
- need better acquirer/provider sharing processes in both directions

SE modernization will depend heavily on establishing managed authoritative high-confidence data sources, typically known as authoritative sources of system data and models, and the means to have it used throughout the enterprise. Collaboration within and between enterprises requires establishing high assurance interfaces between multiple applications in a digital engineering ecosystem. These interfaces need to use known, standards-based data exchange mechanisms, not peer-to-peer proprietary vendor interfaces.

EXEMPLAR REFERENCE IMPLEMENTATIONS INITIATIVE

In the WRT-1051 report we recommended establishing a set of exemplar reference implementations (ERIs) to inform development of DE ecosystems. The goal of these ERIs will be to mature data standards, establish data exchange methodologies between applications, and baseline the needed interface capabilities. Note that the goal is not to build a master DE ecosystem for any program to use, we do not believe a single implementation will be either feasible of cost effective. The ERI initiative will demonstrate the digital capabilities and technology to transition to service program offices as adaptable technology supporting and formalizing the development and integration of models for enterprise and program decision making. The ERI initiative will focus on capturing and retaining digital engineering artifacts using shared semantic data models (ontologies) between applications to support data exchange, product control, operational configuration, and traditional document production. The ERI should be a family of architectures that implement modernized, digital engineering and acquisition practices and policies consistent with the differing acquisition pathways, providing a consistent, coherent, and controlled environment that is standards-based, scalable, and federated.

Figure 23 provides an OV-1 view of the proposed ERI. Note that the primary focus of this diagram is the data exchange mechanisms between the different functional areas, processes, and disciplines in a typical large program, each of which will still likely be operating with their own models.

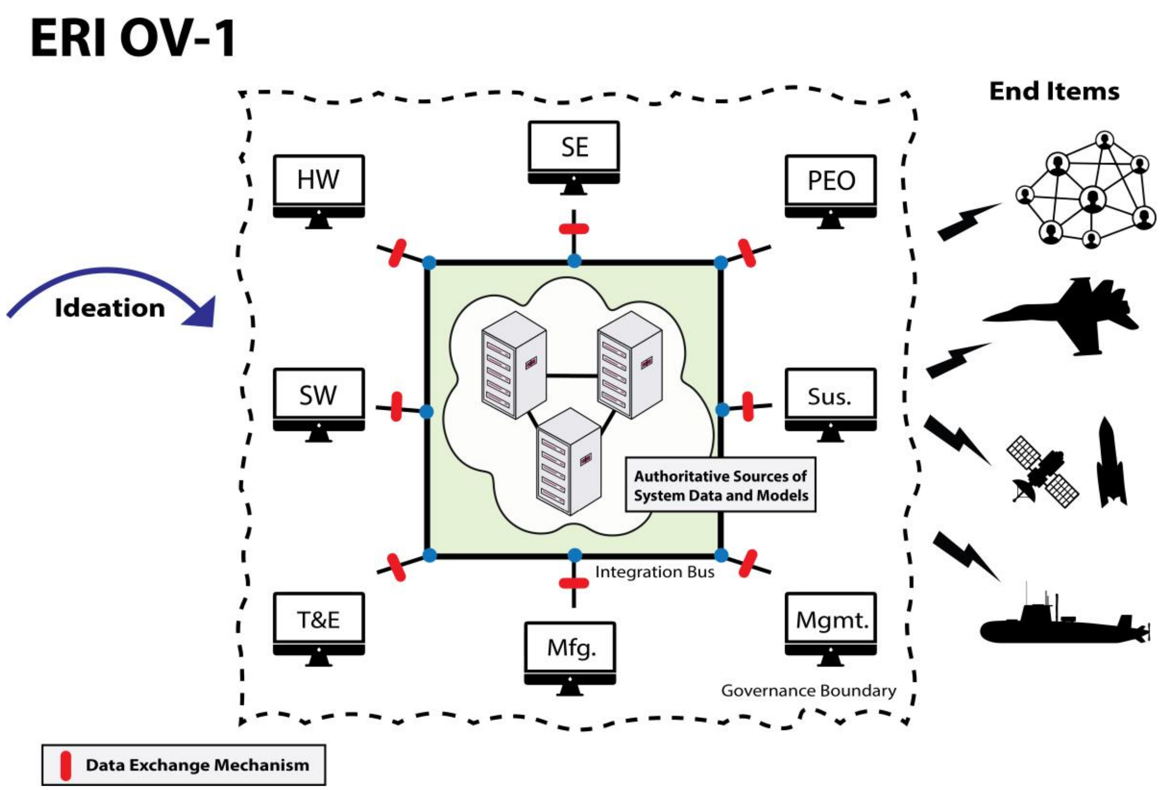


Figure 23. ERI OV-1.

Figure 24 from the WRT-1051 report gives an idea of the number and types of tools that might exist in an ecosystem, and the major SE activities that they support. This figure is extended from one presented by MITRE at NDIA [3]. This is not an exhaustive list—as other applications could be included—but does illustrate each relationship with the authoritative sources of system data and models. In studying the figure, one can also easily see how the types of tools and major SE activities may vary between types of systems – not every program will need either the same environment and it would not be cost effective to require all of these tools. The ERI must focus on the maturing the relationships between these tools and activities, not a universal ecosystem. These relationships will be the data exchanges between the applications and the authoritative sources of system data and models.

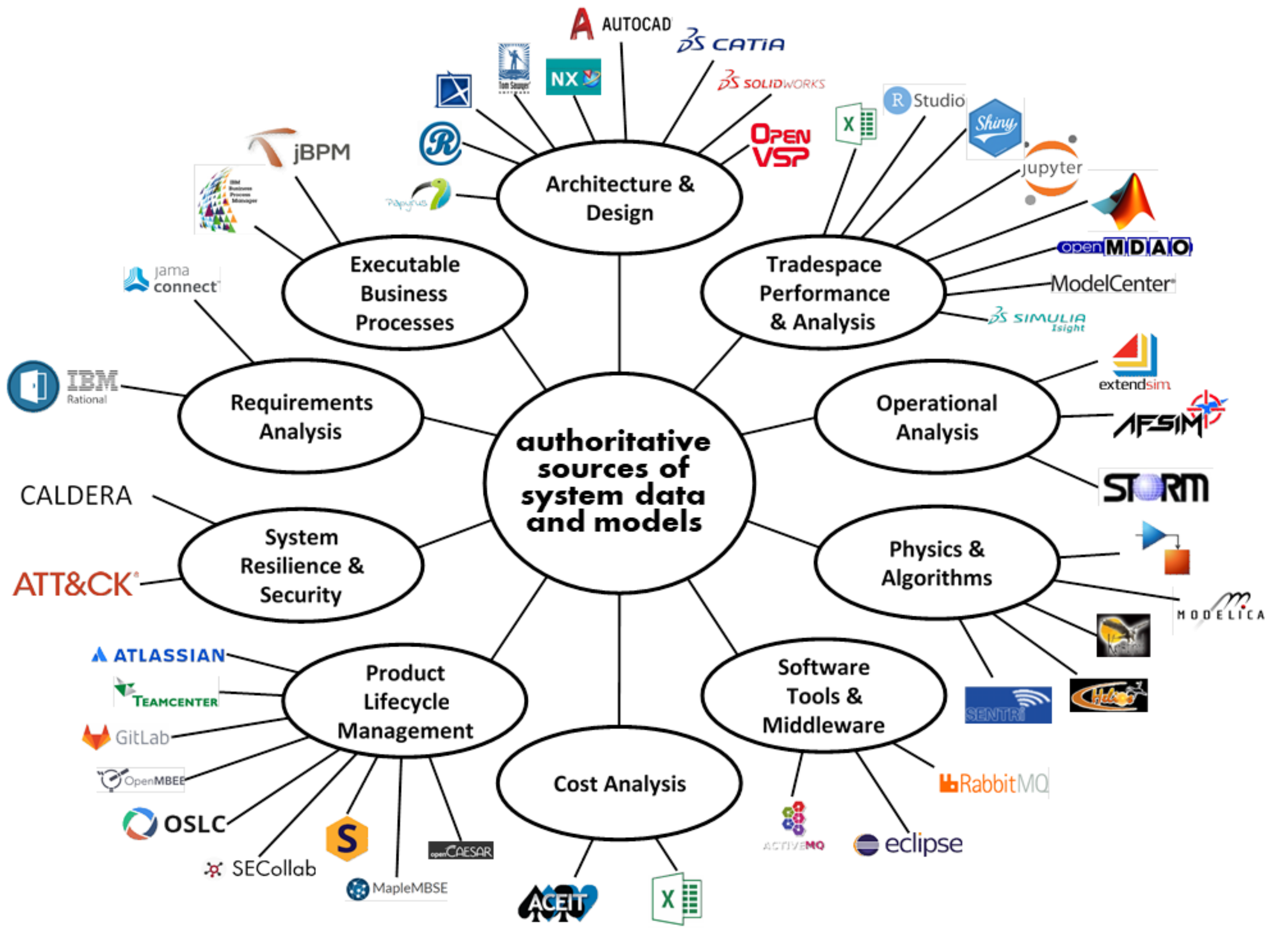


Figure 24. Digital Engineering Toolchain Components.

Applications within the engineering and acquisition lifecycles would provide to list a few: modeling, simulation, analysis, configuration and change management, version control, data/model conversion, and data presentation. Collectively, these applications exist today within engineering tool ecosystems but lack the data exchange interoperability to improve the decision-making speed and accuracy. Interoperability requires an infrastructure for exchanging data between applications with the application's unique data storage needs. Stakeholders need applications that share information or work interactively with other applications.

The situation is less of a barrier in disciplinary areas like mechanical and electrical engineering and software engineering. Currently in MBSE available applications were independently developed to be standalone or work with a vendor's application suite but not with other vendors. In the current application market, vendors have been motivated to provide interfaces that benefit the vendor's application (i.e., increase that vendor's market share). Some efforts have been nominally successful in creating a vendor consortium for interfaces[5], but the entry into the consortium is through vendor contributions. Applications used in systems engineering today lack a MOSA strategy for data exchange between vendor applications, and, in fact, there are no significant ongoing efforts to address this. The ERI initiative will help to achieve the MOSA attributes and provide solutions to meet data exchange challenges. Establishing links between DE and MBSE applications is crucial, creating a digital engineering ecosystem that transforms digital artifacts and provides data exchange mechanisms flowing these artifacts from one application to another.

The vision for the ERI initiative was described in the WRT-1051 report was described as a service oriented architecture to provide flexibility and adaptability between government and industry tool suites. Data exchange between applications will technically be a set of collaborative REST[6] APIs for query and response. While this is the open system method, it is insufficient to create the necessary digital threads for data flow between functional organizations and applications and mature the modeling methodologies necessary for effective data and model sharing. The goal of the ERI initiative will focus on determining and demonstrating data relationships in a flow of digital artifacts. The ERI will be a testbed for development. It would have an orchestrator for transforming data by creating queries requesting the correct data from the application using the REST API and ensuring that the resulting data flow to a receiving application in the form and rate needed. An ERI will use a publish-subscribe (i.e., Pub-Sub) messaging pattern. Ideally, the ERI Pub-Sub server is hosted as a service accessible to all applications on the network (i.e., cloud-based). An ERI will have a data modification language to perform the required data transformations from a parent data source to one or more child receivers. There are simple, known data transformations that can be immediately used, but no "out of the box" set of data transformations directly implementing complex workflow data transformations for the DoD infrastructure and environments for engineering, acquisition, test, logistics, and financial activities. The ERI creates the required functionally correct digital threads of digital artifacts, with data exchanges performed on data sets under configuration management for collaboration and communication across stakeholders.

The development of the reference implementations and common patterns over time would follow the vertical of the SE Modernization roadmap, shown in Figure 25. We reference OpenMBEE[7] as a starting place because this tool recognized that data and model management were the underlying foundations of any digital ecosystem. MOSA is noted next because DoD MOSA regulations must also apply to the tool ecosystems used to develop and maintain the systems if the DoD wants their systems to be adaptable and cost effective over the full system life. As a next step down this roadmap we propose a follow-on project focused on determining a set of concepts of operations (CONOPS) and

[5] Open Services for Lifecycle Collaboration, OSLC, https://open-services.net/

[6] Representational state transfer (REST) is a software architectural style describing a uniform interface between decoupled components in an Internet Client-Server architecture. REST defines four interface constraints: a)Identification of resources, b)Manipulation of resources, c)Self-descriptive messages, and d) hypermedia as the engine of application state (Roy T. Fielding from his dissertation)

[7] OpenMBEE (Open Model Based Engineering Environment), https://www.openmbee.org/.

use cases that reflect classes or patterns of ERIs for different types of systems development and support

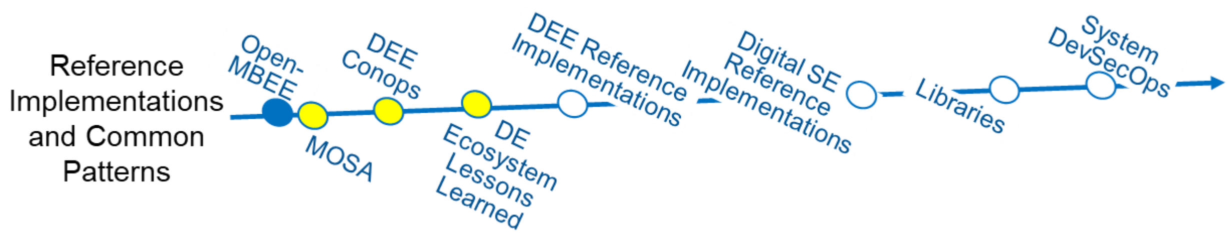


Figure 25. Reference Implementations and Common Patterns Roadmap.

CONCEPT OF OPERATIONS FOR DIGITAL ENGINEERING ECOSYSTEMS: EXEMPLAR REFERENCE IMPLEMENTATIONS

DE can reduce costs, speed development, and improve the quality of acquired systems. However, evolving standards, proprietary vendor systems, and a lack of experience implementing DE on DoD acquisition programs has impeded its transition. DoD Program Offices lack well-defined reference implementations that enable them to easily select and tailor methods and tools for their DE infrastructures. Approaches must be federated across the DoD enterprise as different types of programs and systems will require different types of reference implementations.

The ERI initiative as a whole will demonstrate DE capabilities and technologies across the multiple sets of tools and methods, aligned to different DoD systems and portfolios. The ERIs will capture and retain DE artifacts using shared semantic data models (digital ontologies) between applications using data exchange, product control, operational configuration, and traditional document production. The ERIs will implement digital engineering practices and policies, providing a consistent, coherent, and controlled environment that is context-independent, scalable, and federated.

This project is established to define the initial Concepts of Operations and Use Cases for a federated set of ERIs most relevant to DoD acquisition and engineering, including Major Capability Acquisition programs, Middle-Tier Acquisition programs, Software Acquisition programs, and related hybrid combinations.

Research Tasks:

Task 1: Stakeholder Engagement

- Engage critical stakeholders from the Military Service leadership program offices, OUSD(R&E), OUSD(A&S), and their relevant system engineering and acquisition offices to understand their priorities and expectations with regard to a DE ERIs;

- Engage critical stakeholders from the Defense Industrial Base (DIB) – prime contractors and others in the supply chain;

- Engage the critical tool vendors active in various aspects of the DE environment;

- Develop an initial report documenting classes of stakeholder and business concerns that would define initiation of an ERI in an enterprise or program office.

Task 2: DE CONOPS

- Develop a synthesis of existing DE ERIs within Programs and Industry;

- Develop a Concept of Operations, use cases, and a set of initial reference patterns for evolving DE ERIs. This document should capture the functional capabilities of different types of ERIs, the services they would support, a current gap analysis based on the synthesis of existing ERIs, and the use cases that would define selection of one ERI versus another. The CONOPS shall address Major Capability Acquisition programs, Middle-Tier Acquisition programs, Software Acquisition programs, and related hybrid combinations.

- Use these CONOPS to recommend a strategy for deliberate development

PART 4: CONTROLLING THE DIGITAL AGILE LIFECYCLE: MOSA AND GOVERNMENT REFERENCE ARCHITECTURE

A primary goal of the DoD's application of SE Modernization focus areas is to gain more control over the full lifecycle of DoD systems – the "experience of the system" as noted in the Supra-System lifecycle model. As is well documented, the defense acquisition system uses three processes to manage their systems: the Planning, Programing, Budget and Execution (**PPBE**) provides the resources, the Joint Capabilities Integration and Development System (**JCIDS**) defines or "learns" what needs to be developed and/or acquired and approves the programs, then the **Acquisition Process** manages the programs that result. In the SEMOD roadmap we noted a roadmap vertical defined as "**controlling the digital agile lifecycle**" noting that across all three processes the government needs to develop or acquire and maintain the digital artifacts it needs to control the whole digital and agile lifecycles of its systems. At the center of this vertical is the Government Reference Architecture (GRA). GRA's are government developed, owned, and maintained authoritative sources of data and models that guide system design, development, production, and sustainment in an acquisition program.

GRA, MOSA, AND THE SUPRA-SYSTEM LIFECYCLE MODEL

Architecture: the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.

Reference Architecture: an authoritative source of data and models that guides and constrains the instantiations of multiple architectures and solutions. The goal is to provide templates for solutions in a particular domain that stress commonality.

Government Reference Architectures: a set of authoritative data, models, and existing solutions that guide system design, development, production, and sustainment in an acquisition program. A GRA should exist at the mission level and system of systems/family of systems level but can provide standardized approaches at any level of a system.

Architectural Description: a collection of artifacts used to document an existing or proposed architecture.

Figure 26 shows a conceptual set of relationships between a GRA, MOSA, and the Supra-system model. Although the DoD MOSA guidance is clearly defined, the activities that a program undertakes to actual conduct and measure MOSA remain somewhat undefined. Also, MOSA guidance has evolved over time and is no longer clear on the relationship between system architecture as a disciplinary area of SE (shown in gold in the figure) and MOSA (shown in gray in the figure) as an approach to satisfy some (but not all) of the architectural qualities. This section of the report provides an overview, a set of research conducted by the UAH team on the state of MOSA goals, measures and processes, and a recommendation for a research vision to enable digital agile lifecycle through reimagining of architecting.

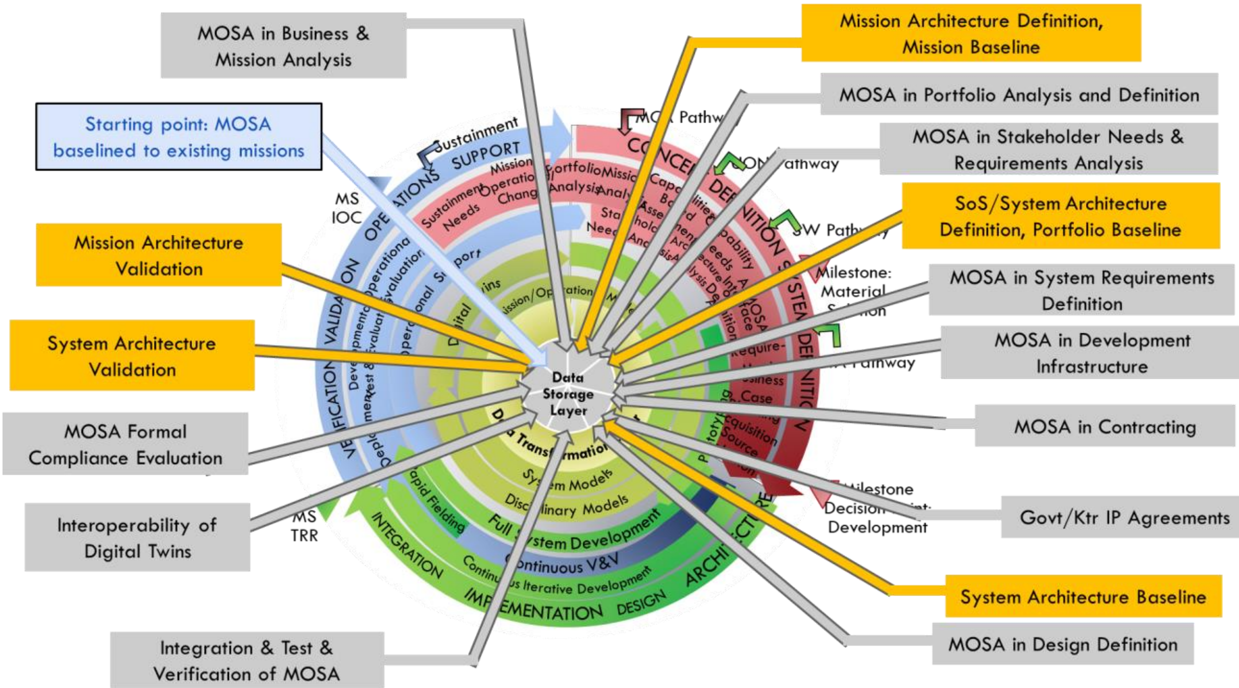


Figure 26. GRA and MOSA activities arrayed around the Supra-system Lifecycle Model.

Collected stakeholder needs include:

- need effective configuration management processes for data and models
- need to develop intellectual property management processes
- need better structural and process approaches for government data/model ownership, including between government functions
- need processes for acceptance and validation of authoritative data and models
- need better decision processes for establishing program data/model needs
- need tools/guides that support model fidelity design for interoperability
- need modular open systems approaches for data/model infrastructures
- need modular open systems approaches for tool infrastructures
- need automated evaluation and test strategies and tools for models and simulations
- need standard contracting approaches and templates for collaborating around models

GRA, MOSA, AND MISSION ENGINEERING

The answer to these needs cannot be easily associated with a single engineering or program lifecycle, as these are affected also by changes in the Supra-system Lifecycle both in the reason for a change to a system, and the larger ecosystem supports open standards and published interfaces. The National Defense Industries Association in 2022 published “Modular Open Systems Approach: Considerations Impacting Both Acquirer and Supplier Adoption.” This guide states:

“MOSA is highly dependent on the SoS’s taxonomy level of the system or component under consideration. The nature of the problem that MOSA is attempting to help solve will vary depending on whether the perspective is at the Joint Force level (where ships and Divisions are modules), service-unique perspective, platform perspective (major components or subsystems) or system level perspective (addressing hardware and software modules).”

These statements imply success of MOSA in DoD engineering and acquisition affects the full life of the system and is a product of all of the supra-system and engineering/program lifecycle activities.

As noted in the model, development of a GRA and associated MOSA approaches must necessarily begin with the previously fielded systems relevant to a service or joint mission. Use of existing interfaces and standards are either a conformance or a divergence point with existing systems in a mission or SoS. These decisions around whether or not to conform or diverge are necessarily associated with the supra-system lifecycle, not a program lifecycle. Mission Engineering is the stage of any systems lifecycle that is most likely to set architectural and associated MOSA requirements. The DoD Mission Engineering Guide (2020) defines as major products from ME analyses the following:

1. Documented results in the form of analytical reports, curated data, and models for continued reuse and further analysis;

2. Visualizations and briefings to inform leadership on key decisions; and

3. Government Reference Architectures (GRAs) (in the form of diagramed depictions of missions and interactions among elements associated with missions and capabilities)

The ME Guide also notes that “*Digital engineering principles should be used when conducting ME as they can help promote consistency in the ME process through the effective use and reuse of curated data and models along with identification and utilization of digital tools throughout ME analyses. Digital engineering is an essential foundational element of ME that allows for sustainment of mission threads (MTs) and architectures, integrated analytical capabilities, common mission representations, and an extensible set of tools*.” In the current time, ME and DE integration is not a mature process and the definition of a GRA as “a set of authoritative data, models, and existing solutions that guide system design, development, production, and sustainment in an acquisition program” is still an area for development. This current shortfall is at the heart of the SE Modernization roadmap of Figure 27.

Figure 27. Controlling the Digital Agile Lifecycle Roadmap.

One of the research areas of study conducted on this project sought to better understand the activities that were conducted across the MOSA guidance using literature review and analysis. There is a lack of literature on MOSA activities other than a recent NDIA initiative to define MOSA activities and metrics. This may be partly related to the focus of MOSA as an acquisition program activity instead of a way to manage the full life of the system. This research is presented in the next section.

MODULAR OPEN SYSTEMS APPROACH (MOSA) GOALS, MEASURES, AND PROCESSES: A LITERATURE ANALYSIS (UAH)

MOSA has been interpreted in multiple ways by different organizations. Research is needed to identify areas of convergence and divergence in the community as it relates to MOSA. This section addresses six research questions relating to the goals, measures, relationships between measures, processes, relationships between processes, and input and output data used in MOSA implementations. Literature related to MOSA was reviewed, response categories were developed for each research question, and the frequencies of each response category were examined. Some common response categories were identified, but consensus among the community for any research question remained elusive. Further work is proposed to compare results to practice in engineering organizations and to identify where further development of MOSA concepts may be most needed.

MOTIVATION AND RESEARCH QUESTIONS

When a new approach enters SE practice, its precise meaning, and from there its precise goals, methods, and data flows, can vary from organization to organization. In cases where the use of an approach is mandated by the DoD, it is of utmost importance to understand what consensus, if any, exists.

In the case of MOSA, there exist some examples of authoritative sources such as the Systems Engineering Guidebook ("Systems Engineering Guidebook" 2022) and various other DoD-sponsored publications ("Program Managers Guide to Open Systems" 2004; "The DoD Open Systems Architecture Contract Guidebook for Program Managers V1.1" 2013; "Modular Open Systems Approach (MOSA) Reference Frameworks in

Defense Acquisition Programs" 2020). This report uses sources of academic literature on the topic, including these three sources, in an attempt to answer the following questions:

RQ1: What are the goals of MOSA?
RQ2: What are the measures of MOSA?
RQ3: How do the measures of MOSA relate to one another?
RQ4: Are there specific processes for MOSA?
RQ5: How do MOSA processes relate to one another?
RQ6: What are the necessary input and output data for the processes?

The goal of this report is to help program managers, engineers, and SE practitioners in general to better understand the varying interpretations of MOSA that have emerged and work towards a more consistent and nuanced MOSA practice.

BACKGROUND

The MOSA has become prevalent in defense and industry acquisition projects. MOSA was codified in the 2017 National Defense Authorization Act, which requires MOSA implementations meet certain requirements to obtain Milestone B approval (SE Guidebook 2022). The move towards MOSA has been motivated by the DoD's desire to accomplish:

- Increased interoperability, including SoS interoperability and missing integration;
- Enhanced competition;
- Facilitation of technology refresh and evolutionary upgrades;
- Increased innovation;
- Potential cost savings or cost avoidance;
- Reduced time to field capability to the warfighter.

This report contributes to the SE body of knowledge along three avenues. The first is by providing a methodology for answering research questions like the ones posed in above. Other works have performed reviews of common MOSA measurements and processes (Geier 2022a; 2022b) but have not provided a repeatable and rigorous methodology for their work. The second avenue is that this report examines not only the goals, measures, processes, and data that are referenced in the literature, but how frequently they are referenced. The frequencies of different MOSA concepts give a more detailed picture of the MOSA concepts seeing the most use or the most neglect. Finally, while it is not addressed directly in this report, the data collected could be used to examine trends in the use of different MOSA concepts based on when concepts appear, become more heavily cited, or disappear from the literature. Hence, the database that is formed in this research will be useful for additional research questions.

METHODOLOGY

SOURCE IDENTIFICATION

The sources for this review were predominantly found through Google Scholar. Table 1 lists the keywords that were used in different combinations to identify sources for review. The phrase “modular open systems” was used with “MOSA” since “MOSA” on its own can be found in irrelevant papers (as a surname, for instance). The terms “architecture” and “approach” were not included since, with “MOSA” as part of the search, their inclusion was expected to generate more confusion in the results than clarity.

Table 1. Summary of Google Scholar search keywords.

Modular	Open	Systems
MOSA	“Systems Engineering”	“Modular open systems”

The Google Scholar results were then examined for academic sources such as journal papers and dissertations. Sources such as conference presentations were omitted due to their relative reduced details. Papers were specifically sought from the National Defense Industrial Association (NDIA) due to their recent MOSA centric initiatives. NDIA was used as a search term once a strong core of sources were gathered.

Each source had to at least mention MOSA, and appear to answer at least one research question. In addition, sources not identified through Google Scholar searches that were highly cited in the SE community, like the DoD’s System Engineering Guidebook (2022), were also added for review.

RESEARCH QUESTION RESPONSES

Once the sources were identified, they were distributed to the 4-person research team for review. The distribution was such that two team members reviewed each source for how the source addressed any of the six research questions. The team members then put their independent findings for each research question (in the form of direct quotes or summaries for long passages or ideas that appeared in multiple passages) into a shared database. The independent findings were then compared in a team meeting, with the consensus findings being recorded in a combined sheet. Some papers with only passing mentions of MOSA were reviewed by the first author alone.

REFINEMENT OF KEY TERMS

RQ1 - Goals and benefits

Only a small number of sources explicitly discuss the “goals” of MOSA. It is much more common for sources to mention MOSA benefits, and so for the purposes of this paper, they were considered synonymous.

RQ2 - Measures and goals

Measures were defined as tools to quantify the quality of a MOSA implementation. Goals were defined as the main objectives to achieve by using MOSA. A source could reference both measures and goals at the same time. For example, cost reduction might be one of the goals of MOSA and cost could be a measure.

RQ4 - Principles, tasks, and processes

Principles, tasks, and processes shared a fair amount of overlap in some of the sources examined. Some of the principles encountered were framed as imperatives, steps that practitioners must follow for proper MOSA implementation. Because of the conceptual overlap, some principles and tasks are categorized as processes.

RQ6 - Inputs and outputs

Distinctions were made between inputs and measures, and outputs and goals. Both distinctions drew on the fact that inputs and outputs need to be data used in some process. A measure could be cost, but balance sheets would be input.

Response categorization

Using the combined response database, the quotes for each research question were examined. The overarching category or categories were extracted and put into a table. Each source's extracted quotes were then examined and placed into the relevant category or categories. This two-pass approach helped to create categorizations that best represented the collected quotes. Because the categories are based on statements by the source authors, there may be instances where a category may appear to be a subcategory of another (such as modularity may be a subcategory to business or technical indicators). Without elaboration by the authors on what is intended by the categories the terms are taken at face value. The sources can be recategorized in future work to examine sub-research questions.

In RQ2 and RQ4, the types of responses were also partitioned into "common" and "novel" categories. The dividing line between common and novel was that response types that arbitrarily had less than half of the most common response were considered novel. The common/novel distinction was made in order to highlight both the most frequent response types as well as rare ones that still deserved discussion.

Results and discussion

42 sources were identified through the literature search. The sources examined are listed in Appendix B.

Research question 1: What are the goals of MOSA?

Frequency and categories

The goals of MOSA were mentioned in 32 out of 42 (or roughly 76%) of the sources examined, making RQ1 the most commonly addressed research question. Ten categories were identified, five of which qualify as “commonly cited” and five of which qualify as “novel” by the criteria detailed in the methodology.

Commonly cited goals

Cost and affordability, grouped together, are the most commonly referenced goals, appearing in 23 of the 32 (or 72%) references to MOSA goals. This is followed by *upgradeability/refresh* (69%), and *capability/performance* and *interoperability*, each being referenced by 53% of the RQ1 sources.

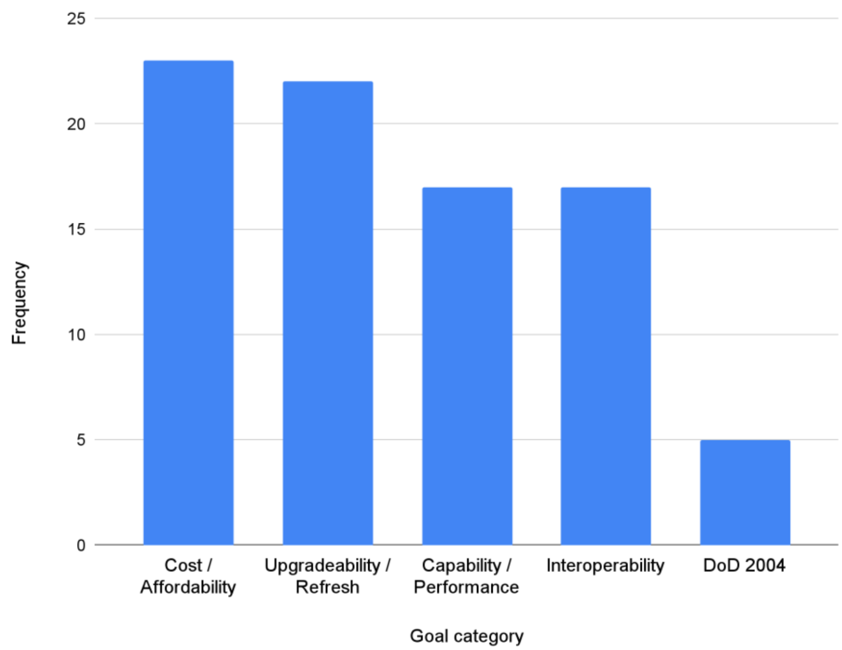


Figure 28. Frequency of Commonly Cited MOSA Goals.

The “DoD 2004” category in Figure 28. Frequency of Commonly Cited MOSA Goals.Figure 28 is included as several sources reference the list of benefits seen in Figure 29, which is from a DoD publication from 2004 (“Program Managers Guide to Open Systems” 2004). The “DoD 2004” list overlaps with other commonly cited goals. For example, *ease of change* in “DoD 2004” overlaps with *upgradeability/refresh* in Figure 28. Because of the high level of overlap, “DoD 2004” was grouped with commonly cited goals instead of the novel goals.

