



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

**SYSTEMS ENGINEERING
CAPSTONE REPORT**

**A STUDY OF MBSE THROUGH THE DEVELOPMENT
OF MODELING AND DATA EXCHANGE PROCESSES**

by

William Emeny, Lance Lowenberg, Lynn P. Nguyen,
Ryan C. Robar, Michael H. Rubow, and Dustin G. Talley

December 2020

Advisor:

Brigitte T. Kwinn

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE December 2020		3. REPORT TYPE AND DATES COVERED Systems Engineering Capstone Report
4. TITLE AND SUBTITLE A STUDY OF MBSE THROUGH THE DEVELOPMENT OF MODELING AND DATA EXCHANGE PROCESSES			5. FUNDING NUMBERS	
6. AUTHOR(S) William Emeny, Lance Lowenberg, Lynn P. Nguyen, Ryan C. Robar, Michael H. Rubow, and Dustin G. Talley				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) The DoN is undergoing a digital transformation that is set to address the needs of sustaining fleet assets for extended periods of time, while maintaining a superior lethality. Within the engineering domain, the DoN is starting to identify MBSE tools and concepts to streamline processes and enhance capability. The capstone looked to lay the foundation for a conceptual system model development process that utilizes SysML and OOSEM to produce system model data and artifacts derived from a single scenario. During the digital transformation, communication of system model data to stakeholders was identified as a need and a SysML tool was used to generate model-based documentation from a formatted Microsoft Word document. With incoming digital product support capabilities from the MBPS program, communication from an MBSE environment is critical and requires XML formatted data. Using the information collected in the completion of the scenario, it was discovered that SysML elements will lose their SE-specific stereotypes when converted directly into XML format. To counter this the capstone developed UML instances derived from the S3000L UML class-based data model to be converted into XML format. The findings and developments of this capstone support the ability for organizations to standardize the way system modeling data is developed, collected, and communicated to other systems external to the engineering domain.				
14. SUBJECT TERMS Department of Navy, DoN, model-based systems engineering, MBSE, model-based product support, MBPS, extensible markup language, XML, systems modeling language, SysML, system modeling, digital engineering, unified modeling language, UML, object-oriented systems engineering methodology, OOSEM, digital transformation, model-based engineering, S-series specification, S3000L, digital integration, model-based enterprise			15. NUMBER OF PAGES 111	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified		18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**A STUDY OF MBSE THROUGH THE DEVELOPMENT OF MODELING
AND DATA EXCHANGE PROCESSES**

William Emeny, Lance Lowenberg, Lynn P. Nguyen,
Ryan C. Robar, Michael H. Rubow, and Dustin G. Talley

Submitted in partial fulfillment of the
requirements for the degrees of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

and

MASTER OF SCIENCE IN ENGINEERING SYSTEMS

from the

**NAVAL POSTGRADUATE SCHOOL
December 2020**

Lead Editor: William Emeny

Reviewed by:
Brigitte T. Kwinn
Advisor

Accepted by:
Ronald E. Giachetti
Chair, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The DoN is undergoing a digital transformation that is set to address the needs of sustaining fleet assets for extended periods of time, while maintaining a superior lethality. Within the engineering domain, the DoN is starting to identify MBSE tools and concepts to streamline processes and enhance capability. The capstone looked to lay the foundation for a conceptual system model development process that utilizes SysML and OOSEM to produce system model data and artifacts derived from a single scenario. During the digital transformation, communication of system model data to stakeholders was identified as a need and a SysML tool was used to generate model-based documentation from a formatted Microsoft Word document. With incoming digital product support capabilities from the MBPS program, communication from an MBSE environment is critical and requires XML formatted data. Using the information collected in the completion of the scenario, it was discovered that SysML elements will lose their SE-specific stereotypes when converted directly into XML format. To counter this the capstone developed UML instances derived from the S3000L UML class-based data model to be converted into XML format. The findings and developments of this capstone support the ability for organizations to standardize the way system modeling data is developed, collected, and communicated to other systems external to the engineering domain.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION AND BACKGROUND.....	1
A.	INTRODUCTION.....	1
B.	PROBLEM OVERVIEW.....	2
C.	BACKGROUND	2
D.	PROBLEM STATEMENT	5
E.	PROJECT OBJECTIVES.....	6
F.	PROJECT SCOPE.....	6
G.	CHAPTER SUMMARY.....	7
II.	PROBLEM ANALYSIS	9
A.	STAKEHOLDERS	9
B.	IMPORTANT DEFINITIONS AND TERMS	11
1.	Model-Based Systems Engineering	11
2.	Model-Based Product Support	11
3.	Architecture Framework:	11
4.	Enterprise Technical Reference Framework (ETRF).....	12
5.	Digital information technology (IT) transformation	12
6.	Systems Modeling Language (SysML).....	14
C.	LITERATURE REVIEW SUMMARY	14
1.	Purpose.....	14
2.	U.S. Navy Legacy Approach	15
3.	Capability Gaps in Legacy Approach.....	17
4.	DoD Model Based Systems Engineering.....	18
5.	USN Model Based Product Support.....	19
D.	CHAPTER SUMMARY.....	25
III.	SYSTEMS ENGINEERING APPROACH	27
A.	SYSTEMS ENGINEERING PROCESS.....	27
B.	SCRUM FRAMEWORK	28
C.	CONFIGURATION MANAGEMENT	30
D.	CHAPTER SUMMARY.....	32
IV.	MODEL DEVELOPMENT	33
A.	SYSTEM MODEL DEVELOPMENT PROCESS	33
1.	Simulation Scenarios	34
B.	OBJECT ORIENTED SYSTEMS ENGINEERING METHOD	35
C.	PROBLEM DEFINITION AND ANALYSIS	37

D.	MISSION REQUIREMENTS GENERATION.....	39
1.	Simulation Results	43
E.	SYSTEM SPECIFICATION DERIVATION	50
1.	Simulation Results	54
F.	MODEL SUMMARY	60
G.	CHAPTER SUMMARY.....	61
V.	MODEL FINDINGS.....	63
A.	DATA EXCHANGES BETWEEN DOMAINS	63
B.	USEFUL ENGINEERING ARTIFACT CREATION	66
1.	Artifacts Supporting Logistics Support Analysis (LSA) & S3000L.....	66
2.	Model-based Documentation Generation.....	69
C.	CHAPTER SUMMARY.....	70
VI.	CONCLUSIONS AND RECOMMENDATIONS.....	71
A.	A SUMMARY OF PROJECT OBJECTIVES.....	71
B.	DEFINING, DEVELOPING, AND IMPORTANCE OF THE SYSTEM MODEL	71
C.	CONCLUSIONS	74
D.	RECOMMENDATIONS.....	75
APPENDIX.	LITERATURE REVIEW MATRIX	77
1.	Research Questions.....	78
2.	Types of Literature Reviewed.....	79
3.	Topics of Review	79
	LIST OF REFERENCES	83
	INITIAL DISTRIBUTION LIST	87