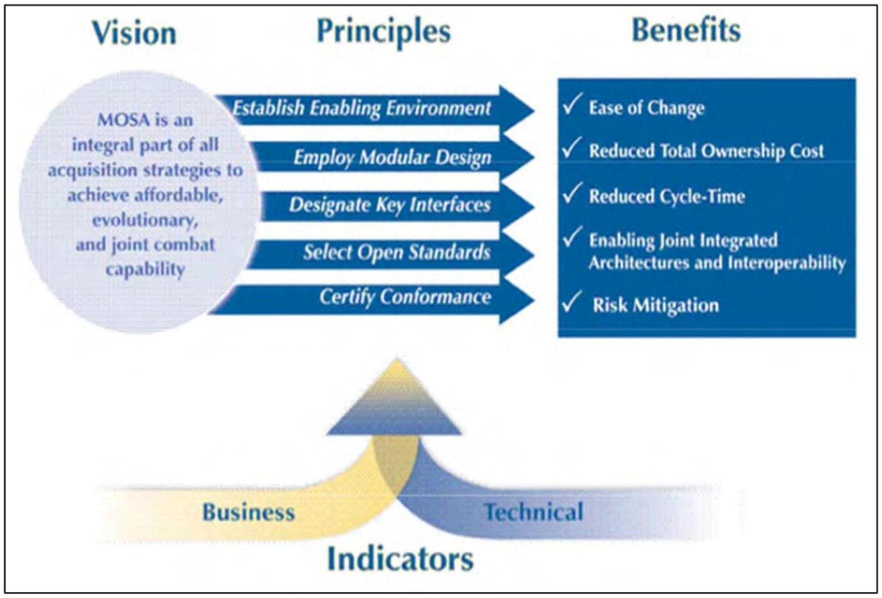


Figure 29. Modular Open Systems Approach (MOSA) (Program Managers Guide to Open Systems 2004).

Novel goals

As shown in Figure 30, *mission flexibility, acquisition, security, logistics*, and *maintainability/sustainment* constitute the set of novel MOSA goals. *Acquisition* is perhaps the least obvious category, and it refers to MOSA either easing the process of acquisition or serving as an acquisition exemplar to other programs.

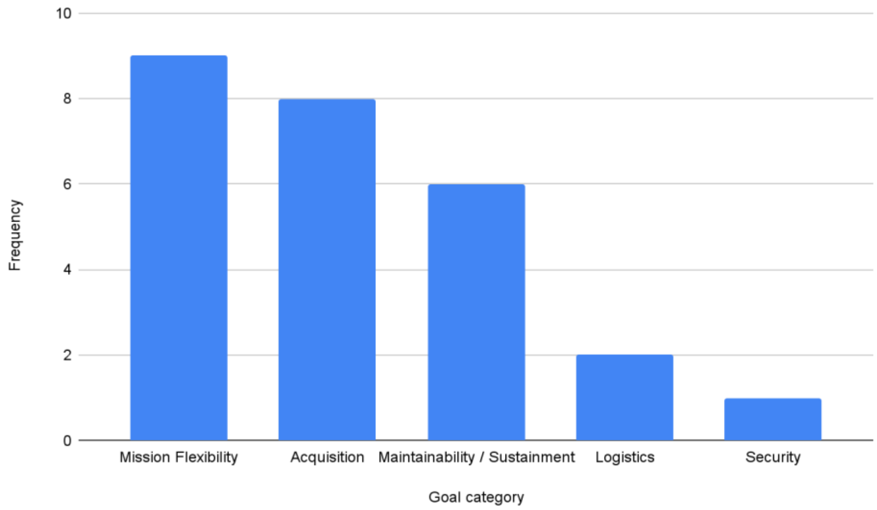


Figure 30. Frequency of Novel MOSA Goals.

Takeaways

The common MOSA goals appear closely connected to what customers would value in delivered systems. The novel MOSA measures appear to reflect more long term or “behind the scenes” aspects of systems, such as acquisition, maintenance, and logistics.

If the frequency of goals in the examined sources reflect the goals' relative importance to engineering organizations, then it may be possible to infer what engineering organizations are looking to get most out of implementing MOSA. However, it must be understood that each organization will prioritize different goals, and the question that truly matters is if these goals align with the government's intention of MOSA.

RESEARCH QUESTION 2: WHAT ARE THE MEASURES OF MOSA?

Frequency

Measures for MOSA were mentioned in more than half of all of the papers (22 out of 42, or approximately 52%) making it the third most addressed research question. Nine categories were identified, three of which qualify as "commonly cited" and six of which qualify as "novel" by the criteria detailed in the methodology.

Commonly cited measures

The three most commonly cited measures of MOSA were *modularity*, *cost*, and *Openness / Open Architecture Assessment Tool (OAAT)*, as seen in Figure 31. The measure of *cost* (mentioned in 6 out of 22 papers) parallels common goals of MOSA in Figure 28. *Modularity* (mentioned in 8 out of 22 papers) is reflected in the common MOSA goals of *upgradeability / refres*h, and *interoperability*.

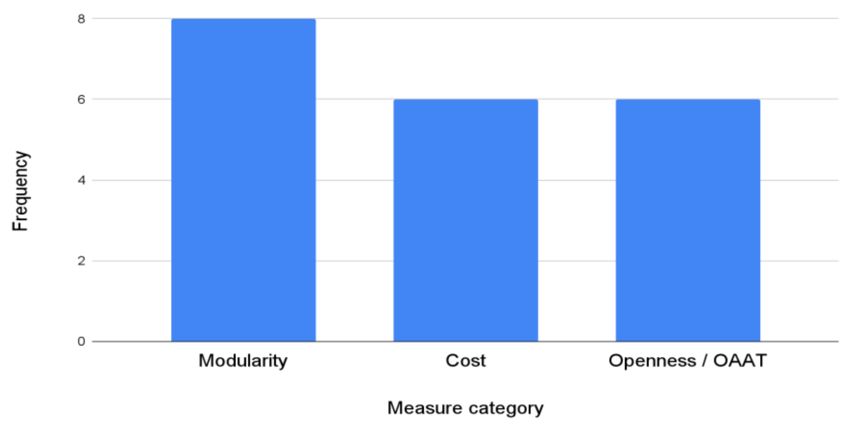


Figure 31. Frequency of Commonly Cited MOSA Measures.

Openness / OAAT (mentioned in 6 of 22 papers) may also parallel *upgradeability / refresh* and *interoperability* in Figure 28 and addresses the "O" in MOSA. OAAT consists of several measures that are combined to give an overall measure of openness. The measures that contribute to openness, labeled as indicators, are described by Sims 2012 as falling into technical and business areas. The key technical indicators are described as:

- Interoperability – The use of standardized data and functional models is essential for the exchange of information between readily separate systems.
- Services – A service is a software component described by metadata, which can be understood by a program. These metadata are published to enable re-use of

the service by remote entities that require no knowledge of the service implementation beyond the published metadata.

- Maintainability – The ability to keep a system operable for a long period of time. This is facilitated by COTS components and the use of open standards.
- Extensibility – The ease with which changes can be made to the system.
- Composability – The extent to which components can be selected and assembled in various ways to meet user requirements.

With key business indicators being described as:

- Is open systems language included in the project documentation?
- Have program personnel been trained in open systems?
- Has an individual been designated as being responsible for open system implementation?
- Is there a plan for implementing an open system with metrics defined to measure progress?
- Does the government own the controlling performance and interface specifications?

Novel measures

Seven novel measures were identified in the examined sources. Of these, the most mentioned were *multiple* and *defined interfaces*. *Multiple* was placed in novel measures despite having half the mentions of *modularity* since it is not a single measure, but a collection from several sources.

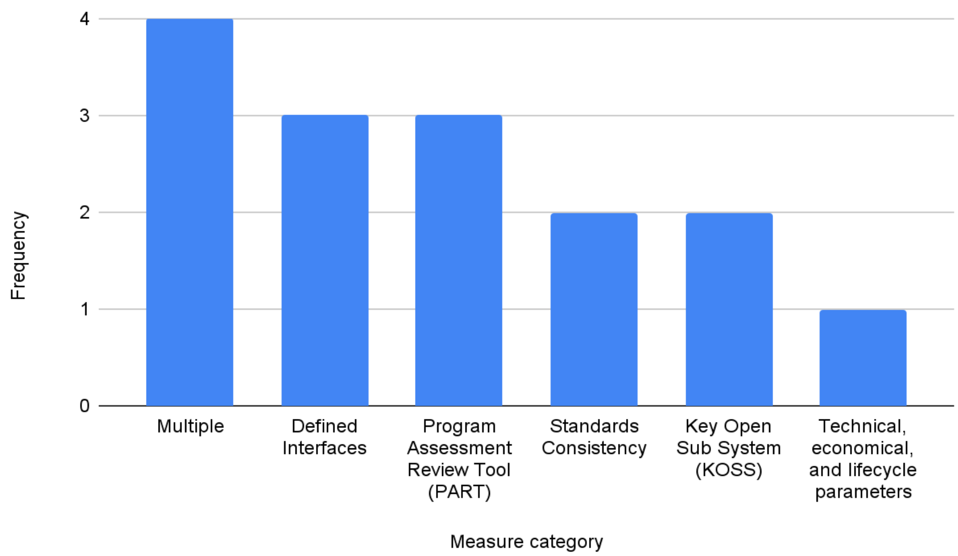


Figure 32. Frequency of Novel MOSA Measures.

Multiple is a category for sources that listed multiple measures at once. Three of the four sources categorized as having multiple measures are seen in Tables 2, 3, 4, and 5. Table

5 shows a list of particularly well-formed measures for MOSA originally created for evaluating a next-generation cockpit.

Table 2 **- Multiple measures in Shaver et al. 2016**

Measures
Decoupling
Cohesion
Portability
Composability
Information reduction

Table 3 **- Multiple measures in "Program Managers Guide to Open Systems" 2004**

Measures
Percentage of key interfaces defined by open standards
Percentage of obsolete modules
Percentage of modules that can change without major system redesign
The number of latest technologies successfully migrated to a program as a result of adherence to MOSA principles

Table 4 **- Multiple measures in Colombi et al. 2015**

Schedule urgency level of integrations
Technology readiness level
Threat environment
Interface maturity level
Information protection level
User community of connected systems
Number of functionally equivalent connected systems
Number of functionally different connected systems
Number of integrations at the interface

Table 5 - **Multiple measures in Maier et al. 2020**

Value measure	***Type of scale used to measure***	***Definition***
Rapid component interchange	Constructed	The ability to swap components within the system.
Security components	Natural	The number of layers and fail-safes dedicated to ensuring system security.
Accessibility	Constructed	The access the Army has to all system data. The inverse of proprietary data.
Compatibility	Natural	The number of other open architecture standards it is compatible with.
Ease of operability	Constructed	Training required to transition from base case to a new system.
Defined interfaces	Natural	The number of persistent systems within the cockpit.
Longevity	Constructed	The ability of the standard to accommodate new technologies.
Applicability	Natural	The number of aircraft systems encompassed by the open architecture
Functionality	Constructed	The current level of development of the standard.

Technical, economical, and lifecycle parameters are defined by example by Heydari, Mosleh, and Dalili 2016 as shown in Table 6.

Table 6 - Multiple measures in Heydari, Mosleh, and Dalili 2016

Technical parameters	***Economical parameters***	***Lifecycle parameters***
Probability density for time to failure	Modules in demand at a given time	Total operation time
Time to availability of an upgrade	Launch and operational cost of a module	Budget
Maximum number of modules allowed	Rate of value generation for various module types	Maximum time to initial deployment
Maximum communication bandwidth		

The *Key Open Sub System (KOSS)* category refers to an approach by that name that "defines subsystems/components that have the potential to yield the greatest benefit to life-cycle affordability by applying MOSA principles" (Srivastava and Rice 2014).

Takeaways

Given the results of RQ1 (MOSA goals), it is consistent for two of the top measures of MOSA to be *cost* and *modularity*, with *openness / OAAT* following from the "open" found in "MOSA". It follows less readily that there would be 6 novel measures to only the 3 common measures. This may indicate that *cost, modularity*, and *openness / OAAT* are not sufficient to assess all the different aspects of a MOSA implementation and so researchers have been proposing additional ones. Even within the novel measures, there is no clear set of measures that dominates, which may imply that 1) MOSA measures are still an open question or 2) that the community has settled on *cost*, *modularity*, and *openness / OAAT* as the acceptable measures.

In addition, the measures that have been identified in the sources generally lack any specificity to enable quantification. For example, "Modularity" doesn't say what the specific quantity is that is being measured, and so multiple sources may say modularity and mean different things. Even for a relatively straightforward measure such as cost, it is not always clear what cost is being measured (development, acquisition, operation, etc.). This vagueness can be detrimental to the communication of measures between organizations or even within an organization.

Research Question 3: How do the measures of MOSA relate to one another?

Discussion of qualitative and quantitative relationships

The relationships between different MOSA measures were categorized as either qualitative or quantitative. Qualitative relationships occurred when a source mentioned a conceptual relationship between measures. Quantitative relationships were when the source in some way combined the numerical values of measures.

Frequency

Of the 42 papers examined, nine (or approximately 21%) address the relationship of measures. One of those nine sources discusses both quantitative and qualitative relationships, and counted for both.

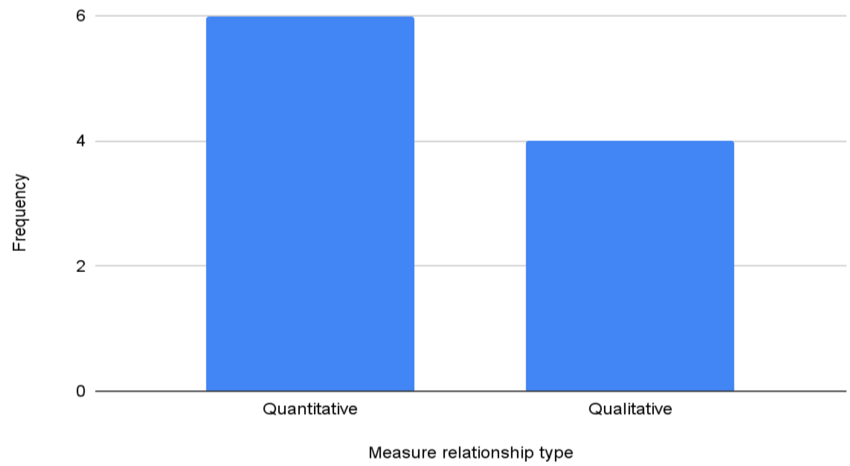


Figure 33. Frequency of MOSA measure relationship types.

Qualitative relationships

The qualitative relationships tend to describe the measures as "synergistic," such as needing to all be present to meet government requirements. The relationship between measures in one case ("Modular Open Systems Approach: Considerations Impacting Both Acquirer and Supplier Adoption" 2020) was described directly as a synergy gained when systems are evaluated and optimized using modularity and openness.

Examples of qualitative relationships can be found in several of the examined sources. The *Systems Engineering Guidebook* ("Systems Engineering Guidebook" 2022) says that programs must meet the requirements of MOSA in order to receive Milestone B approval. Shaver et al. discusses a conceptual tradeoff between coupling, cohesion, and information reduction as measures of modularity (Shaver et al. 2016).

Quantitative relationships

Perhaps the clearest example of quantitative relationships occurs in the Open Architecture Assessment Tool (OAAT), where measures along two dimensions (Business/Programmatic and Technical) are combined into an overall openness score (Rendon 2008). Other sources, however, have doubted the efficacy of this approach (Srivastava and Rice 2014).

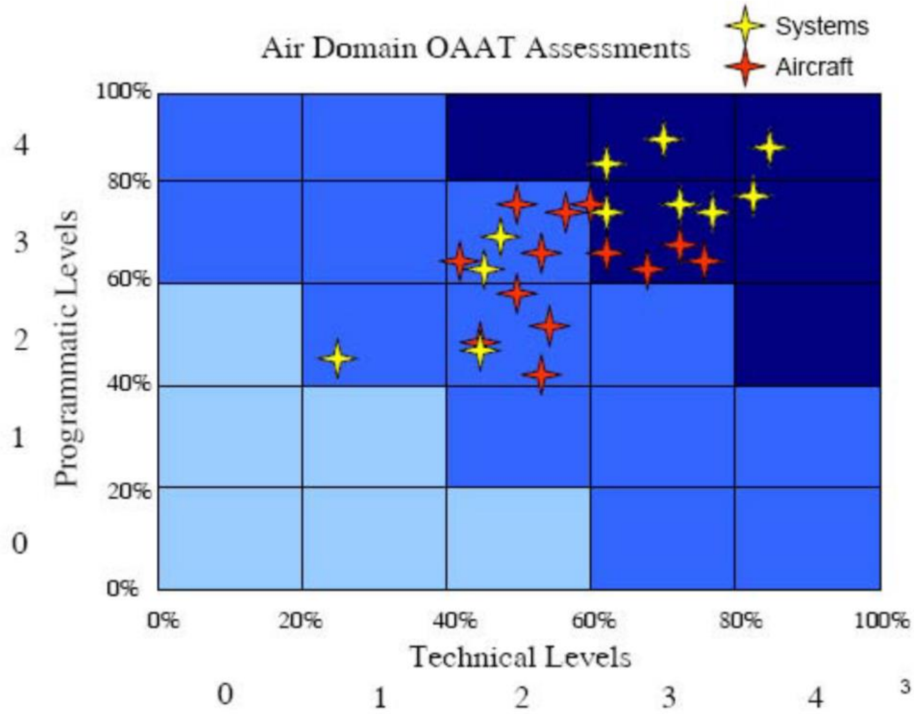


Figure 34. Aircraft Domain OAAT results, with light blue being low openness and dark blue being high openness (Rendon 2008).

Other sources that numerically combined MOSA measures included the use of value-based design (Colombi et al. 2015) and MOSA decision support tools (Dai, Guariniello, and DeLaurentis 2022).

Takeaways

RQ3 is the least discussed research question (tied with RQ5). The lack of discussion may be indicative that the relationships between measures is not often considered by engineering organizations. If approaches like OAAT lack efficacy and approaches like MOSA decision support tools are still in development, this may be an area for further research focus. Understanding the limitations of the number of sources, there still appears to be fairly broad interpretations of how the MOSA measures should relate, with no clear quantitative or qualitative conclusion. It is likely that the MOSA measure relationships are both quantitative and qualitative, and it is the tool that is being used that assumes a type of relationship.

Research Question 4: Are there specific processes for MOSA?

Frequency

This research question is the second most frequently addressed by the sources examined, with it being addressed by 27 of the 42 sources (or approximately 64%). A source may be categorized multiple times depending on their response.

Commonly cited processes

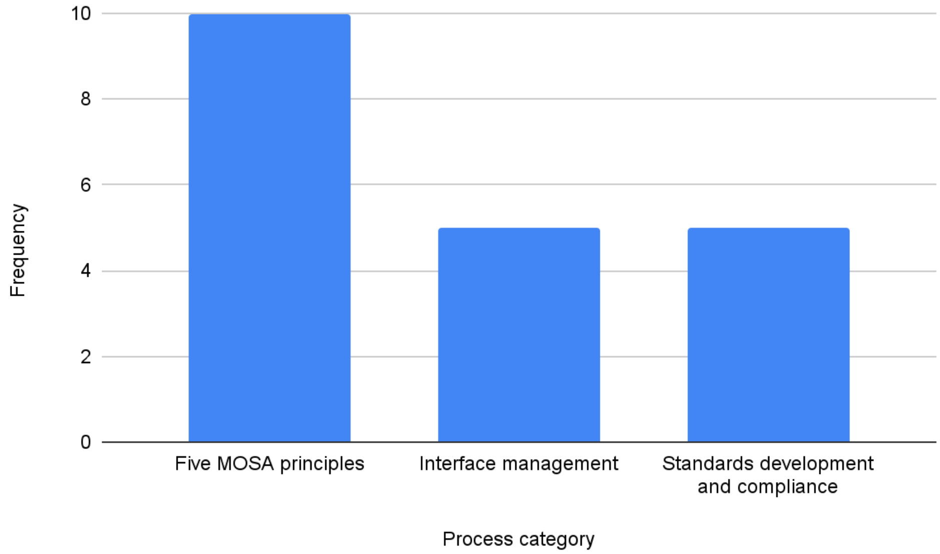


Figure 35.Frequency of common MOSA processes.

The most commonly referenced MOSA process in the sources examined (10 out of 27, or 37%) were the *five MOSA principles* seen in Figure 29. The DoD Program Manager's Guide ("Program Managers Guide to Open Systems" 2004) elaborates on the five MOSA principles, which are listed here and elaborated in Appendix 5.12:

1. *Establish an enabling environment*
2. *Employ modular design*
3. *Designate key interfaces*
4. *Use open standards*
5. *Certify conformance*

The DoD Systems Engineering Guidebook ("Systems Engineering Guidebook" 2022), however, appears to put a slightly different spin on the five MOSA principles, discussing them in more specific detail and distinguishing between the expectations for program managers and systems engineers. Omitting items specifically related to intellectual property, the Systems Engineering Guidebook expects program managers to:

- Establish supportive requirements; business practices; and technology development, acquisition, T&E, and product support strategies for effective development of open systems.
- Map modular open systems strategy and functional architecture to Statement of Work (SOW) requirements, Data Item Descriptions (DIDs) and Contract Data Requirements List (CDRL) items consistently across the enterprise.
- Ensure compliance.
- Consider including MOSA as one of the evaluation criteria for contract proposals.

- Determine the appropriateness of MOSA by considering software constraints, security requirements, and procedures, ... , life cycle affordability and reliability of widely supported and consensus-based standards, as well as other relevant factors such as environmental constraints (e.g., temperature, humidity) and Environment, Safety, and Occupational Health (ESOH) considerations.

Whereas systems engineers are expected to:

- Employ an overall plan for MOSA that supports the system functional architecture and uses prescribed USD(R&E) business case analyses.
- Ensure the system functional architecture is structured to accommodate Open Systems Architecture (OSA) where feasible because of the high potential for reduced risk and cost.
- Assess performance.
- Balance current implementation of MOSA with performance and evolving technology at the physical level; MOSA establishes a technical baseline that may support modular architecture, but formally constrains the interfaces between modules where interfaces close to current performance limits may quickly become obsolete.
- Evaluate the technical appropriateness of MOSA by considering software constraints, security requirements and procedures, availability and cost of data rights, life cycle affordability and reliability of widely supported and consensus-based standards, as well as other relevant factors, such as environmental constraints (e.g., temperature, humidity) and ESOH considerations.

The distinction between program managers and systems engineers was not seen in other sources and provides an important reminder of the different organizational aspects of implementing MOSA.

Interface management and *standards development and compliance* were seen in five sources each. As seen in Colombi et al. 2015 in Figure 36, these processes can occur concurrently. The process in Figure 36 evaluates an interface, but also describes how to choose between implementing, investing, or considering open or closed standards.

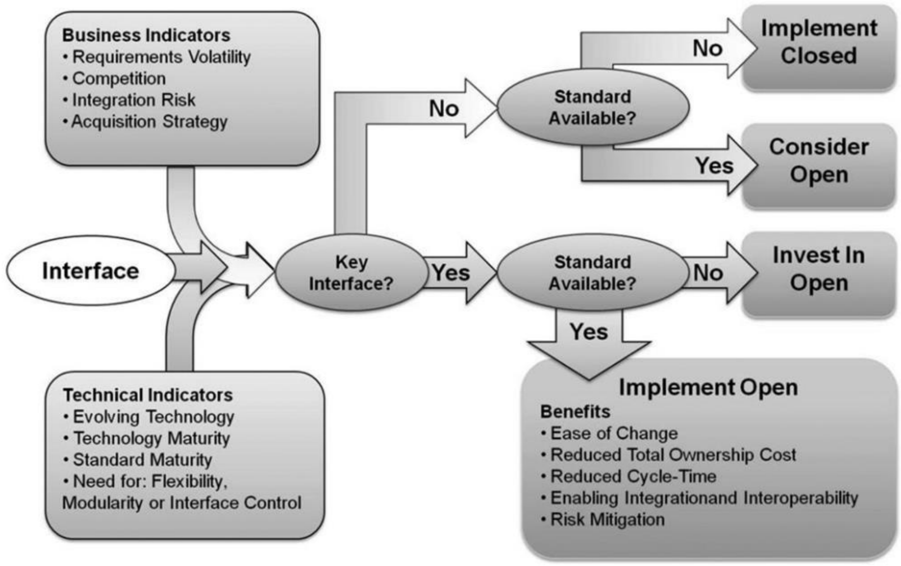


Figure 36. Interface evaluation flowchart (Colombi et al. 2015).

Novel processes

The remaining MOSA processes are referenced only once or twice. In total, nine MOSA processes can be described as novel with 12 references total, as shown in Figure 37. Several of the Figure 37 processes are described in more detail in the following subsections.

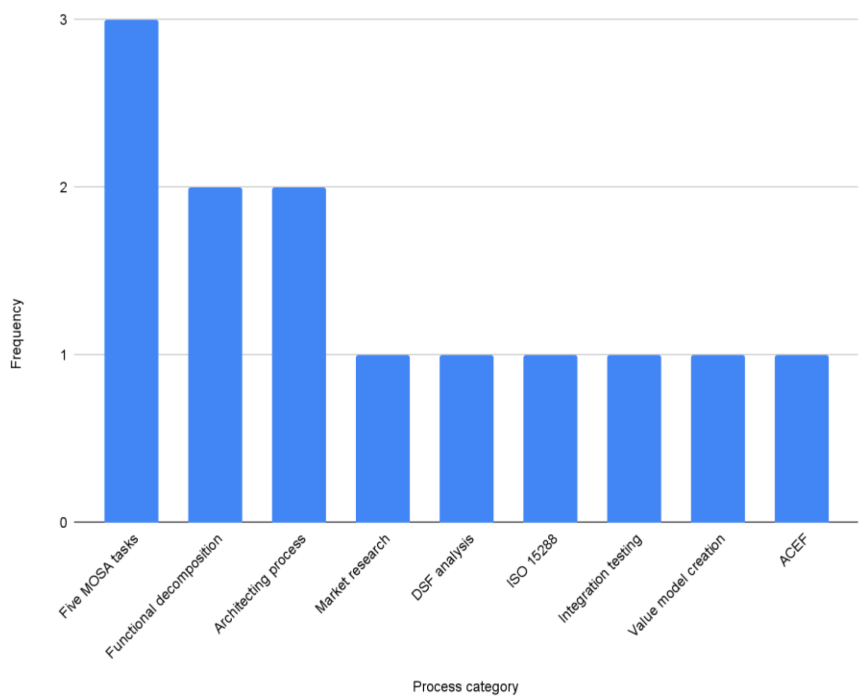


Figure 37. Frequency of novel MOSA processes.

Functional decomposition

Two sources mention *functional decomposition*, illustrated by Rose et al. in Figure 38. The overall goal of functional decomposition is to move as close as possible to having each modular architecture function mapped to a single module.

Figure 38. Initial and end states for functional decomposition (Rose et al. 2014).

Architecting process

Architecting process was the name given to attempts, such as by Connah et al. 2012 in Figure 39, to develop an architecture as an explicit goal.

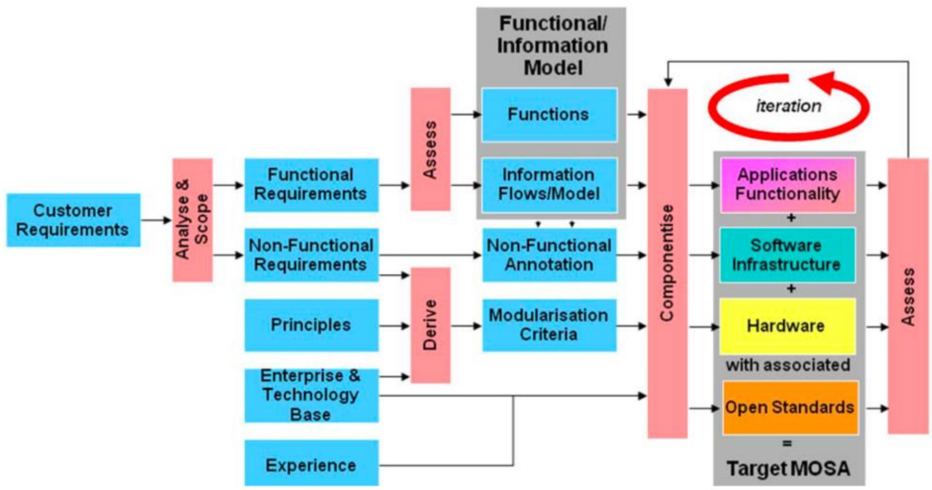


Figure 39. MOSA architecting process (Connah et al. 2012).

Five MOSA tasks

In addition to the five MOSA principles, there are also five MOSA tasks, categorized as *five MOSA tasks* (Aragon 2008; "Program Managers Guide to Open Systems" 2004; Sims 2012). The tasks are similar to the principles, but are called out distinctly by some sources. The five MOSA tasks consist of ("Program Managers Guide to Open Systems" 2004):

1. Identify and analyze capabilities and strategies that could most effectively be pursued by open system design solutions.
2. Assess the feasibility of open systems design solutions
3. Establish performance measures to assess MOSA implementation progress
4. Use MOSA principles to develop an open architecture

5. Identify and resolve MOSA implementation issues and report the unresolved issues to Milestone Decision Authority.

Decision Support Framework (DSF)

The Decision Support Framework (DSF) (categorized as DSF analysis) is an attempt to create executable software that would "provide key information to program managers and other stakeholders to guide MOSA-related decisions throughout the acquisition life cycle" (Dai, Guariniello, and DeLaurentis 2022). It contains several sub-processes, like quantitative and qualitative analysis. Visual examples of DSF are seen in Figure 40.

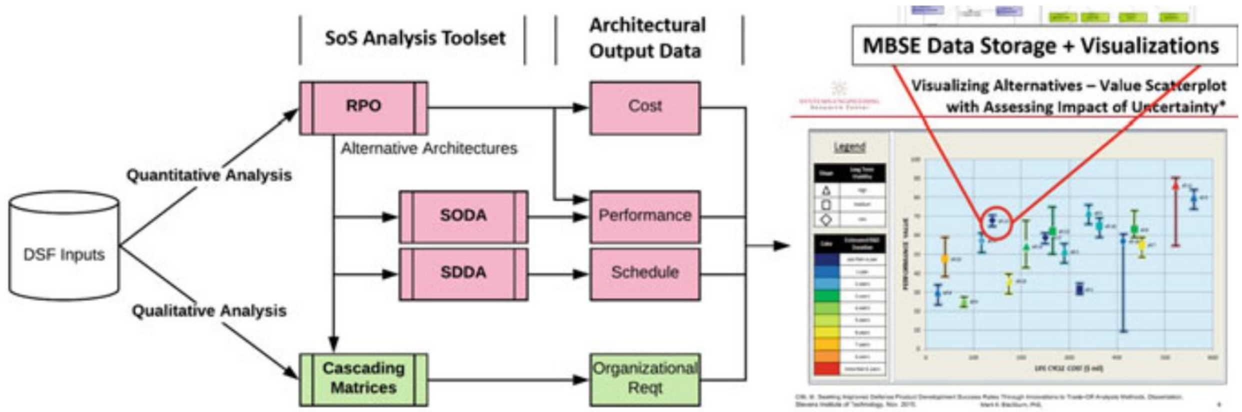


Figure 40. The Decision Support Framework (DSF) MOSA workflow (Dai, Guariniello, and DeLaurentis 2022).

RPO, SODA, and SDDA are tools from the Purdue Analytic Workbench.

Architectural Cost Effectiveness Framework (ACEF)

The Architectural Cost Effectiveness Framework (*ACEF*) proposes 6 steps for estimating MOSA relevant costs (Barrett 2017).

1. Establish business strategy
2. Develop program reference architecture
3. Create alternatives
4. Compare alternatives
5. Account for uncertainty
6. Examine results and update reference architecture

Takeaways

The sources examined show that there are three common MOSA processes, but the sources also show that there are at least nine that researchers have tried or are currently exploring. The existence of so many novel processes may indicate that the common processes still have shortcomings. It is possible that the common processes are too broad and general, so novel processes may be required in order to manage the more nuanced aspects of a MOSA implementation. Understanding how researchers and practitioners are attempting to augment common MOSA processes could provide an avenue for future research into how to evolve standard MOSA processes in the future.

RESEARCH QUESTION 5: HOW DO THE MOSA PROCESSES RELATE TO ONE ANOTHER?

Process relationship categories

Based on the data collected from the sources that had relevance to RQ5, the MOSA process relationships were sorted into four categories:

- ***Sequential*** - The processes are related by the fact that they proceed in a particular order
- ***Synergistic*** - The processes all contribute to achieving an overarching goal
- ***Hierarchical*** - The processes occur at different levels of the system architecture
- ***Organizational*** - Different roles in the organization carry out different processes

Frequency

The relationships between processes were discussed in 9 of the 42 sources examined, or by about 21% of sources. This makes RQ5 tied with RQ3 as the least addressed of the research questions.

Process relationships

From the 9 sources that contained data on process relationships, the single most frequent category was *sequential* relationships between processes, which was mentioned by five of the sources, seen in Figure 41. The three remaining processes relationship categories have four mentions in total, with *synergistic* relationships accounting for two of the four mentions.