LIST OF FIGURES

Figure 1.	Engineering MBSE Models Source: US DoD (2018).	19
Figure 2.	Capstone System Engineering Process.	28
Figure 3.	Scrum Framework. Source: Scrum.org (2020).	30
Figure 4.	Overall System Model Development Process.....	34
Figure 5.	OOSEM Specify and Design Process. Adapted from Friedenthal, Moore, and Steiner (2012).	36
Figure 6.	Problem Definition and Analysis Activity Diagram.....	37
Figure 7.	Analysis of the Stakeholders.....	39
Figure 8.	Mission Requirements Activity Diagram.	40
Figure 9.	Mission Requirements Activity Diagram Expanded.	41
Figure 10.	Mission Requirements Activity Diagram Expanded 2.	41
Figure 11.	Problem Definition and Analysis Expanded.....	42
Figure 12.	Stakeholder Viewpoint (Example).....	44
Figure 13.	Contextual Architecture.	45
Figure 14.	Stakeholder Needs Model.	46
Figure 15.	Example of a Requirement Traceability Matrix.	46
Figure 16.	Example MOE Table.	47
Figure 17.	Engineering Analysis Criteria/Selection Criteria.....	48
Figure 18.	Boundary Decomposition.	49
Figure 19.	Mission Requirements: Requirements Model.....	49
Figure 20.	DEE Use Case Diagram.....	50
Figure 21.	System Specification: Derivation Activity Diagram	51
Figure 22.	System Specification: Decompose Architecture.....	52

Figure 23.	System Specification: Black Box System Specification.....	53
Figure 24.	System Specification: Functional Requirements.	54
Figure 25.	External System (MBPS) Model.	56
Figure 26.	Mission Free Form Diagram: Critical/Common Mission Scenarios.	57
Figure 27.	Behavior Diagram: Updating SE Data Points and Artifacts.	57
Figure 28.	Interface Diagram.	58
Figure 29.	DEE System Specification.....	59
Figure 30.	System Specification: Function Requirements.	60
Figure 31.	In-Service Data Analysis Process as an Example of S3000L Feedback. Source: ASD/AIA (2018).	63
Figure 32.	S3000L Data Exchanges. Source: ASD/AIA (2018).	64
Figure 33.	User Capacity MOE Specification in SysML.....	65
Figure 34.	XML Data Table of User Capacity.....	65
Figure 35.	Example of Producing a Class Instance in a System Model with SysML.....	66
Figure 36.	The Uses of Different Domain Data Sets for S3000L Processes. Source: ASD/AIA (2014).	67
Figure 37.	S3000L FMEA Development Process. Source: ASD/AIA (2014).	68
Figure 38.	Example of Functional Design FMEA within the System Modeling Tool.....	68
Figure 39.	FMEA BDD within System Modeling Tool.....	69
Figure 40.	High-Level Concept of Generating Model-Based Documents from the System Model.	70
Figure 41.	Proposed Conceptual System Data Model.....	72
Figure 42.	Transformation of a Conceptual Action into a Logical Sequence of Signals.....	73

LIST OF TABLES

Table 1.	Stakeholder Description.....	10
Table 2.	Alignment of DoD CIO Objectives to JIE Capability Objectives and Initiatives. Source: USDoD (2019).....	13

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AEL	allowance equipage list
APL	allowance parts list
ASD/AIA	AeroSpace and Defense Industries Association of Europe and Aerospace Industries Association
ATIS	Advanced Technical Drawing Information System
AWARE	Warfighter analytics readiness environment
CDMD-OA	Configuration data managers database-open architecture
CI	Configuration Item
CSOSS	Combat systems operational sequencing system
DE	digital engineering
DoD	Department of Defense
ECP	engineering change proposal
EOSS	engineering operational sequencing system
ETRF	enterprise technical reference framework
FFD	free form diagram
FTA	fault tree analysis
ICAPS	interactive computer aided provisioning
IETM	interactive electronic technical manual
ILS	integrated logistics services
IT	information technology
LOG IT	Logistics IT
LORA	level of repair analysis
MBPS	Model Based Product Support
MBSE	Model-based systems engineering
MIP	maintenance index page
MRC	maintenance requirement card
MRDB	material readiness database
MRO	maintenance, repair, and overhaul
NAVLOGTD	Navy logistics technical data
NDART	Navy data acquisition requirements tool

NDE	Navy database environment
NMRO	Navy MRO
NOBLE	Naval operational business logistics enterprise
NSDSA	Naval systems data support activity
NSEDR	Naval ships engineering drawing repository
NSWC PHD	Naval Surface Warfare Center, Port Hueneme Division
OMMS-NG	organizational maintenance management system-next generation
ORDALT	ordnance alteration
PMS	planned maintenance system
PMSMIS	planned maintenance system management information system
PPI	provisioning-parts information
PS	product support
PTD	provisioning technical data
RCM	reliability centered maintenance
RMA	reliability, maintainability, availability
RTM	requirement traceability matrix
SCD	ship change document
SCIP	ship change installation procedure
SME	subject matter expert
SOI	system of interest
SysML	systems modeling language
TDMIS	technical data management information system
TDP	technical data package
TM	technical manual
UML	unified modeling language
3D	three-dimensional

EXECUTIVE SUMMARY

Currently, there is an initiative to transform legacy logistics information technology (IT) systems to use a model-centric approach to support products that aims to increase system uptime and reduce support costs. Model Based Product Support (MBPS) is a single piece of a larger digital readiness vision that includes new capabilities, such as predictive analytics, data-as-a-service, platform-as-a-service, process automation, and the integration of data across multiple platforms (National Shipbuilding Research Program [NSRP] 2019). This vision of a logistics digital transformation is shown in Figure 1. The new integrated product lifecycle management (PLM) platform supports the sharing of a standardized data model that enables the capability to perform logistics support analysis. The PLM platform inside the product support (PS) domain would have conduits with the engineering, maintenance, training, and other system lifecycle communities to support better logistics models and better supported systems (NSRP 2019).

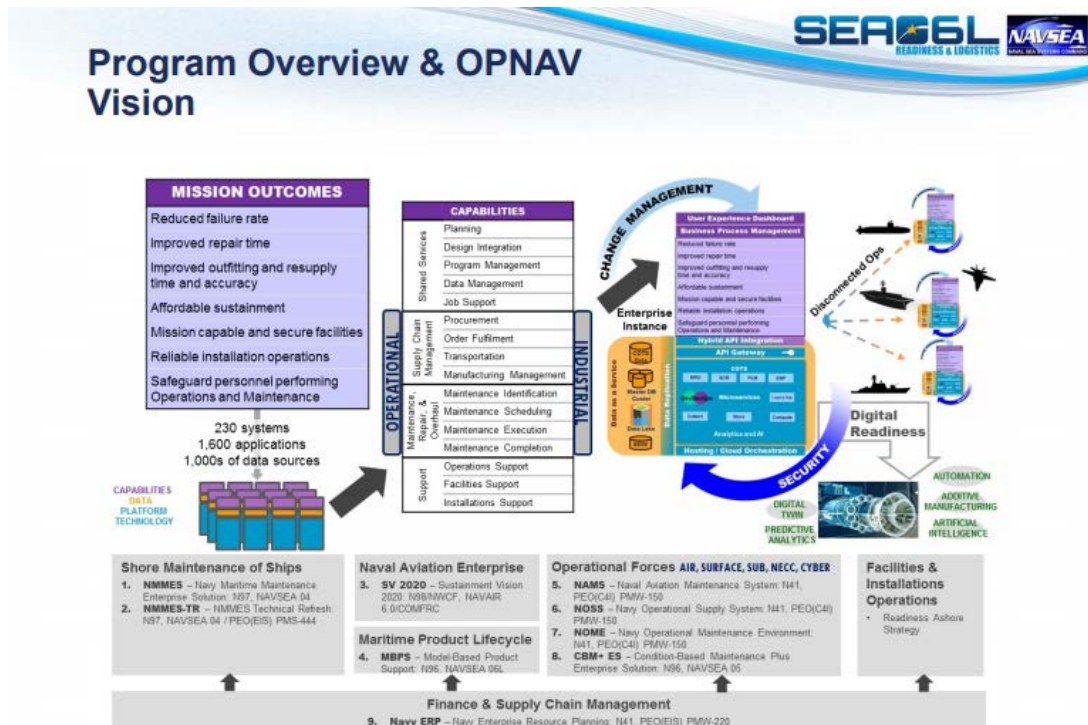


Figure 1. Logistics Digital Transformation Vision Overview. Source: NSRP (2019).

The current transformation occurring in the PS domain is also being pursued within the engineering domain with the exploration and implementation of model-based systems engineering (MBSE) concepts. Department of Defense (DoD) strategic documents have expressed the need that as systems become more complex, the DoD will require more robust engineering practices to develop weapon systems and maintain superiority over our enemies (Engineering 2018). For many years, the DoD has relied on document-based, stovepiped engineering processes and is now looking to incorporate digital engineering practices to work more efficiently. The incorporation of digital engineering will require investment in new methods, processes, and tools in order to enable systems to become more lethal and affordable (Engineering 2018). The Department of Navy (DoN) has embraced the goals set by the DoD Digital Engineering Strategy by developing its own set of high-level strategic documentation that discusses high-level implementation strategies and their alignment to the DoD documentation (Department of Navy (DoN) 2020).

One of the alignment goals set in the DoD Digital Engineering Strategy and envisioned in the DoN Digital Systems Engineering Transformation Strategy is the formalization of the development, integration, and use of models. Using the system modeling language (SysML) and SysML tools, the capstone group built a conceptual system model development process based off the object-oriented systems engineering methodology (OOSEM). The OOSEM is a top-down, scenario-driven approach that leverages object-oriented concepts and other modeling techniques to support in the development of a more flexible and extensible system architecture that can accommodate the constant change in requirements or technologies (Friedenthal, Moore, and Steiner 2012). The developed process encapsulates system modeling data within what is known in SysML as blocks, analogous to classes within the unified modeling language (UML).

The conceptual system modeling process was developed and an example scenario was completed in which an organization has a need to develop and implement a model-based system engineering environment; henceforth named the Digital Engineering Environment (DEE), locally within the organization. The scenario walks through the development of the conceptual system model and pieces of the logical system model prior to a request for proposal (RFP) where vendors would bid on to develop a physical product

based off the information presented to the vendor in the conceptual system model. The conceptual data model, shown in Figure 2, displays the type of models and artifacts that make up the system model and how they contribute to the development of the system of interest. The information and artifacts captured in the data model are developed within the system modeling process described in this capstone report.

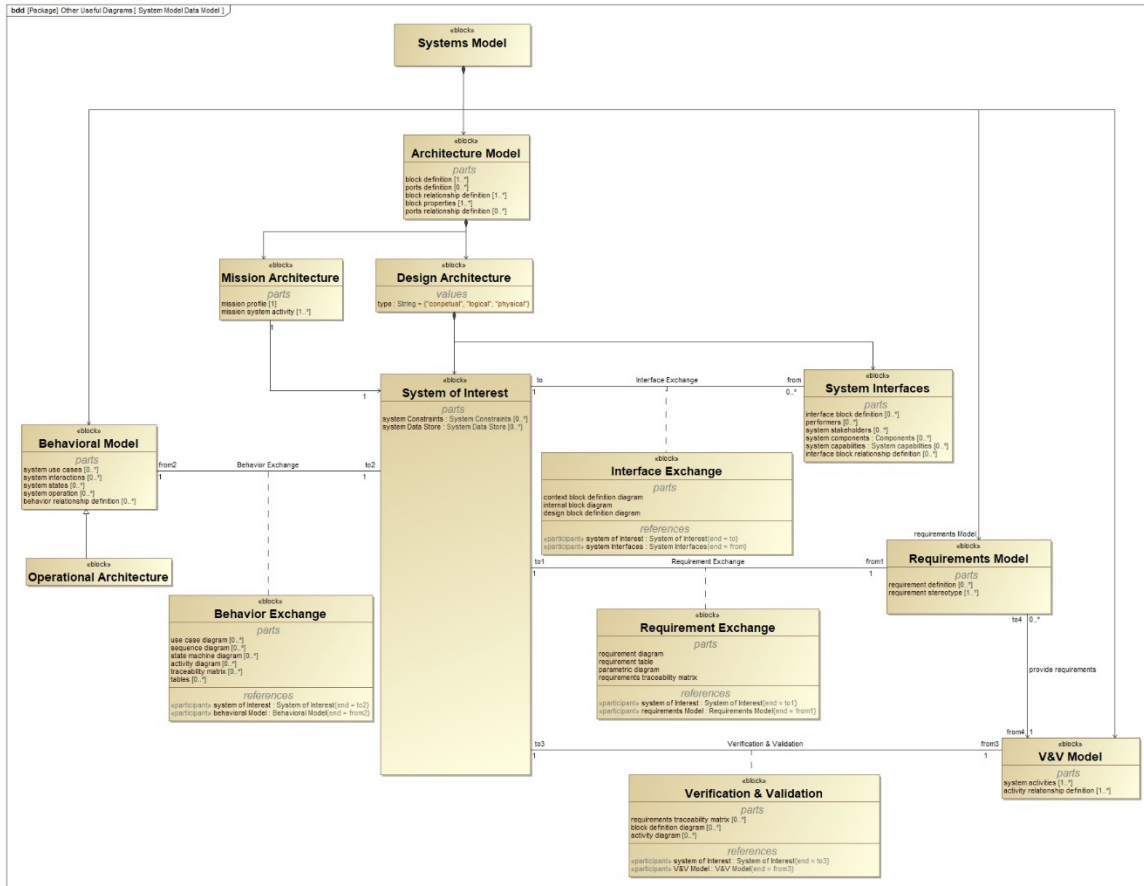


Figure 2. Proposed Conceptual Data Model for Developing a Systems Model.

System model data collected over the design and development phases of a system must be capable of being consumed and of use to the PS domain to enable the reuse of system data for supportability analyses. The MBPS program overview presentation displayed the program’s use of the S-Series specifications developed by the AeroSpace and Defense Industries Association of Europe and Aerospace Industries Association (ASD/AIA). These specifications layout an extensible markup language (XML) schema with data

classes useful for different types of PS efforts, including provisioning, maintenance task analysis (MTA), level of repair analysis (LORA), software support analysis, and other logistics support analyses. There is not a current mapping between the data elements within SysML to the UML data elements within the S-Series specification; however, the developers of the specifications have developed a data model, which can be consumed and useful to a model developed in a SysML toolset. As shown in Figure 3, element instances contain the useful PS data which, if contained within an isolated model, could be manually translated into XML and exported to the S-Series database for analysis use.