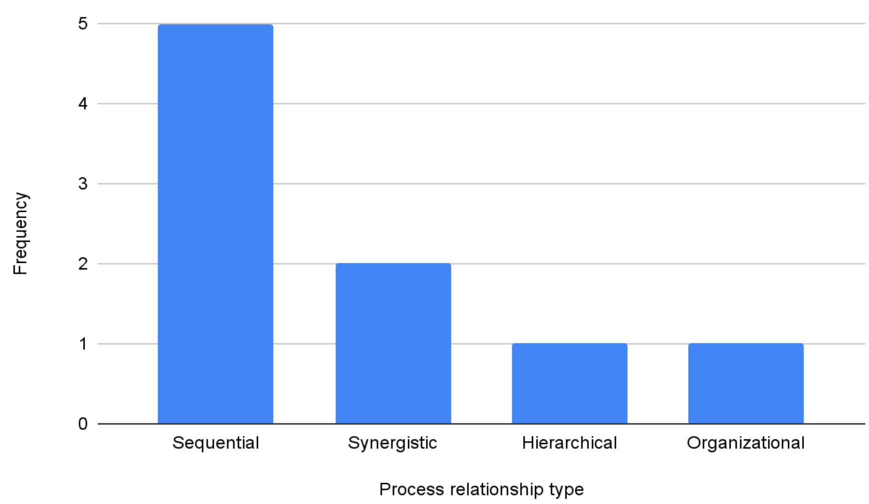


Figure 41. Frequency of MOSA process relationships.

Takeaways

Given the sparsity of coverage in the examined sources for RQ5, it does not appear that relationships between MOSA processes are a common consideration. When the relationships between processes are examined, the most common relationship (*sequential*) may be a reflection of the common approaches used in industry, such as waterfall development, where processes flow from one to another. More research would be needed to understand why sequential process relationships are more common and if sequential processes lead to more or less successful MOSA implementations than other kinds of process relationships.

Research question 6: What are the necessary input and output data for the processes?

Frequency

13 of 42 (31%) of the sources reviewed mentioned input and/or output data.

Inputs

The most common categories of input data were *Others*, *Business and technical strategy documents* and *Requirements*. *Business and technical strategy documents* and *Requirements* were each mentioned in five of the sources. Outside of these three categories, all other data types are mentioned by at most two sources and cover areas such as *market research findings* and *government and industry subject matter expert (SME) input*.

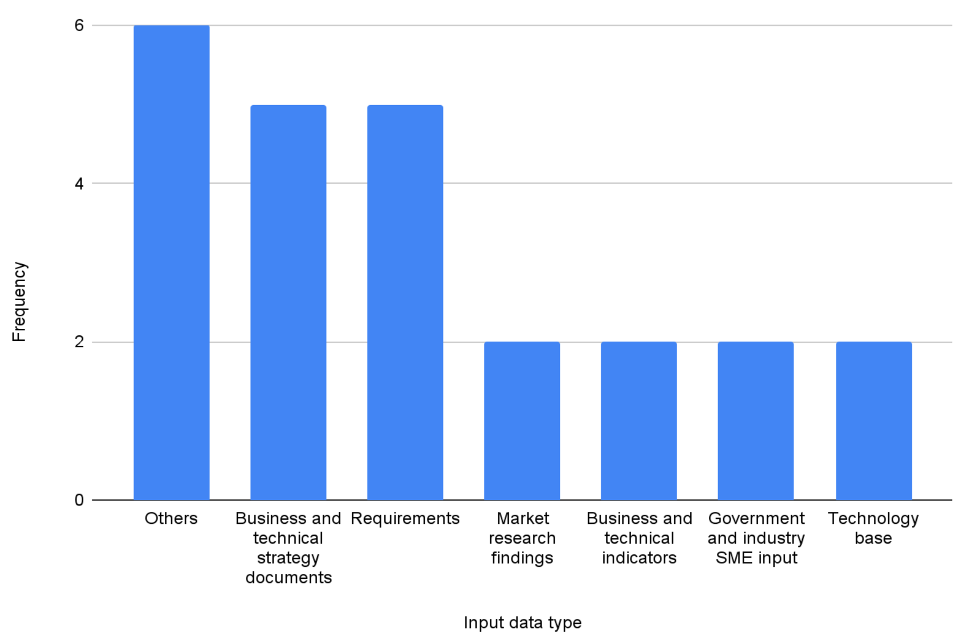


Figure 42. Frequency of MOSA process input data types for sources examined.

The *Others* category in Figure 42 refers to a collection of input data types that are only seen once in the examined sources. Some of these include:

- Technical architecture guidance, constraints from prior/legacy HW/SW components, legacy interfaces, life cycle data (Gaska 2012), Figure 43.

- Concept of Operations, a description of capability gaps to be fulfilled, and a library of candidate systems to be selected (Dai, Guariniello, and DeLaurentis 2022)
- Acquisition strategy, analysis of alternatives, program protection plan, and enterprise architecture ("Systems Engineering Guidebook" 2022)
- Principles and experience (Connah et al. 2012)
- Relative weights for value modeling (Maier et al. 2020)
- A major component set, qualitative obsolescence measures, estimates for the relative cost of change of system components, and the relative capability improvement offered by components (Sims 2012)

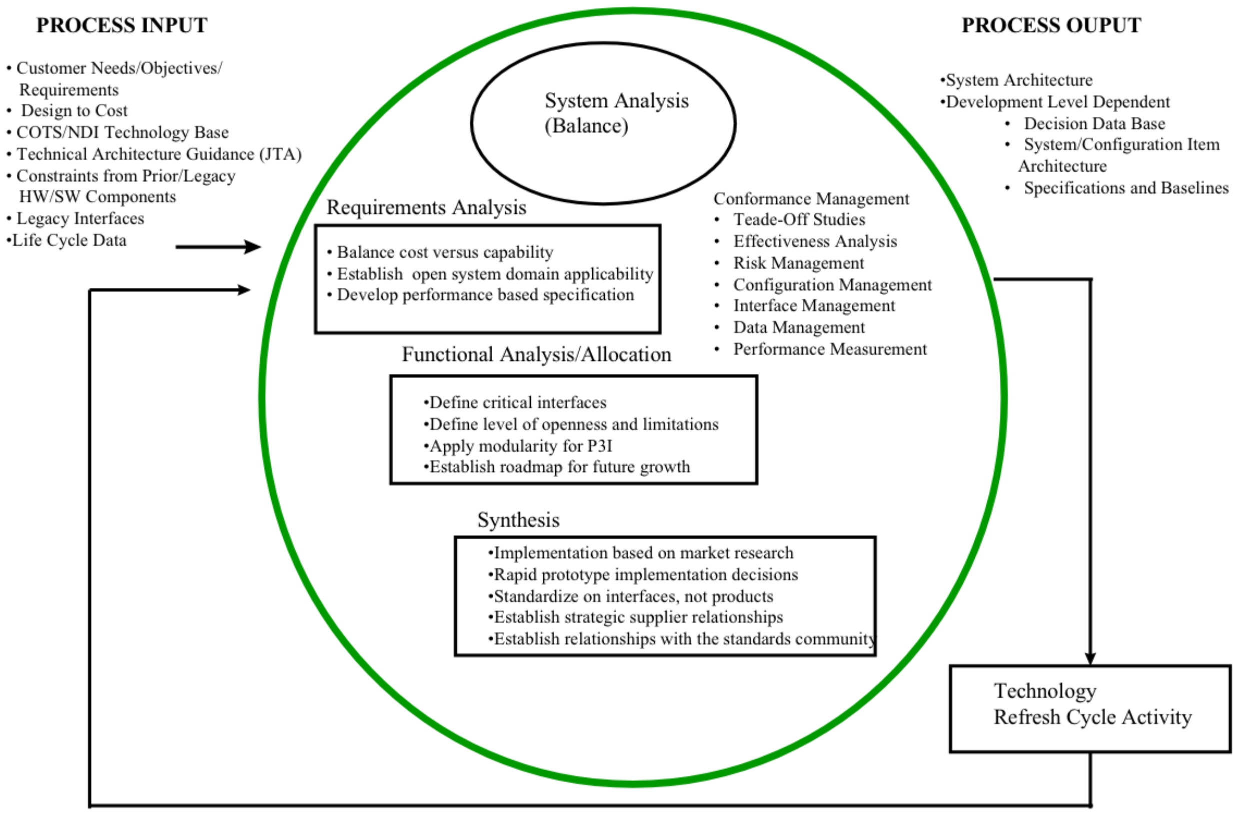


Figure 43. Open System Architecture System Engineering (Gaska 2012).

Outputs

Output data is mentioned in 9 of the 13 (69%) examined sources that address input or output data. The most common type of output data is that of a standard, or set of standards. *Standard* is followed by *system architecture*, and then multiple single mention categories.

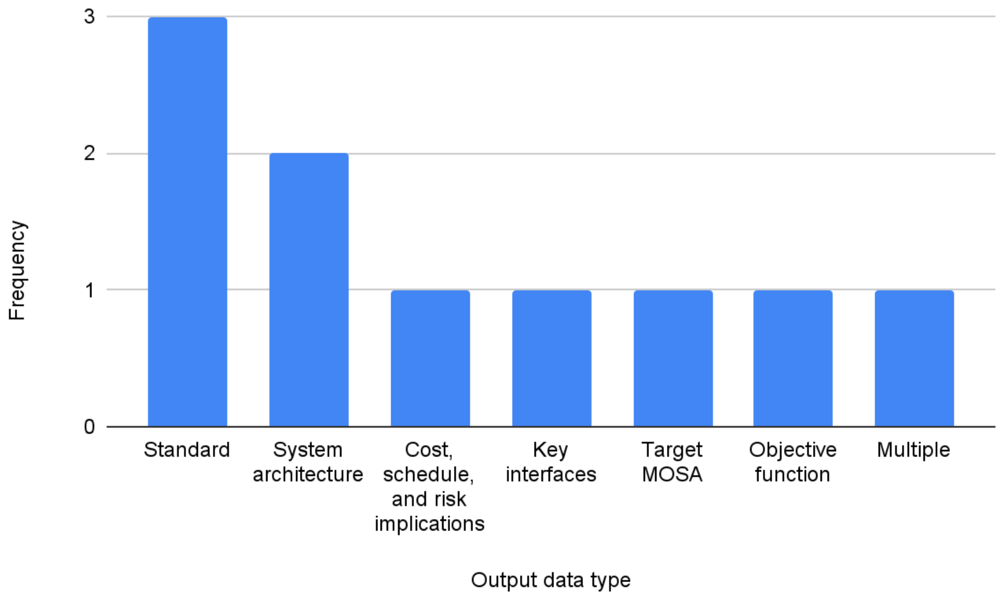


Figure 44. Frequency of MOSA process output data types for the sources examined.

The *Multiple* category in Figure 44 includes the process outputs from Figure 43 that are not system architecture. The other process outputs are:

- Decision database
- System/configuration item architecture
- Specifications and baselines

It should also be noted that the *Multiple* outputs are dependent on development level.

Takeaways

The sources examined suggest many potential types of input and output data for MOSA processes, but lack a clear consensus. Only 13 (31%) of all sources mention input data with only 9 (21%) mentioning output data. Within the sources that mention input or output data types, no one type of input or output data has relatively high frequency, with many data types only being mentioned in one or two sources. A few sources demonstrated the connecting processes from input to output data (such as seen in Figure 45), but most sources do not make such explicit connections. It may also be the case that *requirements* and *business and technical strategy documents* are the most common types of input data in the sources examined because they are the most readily available in engineering organizations. Further research would be needed to see if availability of data influences the representation of data in the literature.

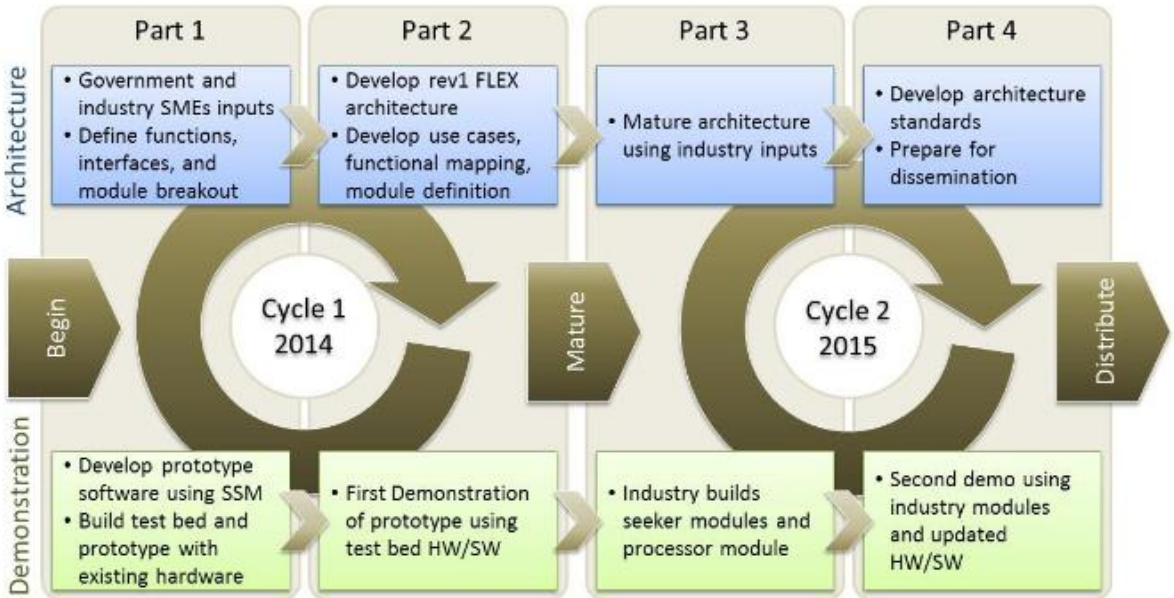


Figure 45. Flexible Weapon's Two-Cycled Development Plan, showing process inputs and outputs (Rose et al. 2014).

CONCLUSION

This report presents findings from analyzing 42 academic sources discussing MOSA in order to answer 6 research questions regarding community consensus on MOSA. This report provides knowledge of what MOSA is to program managers and practitioners of systems engineering with the goal of working towards a more consistent and nuanced MOSA practice.

RQ1: What are the goals of MOSA?

The reviewed literature suggests that the common goals of MOSA are related to customer valuations of delivered systems (Cost/Affordability, Upgradeability/Refresh, Capability/Performance, Interoperability).

RQ2: What are the measures of MOSA?

There were some conflicting ideas presented as to how MOSA should be measured, but the most prevalent were total cost and modularity, along with many less-referenced measures. The measures identified in the sources generally lack the specificity needed to perform a quantification of MOSA, which may lead to miscommunications when discussing measures.

RQ3: How do the measures of MOSA relate to one another?

Only 21% of sources mentioned MOSA measurement relationships, with a split between quantitative and qualitative relationships. This suggests that the relationships between measures is not often considered by engineering organizations

RQ4: Are there specific processes for MOSA?

The five MOSA principles were mentioned across ten of the sources reviewed with eleven additional processes or greater context to the MOSA principles also identified. This indicates that MOSA processes do not have consensus in the community but MOSA principles are becoming dominant.

RQ5: How do MOSA processes relate to one another?

Similar to RQ3, only 21% of sources mentioned relationship of MOSA processes, indicating a potential lack of interest, knowledge, or some other cause. The result of *sequential* being the most frequent may be due to the use of common approaches in practice.

RQ6: What are the necessary input and output data for the processes?

The sources examined suggest many potential types of input and output data for MOSA processes, but lack a clear consensus. Processes were said to require inputs such as system requirements and specifications, and would produce various outputs such as standards and system architectures.

Overall, there did not seem to be much consensus on the research questions or wide enough coverage of some research questions to make conclusions about their interpretations in engineering organizations. While some responses to research questions were more common than others, there were no research questions with strong consensus across a majority of sources.

FUTURE WORK

In the discussions for each research question, several possible future research questions arose.

Future research questions, grouped by this report's RQs below, address areas like comparing the results of this report to practice in engineering organizations and which MOSA areas (goals, measures, processes, and measure and process relationships) are in most urgent need of development.

- ***Follow on Questions related to RQ1*** - Does the frequency of goals in the examined sources match the frequency of their usage in engineering organizations?
- ***Follow on Questions related to RQ2*** - Are measures for MOSA still an open question? Do the common measures represent the acceptance of *cost*, *modularity* and *openness / OAAT* as measures in engineering organizations?
- ***Follow on Questions related to RQ3*** – To what degree do architecting and systems engineering tools influence the relationships between measures?
- ***Follow on Questions related to RQ4*** - Could the attempts by researchers and practitioners to augment common MOSA processes inform how to evolve standard MOSA processes in the future?
- ***Follow on Questions related to RQ5*** - Why are sequential process relationships more common than other types of process relationships? Do sequential processes lead to more or less successful MOSA implementations than other kinds of process relationships?

- ***Follow on Questions related to RQ6*** - Does availability of data types in engineering organizations influence what types of data are most represented in the literature?

Future work should also address methodological weaknesses present in the approach used in this research. First, due to the fact that the categories for each RQ were drawn independently from the sources examined, there is no guarantee that the categories do not overlap. For example, in RQ5, it is likely that processes exhibiting a hierarchical relationship have sequential characteristics as processes move down the hierarchy. Second, there is inherent subjectivity in how the categories were selected, so a more objective approach to category selection may help to improve the repeatability of this report's approach.

The database generated for this report also creates the possibility of tracking the rise and fall of MOSA concepts over time. Future research could help to understand which of the concepts examined in this report are still relevant, are no longer relevant, were never relevant, or may be rising to relevance.

A Research Vision to Enable Digital Agile Lifecycle through Reimagining of Architecting

The overall research objective of the proposed work is to reimagine government reference architectures (GRAs) to enable a digital agile lifecycle similar to the Supra-system lifecycle view in Figure 26. The proposed research has four primary objectives:

A. Identify and represent GRA Data
B. Develop GRA assessment and validation tool
C. Form an Architecture Decision Guide
D. Identify agile lifecycle needs

These objectives will be best accomplished through interdisciplinary approaches due to their inherent incorporation of humans, organizations, policy and law, physical artifacts, etc. The four objectives each require an interdisciplinary approach to be accomplished. The general approach for each objective is outlined below.

Objective A: Identify and represent GRA Data

Data are the building blocks for digital lifecycles. As such, the definitions, form, and impact of the data of a critical engineering artifact, such as the architecture, needs to be identified. In the proposed research, tools are formed to aid the stakeholders in eliciting, understanding, representing, analyzing, and communicating this data. The expected outcomes of the objective's methods are a GRA Data Glossary, a GRA Stakeholder Data Representation Plan, and a GRA Stakeholder Data Impact Guide.

Methods:

- GRA Data Glossary

- Identify the data required as inputs and outputs of an architecture
    - Determine accuracy of the data required
    - Determine the form of the data
    - Determine the criteria that a data must meet to be acceptable
    - Identify which data sources should be prioritized over others
- GRA Stakeholder Data Representation Plan
    - Identify/Form strategies for obtaining group consensus on key architectural preference terms (typically in the form of -ilities)
    - Identify/Form strategies to drive stakeholders to express preferences in the form of metrics and value models rather than subjective terms
    - Develop tools for stakeholders to envision and describe, both qualitatively and quantitatively, future states that the architecture may be used in
- GRA Stakeholder Data Impact Guide
    - Identify the sensitivities of key metrics to architecture data
    - Develop tool for recording and visualizing sensitivities

Objective B: Develop GRA assessment and validation tool

There is a significant lack of research on ways to determine the goodness of an architecture. Due to this, architectures are often approved or disapproved through heuristics or instinct, neither of which provide desired rigor. The proposed research develops tools to assess and validate GRAs. The expected outcomes of the objective's methods are a GRA Data Glossary, a GRA Stakeholder Data Representation Plan, and a GRA Stakeholder Data Impact Guide.

Methods:

- GRA Assessment Tool
    - Identify criteria for GRA acceptance
    - Identify architecture characteristic measure for MOSA completion
    - Form tool for quantifying criteria and performing measurements on architectures
- GRA Validation Plan
    - Identify criteria for GRA validation
    - Identify data and models necessary to determine GRA validity

Objective C: Form an Architecture Decision Guide

Architectures are the result of decisions made by an architect. The types of decisions that an architect can make, and their options in those decisions, are not always apparent. The proposed research develops tools to aid architects in identifying decisions they can make to design the architecture, both based on historical data and creativity. The expected outcome of the objective's methods is an Architecture Decision Guide.

Methods:

- Architecture Decision Guide
    - Identify types of decisions an architect can consider when designing an architecture

- Identify example decisions an architect can consider
- Identify example alternatives an architect can select for each of the decisions
- Form a tool to aid the architect in identifying decisions and alternatives in designing an architect (leveraging research in creativity)

Objective D: Identify agile lifecycle needs

A true agile process is highly iterative with constant improvement. This contradicts the traditional architecting process where the architecture is relatively static once it is passed to the system stakeholders. An agile lifecycle must have architectures that are able to change. The proposed research develops tools to aid architects in understanding the relationships between the architecture and systems and tools to understand the impacts of their changes have throughout the lifecycle. The expected outcome of the objective's methods is a formation of Agile Methods and a Lifecycle Impact Model.

Methods:

- Agile Methods
  - Identify engineering and organizational gaps between architecture and systems
  - Identify/adapt/form system methods for bridging gaps
  - Identify policies for modification of architecture during system development
  - Develop tool for communicating architecture changes to system engineers
- Lifecycle Impact Model
  - Identify potential impact types architectural change has on systems
  - Identify potential impact types between systems
  - Form model structure for analyzing potential impacts due to changes in the architecture
  - Develop review gates for architecture modification process

The approaches outlined above should be executed by an interdisciplinary team consisting of experts in Psychology, Computer Science, Philosophy, and Engineering. Incorporating these diverse disciplinary perspectives in the approaches is a mechanism for identifying gaps that would otherwise be missed. For example, while eliciting stakeholder data is an activity performed by architects and engineers, it is psychology who can provide the necessary rigor. While computer science is most adept at developing visualizations of sensitivities, psychology can identify which visualizations improve effectiveness of the architects through human studies. By structuring the approaches around a diverse team there is inherent innovation which will be produced from the interdisciplinary perspective.

In addition, there is limited research that spans the architectural and engineering activities addressed by the objectives. This research would holistically develop tools that inherently work together to enable a digital agile lifecycle.

References

Aragon, Lasumar. 2008. “Technology Insertion Considerations for Complex System of Systems Development.” Naval Postgraduate School. https://apps.dtic.mil/sti/citations/ADA488833.

Barrett, Donald A. 2017. “DEVELOPMENT OF THE ARCHITECTURE COST EFFECTIVENESS FRAMEWORK AND APPLICATION TO OPEN SYSTEMS ARCHITECTURES.”

Bayrak, Dr Emrah, Dr Mert Egilmez, Henry Kuang, Xingyu Li, Jong Min Park, Panos Papalambros, Bogdan Epureanu, et al. 2015. “MARVEL – A Modular Vehicle Fleet Simulation Tool.”

Bonilla-Ortiz, Giselle, and Dinesh Verma. 2020. “The Need for a Secure Modular Open Systems Approach (MOSA): Building the Case Using Systems Thinking Methodologies.” In 2020 IEEE Systems Security Symposium (SSS), 1–4. https://doi.org/10.1109/SSS47320.2020.9197726.

Boudreau, Michael. 2007. “Acoustic Rapid COTS Insertion: A Case Study in Modular Open Systems Approach for Spiral Development.” In 2007 IEEE International Conference on System of Systems Engineering, 1–6. https://doi.org/10.1109/SYSOSE.2007.4304229.

Boulet, Michael, Tate DeWeese, Andrew Bird, Ryan Kreiter, and Calvin Cheung. 2020. “An Open Data Architecture for Ground Vehicle Data-Driven Autonomy Development and Validation.”

Brabbs, John, Scott Lohrer, Paul Kwashnak, Paul Bounker, and Mark Brudnak. 2019. “M&S AS THE KEY ENABLER FOR AUTONOMY DEVELOPMENT, ACQUISITION AND TESTING.”

Colombi, John M., Matthew J. Robbins, Joshua A. Burger, and Yvette S. Weber. 2015. “Interface Evaluation for Open System Architectures Using Multiobjective Decision Analysis.” Military Operations Research 20 (2): 55–69.

Connah, Jessica, Abigail Solomon, John McInnes, Olwen Worthington, and Dale Chambers. 2012. “Openness in Military Systems.” In 2012 Military Communications and Information Systems Conference (MCC), 1–6.

Dai, Michael, Cesare Guariniello, and Daniel DeLaurentis. 2022. “Implementing a MOSA Decision Support Tool in a Model-Based Environment.” In Recent Trends and Advances in Model Based Systems Engineering, edited by Azad M. Madni, Barry Boehm, Daniel Erwin, Mahta Moghaddam, Michael Sievers, and Marilee Wheaton, 257–68. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-82083-1_22.

Davendralingam, Navindran, Cesare Guariniello, Shashank Tamaskar, Daniel DeLaurentis, and Mitchell Kerman. 2019. “Modularity Research to Guide MOSA Implementation.” The Journal of Defense Modeling and Simulation 16 (4): 389–401. https://doi.org/10.1177/1548512917749358.

Domerçant, Jean Charles. 2017. “A Functional Modularization Approach for Architecting Defense Open Systems: In Support of Strategic Open Architecture Reuse.” In 2017 Annual IEEE International Systems Conference (SysCon), 1–7. https://doi.org/10.1109/SYSCON.2017.7934748.

Dorny, Jonathan, and Ryan O’Leary. 2010. “PLATFORM HEALTH MANAGEMENT INFORMATION SHARING VIA SERVICE ORIENTED ARCHITECTURE.”

Dziuk, Madaline, Lewis Edinburgh, Shawn Reynolds, Jim Rousseau, and Jason Boots Winn. 2022. “Solution Approaches from Industry for Flightline of the Future and Next Generation Test Systems.” In 2022 IEEE AUTOTESTCON, 1–5. https://doi.org/10.1109/AUTOTESTCON47462.2022.9984798.

Gaska, Thomas. 2012. “Optimizing an Incremental Modular Open System Approach (MOSA) in Avionics Systems for Balanced Architecture Decisions.” In 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), 7D1-1-7D1-19. https://doi.org/10.1109/DASC.2012.6382420.

Geier, Nadine. 2022a. “OUSD(R&E) MOSA Assessment Criteria.” Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)).

———. 2022b. “OUSD(R&E) Review of MOSA Tools and Practices.” Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)).

Heydari, Babak, Mohsen Mosleh, and Kia Dalili. 2016. “From Modular to Distributed Open Architectures: A Unified Decision Framework.” Systems Engineering 19 (3): 252–66. https://doi.org/10.1002/sys.21348.

Hinton, Mark, Gautam Vallabha, Chris Cooke, Christine Piatko, Michael Zeher, Peter Gayler, and Geoffrey Osier. 2018. “DEMONSTRATION OF A COMMON WORLD MODEL.”

Hoehne, Oliver. 2016. “The SoS-VEE Model: Mastering the Socio-Technical Aspects and Complexity of Systems of Systems Engineering (SoSE).” INCOSE International Symposium 26 (1): 1494–1508. https://doi.org/10.1002/j.2334-5837.2016.00241.x.

Loren, Jeff. 2004. “USAF Systems Engineering -- Revitalizing Fundamental Processes.” In USAF Developmental Test and Evaluation Summit. Woodland Hills, California: American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2004-6855.

Maier, Andrew, Brandon Vega, Hunter Powell, Lacey Swafford, Matthew Ward, and Michael Parrish. 2020. “Integrating Open Systems Architecture Models for Next Generation Cockpits: A Systems Engineering Approach.” New York.

“Modular Open Systems Approach: Considerations Impacting Both Acquirer and Supplier Adoption.” 2020. National Defense Industrial Association Systems Engineering Architecture Committee.

“Modular Open Systems Approach (MOSA) Reference Frameworks in Defense Acquisition Programs.” 2020. Office of the Under Secretary of Defense for Research and Engineering, Director of Defense Research and Engineering for Advanced Capabilities.

Morgan, Markeeva, Thomas Holzer, and Timothy Eveleigh. 2021. “Synergizing Model-Based Systems Engineering, Modularity, and Software Container Concepts to Manage Obsolescence.” Systems Engineering 24 (5): 369–80. https://doi.org/10.1002/sys.21591.

Naegle, Brad. 2007. “Developing Performance-Based Requirements for Open Architecture Design.” In 2007 IEEE International Conference on System of Systems Engineering, 1–6. https://doi.org/10.1109/SYSOSE.2007.4304230.

Ng, Huat, and Colby Stevens. 2019. “CAPABILITIES BASED TESTING OF DEFENSE MULTI- INTELLIGENCE SYSTEMS.”

“Program Managers Guide to Open Systems.” 2004. Open Systems Joint Task Force.

Rendon, Rene G. 2007. “Using a Modular Open Systems Approach in Defense Acquisitions: Implications for the Contracting Process.” In 2007 IEEE International Conference on System of Systems Engineering, 1–6. https://doi.org/10.1109/SYSOSE.2007.4304231.

———. 2008. “Analysis of Modular Open Systems Approach (MOSA) Implementation in Navy Acquisition Programs:” Fort Belvoir, VA: Defense Technical Information Center. https://doi.org/10.21236/ADA494230.

Rose, Leo J., Jonathan Shaver, Quinn Young, and Jacob Christensen. 2014. “Open Architecture Applied to Next-Generation Weapons.” In Open Architecture/Open Business Model Net-Centric Systems and Defense Transformation 2014, 9096:85–95. SPIE. https://doi.org/10.1117/12.2055266.

Sanders, Greg. 2022. “Open for Business: Business Models for Innovation with Modular Open Systems Approaches.” Technical Report. Acquisition Research Program. https://dair.nps.edu/handle/123456789/4609.

Schapiro, Seth B., and Matthew H. Henry. 2012. “Engineering Agile Systems through Architectural Modularity.” In 2012 IEEE International Systems Conference SysCon 2012, 1–6. https://doi.org/10.1109/SysCon.2012.6189529.

Shah, Parth Devang. 2021. “Modular Open Systems Approach (MOSA) and Its Impact on the US Defense Acquisition Methods and Programs.” D.Engr., United States -- District of Columbia: The George Washington University. https://www.proquest.com/docview/2471375336/abstract/298D769AE0874321PQ/1.

Shaver, Jonathan, Leo Rose, Quinn Young, and Jacob Christensen. 2016. “Challenges in the Application of Modular Open System Architecture to Weapons.” In Open Architecture/Open Business Model Net-Centric Systems and Defense Transformation 2016, 9849:66–72. SPIE. https://doi.org/10.1117/12.2229364.

Sims, Brendan. 2012. “Approaches to Open Technology Systems Specification,” May.

Smith, Montrell F. 2014. “Converting a Manned LCU Into an Unmanned Surface Vehicle (USV): An Open Systems Architecture (OSA) Case Study.”

Srivastava, Niraj, and Michael Rice. 2014. “Strategies for Competition Beyond Open Architecture (OA): Acquisition at the Edge of Chaos:” Fort Belvoir, VA: Defense Technical Information Center. https://doi.org/10.21236/ADA624741.

Stirgwolt, Benjamin W., Thomas A. Mazzuchi, and Shahram Sarkani. 2022. “A Model-Based Systems Engineering Approach for Developing Modular System Architectures.” Journal of Engineering Design 33 (2): 95–119. https://doi.org/10.1080/09544828.2021.1980203.

“Systems Engineering Guidebook.” 2022. Office of the Under Secretary of Defense for Research and Engineering.

Thornton, Adam, Brandon Mieners, Donald Poole, and Mark Russell. 2020. “Latent Dirichlet Allocation (LDA) for Anomaly Detection in Ground Vehicle Network Traffic.”

VanderLeest, Steven H. 2017. “Designing a Future Airborne Capability Environment (FACE) Hypervisor for Safety and Security.” In 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), 1–9. https://doi.org/10.1109/DASC.2017.8102056.

Wilson, C. Barry (Charles Barry). 2017. “Improving Value of Strategic Defense Systems Using Modular Open Architecture.” Thesis, Massachusetts Institute of Technology. https://dspace.mit.edu/handle/1721.1/111233.

Zimmerman, Philomena, Monique Ofori, Donald Barrett, Joseph Soler, and Anthony Harriman. 2019. “Considerations and Examples of a Modular Open Systems Approach in Defense Systems.” The Journal of Defense Modeling and Simulation 16 (4): 373–88. https://doi.org/10.1177/1548512917751281.

PART 5: SEMOD POLICY AND GUIDANCE

The direct policy derivation from the focus areas to eventually SE Modernization guidance is shown in Figure 46. Policy Derivation to SE Guidance. The four focus areas each derive from direction published in different annual National Defense Authorization Acts (NDAA). Of the four focus areas, only MOSA requirements have been codified in Title 10 of the US Code, they others have been taken directly into various acquisition policies and guides. They derive authority through DoD Directives 5000.01 and 5000.02 and are applied through DOD Instruction 5000.88 Engineering of Defense Systems (last release November 2020). From there the derivation to acquisition process is through each of the acquisition pathways, and to systems engineering process in the DoD Systems Engineering Guide.