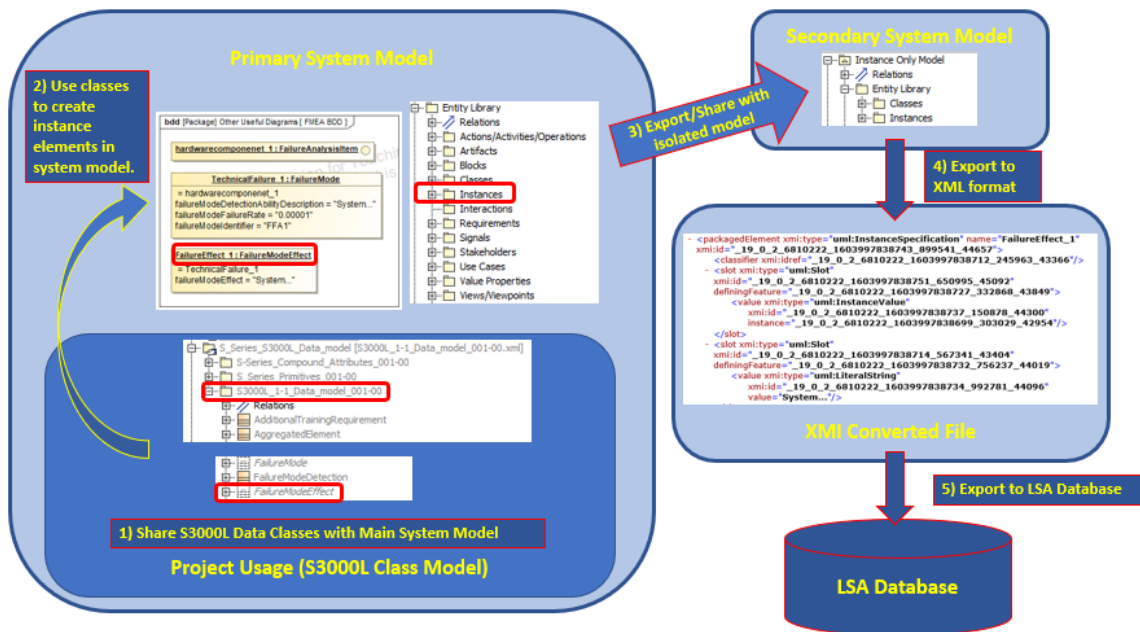


Figure 3. SysML Instances Translated into XML File Format for MBPS Consumption.

Stakeholders were interested in verification of the system design which was supported with the presentation of system model data. Many stakeholders do not have experience in using system modeling tools but are familiar with many of the presentation formats within the model. Many system modeling tools have the capability of developing model-based documentation. Some of the presentation views within the developed model for the capstone’s scenario were utilized to develop a model-based concept of operations (CONOPS). The CONOPS document template was downloaded from public online

sources and configured using the velocity template language (VTL) to place model presentation artifacts into the CONOPS, automatically, upon a click of a button (Department of Veteran Affairs [VA] n.d.).

The model building process does explain the development of a conceptual data model but describes very little work on the development of a logical system model and does not approach the physical model development phase. More research is needed to understand the interfaces with other digital engineering tools and how related data can be used to further define certain aspects of the system model. The process completed a scenario in which useful products were developed for demonstration. To ensure its validity, verification and validation of the proposed process should occur using pilot projects to identify and fix any demonstrated gaps within the process. Future work should include the implementation of another scenario in which a fielded system wishes to undergo a system change. This scenario would require the system model to be updated and used to perform alternative analysis in both the engineering and PS domains.

The resulting scenario provided a collection of data points that represents different SOI viewpoints and that could be used within alternate domains to perform analyses. The conceptual system model in this instance would solely be used to demonstrate a problem and need to a design team or vendor. The instance of a problem would be derived from the technical capability audit (TCA) within the developed process whose following steps would be used to collect data and build presentation views. With the emergence of system of systems (SoS) modeling, it is theorized that existing and anticipated emerging gaps could also be a source of problems in which a TCA could be utilized to determine the necessary solution type (Mohammadi, Elyasi, and Kiasari 2014). Future work could explore the use of a TCA to identify future capability gaps as a second scenario to validate SysML models presented in this capstone. Using SysML and tailoring a process derived from the object-oriented system engineering methodology (OOSEM), enabled the encapsulation of system model data into a single SOI model element to communicate a design's architecture, behavior, requirements, and verification and validation activities. Review of the data developed during the simulation and the S3000L data model shows that there is a need for engineering data (Aerospace and Defense Industries Association of

Europe and Aerospace Industries Association (ASD/AIA) 2014). The capstone presented a way to translate information from SysML into XML, but more work is needed to develop a data mapping to the S3000L XML data model that could lead to an automated conversion process.

References

- AeroSpace and Defense Industries Association of Europe and Aerospace Industries Association. *International procedure specification for Logistics Support Analysis LSA*. Issue No. 1.1. July 2014. <http://www.s3000l.org/docs/S3000L-Issue%201.1.pdf>
- Department of the Navy. 2020. *United States Navy & Marine Corps Digital Systems Engineering Transformation Strategy*. Washington, DC. Office of the Deputy Assistant Secretary of the Navy Research, Development, Test and Evaluation. <https://go.usa.gov/xfQpx>.
- Department of Veteran Affairs, n.d. “Concept of Operations (CONOPS).” Accessed October 5, 2020. <https://www.voa.va.gov/documentview.aspx?documentid=46#:~:text=The%20Concept%20of%20Operations%20or,will%20be%20employed%20and%20supported.&text=It%20is%20used%20as%20a,project%20planning%20and%20decision%20making>.
- Friedenthal, S, A Moore, and R Steiner. (2012). *A practical guide to SysML the systems modeling language* (2nd ed.). San Francisco: Morgan Kaufmann.
- Mohammadi, Mehdi, Mahdi Elyasi, and Mostafa Mohseni Kiasari. 2014. “Developing a Model for Technological Capability Assessment Case of Automotive Parts Manufacturing in Iran.” *International Journal of Innovation and Technology Management* 11 (2) (Winter): 1-19. 10.1142/S021987701450014X.
- National Shipbuilding Research Program. 2019. “Model Based Product Support (MBPS) Overview.” July 18, 2019. https://www.nsrp.org/wp-content/uploads/2019/08/05-MBPS_Overivew_June-2019-Updated_v5.pdf.
- US DoD, Department of Defense Digital Engineering Strategy. June 2018. <https://fas.org/man/eprint/digeng-2018.pdf>.