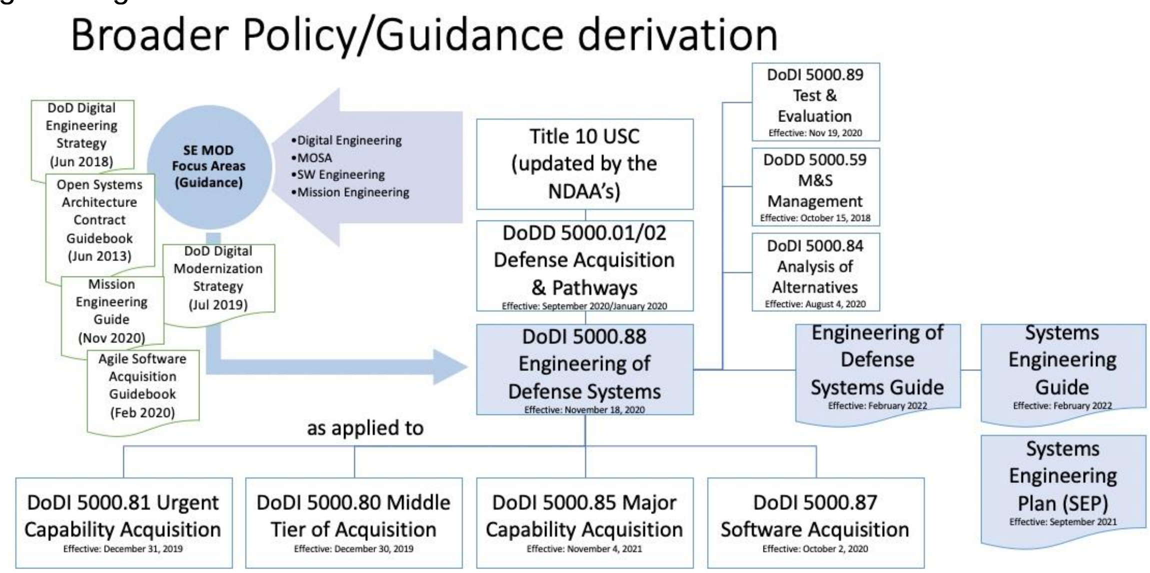


Figure 46. Policy Derivation to SE Guidance.

Based on the lack of a clear integration framework in the policy and guidance, the research team conducted a policy analysis. The initial results were presented in project WRT-1051, the final results were completed on this project. The policy analysis reviewed existing SE policy, identified major gaps, policy flow, and aligned and integrated specific acquisition pathways to develop recommended modifications. Each document was analyzed for cross-references between the documents, including guides and policies. Select DoD 5000 policies and suggested updates according to initial gaps which the team identified and inclusion of SE Modernization focal areas (Digital Engineering, SW-Agile/DevSecOps, MOSA, Mission Engineering). The following policies & guides were reviewed:

1. DoDI 5000.88 "Engineering of Defense Systems" (November 2020)
2. DoDI 5000.85 "Major Capability Acquisition" (August 2020)
3. DoDI 5000.81 "Urgent Capability Acquisition" (December 2019)
4. DoDI 5000.80 "Operation of the Middle Tier of Acquisition" (December 2019)
5. DoDI 5000.87 "Operation of the Software Acquisition Pathway" (October 2020)

6. DoDD 5000.01 and 5000.02 "The Defense Acquisition System"
7. DoDI 5000.84 Analysis of Alternatives (August 2020)
8. DoDI 5000.89 Test & Evaluation (November 2020)
9. DoDI 5000.95 Human Syst. Integration (August 2022)
10. Systems Engineering Plan (September 2021)
11. Systems Engineering Guidebook (February 2022)
12. Engineering of Defense Systems Guidebook (February 2022)
13. DoD Data Strategy (September 2020)

In addition, a number of other policy and guidance documents at the defense and service levels were reviewed for completeness. The current view and the modernized view of systems engineering are not fundamentally different in principles but are undergoing significant change in practice. In our research we found there have been many new practices applied to individual disciplinary approaches to systems engineering but little reintegration into the overall practice.

In review, Congress passed a series of legislative actions through the annual NDAA that target improvements in acquisition execution but also directly focus in on systems engineering. In each of the focus areas the research team was able to derive statements of intent from the policy language that are relevant to systems engineering. These statements of intent are highlighted below in yellow.

- **Mission Engineering (ME)** - The NDAA for Fiscal Year 2017, Section 855, directed DoD to establish Mission Integration Management (MIM) as a core activity within the acquisition, engineering, and operational communities to focus on the integration of elements that are all centered around the mission. ME is the deliberate planning, analyzing, organizing, and integrating of current and emerging operational and system capabilities to achieve desired warfighting mission effects. ME is the technical sub-element of MIM as a means to provide engineered mission-based outputs to the requirements process, guide prototypes, provide design options, and inform investment decisions. Primary guidance for ME is published in the OUSD(RE) Mission Engineering Guide, November 2020.

- **Digital Engineering (DE)** - The NDAA for Fiscal Year 2020, Section 231, directed The Secretary of Defense to establish a digital engineering capability to be used: (A) for the development and deployment of digital engineering models for use in the defense acquisition process; and (B) to provide testing infrastructure and software to support automated approaches for testing, evaluation, and deployment throughout the defense acquisition process. The language additionally stated that the DE capability will provide for the development, validation, use, curation, and maintenance of technically accurate digital systems, models of systems, subsystems, and their components, at the appropriate level of fidelity to ensure that test activities adequately simulate the environment in which a system will be deployed. Primary guidance for DE is published in the DoD Digital Engineering Strategy, June 2018. The DoD DE Strategy defines digital engineering as "an integrated digital approach that uses authoritative sources of system data and models as a continuum across disciplines to support lifecycle activities from concept through disposal."

- **Modular Open Systems Approach (MOSA)** – Unlike the other guidance, MOSA requirements for acquisition programs have been codified into Title 10. Title 10 U.S.C. 2446a.(b), Sec 805 states all major defense acquisition programs (MDAP) are to be designed and developed using a MOSA. Title 10 U.S.C 2320(e) requires ACAT I and II Program Managers to assess the IPR and data rights requirements of their program, create a Technical Data Management Strategy and take steps to secure the Government's appropriate rights consistent with the FAR and DFARS. The code additionally states that "A mandate of OSA is that technical requirements be based to the maximum extent practicable on open standards. Where there are no standards, the OSA methodology creates them. At a minimum, technical standards and related specifications, requirements, source code, metadata, interface control documents (ICDs), and any other implementation and design artifacts that are necessary for a qualified contractor to successfully perform development or maintenance work for the Government are made available throughout the life cycle." MOSA guidance was initially published in the Open Systems Architecture Contract Guidebook, June 2013, and is being updated as of this report.

  With respect to the SE Modernization integration framework, the intent of MOSA policy needs some interpretation. When developing the integration framework, we used the more general intent of MOSA from software and systems literature: to use modular design, control interfaces, adopt open standards, and measure conformance. This centers the goal of MOSA in SE as both a mandate and an enabler to manage adaptability and change.

- **Software Agile and DevOps** - The NDAA for Fiscal Year 2018, Sections 873/874, directed Pilot Program to Use Agile or Iterative Development Methods to Tailor Major Software-Intensive Warfighting Systems. The NDAA for Fiscal Year 2019, Section 868, directed the DoD to commence implementation of each recommendation submitted as part of the final report of the Defense Science Board Task Force on the Design and Acquisition of Software for Defense Systems. The NDAA for Fiscal Year 2020, Section 800, established the Software Acquisition (SWA) Pathway. Primary guidance is provided in the Agile Software Acquisition Guidebook, February 2020. The intent of SWA with respect to SE Modernization can be found int this guide: "Defining the capability need: Agile approaches to software avoid the need for very detailed upfront, predictive requirements capture. That is, they dispense with the idea that through sufficiently rigorous analysis, all of a system's requirements can be determined and specified upfront. In contrast, Agile approaches begin with a high-level capture of business and technical needs that provides enough information to define the software solution space, while also considering associated quality needs (such as security)."

This last statement summarizes the mental model challenges with current versus modernized SE very succinctly: all stakeholder requirements determined up front versus determine stakeholder needs sufficient to define the solution space. Both approaches remain relevant to SE rigor but there is little integration between the two (at least in acquisition processes).

Several other focus areas are relevant to SE Modernization and defined in policy and guidance but are not called out in legislative activities. These include:

- **DoD Data strategy** – DoD acquisition is pursuing a broader digital data strategy as defined in the DoD Data Strategy, September 2020. SE is generally viewed as an engineering and technical discipline but has always been strongly integrated with Program Management activities as well as Enterprise Management. In development of the Integration Framework, we found that several areas of the Data Strategy remain significant pain points with respect to SE Modernization: data as a strategic asset, collective data stewardship, data collection, enterprise-wide data access and availability, data fit for purpose, and design for compliance. In particular, at this point the SE community may be overly focused on "System Models" and underly focused on "System Data." Data architecture, data standards, data governance, and talent and culture are all essential components of SE Modernization but are new concepts to systems engineers.

- **Modeling and Simulation (M&S)** – System models are a combination of descriptive models (requirements, architecture) and computational models (physics, behavior, operations, etc.). In DoD acquisition, much of the descriptive modeling in the past has resulted in documents not models. The integration of descriptive models and computational models is the focus of much of the DoD DE and Model-Based Systems Engineering (MBSE) initiatives.

- **Test and Evaluation (T&E)** – T&E methods and processes will follow a similar transformation using authoritative sources of data and models.

- **Human Systems Integration (HSI**) – Technologies related to autonomous systems and human-machine teaming will evolve the HSI and SE disciplines to be much more integrated.

- **Capability Integration** – The processes to move from ME into the Joint Capabilities Integration and Development System (JCIDS) and then into program requirements and acquisition strategies will also evolve through the integration of authoritative data and models.

- **Sustainment and support** – SE Modernization appears to be evolving separately in the acquisition program development and the program sustainment communities.

The common modernization driver in all of these focus areas, as discussed in the integration framework, **is seamless and efficient transfer of data and models from underlying performance drivers through models to decisions, as well as ease of drilling back down from decisions to data**. This does not mean everything must be connected (that is unlikely to ever happen) but that the process to move up and down the data transformation space is efficient and produces better quality. With this mental model of improved access and flow, a common integration framework can be pursued. Without it, stove-piping of people, processes and tools across lifecycle stages will continue to occur. The purpose of SE Modernization is thus to support more seamless and efficient digital integration of data and models across all program management, engineering, and acquisition process areas. We found this intent to be generally lacking in the current policy and guidance.

Major Policy Gap Areas

The policies were reviewed against the following gaps that the team identified:

1. Much of the policy remains milestone driven. As noted in the Supra-system model, milestone processes and approvers specific to each acquisition pathway are not well defined with respect to continuous processes in a digital environment. In particular most of the engineering guidance continues to use language that is associated with the MCA pathway, with little detail on use in other pathways. **The details of government defined decision milestones and milestone review process should be defined clearly in each AAF pathway**.
2. Application of modernized SE to legacy systems is not well-covered in policy but most of today's implementation examples are legacy systems. This is highlighted in the Supra-system model of Figure 14 as the set of acquisition activities that are derived from the “measure” side of the learn->build->measure set of lifecycles. **Lack of feedback from operations and sustainment back into the next phase of acquisition his makes formal collection of lessons learned difficult.**
3. The breadth/generality of policy at DoD level creates inconsistent flow down to service level. This is intentional to allow flexibility and tailoring in service level guidance but at least some level of compliance needs to be specified to create momentum for adoption of SEMOD activities. As noted in the pain points discussion, effective compliance measures are needed to enforce adoption at program levels and to build momentum for change. The need for services to define these compliance measures at least should be reflected in DoD-level policy.
4. There is an inconsistent level of descriptive detail across documents by focus area that creates confusion. There is also varying sets of terminology and jargon used in different policies and guides that makes integration difficult. This is a general noted gap in our review of the documents. **The ontology effort we conducted in this project took a much deeper look at language consistency across policy areas.**
5. The SE community lacks a desk reference that describes modernization of SE process and focus areas that services can follow prescriptively. This would naturally be the DoD Engineering of Defense Systems Guide and related DoDI 5000.88 policy, which should be evolved over time to capture the core SEMOD concepts. This is noted as a need from interviews and discussions with DoD programs and can be considered as an indication the services would like more prescriptive guidance at the DoD level. **As an exercise, we developed a version of the DoD Engineering of Defense Systems Guide using more of a “how-to” format.**

Policy is considered a statement of intent and is implemented as a procedure or protocol. As such, the policies were additionally reviewed to identify gaps in expression of intent and recommendations for future changes. The highlighted intent statements previously noted for each focus area were used to guide this review. Policy is undefined in the DAU glossary but has a useful clarifying descriptive passage in the DoD Dictionary: "Policy directs and assigns tasks, prescribes desired capabilities, and provides guidance for ensuring the Armed Forces of the United States are prepared to perform their assigned roles. Implicitly, policy can create new roles and requirements for new capabilities." Whether or not the policy clearly articulated intent was a central question. There were four guiding principles when assessing this articulation of **intent**: (1) Provides a clear and concise expression of the purpose of the policy, (2) Provides the desired military & acquisition end state that supports decentralized decision making, (3) Provides focus to the staff, and (4) Helps subordinate and supporting Decision Authorities act to achieve the policy authority's desired results without further instruction, even when the acquisition program does not unfold as initially planned.

We found the documents do not yet communicate the intent of incorporating SE Modernization processes across the policy areas and they should be updated accordingly. When reviewing the clarity of the policy derivation, the team found that the four focus areas derive from various language in NDAA's, but only MOSA has specific referenceable standardized language in Title 10. As such, appropriate standards should be specified in directives and instruction for the other focus areas. Additional recommendations are as follows:

- Establish clear traceability between policy documents and guidance via appropriate cross-referencing
- Identify appropriate standards (to be developed if necessary) to make policy compliance measurable
- Terms used in DoDD and DoDI lack clear, concise and complete definitions. There needs to be a clear taxonomy/ontology developed at least for SE and related SE Modernization activities.
- Systems Engineering is a core technical definition and risk management approach to all Acquisition Pathways. This is reflected in the language of DODD 5000.01 if not in specific directives. There need to be consistent guidance language in each pathway provided and clear intent provided in Engineering of Defense Systems and Systems Engineering guidance reflecting use in each pathway and in sustainment. This will be further assessed in the completion of the policy analysis task.

Following on our review of the DoD 5000 policy documents and Systems Engineering Guide and Plan, the team identified several gaps in the SE Modernization focal areas and in the overall organization of the documents. Some of the gaps identified, such as inconsistent level of descriptive detail, milestone driven and lack of a desk reference that describes the new SE process that services can follow prescriptively, have contributed to a cadre of information that is disjointed and at times difficult to digest. Overall, current policy and guidance have terminology that is independent from each other and jargon across each focus area and acquisition pathway. With this in mind, it was determined that some of the resource documents, such as the Engineering of Defense Systems Guidebook, would benefit from an update which would reorganize the information and content to a more user-friendly and digestible format. This reorganization would improve the flow of information to ensure readers can effectively use the Guide as a resource while also creating consistency and fluidity with collection and presentation of the information. Ultimately, this would support improved planning and activities. By rewriting the Engineering of Defense Systems Guidebook, the team is able to provide a concrete corrective action that supports updating and revising the guides as an example for more effective implementation. The reorganized document is attached to this report as a separate document.

The document reorganization was modeled after the DoD Other Transactions Authority (OTA) Guide. The USD A&S Other Transactions Guide provides conditions for use as well as guidance for planning and executing Research and Prototype Other Transactions (OTs) as well as follow-on Production OTs. This guide provides advice and lessons learned on the planning, publicizing, soliciting, evaluating, negotiation, award, and administration of OTs, to include all three types of OT agreements: Research, Prototype, and Production. While the OTA Guide includes references to the controlling statutory and policy provisions for DoD OT authority, the document itself is not a formal policy document. The Engineering of Defense Systems Guide has a similar purpose.

The OTA guide is separated into the following sections: Overview; Planning; Publicizing, Soliciting and Evaluating; Administration. It also includes Appendices with a Glossary; OT Authority History; Common OT Myths and Facts; Additional Info, Resources, Policies; and IP Considerations. The OTA Guide also has case studies within the document. The OTA guide is organized similar to a How-To document. Overall, this organization of information that the OTA Guide provides is a clear and easy-to-follow format for the reader. This ultimately makes it a useful reference and model for our effort.

Rewriting the Engineering of Defense System Guidebook to model it after the OTA Guide was done as an example to reduce the disconnection of activities that is present in the current version, while providing lessons learned in a new Case Studies section, ultimately providing clear instruction on how to carry out activities. In our reorganized version of the Engineering of Defense Systems Guidebook, the team created a new upfront section titled Large Concepts in Acquisitions. Having the large concepts upfront supports consolidation of information and provides a high-level view but also reduces redundancy that was present in the original document. Other new addition sections such as Lifecycle Phases – MCA Pathway also streamline information and simplify content making pertinent information more accessible. Including additional appendices and sections, such as a Glossary, Resources and Policies, which exist in the OTA Guide but are currently absent from the Engineering Guidebook, will create more ontology agreement and streamline information in this space.

The OT Guide provides a succinct but thorough definitions section. What makes it unique and particularly helpful is that it is not too lengthy but still provides the reader with useful guidance on language and meaning. There is a clear and logical pattern made on what should and should not be included (for example, relevant actors/stakeholders mentioned, software tools, or more “trendy” terms like hackathon and rodeos). The OT guide also uses very simple and easy to understand language. It goes without saying that reading the definition of an unfamiliar word should not prompt readers towards other glossaries or resources in order to understand your definition. Ultimately, when preparing definitions, the when and why of including them should be agreed upon. When rewriting and reorganizing the Engineering Guidebook, additional policy definitions and references might be useful beyond engineering and defense terminology along with further ontology development in this space.

Please refer to Supporting document provided separately for the rewritten example guide.

The next section of the report discusses the research, progress, and next steps related language consistency. In this project we found there were not consistent definitions across the disciplines of military operations, acquisition, and engineering for even widely used terms like “system” and “capability.” This research began the process of defining a formal digital ontology that links together military operations, acquisition, and systems engineering.

PART 6: ONTOLOGIES: TOWARDS AN SEMOD INFORMATION GRAPH

This report derives from two interrelated tasks in the SE Modernization project. The first is the policy analysis and recommendations discussed separately, which noted that there is an inconsistent level of descriptive detail across policy and guidance documents by focus area that creates confusion. There is also varying sets of terminology and jargon used in different policies and guides that makes integration difficult. This is a general noted gap in our review of the documents. The policy review recommended an ontology effort be conducted to identify the more specific recommendations for language consistency across policy and guidance areas. The second is to derive the body of knowledge including investigations of appropriate taxonomy and definitions as well as an initial ontology and set of metadata guidance.

It is worth noting that there is today no published ontology for either systems engineering activities or acquisition activities. As a result there is no ontological basis for today's systems engineering and for acquisition standards and guidance, much less a linking between the two. This work reflects an initial set of research to derive this ontological basis, using standard published ontologies.

DERIVING A FORMAL DOMAIN ONTOLOGY FOR SYSTEMS ENGINEERING AND ACQUISITION GROUPS

One of the Systems Engineering Modernization (SEMOD) pain points is that project managers lack tools and methods for acquisition process integration due to inconsistent terminology. SERC WRT-1058 research completed text analysis of DoD guidelines contents and results showed current DoD guidelines address only a small part of the SE modernization opportunities. Furthermore, initial review of the DoD documents revealed that terms used in DoD guidelines lack clear, concise, and complete definitions. DoD 5000.88 policy states that organizations should "collaboratively perform", and "make data and artifacts available" to increase collaboration and knowledge sharing. However, there are significant gaps in use of terms across DoD instructions and guidelines and all current publications include extensive use of technical jargon. These inconsistencies create the well-known "Tower of Babel" problem where different organizations and groups have their own idiosyncratic terms and concepts by which they represent information they receive. Leveraging ontologies to derive knowledge across different groups including systems engineering, acquisition, and military doctrine would facilitate knowledge integration across different organizational groups that aim to "collaboratively perform" and share data. As data and data models are becoming the core of SE Modernization activities, the need for clear taxonomy and ontologies becomes important. As part of the SERC WRT-1058 research, this paper outlines the background literature and methodology that would establish a decision support system that evaluates the similarity of idiosyncratic representation of concepts generated by different groups. The methodology compares these representations of different groups to a master ontological representation to identify variations and similarities among guidance documents and standards. The methodology has several implications from an application point of view including:

- Facilitate deriving a formal domain ontology of systems engineering modernization and related terms for the systems engineering and acquisition groups
- Improve communication among different groups including acquisition and operational groups as well as systems engineering and acquisition groups
- Automatically analyze gaps in policies and related documents.

Deriving a formal domain ontology for systems engineering and acquisition groups will normalize terminology for human communication as well as support data exchange among various information systems utilized across the acquisition and design groups. Furthermore, it will facilitate automatic analysis of policies and related documents to identify variances and gaps in these documents. These will ultimately facilitate effective transition to digital engineering across acquisition organizations.

The rest of the document is organized as follows. Section 2 summarizes the knowledge sources, lexicons used in the study. Section 3 provides background information about ontologies and reviews some of the related formal ontologies. Section 4 describes the overall methodology and related technologies utilized. Details of the knowledge discovery framework are described in this section as well. Section 5 provides an example of mapping a definition shared across systems engineering discipline and acquisition groups to the Common Core Ontology (CCO) ontology. Section 6 provides an example of comparison of terms across various guidelines and the 5000.88 policy. Finally, Section 7 discusses the challenges and open issues as well as future work linking the study to the SE Modernization Roadmap paths.

KNOWLEDGE SOURCES: LEXICONS

Organizations capture the knowledge areas represented and lexicons used by different groups in various documents. In this study, following standards, guidance documents, and policies are used to analyze similarity of terminology across systems engineering, DoD acquisition, and military doctrines:

- **ISO/IEC/IEEE DIS 15288, Systems and software engineering-System life cycle processes** [13]: The standard describes the processes for the life cycle of engineered systems to facilitate communication among acquirers, suppliers, and other related stakeholders in the life cycle of a system. The document is written from an engineering viewpoint and uses engineering terminology and is applicable for a wide range of engineered systems comprised of hardware elements, software elements, data, humans, processes, services, materials, facilities, and naturally occurring entities [13]. The common process framework described in ISO/IEC/IEEE 15288 adopts a systems engineering approach which is an integrative approach that uses systems science principles, technical and management methods for realization, use, and retirement of engineering systems. The framework presented in the standard can be tailored by an organization, by a project, by an acquirer and supplier, and by process assessors for various purposes.

- **DoD Systems Engineering Guidebook** [7]: The guidebook provides systems engineering guidance and best practices for defense acquisition programs in any of the DoD Adaptive Acquisition Framework pathways. The guidebook is an interim document until DoD publishes the Systems Engineering Modernization policy and guidance. The guidebook can be used by acquisition program managers and system engineers to plan and implement program systems engineering activities across the life cycle of systems, and system of systems.
- **DoD Engineering of Defense Systems Guidebook** [8]: The guidebook provides the activities, processes, and best practices for development of DoD systems and mainly focuses on recommended engineering best practices for the DoD Adaptive Acquisition Framework pathways. The guidebook aligns with the DoD 5000.88 policy and covers engineering disciplines including systems engineering, software engineering, specialty engineering, modular open systems approach, digital engineering, systems security, and technical reviews and assessments.
- **DoD 5000.88, Engineering of Defense Systems policy** [9]: The document provides engineering technical policy in development of DoD systems. The policy covers systems engineering, and other engineering disciplines as well as engineering management approach to guide all technical activities of the development program.
- **AR 70-1, Army Acquisition Policy** [1]: The policy covers research, development, acquisition, and lifecycle management of Army material solutions for warfighting capabilities. The policy implements DoDD 5000.1 and DoDI 5000.2 for Army acquisition programs and applies to all Army areas including the Active Army, The Army National Guard/Army National Guard of the United States, and the US Army Reserve. Systems engineering, development test and evaluation, operational test and evaluation, life cycle sustainment, human system integration, affordability, analysis of alternatives, cost estimation, and other Army acquisition related aspects are covered under this policy.
- **AR 71-9, Army Warfighting Capability Determination** [2]: The policy covers procedures, and responsibilities for determining the required capabilities for warfighting and applies to all Army areas including the US Army National Guard/Army National Guard of the United States, and the US Army Reserve. Requirement elicitation forums, capability documentation, analysis of requirements and capability determination, capability integration process and other capability related activities are covered under this policy.
- **Defense Acquisition University (DAU) glossary** [10]: The document contains acronyms, abbreviations, terms, and definitions used in the systems acquisition process within the DoD. While most of the terms are generic, the glossary includes service specific terms as well.

ONTOLOGIES

Ontology is a representational artifact comprised of terms and relationships between them [3]. The representation of relationships among terms differentiates ontologies from terminologies that only contain a list of lexical entries and descriptions [3]. Ontologies aim to provide an unambiguous description of the concepts and relationships that can exist for an agent or a community of agent, so they can understand, share, and use this description to accomplish some tasks on behalf of users. Having data from different sources and formats mapped into an ontology, helps with processing comparing, and communicating across pervasive knowledge infrastructures [11], [12], [23]. An ontology should be:

- Formal: An ontology should be machine-readable. Web Ontology Language (OWL) is a well-established technology for formal knowledge representation where ontologies can be automatically processed, and inconsistencies can be identified within the model [3].
- Explicit: The types of concepts used, and the constraints on their use should be explicitly defined.
- Shared: An ontology captures consensual knowledge that is accepted by a group and should not be private to some individual.

Ontology design is a cross-disciplinary field with historical roots in philosophy, linguistics, computer science, and cognitive science. Ontologies conceptualize an abstract model of some phenomenon in the real world and identify the relevant concepts of that phenomenon. Ontological commitment on the meaning of vocabulary used to share knowledge is important for effective use of ontologies. This study focuses on identifying the shared knowledge across three groups including DoD acquisition, systems engineering, and the military doctrines. Figure 47 summarizes the key focus knowledge areas of interest for this study.

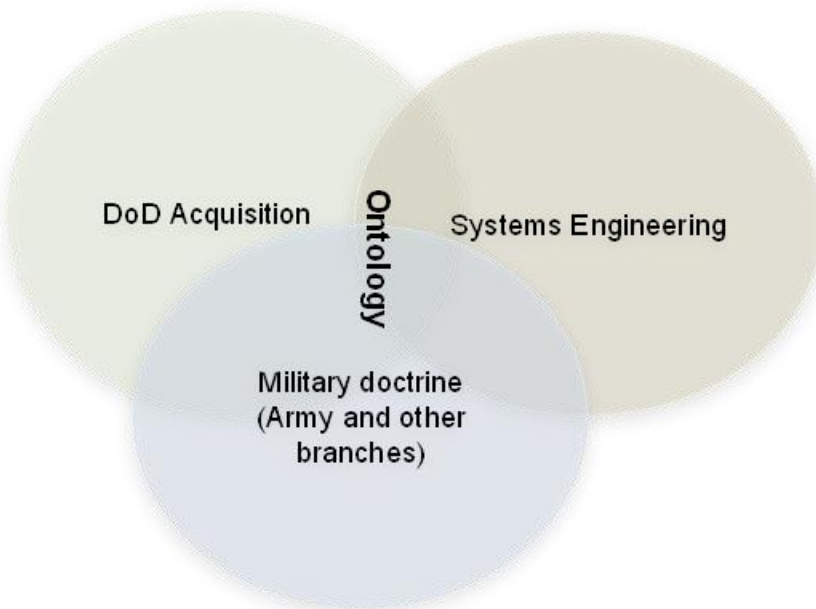


Figure 47. Target knowledge areas of interest.

Acquisition, and military doctrines are intertwined with the knowledge areas in systems engineering discipline. Systematic literature review on ontology-based systems engineering research [25] identified studies that utilize ontologies to support various systems engineering knowledge areas including systems fundamentals, representing systems with models, various engineering system contexts, systems engineering standards, generic life cycle stages, and systems engineering management. The systematic literature review study revealed the need for adoption of formal ontologies for systems engineering discipline as systems engineering is transitioning into a model-based discipline. On the military doctrine side, Joint Doctrine Ontology (JDO) for military information systems interoperability is described in [20]. The JDO is based on the upper-level ontology, Basic Formal Ontology (BFO) which is a formal ontology widely used to create and integrate ontologies in the biomedical domain [3]. The JDO is also a domain-level extension of Common Core Ontology (CCO), a mid-level extension of BFO. CCO is comprised of eleven mid-level ontologies including [21]:

- Information Entity Ontology
- Agent Ontology
- Quality Ontology
- Event Ontology
- Artifact Ontology
- Time Ontology
- Geospatial Ontology
- Units of Measure Ontology
- Currency Unit Ontology
- Extended Relation Ontology
- Model Relation Ontology

The CCO has been used widely by U.S government sponsored projects due to its strength in providing semantics for concepts and relations that are applicable in a wide range of domains without constraining these ontologies to include concepts outside of their specific domain. Some of the CCO derived domain ontologies include aircraft ontology, Airforce aircraft maintenance ontology, Army universal task list ontology, Cyber ontology, Marine Corps Task List ontology, Military operations ontology, mission planning ontology, Undersea warfare ontology [6].

KNOWLEDGE DISCOVERY FRAMEWORK

One of the advantages of increasing intra-department collaboration and cooperation among DoD organizations is to support, promote, and facilitate the overall national objectives and agility required from DoD. This can be achieved only through a shared knowledge base across its departments as well as DoD-industry collaboration, especially given the current pace of technological change and the rapidly evolving threats. In this study, a decision support knowledge discovery framework is described with the purpose of facilitating communication across different organizations to achieve the specific business and operational needs of the DoD while reusing the existing knowledge base that DoD possesses [14].

The knowledge discovery framework acquires abstractions of essential information from the underlying complex documents, present knowledge in a way that maintains coherence and consistency across different parties. The overview of the knowledge discovery framework is shown in Figure 48. In this framework, sources of knowledge across different domains including acquisition, systems engineering, and military doctrine (summarized in Section 2) are converted into knowledge representations. These knowledge representations are then mapped to a formal knowledge base for inferencing, gap analysis, explanation, and justification. The formal knowledge base used in this framework is the CCO which has been widely used as a formal mid-level ontology for driving domain ontologies. Inferencing, explanation, and other analysis methods of knowledge representations ultimately support decision makers in terms of validating knowledge and promotes discussion among different groups in terms of reaching a consensus on the knowledge areas that are shared across these groups.

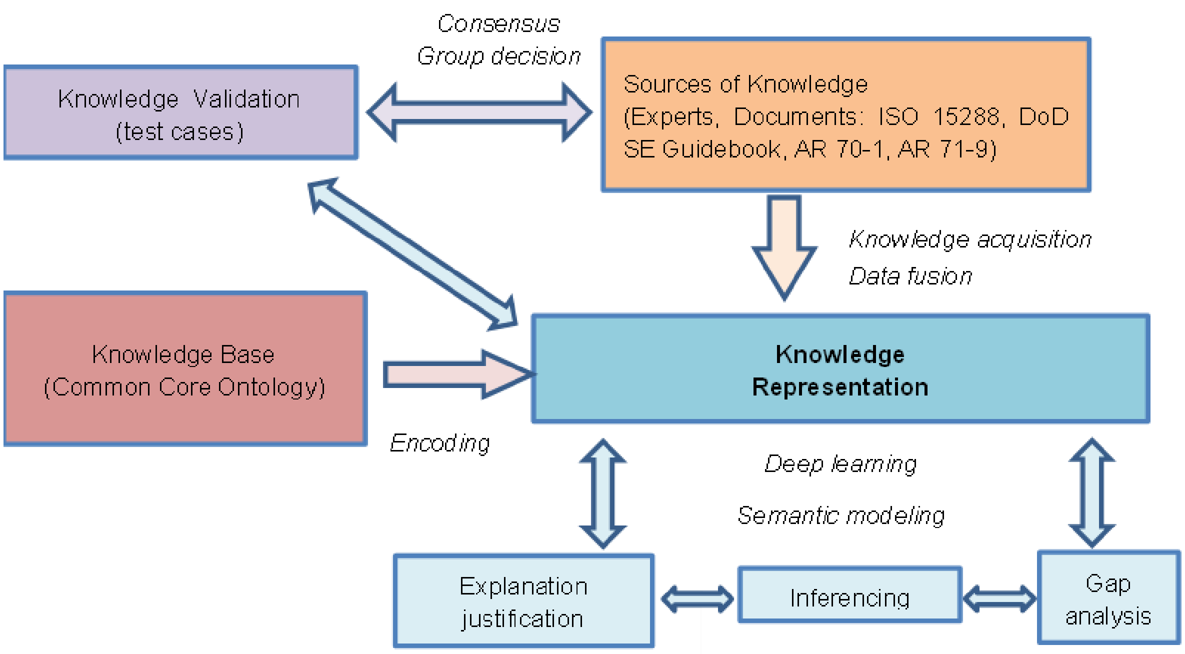


Figure 48. Overview of the knowledge discovery framework.

One of the key components of this framework is the reuse of existing knowledge within DoD. These include:

- Ontologies are one of the most successful ways of representing actionable knowledge in domain specific domains because they can successfully capture knowledge in a formal manner that can be shared across different parts of the organization. However, evolving ontologies are non-trivial tasks that can only be addressed using human in the loop methods. In this approach, DoD experts should have the ultimate decisions on the ontological structure used to depict the shared knowledge. In this framework, CCO is used as a critical knowledge base due to its formal structure and wide range of utility across various DoD organizations [21].

- DoD maintains a very strong repository of specific terminologies. For example, the Dictionary of Military and Associated Terms describes general and universal terms in joint publication glossaries. Terminology Repository for DoD provides awareness on specific and technical terms and definitions that reside outside the DoD Dictionary (universal and general terms). The Terminology Repository describes terms in unclassified and correctly marked issuances and glossaries [15].

Having these key components, the framework uses natural language processing combined with deep learning techniques to develop knowledge representations (semantic maps of documents) and subsequently map them into knowledge bases (ontologies) to present them in a more coherent and consistent fashion [16]. The specific techniques used for the knowledge discovery are shown in Figure 49.