ACKNOWLEDGMENTS

The A-team would like to sincerely thank the leadership and project vision owners at the Naval Surface Warfare Center, Port Hueneme Division, for their support of our capstone project. The A-team would also like to thank the Systems Engineering Department at the Naval Postgraduate School for their support and expertise opinions throughout our capstone. Special thanks to Professor Bridgette Kwinn for guiding us through this journey and for her contributions to this capstone.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION AND BACKGROUND

A. INTRODUCTION

This project demonstrates a process that gives United States Navy (USN) organizations the capability to develop a conceptual system model, whose data can be used to initiate digital twin and digital thread capabilities. The process outlined in the appending pages is meant to be the foundation for creating the conceptual data model that would be created and matured over the life cycle of the system. This process utilizes the early steps of the object-oriented systems engineering methodology (OOSEM) approach, a model-based system design approach, as a guide in its design with the expectation that it will be used to assist Department of Navy (DoN) organizations in better defining and presenting conceptual system needs and requirements to design agents (Friedenthal, Moore, and Steiner 2012). Process gaps within OOSEM were identified and tailored to better suit the needs of our stakeholders. For example, the project implements a data-driven approach to problem definition, something that is not included in OOSEM. To fulfill this capability, the technical capability audit (TCA) was added to the process. The TCA uses both quantitative and qualitative data from questionnaire or survey data to determine the type of problem the organization is facing (Mohammadi 2014). Appended sections further expand upon this with the descriptions and applications of the technical capability audit (TCA) to perform problem analysis and parametric modeling for engineering analysis. At the conclusion of specified steps in the presented process the modeler will have gathered enough data to enable the development of presentation artifacts. The systems modeling language (SysML) was utilized as the data model, while Cameo Enterprise Architecture (CEA) was used to produce SysML presentation artifacts. The produced artifacts were used as the process verification method and was performed using a generalized scenario, performing the outlined steps to create data points and artifacts that can be used to present to the system's stakeholders or to provide information to external systems in order to enable their own capabilities. The report will discuss the steps and artifacts developed through each step of the developed process. A discussion will follow that demonstrates potential

uses for the data to support the development of acquisition documentation and the analysis of data communication with systems external to the systems engineering boundary.

B. PROBLEM OVERVIEW

The Department of Defense (DoD) produced the DoD Digital Engineering Strategy to help spark and align a digital transformation in the engineering community. More recently, the DoN and Marine Corps delivered Digital Systems Engineering Transformation documentation that describes the goals for model-based systems engineering (MBSE) and lays a framework for MBSE implementation (Department of Navy [DoN] 2020). Currently, MBSE is still immature relative to model-based product support and the enterprise technical reference framework (ETRF) and a fully matured enterprise capability may be some time off. In this scenario, it is assumed that the need for better, faster, and centralized tools and process in the system engineering community has been identified and MBSE is the identified solution. With MBSE being as immature within the enterprise as it is, the DoN is still researching for more information on the MBSE subject and trying to identify how it will best be implemented alongside the product support digital transformation. There is not yet a formal standard set of processes, models, data and tools at the DoN enterprise level that align to all of the objectives in the Digital Engineering Strategy and local commands are beginning to develop their own local instances of MBSE environments. The lack of standardization of the processes, data formats and exchanges may lead to systems again becoming isolated and less efficient as their potential.

C. BACKGROUND

As systems experience a never-ending increase in complexity, rapidly changing operational and threat environments, increased budget constraints, and more demanding schedules, the DoD needs more robust engineering practices. Current engineering processes are often document intensive and stove piped. To meet their needs, the DoD is transforming its engineering practices to a digital engineering methodology utilizing model-based approaches, including MBSE (Engineering 2018). MBSE is a subset of digital engineering and can be defined as the use of models to support the activities within systems engineering (SE) process, including requirements, architecture, design, verification and

validation (Giachetti 2020). The implementation of MBSE has been theorized to enable new capabilities within the systems engineering (SE) process (DoN 2020). One of the primary objectives of implementing MBSE is to develop an integrated set of digitally integrated views that enables the capability of automating the engineering assessment of proposed designs. This automated capability would be able to identify risks and gaps through the simulation of operational scenarios. The digital environment would provide feedback data to enable the application of data-driven decision making.

To maximize the effectiveness of MBSE, an organization must find a cohesive set of modeling tools and methods. The process supporting these activities is laid out in the implementation of OOSEM, applying SysML as the model syntax. The OOSEM is a top-down, scenario-driven approach that leverages object-oriented concepts and other modeling techniques to support in the development of a more flexible and extensible system architecture that can accommodate the constant change in requirements or technologies (Friedenthal, Moore, and Steiner 2012). The activities within the OOSEM process reflect those of the fundamental SE process, including needs analyses, requirements analyses, architecture design, trade studies and analyses, and verification (Friedenthal, Moore, and Steiner 2012). The primary output to the OOSEM process is a model of the system of interest (SOI). The collected data on the SOI is captured and encapsulated using a SysML block, an extension of the Unified Modeling Language (UML) class that includes allocated system elements describing different system views. This project explored a system's architecture, behavior, requirements, and verification and validation (V&V) views. Each view contains a set of SysML diagrams, matrices, or tables to create a model of each system model view. These diagrams are presentation mechanisms to display different data sets of the system model to different stakeholders.

Digital transformation inside the DoN is not only an interest within the engineering domain, but within the entire enterprise. The DoN has a vision for digital transformation, and it has begun in the logistics IT domain with the implementation of the ETRF. The ETRF vision will provide a framework that will generate scalable, interoperable, flexible, and fluid technology solutions that will provide access to information and data at anytime, anywhere. One of the major capabilities of the ETRF is the implementation of an integrated

platform as a service (PaaS) environment that will unify all logistics applications internal to the ETRF system and will deploy a set of application programming interfaces (API) to integrate with future and legacy systems. The vision of the ETRF will contain many logistics applications that will be managed by the PaaS. Applications within the ETRF will fall into one of the following four key mission areas: integrated readiness, supply chain management (SCM), maintenance, repair, and overhaul (MRO), or product lifecycle management (PLM) (Accenture 2019).

There are currently two major programs sponsored by the Office of the Chief of Naval Operations (OPNAV), Model Based Product Support (MBPS) and Navy MRO (NMRO), that are developing the applications to meet the objectives of these mission areas. These applications will be developed to deploy new methodologies, including model-based approaches, and replace legacy systems with new systems that utilize digital tools and processes to replace the old capability set. One of these programs is MBPS, which spans across all four of these mission areas and is of special importance to this project. MBPS is an initiative within the Naval Sea Systems Command (NAVSEA) with cooperation from the Program Executive Office (PEO) that will create and implement a digitally integrated environment focused on the support of Naval systems. The MBPS environment will support the production of many artifacts in the support of sustaining engineering, including reliability centered maintenance (RCM) artifacts, level of repair analysis (LORA), readiness at cost analysis, reliability block diagrams, fault tree analysis (FTA), and other product support documentation and analyses. An authoritative source of product support data, that will enable the supportability analyses listed above. The authoritative data structure will be established and MBPS and developed using industry standards to support the communication and exchange of data between systems internal and external to the MBPS environment (National Shipbuilding Research Program [NSRP] 2019). The integration of MBSE and MBPS is of great interest. It has been theorized that this integration could lead to systems that maximize availability, effectiveness, capability, and affordability (Kwon, Page, and Weinstein 2018).

In order to perform cross-platform verification and analysis, data must be accessible by both environments through an authoritative data source. Currently, there are two

identified potential authoritative data sources within the ETRF that are being sponsored for development. Within MBPS, there is the Navy Product Data Management (NPDM) that is being established as the authoritative data source for all system technical data once a system reaches the operation and sustainment phase of the system's lifecycle. The ETRF will also be deploying the agile warfighter analytics readiness environment (AWARE) within NMRO. The AWARE is a data-as-a-service (DaaS) platform to manage and communicate maintenance data from data collected by ship-based NMRO applications to the AWARE. Any data needed by the applications will be stored and transferred through at least one of these data sources. For MBSE, this has been identified as a major integration point between SE and product support (PS) capabilities which, in the future, will communicate and supplement the capabilities of one another (Accenture 2019).

D. PROBLEM STATEMENT

The United States Navy (USN) has produced documentation describing the characteristics of a model-based engineering environment but has not yet realized a solution for a model-based engineering environment and how that environment would be implemented and integrated into the system of systems (SoS) enterprise digital transformation vision (DoN 2020). A need has been identified by the systems engineering community at the Naval Sea Systems Command, Port Hueneme Division (NSWC PHD) to implement a local model-based system engineering (MBSE) environment and to understand how the MBSE data set, capabilities and tools would integrate into the ETRF.

With the MBPS capability set being more mature than the MBSE capability set, this capstone looked to identify potential avenues of implementation that aligned to the high-level objectives within the DoD and DoN strategic documents. With the development of a standard modeling process, the standardization of data sets, presentation artifacts, tool sets, etc. will follow, enabling many of the MBSE capabilities. A standardized set of data of system model data will enable external boundary communication and the development of model-based documentation.

E. PROJECT OBJECTIVES

This capstone team had two high-level objectives: develop a formal process using systems engineering methodologies that would be capable of developing a conceptual system model and compile a final report that will explain the problem space, describe the solution space and how it solves identified issues, describe and explain the processes used, present the developed artifacts, and provide recommendations for future work or action.

The objective of the model is to provide a standard process for organizations to develop a conceptual system model that contains early system architecture, behavior, requirements, and verification and validation models. The conceptual model would be the starting point for a program's digital twin and thread that would mature along with the design to include data from the logical and physical levels of the design. The process and development of system model data enables the capability of producing model-based documentation that supports the development of programmatic documentation from templates. The report will demonstrate and explain the process of how the capstone team developed and produced a model-based concept of operations (CONOPS) from a Microsoft Word template found in the public internet domain.

To ensure the process satisfies the stakeholder objective and requirements, the capstone team applied the process to a development scenario to support the verification of the process. The model will be supplemented by a textual report that will further include explanations of the processes and recommendations for future work.

F. PROJECT SCOPE

The scope of the capstone is set based on the scenario outlined in Section B of this chapter. Verifying the developed process with these scenarios will produce a set of artifacts that will be used to demonstrate to organizations how MBSE can be used. The documented process and developed artifacts are a part of the framework of this report, and the discussions that follow will be based off the development of the system model and the verification methods using the use case scenarios.

G. CHAPTER SUMMARY

This chapter introduced the capstone, overall problem, background information, problem statement, scope, and objectives. This information is used in the understanding of the information and processes that will be discussed in the appended sections.

Having identified the need to utilize MBSE concepts to enhance the DoN's engineering capabilities, the capstone team documented a standard process. The process is used to support an organization's capability to develop conceptual system models. The process was developed using the object-oriented systems engineering methodology as a guide as to what data is required for the development of the system model and the presentation artifacts were produced using Cameo Enterprise Architecture. To provide examples of artifacts to the stakeholders and this report, a fictitious scenario was applied. The appending sections will provide more detailed explanations for each phase of the process and the artifacts that are consumed and produced by each phase.

PAGE INTENTIONALLY LEFT BLANK

II. PROBLEM ANALYSIS

A. STAKEHOLDERS

The primary stakeholders for this capstone are command representatives from the Naval Surface Warfare Center, Port Hueneme Division (NSWC PHD). NSWC PHD submitted a topic area to investigate integrating MBSE to MBPS. The project team began early discussions with the primary stakeholders to understand the stakeholder's needs. The team met frequently with the stakeholders to get feedback on progress, ensure alignment of project objectives, and to guide project development. In addition to the primary stakeholder and for the verification scenario, the team identified alternative stakeholders who do not have a direct involvement in the capstone but would benefit or be influenced by its efforts. The scenario described the development of an MBSE environment named the Digital Engineering Environment (DEE). A summary of each stakeholder, their relation to the capstone, and their stake in the project is below.

Table 1 shows each stakeholder, their stake in the project categorized as end user or adjacent end user, a description of their interactions with the DEE, and a stated need as it relates to MBSE to MBPS integration. An end user is an entity who directly uses the DEE by entering data, accessing models, pulling data, and updating diagrams. An adjacent end user is an entity who does not directly operate the DEE but is impacted by its overall capability. In this example, the Program Office will not use the DEE but will drive policies shaping incoming capabilities. The DEE will be used as a tool to integrate with incoming capabilities. In addition, this process will be validated by using MBSE to MBPS integration as a use case. The last column refers to the specific need of each entity as it relates to the MBSE to MBPS integration effort.

Table 1. Stakeholder Description

Stakeholder	Stakeholder Type	Description	Stakeholder MBSE/ MBPS Needs
NSWC PHD	End User	-Primary end user who will receive project report and associated models. -Guides project development based on stakeholder need and project objectives.	NSWC PHD needs model-based engineering capability and tools to integrate with the product support capabilities of the Enterprise Technical Reference Framework (ETRF)
Program Office	Adjacent End User	-Impacted by system realization if implemented. -Creates official policies and instructions that directs incoming capabilities -Accesses databases for program information	Program Office needs an efficient method to access latest program information and to have that information consistent across databases.
System Engineers	End Users	-Will use project models in the DEE to prepare for integration of incoming capabilities. -Will populate and request MBSE information from DEE such as requirements diagram, behavior diagrams, system models, etc.	Systems Engineers need an integrated environment that supports all of the top down and bottom up technical processes and activities within the SE VEE.
Reliability, maintainability, and availability (RMA) Engineer	Adjacent end user	-Will use DEE to prepare for integration of incoming capabilities related to RMA -Will use the DEE to examine and analyze RMA metrics such as operational availability, mean time between failure MTBF, etc.	RMA engineer needs a system to effectively gather information in relation to complex systems. This includes an integrated database that automatically syncs system information between data lakes.
Logistician	End user	-Will enter logistics data into DEE which will drive updates into external systems/databases -Will participate in IPTs to prepare for incoming logistician impacted capabilities.	Logistician needs a system that accepts logistics data inputs and that drive updates into MBPS system.

B. IMPORTANT DEFINITIONS AND TERMS

Common definitions and terms are used throughout this report. These definitions were researched and established during the literature review. These terms are defined in this chapter to give the reader a general understanding of the topics to be discussed.

1. Model-Based Systems Engineering

The use of models to convey systems engineering concepts and data either in place of or in conjunction with traditional textual methods has gained wide acceptance in recent years. This was introduced by INCOSE in 2007 as follows:

Model-based systems engineering (MBSE) is 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. MBSE is part of a long-term trend toward model-centric approaches adopted by other engineering disciplines, including mechanical, electrical and software (INCOSE 2007).

2. Model-Based Product Support

There is no official definition of model-based product support (MBPS) in literature, but the collective sources support a general definition. Model based product support is a broad term that essentially translates to model based electronic tools and information systems that enable the support of logistics functions such as training, maintenance, operations, and sustainment. Model Based Product Support is the cooperative initiative between NAVSEA and PEO that will provide multiple digital logistics capabilities to the DoN.

3. Architecture Framework:

The Architecture Framework defines how an architecture will be created and subsequently utilized through a set of rules and practices. It is defined by the MITRE Corporation as follows:

An architecture framework is an encapsulation of a minimum set of practices and requirements for artifacts that describe a system's architecture. Models are representations of how objects in a system fit structurally in and behave as part of the system. Views are a partial expression of the system from a perspective. A viewpoint is a set of representations (views and models) of an architecture that covers a stakeholder's issues (MITRE 2015).