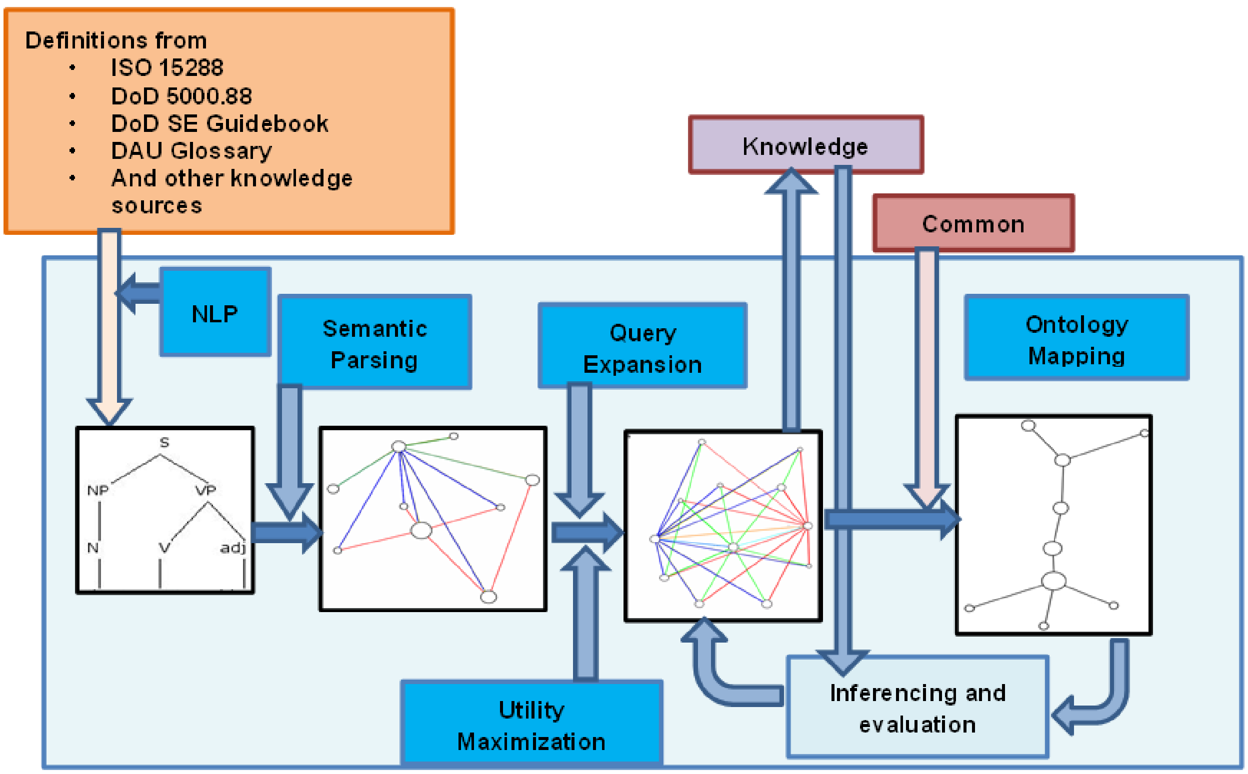


Figure 49. Knowledge discovery techniques.

Semantic maps/networks are structured networks in which concepts (nodes) are connected to semantically similar concepts by edges. Pathfinder network is a method for constructing semantic networks using a distance metric that measures the similarity between any pair of items in the data set. Studies indicate that these networks are valid and reliable measures of science content knowledge [22] and that they are effective in capturing the authentic descriptions/representations of the content in a domain [19].

Furthermore, computer-derived semantic association scores were shown to be significantly correlated with the human derived representations [4]. In this framework, semantic maps/networks are used to convert definitions into knowledge representations. Semantic networks for the term “systems engineering” defined in the DoD Systems Engineering Guidebook, and the ISO 15288 standard are shown in Figure 50. In these networks, the size of the node implies the frequency of usage, and size of the edge implies the strength of connection.

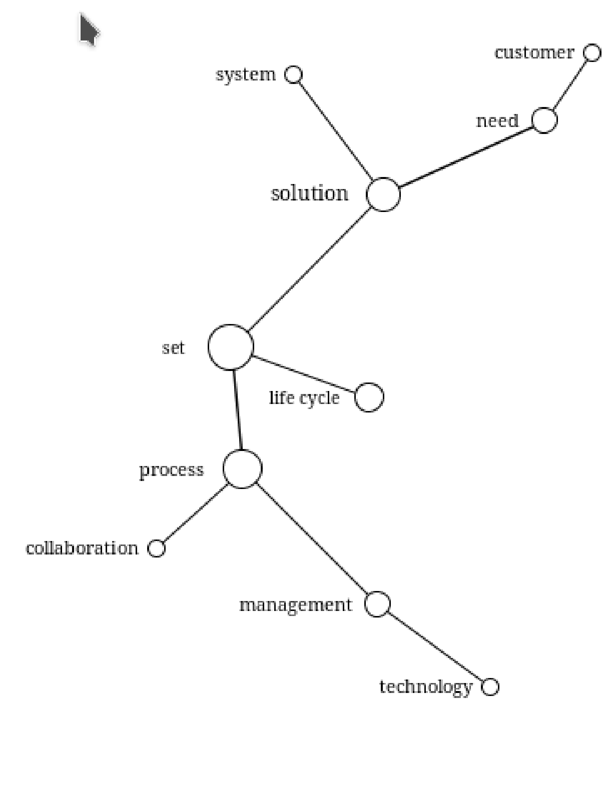
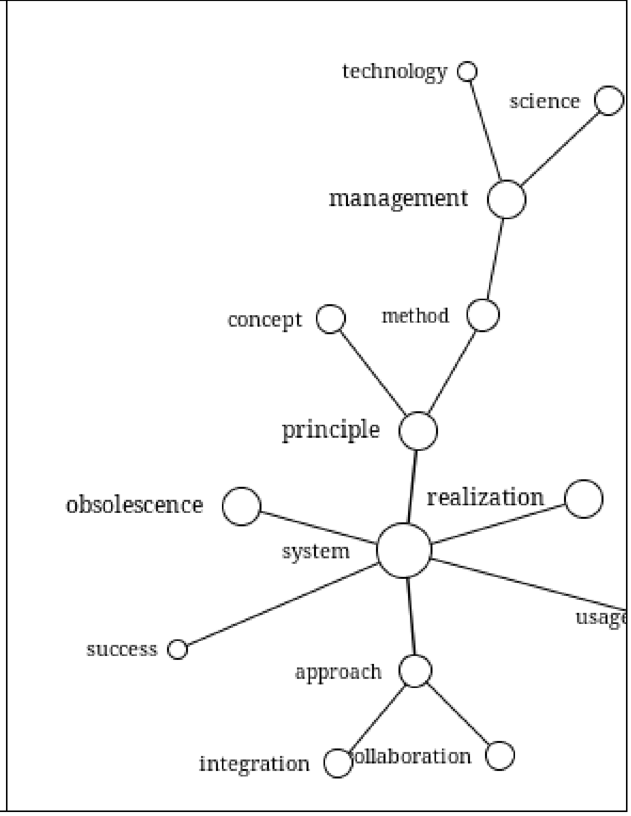
<table>
<tr><th>Systems Engineering (DOD SE Guidebook)</th><th>Systems Engineering (ISO 15288)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Figure 50. Semantic networks for the term “systems engineering” defined in DoD SE Guidebook, and ISO 15288.

Definitions and terms listed in the knowledge resources are processed using natural language techniques such as lemma extraction, coreference identification, name entity recognition, composite words identification [17]. This process stresses out the identification of glossary terms that are relevant to the parts that are exchanging the document. Next, the dependency tree is used to generate a semantic network having the form of a set of subject-action-object triplets. Each word in a triplet is transformed to its canonical form.

To remove idiosyncratic speech that might be included in text, the semantic network is expanded by including relevant terms with similar meaning. For this, deep learning techniques that are trained on domain specific bodies of knowledge are utilized.

The next step is to map each generated term in the semantic network into the existing set of ontologies used by DoD. This would give the capability to vary the granularity of the semantic maps from more abstract to more specialized versions to determine similarities and differences among terms used in documents generated by different organizations. Following section describes the mapping of an example definition to the CCO.

MAPPING OF TERMS TO COMMON CORE ONTOLOGY

To demonstrate the application of the knowledge discovery framework in Section IV, the term "system" defined in the ISO 15288 standard, and in the DoD Systems Engineering guideline is used. System is defined in the ISO 15288 standard as "an arrangement of parts or elements that together exhibit a stated behavior or meaning that the individual constituents do not" [13]. The same term is defined in the DoD Systems Engineering guideline as "A functionally, physically, and behaviorally related group of regularly interacting or interdependent elements" [7]. The dependency tree is used to generate a semantic network of the definitions. A comparison of the semantic networks for the term "system" is shown in Figure 51.

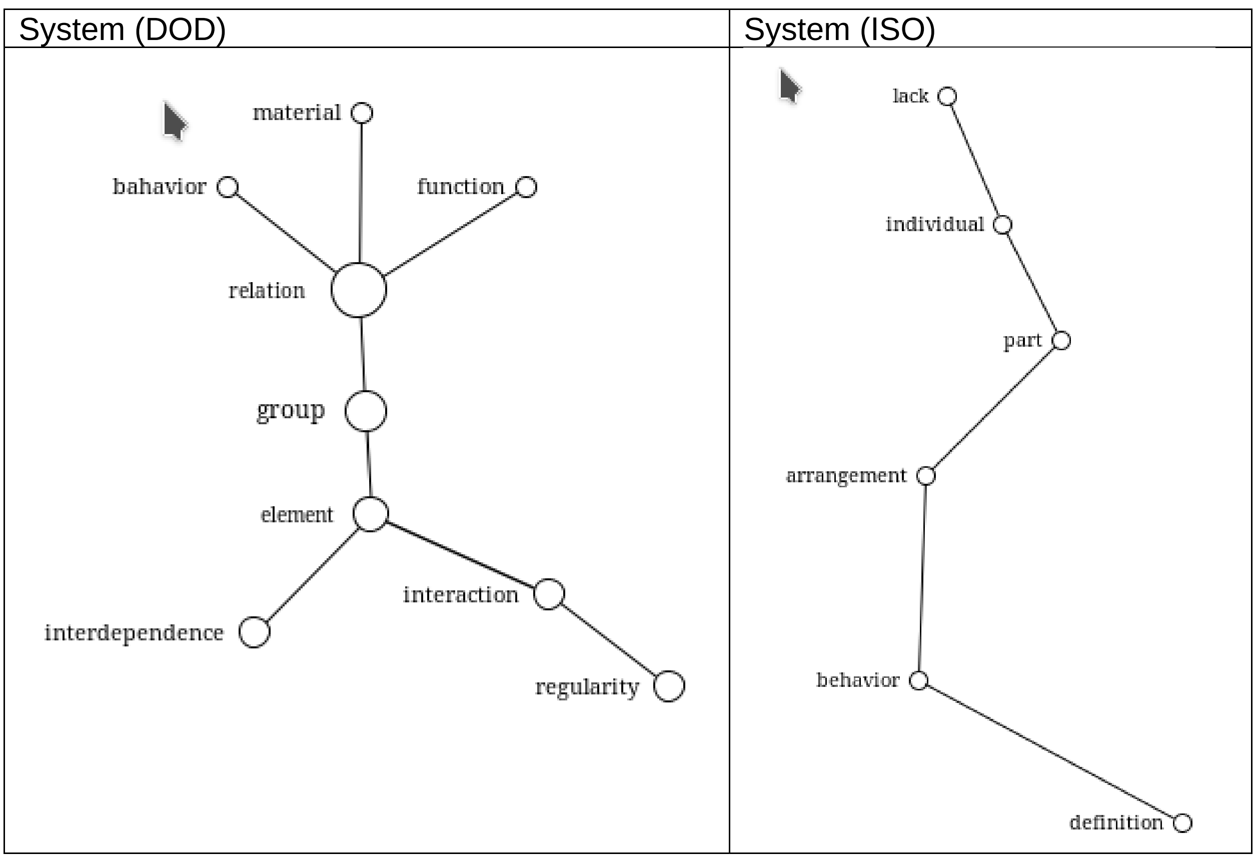


Figure 51. Semantic networks for the term "system" defined in ISO 15288 and DOD SE Guideline.

The semantic networks are expanded by including relevant terms with similar meaning using deep learning techniques. Each node in the network is enriched by using WordNet, a lexical database of English where nouns, verbs, adjectives, and adverbs are grouped into cognitive synonym sets [25]. An example of an expanded network for the DoD SE guideline "system" term is shown in Figure 52. The colored edges capture the WordNet mapping of the relationships to ensure network captures the most relevant semantic of the term. Afterwards, the semantic networks are merged to compare the definitions. Figure 53 shows the merged semantic networks for the "system" definition.

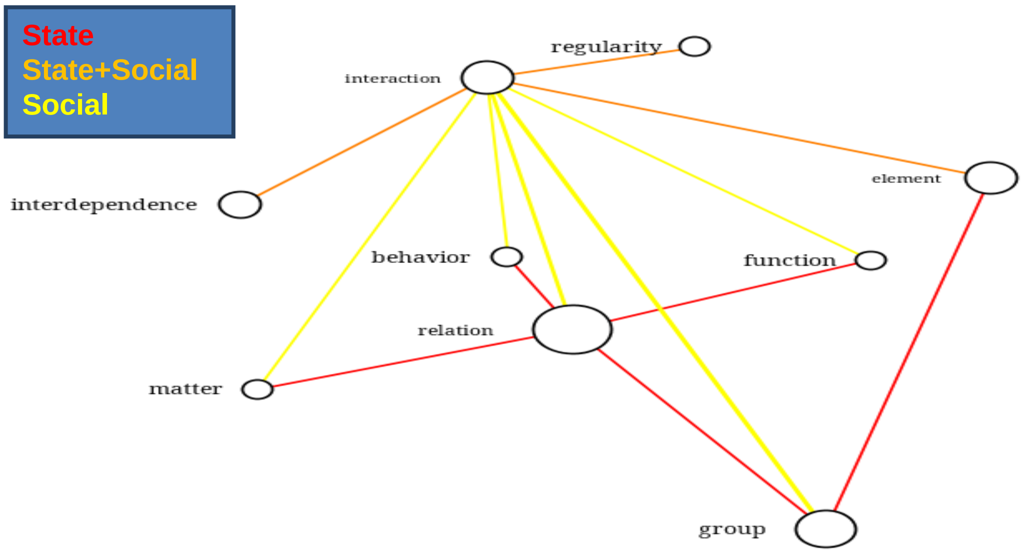


Figure 52. Expanded semantic network for the DoD SE Guideline term "system".

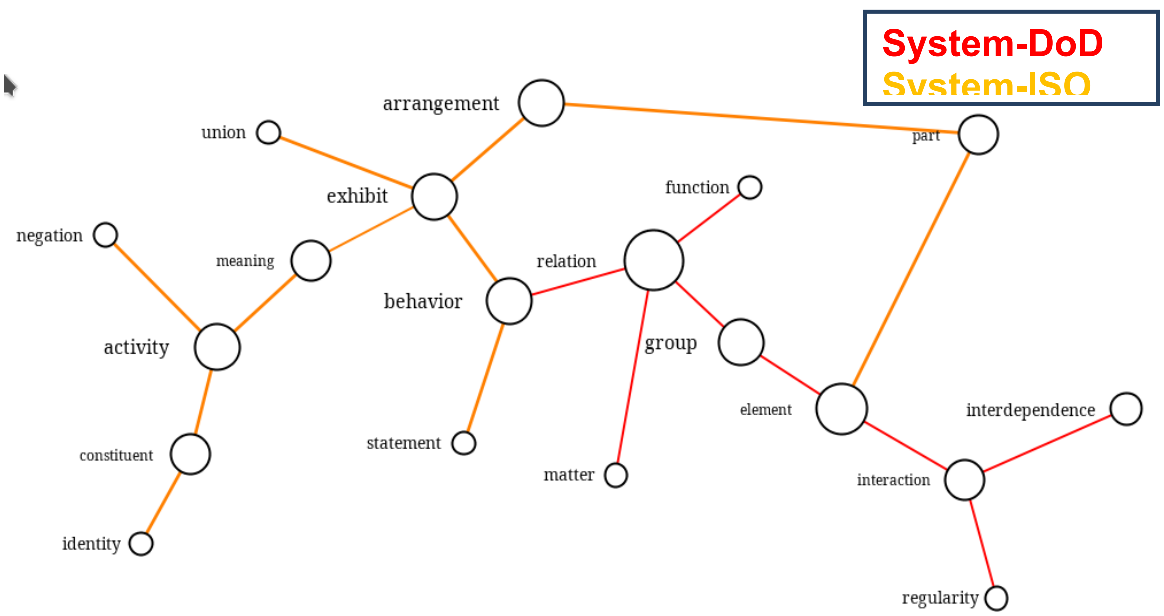


Figure 53. Comparison of semantic networks for the term "system."

In Figure 53, “behavior” node is a shared node in both definitions. “Part” node in the ISO 15288 semantic network and the “element” node in the DoD semantic network are related and linked. All these nodes capture the similarities of the definitions. Further evaluation of the networks identifies differences between the definitions. For example, DoD definition emphasizes interaction and interdependence of elements in a system while ISO definition does not provide these characteristics of a system definition and emphasizes the arrangement of parts. While the merged semantic networks support comparison of original definitions, the comparisons depend on interpretation of the individual evaluating the networks. To eliminate the subjectivity in comparisons, it is important to map the generated semantic networks to a baseline knowledge base such as Common Core Ontology. Thus, each generated term in the semantic networks is mapped into the existing set of ontologies used by DoD, in this case CCO. Table 1 provides the mapping of terms in the “system” definition semantic networks to CCO.

Table 1. Mapping of terms to Common Core Ontology

ISO 15288 definition	DoD definition	CCO mapping
Arrangement	Interaction	Act of Construction
Meaning		Act of Estimation
Statement		Act of Estimation
Union	Group	Act of Association

Once terms are mapped to CCO, definitions can be compared using the CCO mappings. Figure 54 shows the semantic networks of the definitions mapped to CCO.

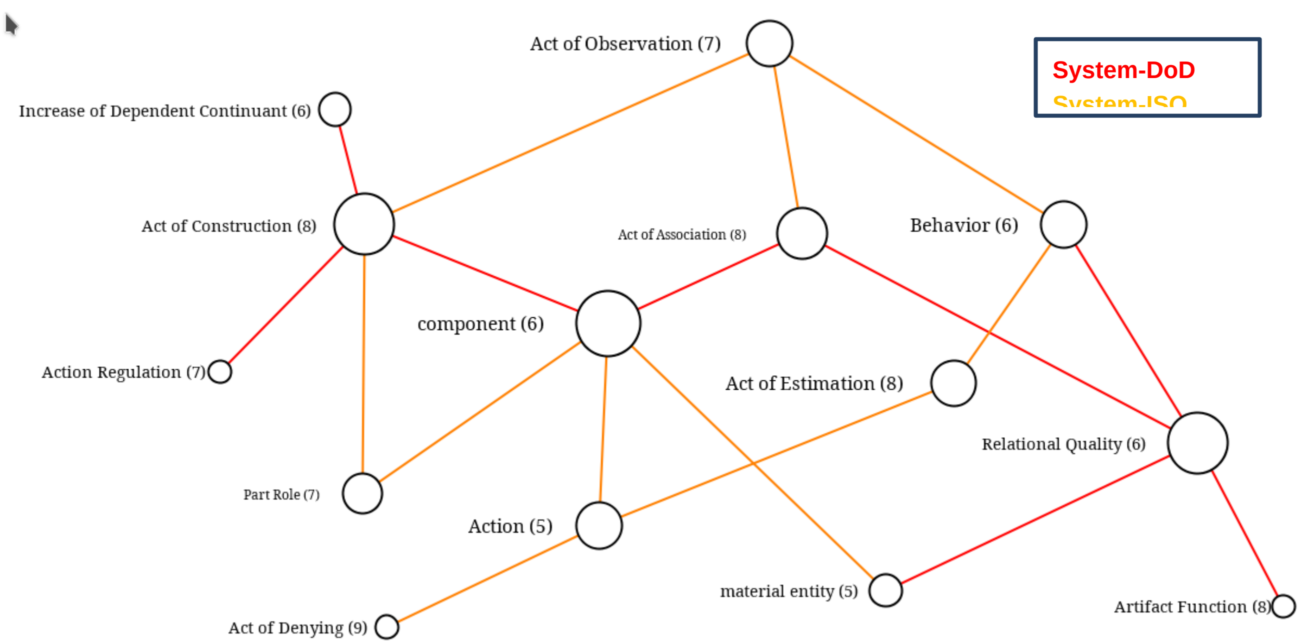


Figure 54. Comparison of semantic networks for the definition “system” based on CCO mapping.

Mapping to CCO eliminates subjectivity in comparing definitions of the “system”. Furthermore, comparing the semantic networks using CCO mappings allows comparisons at various levels. Terms can be abstracted at a higher level to resolve differences among the definitions. Alternatively, terms can be expanded to identify the differences among definitions. Figure 55 and Figure 56 illustrate the comparison of semantic networks based on CCO.

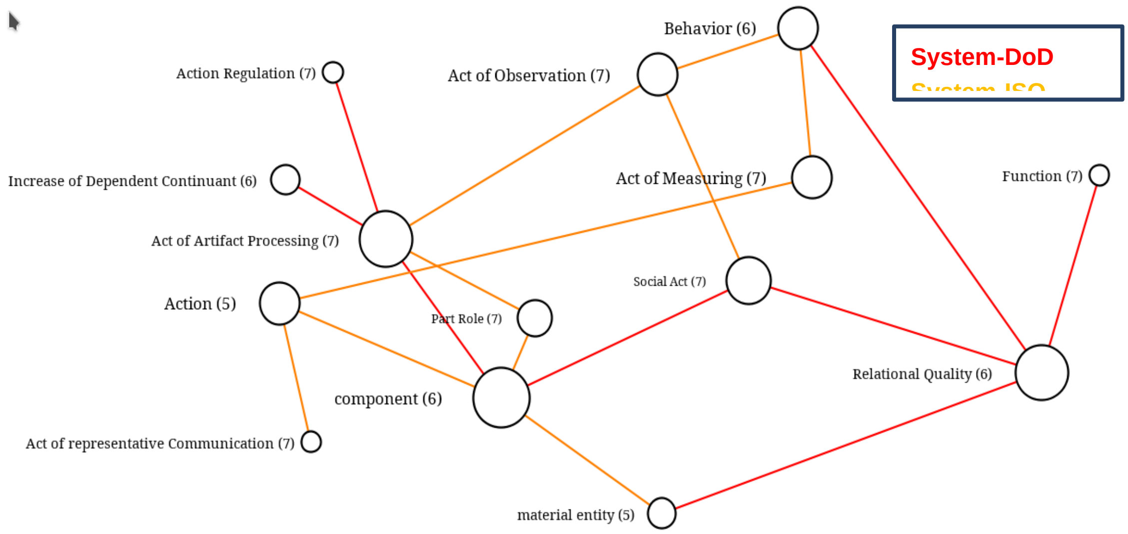
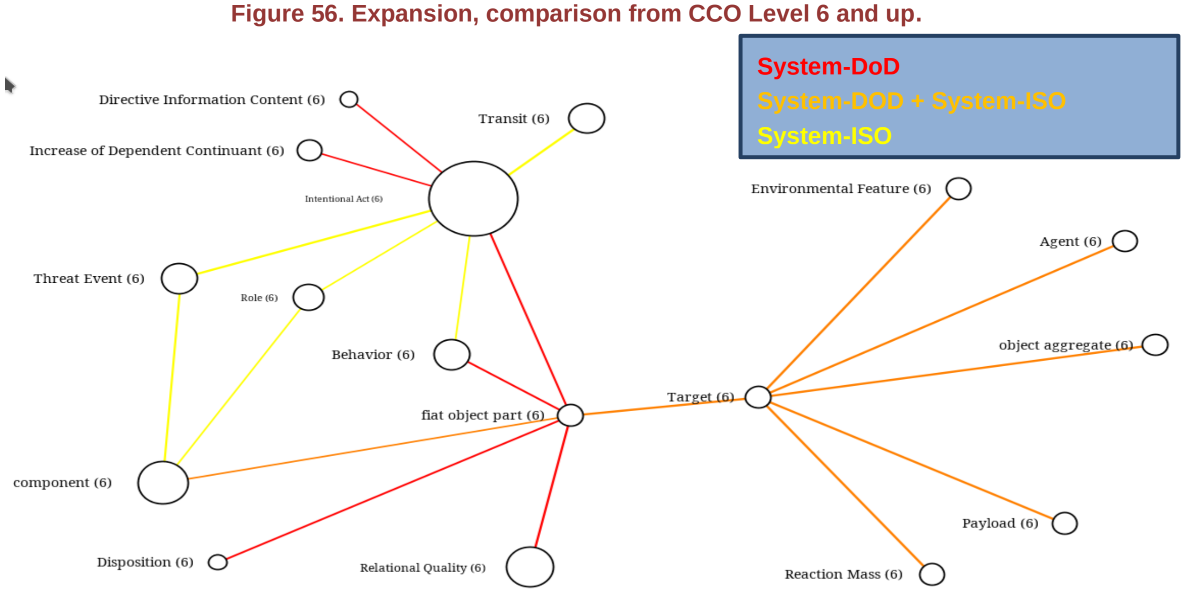


Figure 55.Abstraction, comparison from CCO Level 7 and up.

Figure 56. Expansion, comparison from CCO Level 6 and up.



Comparison of terms across various DoD guidelines

The knowledge discovery framework can be used to analyze terms that are used across various guidelines. The following terms in Table 2 are shared across the DoD 5000.88 policy, the DoD Engineering of Defense Systems Guidebook, and the DoD Systems Engineering Guidebook. The definition of each term is provided in the DAU glossary [10].

Table 2: Terms shares across DoD 5000.88 policy, DoD Engineering of Defense Systems Guidebook, and DoD Systems Engineering Guidebook

Term	Definition
Digital Engineering	An integrated digital approach that uses authoritative sources of data and models about systems as a continuum across disciplines to support lifecycle activities from concept through disposal
Concept of Operations (CONOPS)	A verbal or graphic statement, in broad outline, of a commander's assumptions or intent in regard to an operation or series of operations that is designed to give an overall picture of the operation
Mission Engineering	The deliberate planning, analyzing, organizing, and integrating of current and emerging operational and system capabilities to achieve desired warfighting mission effects. Mission is the objective task, together with the purpose, which clearly indicates the action to be taken.

Following the knowledge discovery framework, terms are converted to semantic networks, and expanded. Afterwards semantic networks of each definition are merged for comparison. Figure 57 shows all terms combined and Figure 58 shows same network mapped to CCO. In Figure 57, red network displays the "CONOPS" definition terms, yellow network displays the "mission engineering" definition, and orange network displays the "digital engineering" definition. The term "operation" is the integration term for "CONOPS" and "mission engineering" definitions. The term "capability" is the integration term for "mission engineering" and "digital engineering" terms. While "digital engineering" and "CONOPS" definitions are not directly related to each other, the terms "operation" and "capability" play a key role in relating these definitions within a specific context. In Figure 58, the same network is mapped to CCO which captures additional nodes of integration for the definitions. For example, in the raw network in Figure 57, digital engineering and mission engineering definitions are not directly related. In Figure 58, "assumption", and "objective" at level CCO level 7 are integration nodes for these definitions.

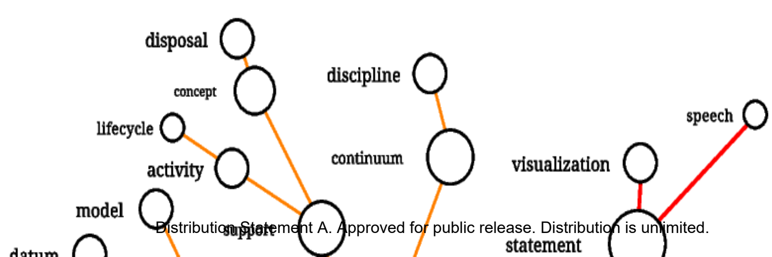


Figure 57. Comparison of terms combined.

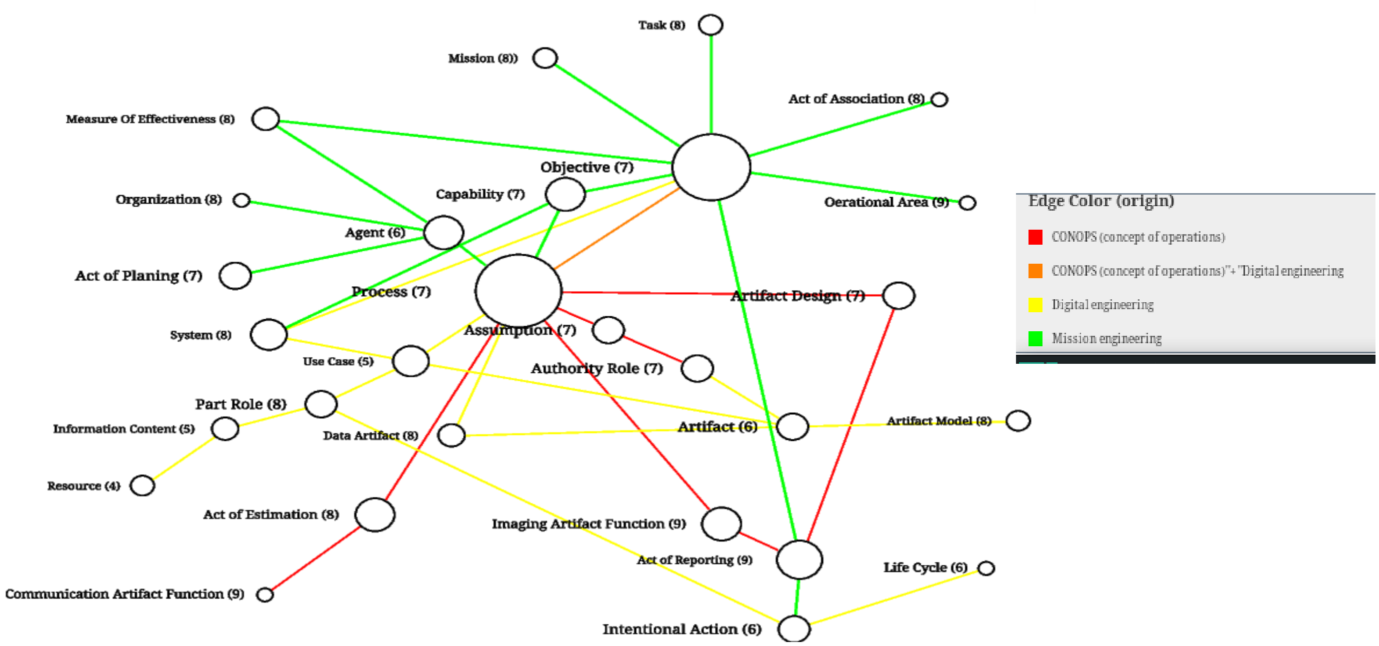


Figure 58. Comparison of terms mapped to CCO.

Current work shown in Figure 59 provided a knowledge discovery framework based on natural language processing and semantic network models to establish a shared knowledge base across DoD acquisition organizations, and ultimately to support convergence of acquisition and systems engineering disciplines. The work also facilitated deriving a domain level ontology for the DoD acquisition organizations by aligning the terms used across various DoD guidelines, policies as well as the ISO 15288 standard to the Common Core Ontology.

Several challenges should be addressed along this research path. Expressivity of input data is an important parameter for effectiveness of the knowledge discovery framework. Definitions that are rich in expressing the meaning of the terms improves semantic networks and the consequent inferencing and analysis. Furthermore, the performance of natural language processing and machine learning techniques improves with availability to additional domain specific knowledge. The framework can be expanded to other test cases including policy gap analysis at various granularities generated by various organizations.

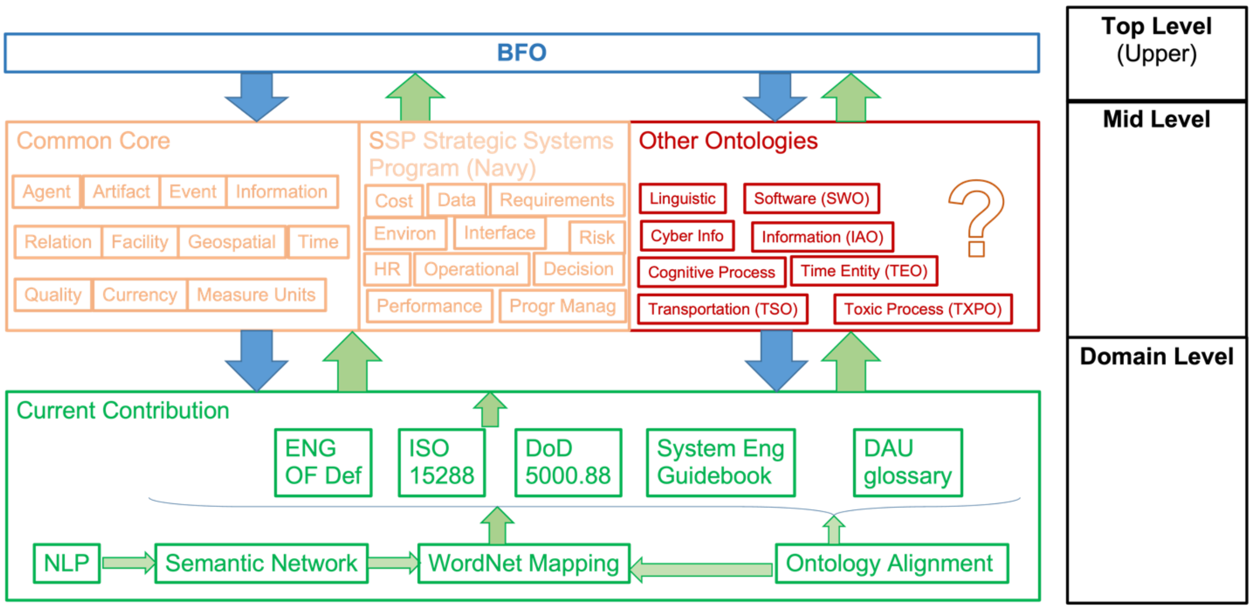


Figure 59. Current work and future direction.