4. Enterprise Technical Reference Framework (ETRF)

The push to consolidate existing systems into a new common logistics platform that leverages new technologies and innovations is necessary in order to adapt to the Navy's changing needs. The following two quotes describe this:

The vision of Enterprise Technical Reference Framework (ETRF) is to enable and accelerate the overall objective of Navy Logistics IT. ETRF provides a digital logistics IT architecture that will generate scalable, interoperable, flexible and fluid technology solutions; maximizing access to information/data via applications anywhere, on any device at any time.(Accenture 2019)

The Enterprise Technical Reference Framework will leverage the Digital Transform Plan Services, Data, Technology, Security and Change Management strategies to provide a framework and roadmap to transform 1600+ current Applications and 5000+ data sources to a common unified logistics IT platform (Accenture 2019).

5. Digital information technology (IT) transformation

The DoD digital IT transformation exists within the Joint Information Environment (JIE) framework that is comprised of a comprehensive Department-wide IT modernization that exists within the DoD Information Network (DoDIN). The JIE purpose is to “improve mission effectiveness, increase cybersecurity, improve interoperability, deliver capabilities faster, and realize IT efficiencies” (US DoD 2019).

The DoD JIE framework is comprised of ten (10) Capability Objectives as shown in Table 2.

Table 2. Alignment of DoD CIO Objectives to JIE Capability Objectives and Initiatives. Source: USDoD (2019).

JIE Capability Objective	JIE Initiatives	DoD CIO Objectives
Modernize Network Infrastructure	Optical Transport Upgrades, MPLS Routers Buildout, ATM Switch and low speed TDM Circuit Elimination, Satellite Communications Gateway Consolidation and Modernization, IPv6 Implementation	<ul style="list-style-type: none"> • Modernize Warfighter C4 Infrastructure and Systems • Modernize DISN Transport Infrastructure • Modernize and Optimize DoD Component Networks and Services
Enable Enterprise Network Operations	Establish global and regional operations centers, Establish the JIE Management Network, Converge IT Service Management (ITSM) solutions	<ul style="list-style-type: none"> • Modernize and Optimize DoD Component Networks and Services • Shift from Component-Centric to Enterprise-Wide Operations and Defense Model
Implement Regional Security	JRSS, JMS	<ul style="list-style-type: none"> • Modernize DISN Transport Infrastructure
Provide Mission Partner Environment (MPE)	Virtual Data Center, Applications and Services, MPE Transport, Mission Partner Gateways	<ul style="list-style-type: none"> • Strengthen Collaboration, International Partnerships, and Allied Interoperability
Optimize Data Center Infrastructure	Data Center Optimization Initiative (DCOI) and Application Rationalization Initiative	<ul style="list-style-type: none"> • Optimize DoD Data Centers
Implement Consistent Cybersecurity Protections	Enterprise Perimeter Protection Capabilities, Operate Securely in the Cloud, Endpoint Security, Data Center Security, Cyber Situational Awareness Analytic Capabilities (CSAAC)/ Big Data Platform (BDP), Identity, Credential, and Access Management (ICAM)	<ul style="list-style-type: none"> • Transform the DoD Cybersecurity Architecture to Increase Agility and Strengthen Resilience • Deliver a DoD Enterprise Cloud Environment to Leverage Commercial Innovation • Deploy an End-to-End ICAM Infrastructure
Enhance Enterprise Mobility	Purebred for Mobile, Defense Enterprise Mobility-Classified Consolidation, DoD Mobile Application Store, Pentagon Mobility	<ul style="list-style-type: none"> • Improve Information Sharing to Mobile Users
Standardize IT Commodity Management	Enterprise Software Agreements, Enterprise License Agreements, Enterprise Hardware Agreements, IT Asset Management, Windows 10 SHB Fourth Estate Network Optimization	<ul style="list-style-type: none"> • Improve IT Category Management • Transform the DoD Cybersecurity Architecture to Increase Agility and Strengthen Resilience
Establish End-User Enterprise Services	Enterprise Collaboration and Productivity Services	<ul style="list-style-type: none"> • Optimize DoD Office Productivity and Collaboration Capabilities (ECAPS Capability Set 1) • Optimize DoD Voice & Video Capabilities (ECAPS Capability Sets 2 & 3)

JIE Capability Objective	JIE Initiatives	DoD CIO Objectives
Provide Hybrid Cloud Computing Environments	Cloud Services	<ul style="list-style-type: none"> • Deliver a DoD Enterprise Cloud Environment to Leverage Commercial Innovation • Optimize DoD Office Productivity and Collaboration Capabilities (ECAPS Capability Set 1) • Optimize DoD Voice & Video Capabilities (ECAPS Capability Sets 2 & 3)

6. Systems Modeling Language (SysML)

The Systems Modeling Language (SysML) is a general purpose MBSE language that uses “graphical modeling for specifying, analyzing, designing, and verifying complex systems that [include] hardware, software, information, personnel, procedures, and facility elements” (Object Management Group n.d). SysML originated from the Unified Modeling Language (UML) 2 framework. Further, SysML “provides graphical representations with a semantic foundation for modeling system requirements, behavior, structure, and parametrics, which is used to integrate with other engineering analysis models” (Object Management Group n.d).

C. LITERATURE REVIEW SUMMARY

1. Purpose

The focus of this capstone involves two types of model-based systems, each with their own unique applications. The first being model based systems engineering, and the second being model based product support. This capstone will take advantage of intrinsic capabilities existing in both modeling efforts. In order to leverage their full capabilities, it is first important to understand what MBSE and MBPS are, the types of functions they can perform, and current applications in both industry and government. The purpose of the literature review was to complete an in-depth analysis of the current state of modeling efforts, starting with defining purpose, explaining functionality, and exploring future

applications. Lastly, it is relevant to this capstone to understand the current process for product-support utilized by the vision owner. This established a baseline understanding of the current operating procedures and gaps in current capability to be identified. The literature review is in the appendix.

2. U.S. Navy Legacy Approach

“Operations and sustainment (O&S) costs make up most of total ownership costs (TOC) for a defense system” (Page, Kwon, and Weinstein 2018). In order to provide equipment life cycle support, over time various legacy systems and repository tools have been created to address and identified deficiencies. MBPS intends to align life cycle support activities from three major functional areas:

1. Drawings and manufacturing modeling data
2. Technical publications and training content
3. Predicted, optimized & sustainable readiness.

The three major areas of supportability can be further decomposed into subcategories exemplifying their respective critical system(s) and application(s). In an overview presentation provided by the MBPS team at NAVSEA06L that was accessed on April 7, 2020, critical system(s) and application(s) currently deployed throughout the DoN are being replaced by the integrated applications of MBPS. Those current legacy systems are discussed below.

The two key areas under the category for drawings and manufacturing modeling data are system shipboard configuration and ship drawings. Described in the MBPS overview briefing, system shipboard configuration utilizes the Configuration Data Managers Database—Open Architecture (CDMD-OA) as well as Revised Alternative Data Flow WEB (RADWEB). CDMD-OA is the authoritative database for all configuration item management and is the civil service tool used to create configuration data packages to update the fleet’s shipboard configuration (NAVSEAINST 4130.12). Described in the MBPS overview briefing “RADWEB acts as the electronic conduit through which shipboard and shore site allowance update and system / equipment maintenance history

data files are passed among various activities (NAVSUP / NAVSUP WSS, NAVSEA, SPAWAR, NAVAIR, Warfare Centers, Regional Maintenance Centers, and Fleet Forces Command and Fleet Operational Units)” (National Shipbuilding Research Program [NSRP] 2019). The MBPS overview briefing further details the functionality “RADWEB enables configuration, maintenance, and allowance data transfer between ship and shore” (NSRP 2019). Ship Drawing requirements are housed in the Navy Ships Engineering Drawing Repository (NSEDOR) database. “NSEDOR stores and maintains all Naval ship drawings utilized by planning yards, fleet activities, Naval Surface Warfare Centers, Systems Commands, and ISEAs” (NSRP 2019).