Current work is an initial step towards establishing a common digital ontology which is essential step towards acquisition and engineering digital convergence (Roadmap shown in Figure 60). Future work should focus on expanding the current work by aligning the other ontologies relevant to the acquisition and engineering domains. DoD and various engineering disciplines already utilize a wide range of ontologies that are derived from BFO and CCO (ontological basis for systems engineering and acquisition node in Figure 60). Aligning these disparate ontologies developed for various purposes will enrich the shared knowledge base which is necessary for digital engineering applications used across the system lifecycle. This will also facilitate semantic integration necessary for future pervasive data and model driven applications.

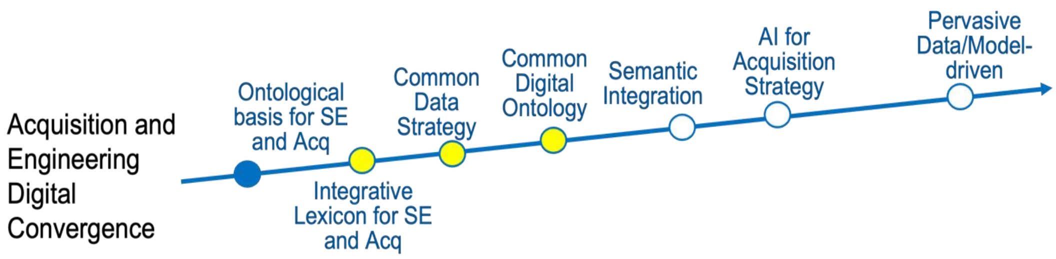


Figure 60. Roadmap path: Acquisition and Engineering Digital Convergence.

REFERENCES

1. ‘Army Regulation (AR) 70-1, Army Acquisition Policy”, available at https://asc.army.mil/web/arn3845_ar70_1_final/
2. “Army Regulation (AR) 71-9, Army Warfighting Capability Determination”, available at https://armypubs.army.mil/epubs/DR_pubs/DR_a/ARN31331-AR_71-9-000-WEB-1.pdf
3. Arp, R., Smith, B., and Spear, A.D. 2015, *Building Ontologies with Basic Formal Ontology*, Cambridge, MA: MIT Press.
4. Barb A. S., Clariana R. B. and Shyu C. R., (2013) "Applications of PathFinder Network Scaling for Improving the Ranking of Satellite Images," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 3, pp. 1092-1099, doi: 10.1109/JSTARS.2013.2242254.
5. Clariana, R.B. (2010). Deriving Individual and Group Knowledge Structure from Network Diagrams and from Essays. In: Ifenthaler, D., Pirnay-Dummer, P., Seel, N. (eds) Computer-Based Diagnostics and Systematic Analysis of Knowledge. Springer, Boston, MA. https://doi.org/10.1007/978-1-4419-5662-0_7
6. Common Core Ontology, available at https://github.com/CommonCoreOntology/CommonCoreOntologies
7. “DoD Systems Engineering Guidebook”, available at https://ac.cto.mil/wp-content/uploads/2022/02/Systems-Eng-Guidebook_Feb2022-Cleared-slp.pdf
8. “DoD Engineering of Defense Systems Guidebook”, available at https://ac.cto.mil/wp-content/uploads/2022/02/Eng-Defense-Systems_Feb2022-Cleared-slp.pdf
9. “DoD Instruction 5000.88, Engineering of Defense Systems”, available at https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500088p.PDF
10. “Defense Acquisition University (DAU) glossary”, available at https://www.dau.edu/tools/t/DAU-Glossary
11. Gruber, T. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5, 199-220.
12. Gruber, T.R.,(1995). Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum. Comput. Stud. 43, 907–928. doi:http://dx.doi.org/10.1006/ ijhc.1995.1081.
13. "ISO/IEC/IEEE Draft Standard - Systems and Software engineering -- System Life Cycle Processes," in *ISO/IEC/IEEE P15288/DIS, March 2022* , vol., no., pp.1-122, 26 April 2022.

14. Joint Chiefs of Staff. “Insights and Best practices Focus Paper – Interorganizational Cooperation”, Fifth Edition Washington, D.C., April 2018, available at https://www.jcs.mil/Portals/36/Documents/Doctrine/fp/interorgan_coop_fp.pdf
15. Joint Chiefs of Staff. “Department of Defense Dictionary of Military and Associated Terms.” Washington, D.C.: Joint Chiefs of Staff, U.S. G.P.O., November 2021, available at https://irp.fas.org/doddir/dod/dictionary.pdf
16. Kilicay-Ergin N; Barb A. Multi-level Evaluation of Smart City Initiatives Using the SUMO Ontology and Choquet Integral,” Human Centred Intelligent Systems. Smart Innovation, Systems and Technologies. Zimmermann A, Howlett R, Jain L, editors. Singapore: Springer; 2021, p.27-39.
17. Kilicay-Ergin N; Barb A. (2020). Smart City Document Evaluation to Support Policy Analysis. 2020 IEEE International Systems Conference; 2020 August 24; Montreal, Canada.
18. Musen, M.A., (2015). The protégé project. AI Matters. 1, 4–12. doi:http://dx.doi.org/10.1145/2757001.2757003.
19. Novak, J. D. & Gowin, D.B. (1984). Learning how to learn. New York: Cambridge University Press.
20. Morosoff P., Rudnicki R., Bryant J., Farrell R., Smith B., (2015). Joint Doctrine Ontology: A Benchmark for Military Information Systems Interoperability. Semantic Technology for Intelligence, Defense and Security (STIDS), CEUR.
21. Rudnicki R., (2019). An Overview of the Common Core Ontologies. available at https://www.nist.gov/system/files/documents/2021/10/14/nist-ai-rfi-cubrc_inc_004.pdf
22. Ruiz-Primo, M. A., Shavelson, R. J., Li, M., & Schultz, S. E. (2001). On the validity of cognitive interpretations of scores from alternative concept-mapping techniques. Educational Assessment, 7, 99-141.
23. Sellar, S., & Thompson, G. (2016). The Becoming-Statistic: Information Ontologies and Computerized Adaptive Testing in Education. Cultural Studies, Critical Methodologies, 16(5), 491–501.
24. Transforming Systems Engineering Through Model-Centric Engineering, Technical Report SERC-2019-TR005, available at https://apps.dtic.mil/sti/pdfs/AD1073187.pdf
25. WordNet, A Lexical Database for English, available at https://wordnet.princeton.edu
26. Yang L., Cormican K., and Yu M., (2019) Ontology-based systems engineering: A state-of-the-art review. Computers in Industry, 111, 148-171.

PART 7: SE MOD LESSONS LEARNED AND ADOPTION FRAMEWORK

This research was published in a paper entitled "Framework for and progress of adoption of digital and model-based systems engineering into engineering enterprises" as part of the 2023 Conference on Systems Engineering Research. The lessons learned were published separately and are included in this report.

FRAMEWORK FOR AND PROGRESS OF ADOPTION OF DIGITAL AND MODEL-BASED SYSTEMS ENGINEERING INTO ENGINEERING ENTERPRISES

The Systems Engineering Research Center (SERC) conducted a sustained series of research tasks leading to codification of a framework and lessons learned for adoption of Digital Engineering (DE) and Model-Based Systems Engineering (MBSE). DE and MBSE are separate but jointly evolving strategies. DE is defined as "an integrated digital approach that uses authoritative sources of system data and models as a continuum across disciplines to support lifecycle activities from concept through disposal."[1] MBSE is defined as "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases."[2] Successful adoption will thus be characterized by two general characteristics of the organization: *its ability to conduct systems engineering and related systems modeling, and its effectiveness at digital transformation*. These two characteristics have not necessarily been the core of organizational adoption strategies to date and resulting published literature.

System level modeling has been in practice since there was systems engineering. The purpose of a systems level model is synthesis: the behavior and performance of the whole. Conversely, much of the up-front SE process is focused on analysis: necessary decomposition of system structure/function and work breakdowns so a system can be created by development teams. Good SE organizations maintain and use models describing a holistic view of system behavior and performance for the team in support of all the other disciplines that do the detailed design work of their functions and performance.

With the Systems Modeling Language (SysML) and MBSE tools, we now have a structured digital language and toolset to describe and visualize that system model in a form that can be digitally connected to the decomposed analysis models. A good holistic system model provides insight to the development team that cannot be gained from lower-level analysis models, and the digital infrastructure creates connectivity of the systems level behavior and performance down lower level components and disciplines. There is nothing in the SysML language or the MBSE tools that cause these, only how they are used. Thus, no amount of investment in tools and training will drive adoption unless the workforce sees value being created from this in their daily work. The value of a digitally connected system model has two components: it provides insight on the end product performance that improves my decision-making, and it creates digital connectivity to data and analysis models that improves my efficiency.

Cultural change involves people showing other people better ways to accomplish their work. Really talented systems modelers are needed to create models that involve and lead the team to make faster and better decisions, and really good software/information technology people are needed who transparently improve the team's efficiency through the digital connectivity. The workforce needs to see these people and these benefits before they will embrace adoption of the tools and methods. DE/MBSE transformation requires an organizational design process that builds the SE, MBSE, and digital skills around a set of informal SE leaders that grow workforce knowledge and skills in all three areas together, integrates the digital infrastructure with a further set of informal software/IT leaders, who demonstrate the benefits of this transformation, in a community that builds awareness over time. Much of the focus on DE/MBSE transformation has been on organizational enablers and organizational change management, while many anecdotes for success speak of the talented systems engineers that adopted the digital strategies and led their organizations to change.

The SERC research on DE/MBSE adoption found many factors that must be addressed for organizations to achieve this transformation. Table 1 organizes the 12 most prominent factors in our research, organized across three categories: Organizational design, Organizational enablers/barriers, and Organizational change management. Current research is building a set of detailed lessons learned around each. these factors were selected and categorized through a set of research activities summarized in section 2. The team first conducted an enterprise analysis using interviews to identify what might result from future adoption of DE/MBSE (section 2.1). Following that, the team used survey and literature reviews to broadly categorize DE/MBSE benefits and adoption factors (section 2.2). That study led to a causal analysis that selected primary measurable benefits and adoption factors, resulting in the model in Table 1 (section 2.3). Finally, additional literature review and interviews were used to conduct an initial validation of Table 1 (section 2.4).

Table 2. Organizational Adoption Factors.

Organizational design	Organizational Enablers/Barriers	Organizational Change Management
Workforce knowledge / skills (SE domain, MBSE tools, digital strategies)	Leadership support / commitment	DE/MBSE methods / processes (maturity): MBSE terminology and libraries
Integration to support the digital implementation (tool infrastructure)	Training & categories of training	Change management process design (lessons learned, communicating success)
Demonstrated benefits/results Programs/projects using methods & processes	Resources for implementation (cost to use tools, willingness to invest)	People willing to use the DE/MBSE tools (a primary adoption measure)
People in model building roles	Tool Infrastructure: user experience with them and stakeholder buy-in	Greater use of DE/MBSE tools (overcoming resistance)

RESEARCH SUMMARIES

The SERC has analyzed through various surveys, interviews, and literature review the detailed benefits, enablers, barriers, change strategies, and lessons learned that are related to DE/MBSE adoption. As the SERC is a U.S. Department of Defense (DoD) funded research center, the central focus has been on the defense acquisition system and its related industrial base. However, the interviews, surveys, and literature analyses have been conducted broadly across government agencies, industry, and academia.

Enterprise System Analysis

The initial research task, entitled “Digital Thread Enabled Acquisition,” interviewed over 25 stakeholders currently involved in DE initiatives across multiple DoD agencies, the National Aeronautics and Space Administration (NASA), and other DoD research centers. Industry was (intentionally) not part of the interview process. These interviews were then used to develop five conceptual models, represented as systemigrams, describing what that future DoD acquisition enterprise might look like given success of their emerging SE strategy.

In June 2018, the Deputy Assistant Secretary of Defense for Systems Engineering released the DoD Digital Engineering Strategy, a comprehensive strategy for the transformation of DoD engineering methods, processes, and tools to the digital age. The strategy outlines five strategic goals for the transformation, targeted to “promote the use of digital representations of systems and components and the use of digital artifacts as a technical means of communication across a diverse set of stakeholders, address a range of disciplines involved in the acquisition and procurement of national defense systems, and encourage innovation in the way we build, test, field, and sustain our national defense systems and how we train and shape the workforce to use these practices."[1] The SERC project was conducted in parallel with and independent of the development of that strategy. Its purpose was to understand how the strategy might evolve and change the way the DoD conducts engineering development, acquisition, and sustainment of new and existing systems. The full report, "Enterprise System-of-Systems Model for Digital-Thread Enabled Acquisition," can be accessed on the SERC website[3].

This research conceptualized the changes to workforce and culture necessary to achieve the strategy. A number of qualitative characteristics of DE/MBSE adoption emerged from this study, which are listed in Table 2. These are just initial interview statements and are listed for completeness and provided as qualitative information to familiarize the reader with concepts of organizational DE/MBSE adoption. The research discussed in the following sections will generalize these to the adoption model of Table 1.

Table 3. Qualitative statements of organizational adoption.

Organizational design
Start with SE: Good SE will enable success with MBSE, MBSE itself will not create value.
Multidisciplinary value: Systems Engineers work across disciplines, DE and MBSE must create added value across disciplines.
Digital literacy: Experienced SE's may not be comfortable with DE, younger engineers may bring a digital culture with them. Create knowledge transfer between them.
Systems knowledge: is a unique value of DE/MBSE. A good system model can create/maintain systems knowledge to improve awareness of other disciplinary engineers.
Model quality: models must demonstrably improve system understanding and decisions.
Model abstractions: commensurate to the roles and uses of DE/MBSE. Models must communicate decisions at all levels.
Digital infrastructure: will make finding and using data more efficient, everything will be on the desktop when needed.
In-house software/information technology (IT): skills are necessary for speed and flexibility of tool integrations.
New data management processes: must be created and will bring new roles/skills to DE/MBSE.
Organizational Enablers/Barriers
Training: must be commensurate to the roles and uses of DE/MBSE.
Leadership: must clearly provide strategy and intent, investment and resources, and the messaging associated with the value of organizational change.
Communication and messaging: must clearly articulate value to the workforce (benefits) and maintain awareness.
Continuous assessment: of enterprise capabilities and maturity.
Automation: of time-intensive tasks will improve value and adoption.
Digital collaboration platforms: will improve cross-program and cross-disciplinary interaction.

Digital modeling and collaboration: improve distributed development, ownership, maintenance, etc.
Organizational Change Management
DE initiatives: are essential to adoption. These include initiatives to share data and models, methods/ processes/tools (communities of practice), and successful implementations (pathfinder/innovation projects).
Lessons learned: DE initiatives create lessons learned and success stories that increase awareness of DE/MBSE value. In particular how a quality systems model increases insight and improves decisions.
Champions: there are likely to be a few experienced practitioners in the DE/MBSE strategy that attract followers to the change activities.
First-hand MBSE benefits awareness: will incentivize more people to become systems engineers.
Awareness of DE/MBSE limitations: will accelerate learning curves.
User experience: with digital tools must be positive. Digital tools must provide effective visualization for decisions, support collaboration, be customizable, and promote new uses that add value.
Measurement: validate that DE/MBSE improves efficiency, agility, and quality.

DE/MBSE Benefits, Adoption, and Maturity

As a transformation strategy, each of the qualitative aspects of adoption must be assessed over time, which also implies a measurement strategy. The enterprise analysis task identified a candidate set of metrics but did not document any objective evidence for these. A second research task was undertaken to formalize these measures. At the start of the research effort, the hope was to identify and document best practices across the DoD, defense industry, and other industries related to measurement of the DE enterprise transformation, metrics for success, and standard success guidance. It quickly became clear that best practices for assessment and measurement do not yet exist in the DE and MBSE communities, and the transformation process is not yet mature enough across the community to standardize best practices and success metrics. Given the state of the practice, the research shifted to a set of efforts to define a comprehensive framework for DE/MBSE benefits and expected value linked to the ongoing development of enterprise capabilities and experienced transformation “pain points,” enablers, obstacles, and change strategies.[4]

Given that the value is expected to come from the foundations of good SE and related MBSE activities, supported by DE, the research team set out to broadly identify the benefits and success factors of MBSE. The research inductively characterized two frameworks that categorize MBSE and related DE benefits and adoption strategies that can be universally applied to a formal enterprise change strategy and associated performance measurement activities. The first framework categorized 48 benefit areas linked to four digital transformation outcome areas: quality, velocity/agility, user experience, and knowledge transfer. The second framework categorized 37 success factors linked to organizational management subsystems encompassing leadership, communication, strategy and vision, resources, workforce, change strategy and processes, customers, measurement and data, workforce, organizational DE processes related to DE, and the organizational and external environments. The two frameworks were developed from literature reviews and a survey of the systems engineering community.

The survey was designed using the INCOSE Model-Based Enterprise Capability Matrix.[5] The Capability Matrix was developed to help organizations that have already made the decision to implement DE/MBSE capabilities assess and grow these capabilities in a comprehensive and coherent manner. The survey consisted of 23 rated questions linked to the 42 capabilities in the INCOSE Capability Matrix, another 12 free-text questions, as well as a set of demographic questions. Results were published on the SERC website along with the metrics research at https://sercuarc.org/results-of-the-serc-incose-ndia-mbse-maturity-survey-are-in.[6]

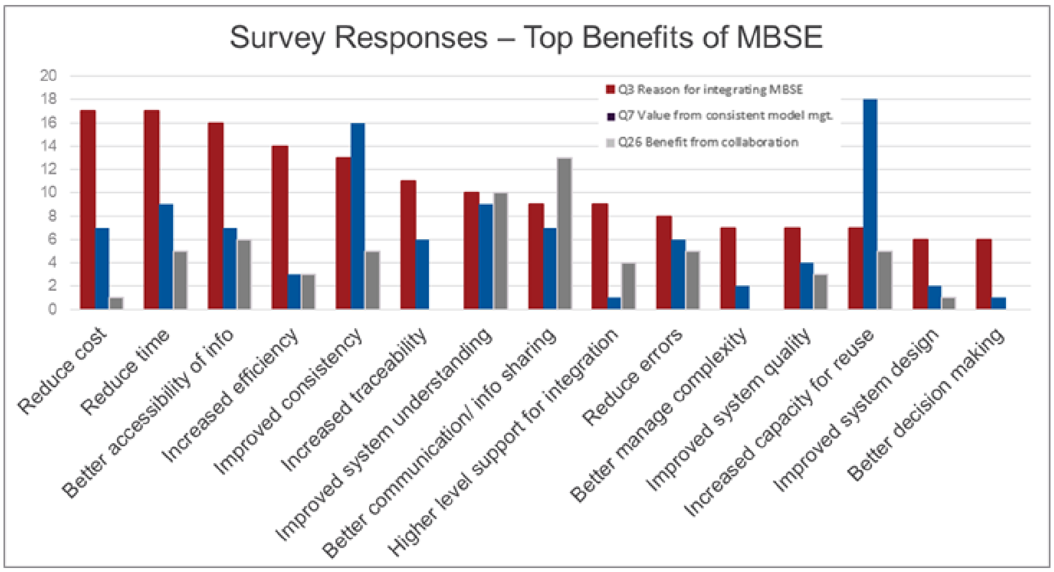


Figure 61. Top cited benefits from the survey.[6]

Figure 61 lists the 15 most cited of the 48 benefits collected. For a detailed analysis of the benefits, refer to research conducted by Henderson and Salado.[7] It is critical that adoption strategies are linked to benefits (the benefits are realized by adoption) and the benefits are measurable. This led to a further research task to define causal linkage of adoption strategies, discussed in the next section.

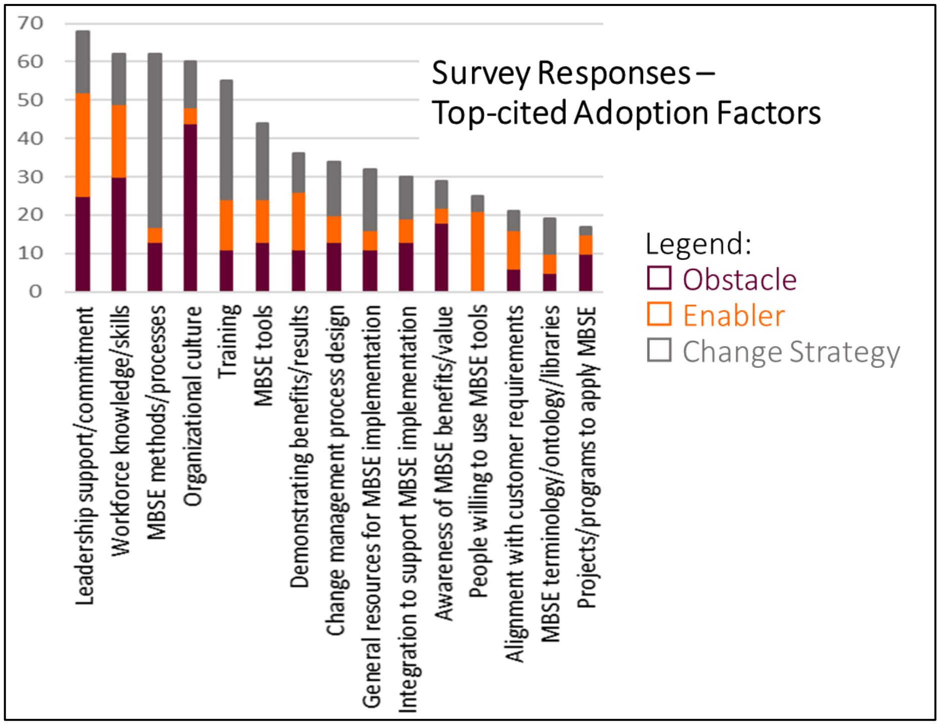


Figure 62. Top cited adoption factors from the survey.[6]

The 15 most cited adoption factors from the 37 found in the survey are shown in Figure 62. These were derived from three free-text survey questions requesting the top enablers, barriers, and change management strategies for organizational MBSE adoption. It should be noted that the survey did not directly request input on strategies for organizational design, these were derived from the further research.

DE/MBSE Measurement Framework

The research on benefits, adoption, and maturity resulted in a follow-on research task to formally document DE/MBSE measures.[8] DE is a subset of the larger aspects of enterprise digital transformation. Gartner reported four common characteristics for good enterprise level digital transformation metrics: adoption, usability, productivity, and new value.[9]

1. Measure people adoption, and enterprise process adoption
2. Analyze breadth of usability, and issues with usability
3. Measure productivity indicators
4. Generate new value to the enterprise (quality and knowledge transfer)

This research derived a set of metrics across these categories from the survey and literature review, organized into five categories: organizational adoption, user experience with the methods and tools, productivity in terms of time and effort, and value including both improved product and process quality and improved knowledge transfer across groups. The top-cited metrics in each category are listed in Table 3.

Table 4. Most cited DE/MBSE metrics derived from survey and literature review.[9]

CATEGORY	MOST CITED MEASURES
Organizational Adoption	Projects using DE/MBSE Methods and Processes
	People Willing to Use the Tools, related to organizational roles
	Leadership Support and Commitment
	People Willing to Use the Tools, related to organizational knowledge and skills
	Change Management Process Design
User Experience	Improved System Understanding
	Better Able to Manage Complexity
Productivity	Increased Consistency
	Reduced Time
	Improved Capacity for Reuse (Reusability)
	Increased Efficiency
	Improved Collaboration
Product and Process Quality	Reduced Cost
	Reduced Defects/Errors/Rework
	Increased Traceability
	Higher Level of Support for Integration
	Improved System Quality
Improved Knowledge Transfer	Better Accessibility of Information
	Better Communication/ Information Sharing

The first two categories are related to adoption and the others are related to benefits. The research provided a basis for a comprehensive measurement model of both the DE/MBSE activities that provide value and the other activities that promote adoption. However, the measures as collected in Table 3 are a mix of higher level statements, activities, low level measures, aggregated measures, and outcome measures. The next phase of the research concentrated on building a causal model that related all the potential benefit and adoption measures together.

Figure 63 is a partial causal map of DE/MBSE benefits and success strategies derived from this analysis. In the figure, the green boxes at the top refer to “primary benefits” of DE/MBSE, those benefits that come directly from the improved capability or additional features of MBSE. The primary benefits are defined in Table 4. These can also be seen as the actionable benefits or those that could be directly influenced in the organizational design of DE/MBSE methods and processes.

The black boxes are the secondary benefits that are the effects or results of those primary benefits. The causal links between primary and secondary benefits are not shown in this diagram in order to simplify the figure, but they should inform an organizational measurement strategy. In addition, only the secondary benefits with direct causal linkage to adoption factors are shown. Refer to Henderson, McDermott, Van Aken, Salado 2022[10] for the full model.

The adoption factors are shown in the blue boxes and links of Figure 3.[10] In this figure all the direct links between adoption factors and benefits are shown, with the green arrows representing adoption factors with causal influence on a primary benefit and the black arrows representing adoption factor causal influence on a secondary benefit.

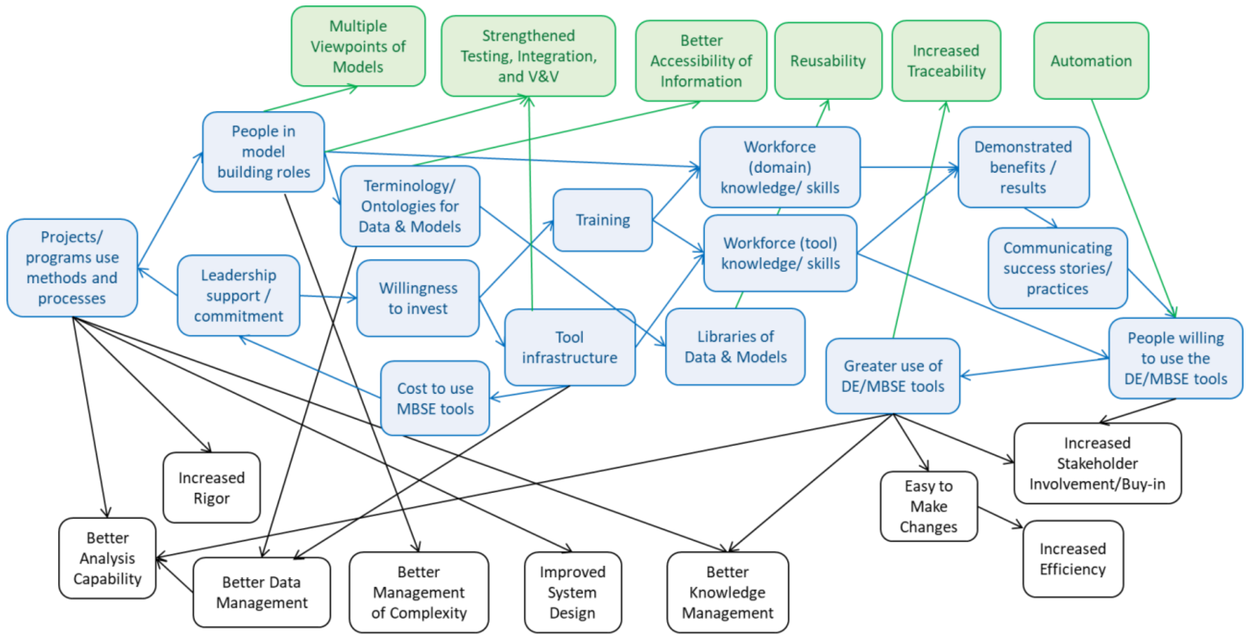


Figure 63. Benefits causal map with adoption measures added.[10]

Table 5. Benefits from DE/MBSE that come directly from implementation.[10]

Direct Benefits	Definition
Higher level support for automation	Use of tools and methods that automate previously manual tasks and decisions
Early V&V	Moving tasks into earlier development phases that would have required effort in later phases
Reusability	Reusing existing data, models, and knowledge in new development
Increased traceability	Formally linking requirements, design, test, and so forth through models
Strengthened testing	Using data and models to increase test coverage in any phase
Better accessibility of information	Increasing access to digital data and models to more people involved in program decisions
Higher level support for integration	Using data/models to support both the integration of information and system integration tasks
Multiple viewpoints of model	Presentation of data and models in the language and context of those that need access

The blue boxes in Figure 63 were used to categorize the organizational adoption factors of Table 1, which is repeated here as Table 5 for ease of reference. Actual text linkage to Figure 3 adoption factors are shown in bold text.

Table 6. Organizational adoption factors (equivalent to Table 1).

Organizational design	Organizational Enablers/Barriers	Organizational Change Management
Workforce knowledge / skills (SE **domain**, MBSE **tools**, digital strategies)	**Leadership support / commitment**	DE/MBSE methods / processes (maturity): MBSE **terminology** and **libraries**
Integration to support the digital implementation (**tool infrastructure**)	**Training** & categories of training	Change management process design (lessons learned, **communicating success**)
Demonstrated benefits/results **Programs/projects using methods & processes**	Resources for implementation (**cost to use tools, willingness to invest**)	**People willing to use the DE/MBSE tools** (a primary adoption measure)
People in model building roles	**Tool Infrastructure**: user experience with them and stakeholder buy-in	**Greater use of DE/MBSE tools** (overcoming resistance)

Adoption Experiences in Organizations: Lessons Learned

The final research task (to be published) compiled lessons learned from published case studies and practitioner interviews. Lessons learned through DE/MBSE adoption efforts are a useful means of communicating best practices and recommendations. The statements from these interviews and publications informed the final adoption framework of Table 1. The literature review has compiled over 600 lessons learned statements from 46 papers discussing MBSE lessons learned. This research also conducted 18 interviews across 18 different organizational units conducting DE/MBSE transformation.[11] The data collected in this research informed and completed the classification of the organizational adoption factors of Table 1.

Organizational design:

- Workforce knowledge / skills (SE domain, MBSE tools, and digital strategies): this must begin with SE domain knowledge, then mature the MBSE tool knowledge to gain the benefits, and in parallel develop the knowledge of the digital strategies needed to integrate data and models efficiently. If the organizations do not have the requisite SE knowledge introduction of MBSE tools may aid that development, but only if used to provide the holistic knowledge to other disciplines need for better decision making. Use of standalone MBSE tools without the underlying software/IT knowledge/skills will also prevent success.
- Integration to support the digital implementation: organizations need coordination mechanisms across both the technical and organizational levels in order to communicate the domain, tool, and digital strategies. The research has codified nine coordination mechanisms: Lack of a coordination mechanism; formal hierarchies; standard rules/procedures; direct informal contact; permanent coordinating teams/ roles; implicit coordination through technology; task forces; liaison roles; and matrix structures.[11] Organizations must employ sets of these to facilitate adoption.
- Demonstrated benefits / results: interviews and literature offer different perspectives on whether to introduce the DE/MBSE transformation all at once or to start small with selected programs and expand. Perspectives also differ as to introduction off-cycle (in a non-project sandbox) or on-cycle on productive projects.[12] The majority of the adoption lessons learned suggest an on-cycle approach and a strategy to start small and build over time.[11]
- Awareness of benefits / results: people must understand the depth and breadth of the SE processes defined in the MBSE tools and their value. This will come from the results of the other three organizational design strategies noted here.

Organizational enablers/barriers:

- Leadership support / commitment: for organizations who were not able to successfully adopt MBSE, the main cause was management did not support it. But when there is management support, the adoption effort is vastly improved.[11] Many of the adoption factors in Figure 2 must be "led," and may possibly inform a leadership checklist (a subject for further research).
- Training & categories of training: all of our interviews, surveys, and the associated literature agree that training is critical to adoption. The lessons

learned research identified four separate training categories: Reviewers (all who need to know how to use models to make decisions); Developers (people building and maintaining the models); Architects (senior engineers assisting with the content of the models); and Administrators (software and information technology support managing the infrastructure).[11]

- Resources for implementation: both the related effort/cost and the willingness to invest are critical factors.
- Tools and user experience with them: at this point the DE/MBSE related tools are immature, rapidly changing, and must be experimented with to support the transformation. A key lesson learned is to not make the tools a sole focus of the adoption effort.[11]

Organizational change management:

- DE/MBSE methods/processes (maturity): many interviewees report having an enterprise level standardization group and framework is critical, but it can be slow and difficult to develop. It is necessary for integration across organizational boundaries.[11]
- Change management process design: although change management strategies may differ across organizations, most cited the establish of a core MBSE team that could demonstrate benefits to the rest of the organization.[11]
- People willing to use the DE/MBSE tool: many of the benefits of a DE/MBSE transformation will be lost if artifacts have to be produced and activities have to be conducted outside of the digital environment and related models. The number/percentage of people willing to use the DE/MBSE tools will be a critical leading indicator of successful adoption.[10]
- Greater use of DE/MBSE tools: DE/MBSE is almost universally discussed as being a major cultural shift for organizations. Resistance to change will be a common problem in the transformation, requiring a good change management process design and an evolution over time. Management literature on cultural change is consistent with the adoption model presented here – critical behaviors must spread from “champions” who obtain the three-fold domain knowledge and skills; these informal leaders will motivate the change based on their demonstration of value to others; and the use of DE/MBSE must align with the traits or “identity” of the organization.[13] Measuring growth in use of the tools is the full outcome measure.

DERIVED GUIDANCE AND DISCUSSION

This research initially set out to qualitatively understand the complexity of enterprise change needed to achieve the full benefits from fully adopting DE/MBSE methods and processes. In this process we learned that the benefits and measures of success for DE/MBSE adoption were not understood by the community, and needed to be defined and formalized by further research. Through survey and literature review, a series of research projects were able to first define and categorize DE/MBSE benefits and adoption factors, then create recommendations for both measurement of benefits and applications of the adoption factors and strategies discussed here.

As we have determined from this research, there are a set of organizational adoption factors that can be defined, tracked, and measured as part of an organizational DE/MBSE transformation. Too many enterprises are entering this transformation without a full understanding of the necessary factors and mechanisms of DE/MBSE adoption. Many do not have the basic SE capacity and experienced people in model building roles to successfully accomplish the transformation. Others jump into the transformation without sufficient understanding of the factors that need to be addressed for successful adoption of the digital and MBSE methods and tools and lack the organizational leadership and change management support to make it to the finish line. There appears to be a general lack of focus on the two primary values of DE and MBSE: models that provide holistic insight on the end product behavior and performance so as to improve team decision-making, and the digital connectivity to data and analysis models that improve the efficiency of the development, management, and support teams.

The series of research efforts conducted by the SERC can be combined at this point to provide a comprehensive model of the factors needed for organizational adoption of DE/MBSE. At this point the research may not yet be a “checklist” for DE/MBSE adoption, but it does provide a view into the primary factors that must be addressed. There remain very few organizations and programs that are actively measuring their DE/MBSE methods and processes and organizational adoption journeys. This work can provide a framework for collection of those experience over time.

ACKNOWLEDGEMENTS

A large part of the interviews, survey analysis, and literature reviews cited in this research were conducted by Kaitlin Henderson, a PhD student in the Grado Department of Industrial and Systems Engineering at Virginia Tech. Many thanks to Kaitlin who successfully defended her PhD with the work, we would not be here without you.

References to this section

1. DoD. Department of Defense Digital Engineering Strategy. Office of the Deputy Assistant Secretary of Defense for Systems Engineering; 2018.
2. Model-Based Systems Engineering (MBSE) (glossary) - SEBoK. sebokwiki.org. Retrieved 2022-12-10.
3. McDermott T, Collopy P, Paredis C, Nadolski M. Enterprise System of Systems Model for Digital Thread Enabled Acquisition. SERC-2018-TR-109. Stevens Institute of Technology Hoboken; 2018. https://sercuarc.org/publication/?id=197&pub-type=Technical-Report&publication=SERC-2018-TR-109-Enterprise+System-of-Systems+Model+for+Digital-Thread+Enabled+Acquisition. Retrieved 2022-12-10.
4. McDermott T, Hutchison N, Van Aken E, Salado A, Henderson K, Clifford M. Task Order WRT-1001: Digital Engineering Measures. Hoboken, NJ, USA: Systems Engineering Research Center (SERC); 2020.
5. INCOSE Model-Based Enterprise Capability Matrix and User’s Guide, Version 1.0, January 2020.

6. McDermott T, Hutchison N, Clifford M, Van Aken E, Salado A, Henderson K. Benchmarking the Benefits and Current Maturity of Model-Based Systems Engineering across the Enterprise. SERC-2020-SR-001. Systems Engineering Research Center; 2020.
7. Henderson K, Salado A. Value and benefits of model based systems engineering (MBSE): Evidence from the literature. Systems Engineering. 2021;24(1):51-66.
8. Moore S. How to measure digital transformation progress. Information technology insights. 2019. Accessed December 14, 2021. https://www.gartner.com/smarterwithgartner/how-to-measuredigital-transformation-progress. Retrieved 2022-12-10.
9. Henderson K, McDermott T, Van Aken E, Salado A. Application of Digital Engineering Measures. SERC-2021-TR-024. Stevens Institute of Technology Hoboken; 2021.
10. Henderson K, McDermott T, Van Aken E, Salado A. Towards Developing Metrics to Evaluate Digital Engineering. *Systems Engineering*. 2022;1–29
11. Henderson K, McDermott T, Salado A. MBSE Adoption Experiences in Organizations: Lessons Learned. To be published.
12. Chami M, Aleksandraviciene A, Morkevicius A, Bruel JM. Towards solving MBSE adoption challenges: the D3 MBSE adoption toolbox. Paper presented at: INCOSE International Symposium 2018.
13. Katzenbach JR, Steffen I, Kronley C. “Cultural Change That Sticks.” Harvard Business Review. July 2012.

PART 2. LESSONS LEARNED

The SERC research on DE/MBSE adoption found many factors that must be addressed for organizations to achieve this transformation. Table 1 in the previous section organized the 12 most prominent factors in our research, organized across three categories: Organizational design, Organizational enablers/barriers, and Organizational change management. This research collected a set of detailed lessons learned around each from existing literature. The numbers following each statement reflect the reference that the statement was taken from.

ORGANIZATIONAL ENABLERS/BARRIERS

Leadership Support/Commitment:

1. “Successful implementation and adoption of MBSE is not a single, discrete event. Successful implementation requires a time-phased transformation in a complex System-of-Systems enterprise environment. As such, a coordinated vision across the entire enterprise is essential.” 35
2. “Leadership sets direction, supports staff development, organizes for project and infrastructure development, support and sustainment (ask for artifacts, implement MBSE-based reviews, establish and reward milestones, measure progress).” 42

3. "Unity of Leadership is Essential. In the first infusions, management support for the effort must to be clear and consistent. Management must be willing to pay the startup costs and to give time for the effort to pay dividends. In addition, the engineering leadership must be reasonably unified in their willingness to work together to figure out how to do this." 4
4. "Executive Level Sponsorship. Although increased MBSE popularity has strengthened executive support, there are still conflicting MBSE adoption goals between short-term driven employees who care about low adoption cost, with others aiming at more adoption quality and long-term solutions." 13
5. "Failure to consider the broader engineering organizational needs, as well as those of the enterprise itself, results in lack of enterprise leadership support, including MBSE adoption decisions and choices that fail to operate within the overall engineering environment and fail to achieve their goals even within the systems engineering organization." 35
6. Organizations should be "aware of potential target programs and [have] just-in-time availability of mature tools, training, support, expertise, tailoring approaches, and troubleshooting prior to actually beginning the engineering. It also requires the encouragement of program leadership to recognize that the competitive environment likely requires changes to the development process for the sake of improved cost, schedule, and capability." 11

Training and Categories of Training:

7. "All engineers should get, at least, basic training in MBSE." 1
8. "New practitioners need training in the language methods and tools of MBSE. The training should also be adapted to different members of the project team. In particular, a small core modeling team may require more significant MBSE training, while the larger project team may only require sufficient training to understand the modeling artifacts. After the initial training, ongoing mentorship is essential to provide the support needed to help the team climb the learning curve." 19
9. "Everyone Needs Training, but to Different Levels... three groups need to receive training commensurate with their level of interaction with the models. Different levels of modeling familiarity are required, thus resulting in different levels of training... we have constructed a set of classes that addresses all three user-type groups. The classes are sequential, each one building on the one before it. We start with the basics: Architecting, AFT, MBSE and SysML familiarization for everyone. The second class is the advanced SysML and Tool-Specific training, for the engineers who will be working with the models (MagicDraw and AFT). Then finally the third class, really a continuing series of special topic sessions, is for those engineers who actually construct, maintain, and analyze the official project models." 4
10. "To master a modeling tool for UML and/or SysML, training and practice must provide some understanding of the definition of these model elements, how they relate, how to apply them efficiently in modeling tasks, but also what not to use of all available features. Even if only a subset of UML/SysML is used in the project,

the users—and in particular the mentors—need to have understanding of the underlying meta-model and its implementation in the tool. The fact that SysML is based on UML, and that UML is in turn based on a “merge” of several other modeling notations/standards, makes the meta-model cumbersome and non-transparent.” 2

User Experience with Tools and Stakeholder Buy-in:

11. “Provide expected SE products to accelerate adoption. Once the MBSE effort has proven that it can support existing processes with familiar products, these stakeholders may become more amenable to exploring more transformational approaches to improving processes and products, enabled by MBSE capabilities.” 6
12. “Engage early and often - If… MBSE has resulted in the discovery of non-obvious issues and problems, the earlier empowered stakeholders are engaged, the earlier these issues can be addressed. Through the use of the model, the implications of a given decision or option can be quickly (relative to not using a model) analysed and then returned to the decision makers for resolution and further guidance if required. The early and regular engagement also militates against producing a perfect solution for the wrong question.” 40
13. “Challenge: Tool Dependency and Integration. Companies need to pick a set of tools and train employees accordingly. Such a decision is not an easy task and there is no tool that satisfies all needs. Moreover, integration between systems modeling tools and others, such as simulation or requirements, is still solved with specific solutions.” 13
14. “Peer Pressure Pays. The outside experts mentioned above could convey to our executive leadership their informed assessment of the state of the industry in a way that we practitioners could not. Their assessments are far more authoritative than ours could be, so that when they warn of JPL falling behind and becoming less competitive if it does not proactively engage with MBSE, the message is compelling and believable. In this way we have found that peer pressure pays in terms of building institutional support for MBSE. Likewise we have found peer pressure within JPL to be an effective driver of infusion. IMCE has organized several lab-wide opportunities for emerging MBSE-based efforts to showcase their work and share lessons learned. The obvious benefit of course has been to cross-fertilize and share learning across many efforts. The additional benefit is the spirit of healthy competition which has been fostered.” 5
15. “The models afforded by MBSE, specifically SysML, were shown to bridge the gap in communication between engineers and medical professionals. This suggests that the same benefit may be seen when working with professionals in other non-engineering domains. The ability to formally model aspects of the system, and display them simply enough for all stakeholders to understand, proved invaluable.” 38

16. “Expose the stakeholders to the model, but not in more detail than absolutely necessary - The model is a tool to enable design decisions to be made not an end in itself. Depending on the stakeholder group and the level of experience of the MBSE practitioners, the early and frequent engagement referred to above, may devolve into a discussion on the minutia of the model or the modelling process rather than being used as a forum to make decisions or discuss options and trades. Some stakeholder groups may, if allowed, become enamored of the model or the modelling process rather than focused on making the decisions the model was created for.” 16
17. “Challenge: Large Models Visualization. Different team members are involved in querying the model contents. Unfortunately, existing tools require additional training effort, and customizing the layout of model elements and diagrams is time consuming. Additional challenges appear in large models, where model navigation and understanding become highly complicated.” 13
18. “the process itself needs to be visible, with a clearly defined purpose and observable benefits of using supporting models – using models or diagrams to communicate complex use cases and system functionality is likely to be beneficial and improve communication of design information, but the process by which this is done must also be clear.” 22
19. “the tool must be used as a communication aid, as well as for storing and executing design information. It would require the development of a system model with the appropriate structure to communicate data between multiple domain-specific tools in order to execute and analyse the validity of the proposed system.” 22
20. “Benefit for project: The ability to embed recognizable graphical icons within the models to represent system elements allowed for the models to be intuitively understood without a great deal of additional explanation. This enhanced communication was perhaps the most outstanding benefit of employing a model-based approach to engineering a system.” 18
21. “Build models hierarchically and eschew "eye chart" diagrams. Each diagram should be built for a specific purpose, and only the information needed to communicate that purpose should be included in the diagram. Additional purposes can be served with additional diagrams or reports. Most of these large diagrams can—and should—be built hierarchically, in many smaller diagrams that each attempt to capture a small number of key concepts. Capturing too many layers of hierarchy in a single diagram makes it difficult to read and understand. A good rule of thumb was found to be to limit the scope of a diagram to what can be viewed on the computer screen without scrolling.” 34
22. “Pilot programs can use available tools for early exploratory work. The principal value of the pilot effort is to provide valuable insight into the MBSE problem space, demonstrate the value of system modeling, architect the conceptual model, and serve as the launching point for the next iteration of the organization’s evolution to an MBSE end state. The models themselves should not be seen as the principal

objective of that effort. And the program should be willing to entertain changing tools if it is clear that the pilot toolset is not the most suitable solution for the longer-term effort." 6

23. "Keep the Focus on Engineering Products. Keeping focused on real engineering deliverables is important to avoid the pitfall of delivering a modeling solution everyone thinks is finished but which doesn't provide the required engineering answers. After all, the engineering deliverables are the whole point of the exercise. Our early attempts at "rolling up the mass" for the mass margin report showed this in stark relief: getting the numbers to add up, make sense, and be reliable turned out to require significantly more modeling and scripting than expected." 4
24. "Problems arose to a higher extent between system engineering and other specialty engineering disciplines. In spite of special training provided for, e.g., safety engineers, the feedback was that the models were very difficult to interpret. A conclusion is that model based projects should be prepared to provide "traditional style" information in the form of documents in order to, at some level, serve specific engineering disciplines when requested." 2
25. "It is not possible to make all stakeholders (of the model content) proficient in interpreting and navigating the model. Consequently, a way of extracting information from the model to traditional documents must be established so that stakeholders/colleagues can view and review model content as effective as with a traditional document approach, without extensive mentor support." 2

Resources for Implementation (Cost to Use Tools, Willingness to Invest):

26. "MBSE does imply heavy investment as well as a steep learning curve, especially in the initial phases, and is beyond the scope of a single project." 41
27. "MBSE adoption requires a substantial upfront investment, especially if it has not been considered before. This also includes determination of an effective investment strategy, accurate cost estimation and quantifying its return on investment." 13
28. Two approaches dominate MBSE adoption: off-cycle (in a sandbox environment) or on-cycle (directly on productive projects). The first approach is considered ideal, as not all companies have the required budget and time. The second approach is much more challenging and introduces additional costs for running projects. Choosing the wrong strategy can negatively impact the benefits of MBSE." 13
29. "… it is clear that the resource needed to align large teams to a coherent modelling methodology has been underestimated. Modelling opens up many new possibilities in terms of capturing design information and some stakeholders see the opportunity to revolutionise their engineering process. Here, the challenge is to communicate the cost associated with meeting ambitious objectives and the risk imposed by setting ambitious objectives that cannot be readily met by the current generation of SysML tools." 24
30. "For instance, companies with high available upfront investment might suffer from having the freedom that each department starts to define its own MBSE adoption solution. This brings later integration and model interchange issues." 10

31. “Good tools are costly; time to implement MBSE before starting the project is costly; until it becomes mainstream or someone demonstrates value at no cost to them, many will resist.” 14
32. “The second challenge mentioned, the additional costs and efforts, requires an organizational proactive approach to put in place an overall infrastructure including training, toolset and MBSE support team. The costs mentioned in the challenge section have to be fully or partly supported by the overall organization rather than by the project itself. Training is already funded by the JSC Human Resource training department. Similarly, tool costs should be subsidized by the overall engineering organization in order to facilitate the insertion into a project and ensure standardization across projects.” 45
33. “Having enough skilled resources to support the MBSE efforts is critical to its successful adoption. Mentors should be made available to partner with the Systems Engineers in order to support the adoption of modeling practices and tools. Effort and budget is required to provide project mentors who can participate as part of the project team and be involved in the system design and integration.” 45

ORGANIZATIONAL CHANGE MANAGEMENT

DE/MBSE Terminology, Methods and Processes (Maturity):

34. “More time should be spent on properly defining how to best use existing methodologies, which therefore appear to cost much more time and resources than necessary. Time must be set aside to build up an MBSE-friendly infrastructure; otherwise MBSE practices will also be considered a burden, because they are not well defined.” 27
35. “Challenge: Method Definition and Extension. It is often necessary to customize an appropriate method according to a defined purpose and scope. It is a challenge to set up the required method, document it and facilitate it with modeling rules, guidelines, tool customizations and training materials. Further challenges arise when new method extensions are needed.” 13
36. “Apply engineering methodologies that support effective modeling of the architecture both horizontally and vertically. The modeling method must support the definition of end-to-end system threads, preserve decomposition, be executable, and provide the capability to associate requirements to the elements within the behavior model.” 9
37. “Tools are perceived as not really mature at least for SysML and language is found too complex so people prefer using MS Visio.” 14
38. “Tools and methodologies issues. projects also seem to create their own “style” in how ‘they’ decide to utilize MBSE on a project.” 14
39. “No common terminology for MBSE… There are no process models that integrate MBSE properly.” 44

40. "Do not use SysML for requirements handling only! since it was not meant to be used as such, it becomes a bit awkward and unhandy as compared to other tools, which are better suited to this kind of use." 32
41. “A well-defined MBSE method is essential. An MBSE method must be clearly defined to support the model development. The method should also provide guidance on how to organize the system model to ensure it can be navigated, managed, and controlled.” 42
42. “By having the project team clearly identify their goals in adopting MBSE, one can better advise them on the method to follow. The method defines the concepts and rules to model the system. Using the method, the engineers can ensure that the models used to describe the structural and behavioral aspects of their systems are created accurately. The JSMT developed a meta-model, as a foundation to the modeling method, to capture the system architecture, hardware interfaces, and command and telemetry interfaces. The tools that extract data from the system models are used to generate multiple target products. This is aimed at providing added value to the project team.” 45
43. “Have the new MBSE processes well documented so you better understand what tool you will need.” 1
44. “The project team found that SysML-based MBSE practice provides a great level of flexibility, providing systems engineers with numerous options for facilitating and capturing systems engineering knowledge throughout the system lifecycle. However, this high-level of flexibility imposes the need to design the model structure carefully and select the most appropriate metamodel, SysML diagrams, and model elements to suit the project. This selection needs to be formalised and supported with model design rationale, in order to ensure consistent practices between team members, and across projects internal or external to an organization.” 16
45. “Models are Meant to be Abstractions. A common misconception of MBSE is that in order for the model to be useful it must describe everything, and describe it to a fine level of detail. This misconception needs to be corrected for an infusion to succeed, because otherwise resources will run out long before the job is complete. A key principle we have followed is to model only as far as we need to answer the question at hand. Assuming this is done on an infrastructure of common languages and tools, then the model can grow over time, as necessary, and each new model element will add synergistically to the body of work.” 4
46. “It is important to realize the necessity of different levels of abstraction, their relation and how to effectively use them.” 27
47. “One must be mindful when conducting the modelling task of the level of model fidelity needed, the purpose of the model, the questions that need to be answered, and not to model for the sake of modeling. Thus, the purposes of creating the model must be clearly defined upfront.” 15
48. “Establish a consistent approach towards the definition of equivalency relationships within the Model Based Engineering Environment. Specifically, a rigorous process

must be in place to establish equivalency relationships, and to modify or remove equivalency relationships when associated artifacts change, undergo versioning, or are removed. Without a consistent process, equivalency relationships can become confused, corrupted, or lost, leading to unreliable traceability throughout the ASoT implementation.” 40

49. “For example, is it possible to have a single, unified model? If not, how should different heterogeneous models communicate? How should different disciplines and attendant models interact with each other? What measures need to be taken to assure common assumptions and consistent semantics across different models from the different disciplines? How should quality attributes be incorporated in system models and how can the models be analyzed in terms of the degree to which they satisfy the quality attributes? What is the best way to capture knowledge, decisions, decision rationale, and expertise of system engineers? Last, since a model is a shared, living representation of multiple domains of interest, how can a consistent “baseline” be established, and how should it be reviewed?” 30
50. “Defining different use cases for a use case diagram has allowed a cyclical requirements analysis. Thanks to this analysis it is possible to correctly translate the business requirements in system requirement.” 31
51. “Closely related is the lack of stopping criteria. Modeling easily becomes addictive. It is easy for an engineer to over-detail the parts he knows well and overlook what he less understands. It should be the opposite: hot spots of a system must be modeled with greater care.” 8
52. “An organization should create its own, domain specific cookbook when introducing MBSE. Domain independent recipes and practices can be collected in a commonly shared cookbook…which provides modeling patterns, recipes, guidelines, and best practices for the application of SysML.” 27
53. “It is also worth underlining the importance of adequate modeling guidelines clearly describing what information should be captured in UML/SysML and what should be captured using traditional methods. In the absence of such guidelines, users have a tendency to add information to the model just because the possibility exists, leading to information inconsistency and redundancy.” 2
54. “Not separating need and solution modeling is a major but common mistake. This actually is not a MBSE-specific problem, as customers and engineers are often too early diving into the solution, overlooking the problem space [12]. Adopting a new approach is an opportunity to change this wrong practice. Not separating need and solution blurs the reading grid of the model and increases the risk of delivering a system that actually does not fully satisfy the original operational needs.” 8
55. “Another major pitfall is keeping several engineering levels in the same model “for the sake of simplicity”. The architectural design of a system is not the same thing as the architectural design of each of its subsystems: Lifecycle are different, contributors are different, design drivers might be different, etc. Performing the

design of a system and a subsystem in the same model is a major error that has led to significant refactoring work 2 or 3 times on … projects these last years.” 8

56. “Build in flexibility to adapt to future needs. Architects of the integrated set of system models should provide some flexibility in the structure of the models to allow them to grow to encompass new applications.” 34

Data/Model Accessibility and Libraries:

57. “There should be methods to capture, store, access, and share both artifacts and the central model.” 42
58. “The project team recommends creating an appropriate structure and defining where each type of textual component should be stored, since there are various locations available.” 16
59. “Ensure that the model is centralized and distributed. This does sound like a contradiction, but it is possible. Mostly it involves making MBSE a reality. A centralized repository for the information as opposed to a file-based system ensures that people are able to access the model as a whole rather than snippets. In addition, because of the pervasive nature of the model elements and the need for cross-references, much of the model is required for most operations. Keeping as much information in the model simplifies traceability considerably and helps ensure completeness, correctness, and consistency. It also provides a means for impact analysis.” 23
60. “In some cases, the technical debt that has accumulated in the models makes reuse of the model untenable, and significant refactoring of the models—if not complete reconstruction from first principles—may be the most effective approach to take in the long run.” 34
61. “Acquire required metadata for each digital artifact needed in order to support access control, search, approval, and recompete. Develop and adhere to a standard for the metadata collected. The ASoT requirements call out collecting artifact expiration dates, country-of-origin, country-of-delivery, information criticality, non-functional requirements such as manufacturing constraints, and cost and scheduling metrics. The ASoT requirements also call out evidence to demonstrate the provenance of digital artifacts, such as the tools used to build or generate the artifact, the contract guidance used to produce the artifact, marking and licensing information (even from previous contracts), template models used to produce the artifact, and analysis results and certification results associated with specific versions of the artifact.” 40
62. “It is easy to fall into the trap of putting semantics into diagrams: the semantics must be instead fully defined by the model behind in order to create more artifacts from the same source in an automated way: this requires an expensive initial effort which pays back many more times later on.” 27
63. “Models Evolve. The model needed in concept formulation is very different than the model needed in detailed design, or in operations. Models need to evolve and grow, and sometimes shrink. This should be the focus of model reuse along the project lifecycle. It also helps to answer the people who will suggest that building a

detailed model of the last flown mission will help you formulate the next. It all goes back the principle of modeling for a purpose, and not more. While the models may change, these changes can be evolutionary and cumulative as long as they are connected by a common set of ontologies and methodologies." 4

64. "the more the analysis can be separated from the model, the more reusable it will be. For our mass analysis we have achieved a high degree of separation of the model from the analysis, and as a result we are able to run exactly the same mass analysis script on all three of our mission option models. The corollary to this is "keep the model aligned with the concept rather than with the analysis". We initially found ourselves adopting modeling patterns which made the analysis scripts easier (drifting back into the Excel mindset). But we soon discovered that in order to further expand and refine these analyses, we would be forced to model in more and more non-intuitive ways. Therefore we discovered, and adopted, the principle that the model should be kept intuitive and aligned with the concept. We are convinced that the extra work required to make the analysis tools work is well worth it in the long run." 4
65. "What constitutes a complete set of models is a fundamental gap that is beyond the purview of MBSE. Also, the specification of model uses and how to use models is an overarching concern for model-based approaches." 30
66. "Focus on the underlying data, not just diagrams. While diagrams are the most visible products of a model, a well-architected model can provide many more insights through queries and automated report generation than the hand-assembled diagrams can by themselves, and additional views can be generated from that model through automated model transformations." 34
67. "At the commencement of every project, it is essential to establish a package structure to enable model elements storage, data accessibility and control, model management, and data exchange." 15
68. "But if the first mission element took longer than expected to analyze … the second and third ones showed the power of developing reusable methods: they each took a fraction of the time of the first. For the first mission element… both model capture and analysis were performed in the SysML model and the mass report took approximately two work-months to complete. The subsequent two concepts we modeled … each took about one half work-month each. And, a subsequent change of … design was accomplished in a fraction of that time. So, our advice is to first focus on description, and then implement analyses. A large part of the benefit accrues as soon as people start using the descriptive models, and this gives time and support to allow the more difficult work of analysis to be done." 4
69. "Separate the Model from the Analysis. … Two troublesome characteristics worth mentioning in this context are: as it is commonly used, the model and the analysis are inextricably intertwined; and by the nature of the tool, the model is forced into a form which facilitates the analysis. It is clear that the more a model can be a self-contained, internally self-consistent, and an intuitive description of the concept, the more informative it will be. Moreover, the more the analysis can be separated from

the model, the more reusable it will be… The corollary to this is “keep the model aligned with the concept rather than with the analysis”. We initially found ourselves adopting modeling patterns which made the analysis scripts easier (drifting back into the Excel mindset). But we soon discovered that in order to further expand and refine these analyses, we would be forced to model in more and more non-intuitive ways. Therefore, we discovered, and adopted, the principle that the model should be kept intuitive and aligned with the concept. We are convinced that the extra work required to make the analysis tools work is well worth it in the long run.” 4

70. “Challenge: Modularity and Reusability. Many organizations still follow an opportunistic and isolated reuse approach, where a set of data is copied and pasted from one context to another. Unfortunately, this still happens even with system models and results in losing the “source of truth” as soon as the copied source or pasted target is changed.” 13
71. “Provide Versioning, Variants, and Backups. If possible this should be done on a whole model basis. Again, this ensures that impact and traceability can be assessed against the whole model. If done on a section by section basis, it becomes more likely that version skew will take place.” 23
72. “Model configuration management is different. Configuration management of these descriptive models is a little different than for typical analytical models. The reason for this is that these descriptive models share some features with analytical models (software) and some features with program documentation (static documents). Like software, these models are contained in electronic files. Like program documentation, these models are intended to describe the system primarily for human understanding across multiple perspectives, and therefore need to reflect the system more precisely than typical analytical models, which only need to capture the essential abstracted characteristics of the system as it pertains to that analysis perspective. However, some of the unique properties of these models may add complexity to the configuration management process. For example, software configuration management typically relies on textual comparisons of software source code and checksum calculations performed on binary files. In contrast, system models are neither textual nor linear in nature, so model comparison will be much more complex. For example, moving a graphical model element within a diagram constitutes a change to the model file, but may or may not represent either a syntactic or semantic change. Because of these unique characteristics, configuration management processes for models will need to incorporate and improve upon the appropriate features from both software configuration management and document configuration management practices.” 6
73. “Another part of the MBSE infusion approach that has been successfully implemented … is the creation of a library of reusable models. By providing exemplary reference models, a project can jump start the model development. During SysML model development, for various projects, models that has the potential for re-use were collected for future projects. [we] designed a preliminary library structure for re-usability. As a result, some common representation and re-

usable elements have been established that are shared with new projects to leverage at project initiation. A library of reusable SysML elements is essential to assisting beginners and expert modelers in adopting MBSE." 45

74. "Acquire the data rights for each digital artifact … Consider technical data, computer software, and computer software documentation data rights and communicate the DoD's desired rights in the solicitation for each procurement based on the TD and CS strategy according to Defense Federal Acquisition Regulations Supplement (DFARS) 207.106 in the Acquisition Planning Phase of the procurement. The Statement of Work and CDRL should identify negotiated data rights for each digital artifact to be delivered… Communicate data rights and distribution marking policy for all types of digital artifacts that the ASoT will manage. Communicate the granularity with which markings are to be applied within diverse types of artifacts. For example, policy might call for data rights markings applied at the level of blocks in a SysML model." 40
75. "Communicate the approved representations for each type of digital artifact. Adopt and adhere to a set of approved representations (languages, formats) to facilitate interoperability between different ASoTs and to simplify the recompete of any digital artifact. During our demonstrations we found that contemporary tools can manage and relate different data representations, but to configure and maintain these tools requires engineering effort." 40
76. "Communicate the security policies that will enforce authorized access to digital artifacts stored in the ASoT. Stakeholders contributing digital artifacts to the ASoT should understand how the ASoT will protect those artifacts. The ASoT requirements call for security policies addressing, for example, information sensitivity, contractual rights, and organizational role." 40
77. "MBSE needs to incorporate artifacts/objects that are maintained outside the MBSE database. For example, detailed design drawings, which are created using specialized tools, are generally maintained in a configuration-controlled external database. Accessing these drawings from within the model, while maintaining consistency of the model with these drawings inevitably leads to parallel systems that need to be individually maintained while remaining mutually consistent." 30

Change Management Process Design:

78. "A successful MBSE adoption effort must address enterprise level challenges, such as: Identification of organizational changes and new skills training required for all constituent organizations within the enterprise needed in a digital environment over the full lifecycle of systems." 35
79. "An organization or project team should not make the transition to MBSE in an ad-hoc manner, but should employ concepts of organizational change in support of continuous improvement. These concepts include clearly identifying the issues to be addressed by MBSE, engaging stakeholders, developing and executing a plan for improvement or transition, and monitoring the results." 19
80. "Efficiency requires reengineering the business process, rather than just building the legacy process into the model." 26

81. “The adoption of MBSE is not isolated to the System Engineering department. It affects the entire set of engineering disciplines and overall business enterprise.” 35
82. “The most challenges related to MBSE adoption are noticed to be based on the human and technological factors. It starts with the awareness and change resistance on both executive and engineering levels within an organization. It goes over having the right MBSE resources to define the purpose, scope and method. Additionally, these challenges need to be addressed from the early phases and directly depends on the executive sponsorship and available upfront investment.” 10
83. “The human factor plays a central role, particularly if key players have different levels of MBSE knowledge and adequate time for training is not granted. Consequently, change is not always accepted, compared to existing approaches, it creates strong resistance due to the lack of expertise to deliver the required artifacts.” 13
84. “Grow capabilities slowly (through a roadmap) to more advanced capabilities.” 36
85. “Bring everyone to adoption (i.e., avoid creating castes).” 1
86. “Operational users get lost in their objectives, face difficulties with the tooling. Damages are there already when help is sought, typically leading to blaming the method and workbench.” 8
87. “The adoption of MBSE and the development and implementation of the digital engineering environment requires systems engineering methods and rigor. If not applied to the definition of requirements, development of the CONOPS, analysis of organizational impacts, and security concerns and the planning of development and integration, the resulting implementation of tools and the information technology infrastructure will likely fall well short of the strategic needs of the business and the functional and performance requirements.” 35
88. “Standardization of language across domains is necessary to aid in communication between teams.” 22
89. “To alleviate the perception of increased risks, targeted presentations are created and directed to new and potential users that explain the benefits of using MBSE, highlights the available models and tools, and presents successful project experiences. These presentations need to clearly emphasize the value proposition of MBSE and provide evidence on how the project can benefit by adopting MBSE.” 45
90. “More time should be spent on properly defining how to best use existing methodologies, which therefore appear to cost much more time and resources than necessary. Time must be set aside to build up an MBSE-friendly infrastructure; otherwise MBSE practices will also be considered a burden, because they are not well defined.” 27
91. “Pilot projects can be used to validate the MBSE approach. A pilot project should be well planned with clear objectives, deliverables, milestones, and sufficient resources to achieve the objectives. In addition, team continuity with effective leadership and stakeholder participation are essential.” 19

92. “As a consequence of not adapting how projects were planned and executed, the MBSE initiative often became an isolated effort, not part of the generic planning process, leading to unclear goals and objectives. Successful implementation is a management issue, and requires managers have the relevant understanding/competence. However, this necessary education/training is often not taking place on all levels.” 26
93. “There is resistance to enforced consistency. Smart people like to do things their own way, which makes collaboration and knowledge transfer more difficult.” 26
94. “Tooling inertia describes phenomena of the current in-house tooling environment that made our participants refrain from adopting MBSE. Tooling inertia includes resistance against learning new tools as well as potential incompatibilities of MBSE tools with current tools, and resistance to integrating new tools.” 44
95. “Inertia cannot be easily offset without a change agent or a technology champion. Understanding where to champion the efforts and at what level of the project, is important to successful adoption of MBSE.” 45

ORGANIZATIONAL DESIGN

Workforce (domain) Knowledge/Skills:

96. “There is a learning curve associated with reprogramming engineers. It can be more difficult to learn a new way of doing a familiar task than to learn it fresh.” 26
97. “Another major inhibiting factor is the availability of skilled practitioners; not just to actually execute MBSE-based projects, but the advocacy and energy of our most talented modelers is now spent in projects (a good thing) without much time left over to educate management or practitioners in other parts of the adoption curve than early adopters (not such a good thing).” 14
98. “The final lesson learned is the addition of new team members to the trade study. As the trade study matures and goes through cycles , team members come and go. One of the biggest challenges for the team is bringing new team members up to speed on the Trade Study itself. Background knowledge ... is critical. Since the Trade Study continues moving forward, bringing any new person into the team would require them to come up to speed on all existing [knowledge], the trade study work done so far, as well as keeping up with current trade study work being done. ... The Trade Study team has found an incredible value of MBSE and this modeling methodology when bringing new team members up to speed rapidly. Tradition SE practices would have a new team member reading thousands of pages of documentation on each of the network processes and systems. Using MBSE and our specific model, new members are able to go to one source for all information they would need to come up to speed on the study as the existing network diagrams have all been created already. At the same time as the new team member is parsing through previous cycle diagrams to learn the working of each network and what the integrated solution might be, the team member can also browse

models which are currently being created so the transition from keeping up to speed on the study, to producing for the study, can be as smooth and painless as possible. This has enabled our team to bring in new members and quickly get them up to production capability for the team, which in turn allows the team to shift off veteran members to further work in any one of the specific cycle programs.” 3

99. “In the early formulation phase in which we find ourselves there is a curious duality. On the one hand, the key work in early formulation centers around conceptual thinking. The spacecraft we propose are mere sketches, and a critical function of models is to describe the design space generously, in which the concept can evolve and take shape. On the other hand, we must always show that our concepts are feasible, and one of the ways we do this is build and analyze a ‘point design’ which we analyze for technical resources, performance, and cost. The models that we build to address these two disparate viewpoints must of necessity be partly conceptual and partly realizational. This should not mislead one into thinking that the space between them must be entirely filled in: it has proven very workable to have some parts of our model be treated as strictly conceptual, and other parts be treated as realizational (e.g., for the mass margin analysis).” 4
100. “Models Evolve. The model needed in concept formulation is very different than the model needed in detailed design, or in operations. Models need to evolve and grow, and sometimes shrink. This should be the focus of model reuse along the project lifecycle. It also helps to answer the people who will suggest that building a detailed model of the last flown mission will help you formulate the next. It all goes back the principle of modeling for a purpose, and not more. While the models may change, these changes can be evolutionary and cumulative as long as they are connected by a common set of ontologies and methodologies.” 4
101. “Models need to be architected and peer-reviewed. The conceptual model and the organization of models into modules can significantly impact their usability, consistency, and maintainability. It is important to have experienced architects familiar with both the problem space and the capabilities of the tools to lead that effort.” 34

Workforce Tool Knowledge/Skills:

102. “the first infusions will not have the benefit of an engineering pool with ubiquitous modeling skills…we found that the best way to get started on the right path was simply to hire as many of the existing cadre of skilled MBSE practitioners as we could afford.” 4
103. “Get Outside Expertise. From the very beginning, visionary managers … brought in world-class outside expertise to teach, advise, and guide our adoption of MBSE. These experts imbued our efforts with a maturity and credibility which would otherwise have been achieved through expensive trial and error. In addition, because some of these experts have also been in positions of executive leadership, they were also extremely helpful in helping executive leadership … understand the value proposition for MBSE. So outside

expertise has proven invaluable both for the quality of the infusion itself, as well as the institutional support for the infusion." 5

104. "Team Organization Matters... The pattern we have found that works well is a three-tiered one involving a small set of core modelers within a larger set of modeling-savvy systems engineers, within a larger set of all project personnel. While we have found that descriptive modeling can be done by almost anyone with the basic training, the additional rigor and consistency needed for quantitative analysis requires us to designate a smaller team of people who are modeling experts and who can apply best practice to the official configuration managed project models. Presently we have a core modeling team of a half dozen or so, within a larger team of 20 or so engineers. The experienced systems engineers provided guidance to keep the modeling focused on providing useful information, as well as mentoring of the core modelers who tended to be more junior. Frequent (daily) interactions were crucial to getting useful products: we were pathfinding so we had to stay very closely in touch. As important as it is to have a core modeling team, it is just as important to avoid fencing them off from the rest of the project. If the models are to be useful to the project, the project must understand and interact with the modeling team regularly, and likewise the modelers must be engaged in the larger engineering effort. Modelers who are also capable systems engineers will naturally employ their modeling skills to deliver engineering products in a model based way." 4
105. "Given the modeling and analysis tools now available, an MBSE methodology is essential — otherwise every student invents their own, and this lesson carries over into the practice of MBSE as well." 37
106. "There is a need for a team of expert modelers that can be provided by the organization to any project. In the past it was identified as a critical core function to develop and maintain a cadre of users for applications such as CAD and Finite Element Analysis since an individual project could not afford the time to train project members on the proper use of these detailed modeling tools. MBSE should be treated in the same manner. The time to train a MBSE modeler and to become proficient at the tool is beyond the ability of any single project. Another problem is that the MBSE modeler on one project may not continue with MBSE after that particular project. The time invested to train that modeler is lost when the project is completed. By having a team of MBSE modelers matrixed into all projects, the skills can continue to improve on each new project." 45
107. "Models are Meant to be Abstractions. A common misconception of MBSE is that in order for the model to be useful it must describe everything, and describe it to a fine level of detail. This misconception needs to be corrected for an infusion to succeed, because otherwise resources will run out long before the job is complete. A key principle we have followed is to model only as far as we need to answer the question at hand. Assuming this is done on an infrastructure of common languages and tools, then the model can grow over time, as necessary, and each new model element will add synergistically to the body of work." 4

108. “Prefer methodological knowledge over domain knowledge - While it would be optimal to have practitioners with a sound understanding of both MBSE methodology and the domain being analysed, this will not be achievable in all instances. If this is the case, it is recommended that an experienced MBSE practitioner be preferred to a domain expert. To extend Logan’s (2011) observation on MBSE enforcing good SE: expert application of MBSE on a group of domain experts will elicit a better product than a less than expert application of Systems Engineering on the same domain savvy personnel.” 17
109. “No matter how much UML/SysML and tool training is performed, there is a critical period in any project where engineers are getting frustrated in their attempts to apply the MBSE techniques. This is partly due to the gap in complexity between the comparably simple examples used in training and the size and complexity of the product under development… experienced mentors were engaged to assist developers to overcome such frustration and to ensure that modeling guidelines were applied by the individual developers. The mentors also had the responsibility to verify and approve implementation of tool/method “add-ons,” such as report generator implementation, document templates, and modeling guidelines. By experience, at least two members in the project team need to have experience in development with the use of object-oriented methods, the modeling tool, and large-scale models. Having two experienced persons reasoning about how to plan the modeling approach and how to partition the model, brings stability and safety to the proposed and adapted method. Initially, in a project, it seems desirable to have one experienced mentor for every 5–7 developers... Naturally, mentor support can be decreased as developers gain proficiency. With less mentor support, the risk increases for a diverging model and/or insufficient stringency or quality in the model/system.” 2
110. “Skill in modeling comes mainly from practical work and the learning curve to become highly productive seems to be 3–6 months, depending on modeling focus, engineering back-ground, and ambition. A wide range of learning curves was observed; one engineer with a good background in object-oriented methods became highly productive in 1 month. Hence, pilot projects for MBSE introduction should be allowed to have longer calendar time allotted, until the first increment of system analysis and design has been completed. For following increments the calendar time can be “paid back” through increased efficiency gained by the MBSE approach. With this initial investment in the “core development” phase, …we believe that there is a great potential for cost and time savings in the following phases.” 2

Integration to Support the Digital Implementation (Tool Infrastructure):

111. “The MBSE implementation itself is a system of interest (SOI) and requires the same systems engineering technical processes and any other SIO for successful design and development. As one SOI within the larger digital engineering system-of-systems, the MBSE SOI will most likely be integrated into an existing network of engineering tools, networks and data repositories.” 35

112. “Operational engineers have spent too much time installing and configuring the engineering workbench, which has impacts on their operational milestones.” 8
113. “Communicate the approved tools that the DoD will require stakeholders to use. Communicate these selections in the solicitation and/or Statement of Work. The ASoT requirements call out the need for a registry of approved modeling and analysis tools and the need to store the model analysis results in a systematic way that supports examination by subject matter experts.” 40
114. “The model building tools allow the import of data from different sources; this accelerates the building of the model. One of the objectives is to leverage existing artifacts that stakeholders have built to accelerate the development of models. Creating a tool, … to generate elements and diagrams from an Excel spreadsheet was the beginning of the toolset development. Modelers utilized this tool to build models directly from existing spreadsheet artifacts they have collected.” 45
115. “Tools to validate the generated models and products were developed to check for adherence to the recommended modeling method. For example, before the SysML Builder plug-in builds the model, it checks the accuracy of the data for import. As more stakeholders were exposed to the modeling method and tools, they requested additional capabilities to extract an increasing number of system design artifacts. Expanding the tool suite and generated products makes MBSE useful to a broader audience and increases the stakeholder involvement.” 45
116. “To build these MBSE tool suites in house, it was necessary that [we] develop a set of processes to guide the specification, development and deployment of the tool suites, and a unique set of SE skills within the cadre of engineers who execute these processes.” 20
117. “Tool selection should be driven by needs. Stakeholder concerns and needs for the models should be solicited first, then the desired outcomes for the project, as these should be the ultimate drivers for the project. The questions that the models are intended to answer to achieve these outcomes should be identified, followed by the products that the model set needs to contain or produce to answer these questions. Then, the methodology to be followed to develop those products through the sequential application of system modeling concepts should be fleshed out to the point where the needs of the modeling language and the tools are made explicit. This process provides a traceable framework for identifying tool needs based on the stakeholders’ needs for the integrated set of models.” 6
118. “Security classification issues complicate model management. While one of the strengths of the MBSE approach is that the model can be queried and transformed to construct a near-infinite variety of customized views, some of these views may result in classified associations of information. This may pose a significant challenge for security review of models.” 6
119. “13. Tools and modeling languages evolve very quickly. These modeling tools and modeling languages are evolving quickly, so any tool assessment, no matter how disciplined the decision process, will also become out of date quickly. The model architects will need to keep up to date with the latest developments in the modeling

tools, explore the new features and capabilities being added with each new release, and understand what old bugs or deprecated modeling constructs have been eliminated. Establishing a reusable tool and language selection criteria provides a repeatable method for assessment." 6

120. "Communicate the interfaces approved for access to other stakeholder ASoTs, such as OSLC. Our tool survey revealed that of the two common approaches for tool integration (either build a custom interface or build to a common standard), building to a common standard is more scalable and better supports future capabilities." 40
121. "Consider … Tools that support standardized, interoperable data representations and interfaces provide flexibility and enable the credible threat of recompete." 40
122. "Invest in open standards. To meet engineering objectives an organization may need to use multiple tool environments. For example, an organization might use MagicDraw for a modeling environment, IBM DOORS for requirements management, and IBM Rational Change Management for change management. Any significant engineering effort generates a vast amount of data, with data overlapping in representation and storage. An ASoT should integrate accumulated data so that query operations can traverse data relationships." 40
123. "Automated Web-Based Model Reports are Critical. One of the issues faced by adopters of MBSE is that the default vendor offerings require a consumer or reviewer of model information to use the vendor tool. Because the tools have a significant learning curve, this can present an insurmountable hurdle to acceptance among non-modelers (i.e., management, sponsors and review board members). As luck would have it, a separate team … had already invested in and developed a solution." 5

Demonstrated Benefits/Results:

124. "In contrast, the discipline specific Subject Matter Experts (SME) generally are not very interested in the MBSE approach since system integration is not their primary responsibility. The benefits to the individual discipline are often overshadowed by their direct responsibility. To win over the SMEs, there needs to be some concrete added value provided to them. By demonstrating some of the tools described below, the SMEs can be convinced to give MBSE a try. For instance, producing documents, and requirements compliance matrices from the information captured in the models, and showing the ability to produce the reports that project management and design engineers utilize during the design process can demonstrate how MBSE based approach can assist the SMEs in their daily activities. Demonstrating the capability to support the communication between all the project stakeholders is also important to obtain acceptance of the MBSE approach. Showing evidence of successful projects that have benefited over time can help the project justify the additional costs." 45
125. "the project team members need to be convinced that the extra effort required to develop the models provides some direct benefits. Generating products such as a parts list, connectivity information, telemetry and command data, requirements and

traceability from the model is a considerable help for the project team. SMEs and other stakeholders can also benefit from these products.” 45

126. “Resolving disconnects is a key benefit of MBSE. These disconnects linger unnoticed because the document-centric approach for knowledge management results in many information stovepipes that may only intersect due to serendipitous circumstances. However, building models based on the available data sources was found to be very useful in exposing the disconnects, driving stakeholders to recognize the disconnects and beginning the process of resolving them. The state of consistency of the program’s technical baseline will determine how much time is spent resolving these issues. If the MBSE effort is being tracked against schedule milestones, it is important to realize that this time spent is more a benefit of MBSE than a cost of MBSE. MBSE found the latent problems—it did not create them.” 34
127. “First Description, Then Analysis. Another common misconception is that models are not really useful until they can be subjected to quantitative analysis. This is simply not the case. Capture and description are powerful and far-reaching first steps. Just describing something in a formal modeling language like SysML immediately improves communications and understanding. The benefits of this would be difficult to overstate.” 4
128. “Change impact assessments. Changes to the conceptual design continually occurred during early supplier engagements. The Operations Concept Definition (OCD) and Maintenance Concept Definition (MCD), System, Subsystem requirements and interface requirements were analysed by various SMEs and refined. The inherent traceability of the MBSE approach significantly assisted these impact assessments with a near end-to-end visibility from project business requirements to functions and interfaces to system and subsystem requirements. This has enabled trade-offs to be made against the user’s operational requirements and commercial off-the-shelf (COTS) products offered by suppliers, with the intent to minimise customisation in products.” 39
129. “Using Enhanced Functional Flow Block Diagrams (EFFBD) for scenario modelling were found to be novel to the … system acquirer, supplier and user, and considerable time was spent on advocating the benefits of using this method. The best EFFBD scenario review results came from preparing the reviewers with a simple flow block diagram example, explaining the purpose of the information captured in them, and keeping the review session numbers low (1 –3 people). Following a few sessions, majority of the reviewers were able to utilise the diagrams to create a shared understanding of socio-technical interactions between multiple operational and technical stakeholders.” 39
130. “sharing the model with all stakeholders and making it the reference. Once a model is recognized as the reference, it is used as a source for other engineering activities and its existence becomes therefore less likely to be challenged. Evangelization, coaching and MBSE commitment from the management are necessary to reach this goal.” 8

131. “Best practice: Using diagram automatic generation when possible instead of manually maintaining diagrams.” 8
132. Demonstrated “ability to link system elements and components to the requirements and provide end-to-end traceability from the final system architecture, back to the original customer goals.” 18
133. “A model-based architecture [is a] valuable communication device, especially for stakeholders who desire system information to be conveyed quickly and efficiently.” 18
134. “…some short time benefits can be easily obtained. For instance, communication between cross-functional teams, single source of information, traceability between requirements and system artifacts etc. Project planning need take this into account and commit sufficient resources in advance avoiding budget over-runs later.” 41

Programs/Projects Using Methods and Processes:

135. “A crucial basis for MBSE adoption is to define a clear purpose and scope (the why and what). Ideally, it must be precisely described before beginning the deployment. However, this is a challenge in real world applications, where modeling can be used in so many ways.” 13
136. “For new programs the challenge is to gain adoption early during the program definition phase. Short time-to-market leaves little time for learning MBSE once the program begins, yet there may be little demand to learn an MBSE approach prior to having a targeted program for application.” 11
137. “Well-defined modeling objectives and scope are critical to MBSE success. The application of MBSE to a particular project should have a well-defined purpose, objectives, and scope, and the scope should be consistent with the planned resources and schedule.” 19
138. “Indisputably the most important of all best practices is to set clear modeling objectives right from the beginning of the project: identification of inputs and outputs of the model. Ideally, these objectives would be captured in a model management plan, also containing the modeling guidelines.” 8
139. “A third lesson learned is the division of modeling between two geographical diverse teams. … Each team has a lead modeler which has some background in MBSE, and thus, their own ideas of how models should be constructed and how information should be represented. This is a problem since one of the main goals of this modeling work is to link information from the software diagrams to the operational process flows, and vice versa. If two teams are not on the same page, as far as modeling methodology, this could present a problem when it comes time for integration/linking… the modeling team has gone through several iterations of methodology discussion before finally settling into a final methodology. Even with everyone seemingly on the same page, the team still finds it important to meet on a regular basis to reevaluate the modeling standards for the trade study and discuss any issues occurring or that we might see on the horizon.” 3

140. “By providing a graphical, navigable model template with standard notation for architectures and behaviour, development time can be reduced and the benefits of more complex modelling features can be extracted over the course of multiple projects. Best practices can be built into the model template structure.” 22
141. “Replace the "vicious cycle" with the "virtuous cycle" (If a model isn’t being used, it won’t get the resources or attention needed to keep it current and relevant. However, if a model isn’t current and relevant, it won’t be used) (virtuous cycle, in which the models are so highly valued and frequently used that the enterprise is compelled to commit the resources to maintain them, which allows them to retain and expand upon that value, and continue to increase in value).” 34
142. “Just Do It. We’ve found that the best way to figure out how to apply MBSE is to do it for real: make the commitment to adopt MBSE as the way to produce (at least some subset of) the project products, and then figure out how to accomplish this. This is in contrast to the suggestion sometimes made by skeptics, that a “safer” or “more gradual” approach would be to conduct a “shadow” or “parallel” pilot that allows side-by-side comparison of benefits and drawbacks, including cost.” 4
143. “Dashboarding. Dashboards were created for the digital systems model (DSM) to measure the level of coverage and were an effective way of communicating progress and areas of improvement to senior management. Model data in the DSM enabled these representations.” 39
144. “Best practice: Documenting the model. Not only should each model element be correctly and textually described, the global model should be given a reading grid providing external readers a logical path to browse the model.” 8
145. “Maintain the project schedule and ensure it is “trackable”. A project schedule that is trackable is one where the project schedule tasks and deliverables correspond to what people are actually doing on the project. This may sound obvious, but I have been unpleasantly surprised by too many project schedules to assume that this is always the case. Regular and short-term deliverables are essential to letting you know when you are falling behind. Finally, contingency planning needs to be done to investigate what to do when things go wrong.” 23
146. “Acquire the digital artifacts the DoD needs to approve and recompete the fielded system. “Knowing what you know” was a recurring theme in our discussions with stakeholders; data does no good if you cannot find it or do not have the rights to use it. Digital artifacts that the DoD requires to recompete the system should exist within an ASoT that is under the DoD’s control. Mark the digital artifacts approved for integration, and associate with each digital artifact the evidence that justifies that approval. Track the system throughout its lifecycle to identify the as-approved, as-built, as-maintained, and as-destroyed versions of the system. Acquire models to represent legacy components.” 40
147. “CM Can Start Modestly. In thinking about the needs of a Configuration Management (CM) system for our models, we found that the Initial exploration in the IMCE Concept of Operations was helpful. Initially setting up the model to support collaboration, we focused on: structuring modules and packages with

collaboration in mind; and we emphasized single owner packages in topically-defined modules. Model access permissions were set loosely for the time being. Lightweight versioning was found to be sufficient: Teamwork was used to track changes to model elements; DocWeb reports captured snapshot of full model and resource reports; reviewed and baselined versions are tagged as such in DocWeb. Quality Control is developing as needed: scripts are now doing some rudimentary model validation; a hand calculation is used before report release as final correctness check." 4

148. "Exploit "network effects" to accelerate adoption (the more a model is used, the more data it contains and integrates, the more valuable it is)." 34
149. "Model the "T" to explore both breadth and depth. Modeling needs and issues driven by the breadth of model scope were found to differ significantly from those driven by depth of modeling detail. Exploring both of these dimensions of modeling early in the model life cycle was found to mitigate the risk that poor decisions made early in the model's life are allowed to propagate far and wide as the model grows." 34
150. "Alongside with model progress monitoring, it is crucial to organize regular model reviews involving not only model contributors but also domain experts. The model cannot be considered as the reference if all stakeholders are not involved." 8
151. "Iterative Design Approach -Through regular stakeholder meetings to confirm the results of the operational and functional analysis, divergent (and at times conflicting) stakeholder expectations were able to be managed. Presenting previous decisions and the resultant analytical consequences enabled convergence to be achieved when describing the physical domain. A strength of using a model based, vice a paper based, approach enabled a relatively fast turnaround on the effect of decisions made by the key stakeholder group." 17

People in Model-Building Roles:

152. "Success here can be traced to the following factors: A clear objective with modeling from the start of the project creating a clear sieve for identifying information within scope of the model. Strict adherence to a tested pre-defined methodology. One single experienced user who took command of methodology definition (considering the constraints imposed by the tools used) and further refinement of the document generator." 24
153. "To counter the pitfall of capturing several engineering levels in the same model, use models as a means to perform co-engineering. In the case of a transition from a system to a subsystem for example, subsystem stakeholders have to be involved in any decision related to their subsystem and must validate that the high-level view the system stakeholders have is accurate, relevant and feasible." 8
154. "Seek the "killer apps" (specifically identified high impact applications of the methodology that can motivate each stakeholder to make the leap from skeptic to advocate)." 6
155. "We let the discovery of the need drive the solution. There was 'top down' innovation but not in the traditional sense of pre-ordained specifications: it

consisted mainly of constant guidance during the modeling process to keep the effort focused on satisfying the end objectives.”: 4

156. “Real Examples are Powerful. Trying to describe to stakeholders and potential collaborators what MBSE looks and feels like has proven to be rather difficult and not very effective. We have found that many people ‘get it’ for the first time only when they see an actual example.” 4
157. “Always take extreme care when showing diagrams to persons not introduced to SysML! Systems Engineers are nowadays used to MS Visio diagrams in which an exact and precise meaning is not associated with each type of arrows and blocks, and in which the purpose of the diagram is supporting a paragraph, or a document. This leads to misunderstandings, confusion and misinterpretations of the SysML diagrams. When a modeller presents a diagram to other engineers, he or she needs to make sure, before even explaining the diagram itself, that the others understood: the purpose of the diagram (e.g. showing architecture and not functions), what each type of element represents (e.g. physical block), and the meaning of each link (e.g. composition).” 32

LESSONS LEARNED REFERENCES TO THIS SECTION

1. Amorim, T., Vogelsang, A., Pudlitz, F., Gersing, P., & Philipps, J. (2019, May). Strategies and best practices for model-based systems engineering adoption in embedded systems industry. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP) (pp. 203-212). IEEE.
2. Andersson, H., Herzog, E., Johansson, G. and Johansson, O. (2010), Experience from introducing Unified Modeling Language/Systems Modeling Language at Saab Aerosystems. Syst. Engin., 13: 369-380. https://doi.org/10.1002/sys.20156
3. Barnes, P. (2012), 11.5.2 A NASA Space Communications and Navigation SysML Profile Adaptation. INCOSE International Symposium, 22: 1711-1725. https://doi.org/10.1002/j.2334-5837.2012.tb01432.x
4. Bayer, T., Chung, S., Cole, B., Cooke, B., Dekens, F., Delp, C., Gontijo, I., Lewis, K., Moshir, M., Rasmussen, R. and Wagner, D. (2012), 11.5.1 Early Formulation Model-Centric Engineering on NASA's Europa Mission Concept Study. INCOSE International Symposium, 22: 1695-1710. https://doi.org/10.1002/j.2334-5837.2012.tb01431.x
5. Bayer, T.J., Chung, S., Cole, B., Cooke, B., Dekens, F., Delp, C., Gontijo, I. and Wagner, D. (2013), Update on the Model Based Systems Engineering on the Europa Mission Concept Study. INCOSE International Symposium, 23: 694-709. https://doi.org/10.1002/j.2334-5837.2013.tb03048.x
6. Blackburn, M.R., McDermott, T., Kruse, B., Dzielski, J. and Hagedorn, T. (2022), Digital Engineering Measures Correlated to Digital Engineering Lessons Learned from Systems Engineering Transformation Pilot. INSIGHT, 25: 61-64. https://doi.org/10.1002/inst.12375

7. Bone, M., & Cloutier, R. (2010, March). The current state of model based systems engineering: Results from the omg™ sysml request for information 2009. In Proceedings of the 8th conference on systems engineering research.
8. Bonnet, S., Voirin, J. L., Normand, V., & Exertier, D. (2015, October). Implementing the mbse cultural change: Organization, coaching and lessons learned. In INCOSE International Symposium (Vol. 25, No. 1, pp. 508-523).
9. Booth, S.T. (2005), 4.2.3 Network Centric Architectures:Are We Up To The Task?. INCOSE International Symposium, 15: 600-609. https://doi.org/10.1002/j.2334-5837.2005.tb00693.x
10. Chami, M., & Bruel, J. M. (2018). A survey on MBSE adoption challenges.
11. Carson, R. S. (2015). MBSE Implementation across diverse domains. Insight, 18(2), 22-23.
12. Carson, R.S. and Sheeley, B.J. (2013), 2.5.1 Functional Architecture as the Core of Model-Based Systems Engineering. INCOSE International Symposium, 23: 29-45. https://doi.org/10.1002/j.2334-5837.2013.tb03002.x
13. Chami, M., Aleksandraviciene, A., Morkevicius, A., & Bruel, J. M. (2018, July). Towards solving MBSE adoption challenges: the D3 MBSE adoption toolbox. In INCOSE International Symposium (Vol. 28, No. 1, pp. 1463-1477).
14. Cloutier, R. (2015). Current modeling trends in systems engineering. Insight, 18(2), 10-13.
15. Do, Q., Lovric, B. and Cook, S. (2011), 4.6.3 An MBSE Feasibility Study using the Systems Integration Sandpit. INCOSE International Symposium, 21: 523-533. https://doi.org/10.1002/j.2334-5837.2011.tb01222.x
16. Do, Q. and Cook, S. (2012), 10.5.1 An MBSE Case Study and Research Challenges. INCOSE International Symposium, 22: 1531-1543. https://doi.org/10.1002/j.2334-5837.2012.tb01420.x
17. Do, Q. and Hartigan, D. (2017), An MBSE Methodology for Capability Systems Definition. INCOSE International Symposium, 27: 613-626. https://doi.org/10.1002/j.2334-5837.2017.00382.x
18. Foster, M. (2011). Developing a heavy lift launch vehicle systems architecture using a model based systems engineering approach. In AIAA SPACE 2011 Conference & Exposition (p. 7154).
19. Friedenthal, S. (2009). SysML: Lessons from early applications and future directions. Insight, 12(4), 10-12.
20. Friedland, B., Malone, R., & Herrold, J. (2016, July). Systems engineering a model based systems engineering tool suite: The Boeing approach. In INCOSE International Symposium (Vol. 26, No. 1, pp. 386-398).
21. Gordon, S., Oswalt, I. and Cooley, T. (2017). Why Spend One More Dollar for M&S? Observations on the Return of Investment. In The Profession of Modeling and Simulation (eds A. Tolk and T. Oren). https://doi.org/10.1002/9781119288091.ch14
22. Gregory, J., Berthoud, L., Tryfonas, T., Rossignol, A., & Faure, L. (2020). The long and winding road: MBSE adoption for functional avionics of spacecraft. Journal of Systems and Software, 160, 110453.
23. Hause, M. (2010), 2.1.2 The UPDM RFC Development Project An Exercise in Model-Based Virtual Team Development or “Practicing What We Preach”. INCOSE

International Symposium, 20: 187-201. https://doi.org/10.1002/j.2334-5837.2010.tb01064.x
24. Herzog, E., Hallonquist, J. and Naeser, J. (2012), 4.5.1 Systems Modeling with SysML – an experience report. INCOSE International Symposium, 22: 600-611. https://doi.org/10.1002/j.2334-5837.2012.tb01359.x
25. Huldt, T. and Andersson, J. (2009), 6.1.3 Model-based Technical Planning: An evaluation of description techniques. INCOSE International Symposium, 19: 894-908. https://doi.org/10.1002/j.2334-5837.2009.tb00990.x
26. Huldt, T, Stenius, I. State-of-practice survey of model-based systems engineering. Systems Engineering. 2019; 22: 134– 145. https://doi.org/10.1002/sys.21466
27. Karban, R., Zamparelli, M., Bauvir, B. and Chiozzi, G. (2012), 10.5.2 Three years of MBSE for a large scientific programme: Report from the Trenches of Telescope Modeling. INCOSE International Symposium, 22: 1544-1558. https://doi.org/10.1002/j.2334-5837.2012.tb01421.x
28. Llano, M.I. and Quappen, G. (2020), MBSE Methodology Applied for a Technology Demonstrator in the Space Transportation Sector. INCOSE International Symposium, 30: 1474-1488. https://doi.org/10.1002/j.2334-5837.2020.00798.x
29. Lu, J., Wen, Y., Liu, Q., Gürdür, D., & Törngren, M. (2018, July). MBSE applicability analysis in Chinese industry. In INCOSE International Symposium (Vol. 28, No. 1, pp. 1037-1051).
30. Madni, AM, Sievers, M. Model-based systems engineering: Motivation, current status and research opportunities. Systems Engineering. 2018; 21: 172– 190. https://doi.org/10.1002/sys.21438
31. Marco, A., Angelo, C., Elena, L.M., Scarl, D. and Elena, V. (2012), 4.4.2 MBSE Feasibility Study to improve PLM Business Solution System Specification and Design. INCOSE International Symposium, 22: 574-588. https://doi.org/10.1002/j.2334-5837.2012.tb01357.x
32. Maurandy, J., Helm, A., Gill, E. and Stalford, R. (2012), 11.5.3 Cost-Benefit Analysis of SysML Modelling for the Atomic Clock Ensemble in Space (ACES) Simulator. INCOSE International Symposium, 22: 1726-1745. https://doi.org/10.1002/j.2334-5837.2012.tb01433.x
33. Mitchell, S. W. (2014). Transitioning the SWFTS program combat system product family from traditional document-centric to model-based systems engineering. Systems Engineering, 17(3), 313-329.
34. Noguchi, R.A. (2019), Recommended Best Practices based on MBSE Pilot Projects. INCOSE International Symposium, 29: 753-770. https://doi.org/10.1002/j.2334-5837.2019.00633.x
35. Papke, B. L., Wang, G., Kratzke, R., & Schreiber, C. (2020, July). Implementing MBSE–An Enterprise Approach to an Enterprise Problem. In INCOSE International Symposium (Vol. 30, No. 1, pp. 1550-1567).
36. Parrott, E. (2016, May). The value of successful MBSE adoption. In No Magic World Symposium 2016 (No. GRC-E-DAA-TN32000).
37. Peak, R., McGinnis, L. and Paredis, C. (2015), Academia as an Educator in the World of MBSE. INSIGHT, 18: 30-34. https://doi.org/10.1002/inst.12018
38. Pihera, L.D., Ender, T., Taylor, B., Bollweg, N.R., Lopez, A., Paden, M.L. and King, S. (2012), 4.2.2 Application of Systems Engineering to Improve Extracorporeal

Membrane Oxygenation (ECMO) Therapy. INCOSE International Symposium, 22: 490-504. https://doi.org/10.1002/j.2334-5837.2012.tb01351.x
39. Roodt, D., Nadeem, M. and Vu, L.-T. (2020), Model-Based Systems Engineering for complex rail transport systems – A case study. INCOSE International Symposium, 30: 1581-1595. https://doi.org/10.1002/j.2334-5837.2020.00805.x
40. Smith, T., Payne, C. and Shackleton, J. (2022), Constructing an Authoritative Source of Truth in a Changing Information Landscape. INSIGHT, 25: 29-33. https://doi.org/10.1002/inst.12369
41. Suryadevara, J., & Tiwari, S. (2018, December). Adopting mbse in construction equipment industry: An experience report. In 2018 25th Asia-Pacific Software Engineering Conference (APSEC) (pp. 512-521). IEEE.
42. Trauth, S. L., Carroll, E. R., Matta, A. P., Salguero, L. M., & Griego, R. M. (2017). MBSE Culture Configuration and Integration At Sandia National Laboratories (No. SAND2017-6561PE). Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).
43. Tsui, R., Davis, D. and Sahlin, J. (2018), Digital Engineering Models of Complex Systems using Model-Based Systems Engineering (MBSE) from Enterprise Architecture (EA) to Systems of Systems (SoS) Architectures & Systems Development Life Cycle (SDLC). INCOSE International Symposium, 28: 760-776. https://doi.org/10.1002/j.2334-5837.2018.00514.x
44. Vogelsang, A., Amorim, T., Pudlitz, F., Gersing, P., & Philipps, J. (2017, November). Should I stay or should I go? On forces that drive and prevent MBSE adoption in the embedded systems industry. In International Conference on Product-Focused Software Process Improvement (pp. 182-198). Springer, Cham.
45. Wang, L., Izygon, M., Okon, S., Wagner, H., & Garner, L. (2016). Effort to accelerate MBSE adoption and usage at JSC. In AIAA SPACE 2016 (p. 5542).
46. Weiland, KJ, Knizhnik, JRL Design thinking, lean startup, and high-technology marketing for human-centered systems engineering. Systems Engineering. 2022; 25: 207– 223. https://doi.org/10.1002/sys.21612
47. Younse, PJ, Cameron, JE, Bradley, TH Comparative analysis of a model-based systems engineering approach to a traditional systems engineering approach for architecting a robotic space system through knowledge categorization. Systems Engineering. 2021; 24: 177– 199.

CONCLUSION

Computer Aided Design (CAD) and Product Line Management (PLM) methods and tools gradually moved the physical design processes from highly manual to mostly seamless and efficient sets of data and workflow. This “Mechanical Engineering Modernization” took decades of evolution. SE Modernization is also going to be a long-term process.

This report summarizes the initial roadmaps and pain points for the journey. The question “why modernize systems engineering” proved to be very difficult to answer originally. Different aspects of systems engineering such as mission integration, digital systems engineering, and agile systems engineering are evolving differently in different disciplines, creating organizational and process barriers in both engineering and acquisition.

A primary goal of this phase of SE Modernization framework was to develop the integration framework that would bring these disciplinary and methodological views together. To do this we had to create a new mental model, The Supra-system Model, and set of visualizations that portray systems engineering in a different light. The basic process areas of systems engineering remain valid, but the lifecycle models and associated digital practices will look very different than the traditional underlying SE process in defense acquisition cycles.

As mentioned at the beginning, this full report was written in seven standalone parts in order to facilitate the use of each part in the SEMOD body of knowledge.

PROJECT TIMELINE & TRANSITION PLAN

This completes the initial phase of SEMOD research under research tasks WRT-1051 and WRT-1058. Transition plans are defined in the SE Modernization roadmaps in Part 2.

ACKNOWLEDGEMENTS

This research was conducted as team effort between SERC researchers and the government SE Modernization team. Thank you to Nadine Geier for sponsoring and leading this work. Thank you to Kelly Alexander for managing the government side of the project. Thank you to Monique Ofori, Edward Moshinsky, and Michael Guba for supporting this work.