The second key area for supportability is technical publications and training content, which encapsulates three critical areas:

- Configuration management and modernization.
- Provisioning parts information.
- Ship and shore technical data viewing.
- Organization and depot maintenance procedures.

Configuration management and modernization artifacts are housed in the Navy Database Environment (NDE). The NDE is a repository for configuration management and modernization data for Naval Sea Systems Command (NAVSEASYS COM). Provisioning Parts Information (PPI) employs the Interactive Computer Aided Provisioning System (ICAPS) where provisioning technical data (PTD) is entered and stored. The PTD provides information used to create the Allowance Parts Listing (APL) for the identified Configuration Item (CI). Allowance Parts Listings consist of parts, or in the supply verbiage, nation stock numbers (NSN’s). Ship and shore technical data viewing is accomplished through the Advanced Technical Data Information System (ATIS). The ATIS provides an Interactive Electronic Technical Manuals (IETM) and compatibility testing, installation, and troubleshooting support (Jorgensen 2006). All technical manual updates must go through ATIS certification before being disseminated to the fleet for use. Organization and depot maintenance procedures artifacts are housed in the Technical Data

Management Information System (TDMIS) and Navy Logistics Technical Data Repository System (NAVLOGTD). The MBPS overview briefing states that TDMIS provides “technical manual / documentation configuration and life cycle management and enables users to research and view selected technical manuals” (NSRP 2019). “[The] NAVLOGTD is the authoritative data repository used to develop, edit, publish, distribute and view technical data for” Engineering Operating Sequencing System (EOSS), Planned Maintenance System (PMS), and Technical Manual (NSRP 2019).

The last key category for supportability is predicted, optimized, and sustainable readiness, which contains readiness/mission models. Readiness/mission models address reliability, maintainability and availability (RM&A). The Materials Readiness Data Base (MRDB) tracks the readiness status of all USN equipment currently deployed. The database maintains a detailed maintenance record of all equipment and systems to assess reliability and readiness (TransSolutions 2012).

3. Capability Gaps in Legacy Approach

These legacy systems and repositories serve their purpose in a singular fashion. Each system is siloed, where a change or update in data in one system does not affect the dependent supportive documentation/artifacts in another. For example, if there is an identified deficiency in a piece of equipment, configuration change artifacts would be generated and stored in NDE. A ship installation drawing (SID) would then be produced for modernization of equipment and stored in NESDR. From the SID, a PTD would be generated, from PTD data being entered into ICAPs. Provisioning technical data is also entered into MRDB to model RM&A to check for new sparing requirements. The NSN changes would also need to be manually input into the TMs. The newly generated technical documentation to address corrective maintenance would be stored in TDMIS. The PMS generated from the new NSNs’ effect on the equipment performance would be stored in NAVLOGTD. It would then take human intervention to consolidate all the supporting documentation and configuration data to transfer to the end item user. Personnel would need to identify all the supportive Technical Documentation, PMS, and Supply Support

data into CDMD-OA to then transfer to the respective ship activity so the end item user can operate and sustain the equipment.

4. DoD Model Based Systems Engineering

The U.S. DoD is currently in the process of shifting many operations within the department to a digital environment as outlined in the 2019 DoD Digital Modernization Strategy (US DoD 2019). In addition to the DoD digital transition, the DoD has a more specific digital engineering strategy that outlines the goals and visions within the engineering discipline. The 2018 Digital Engineering Strategy emphasizes four (4) initiatives as the purpose of digital engineering (DE); 1) policy/guidance, 2) pilots, 3) implementation, and 4) tools. In addition to the initiatives, there is a goal to transform the culture within the workforce to adopt digital engineering capabilities across the life cycle. The expected benefits the initiatives and goals will provide include greater ability for well-informed decision making through heightened insight and transparency, enhanced communications, increased understandability and adaptability in design, increase confidence in system performance, and increased efficiency in engineering acquisition practices (US DoD 2018).

The ability to reach the vision as explained, the digital IT infrastructure requires free flowing but accurate data seen in Figure 1, to be used within the different modeling products, which aids in allowing for accurate models to be viewed and analyzed, resulting with the right decisions to be made. One key to enabling a robust digital IT infrastructure includes the assurance and accuracy of data and models that are to be stored in a centralized location and acts as the authoritative source of truth, (also called the authoritative data source). A centralized authoritative source of data allows for models & data to be retrieved, viewed, modified, manipulated and/or uploaded to the authoritative data source. This allows for different technical and logistical tools to utilize accurate and up-to-date information throughout the life cycle of a system or product

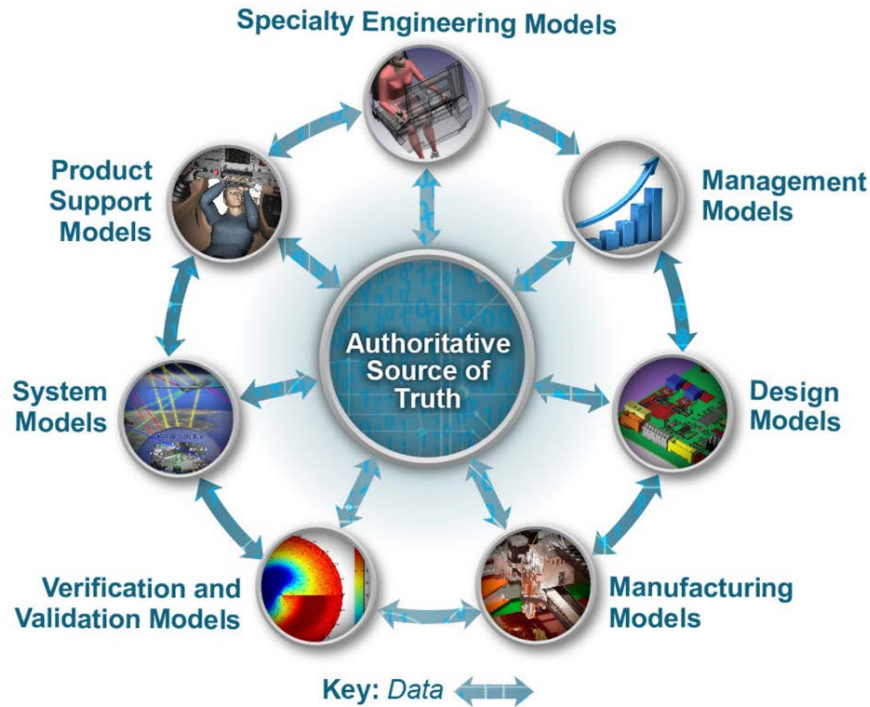


Figure 1. Engineering MBSE Models Source: US DoD (2018).

5. USN Model Based Product Support

Recently there has been a shift in the Navy’s approach to executing PS functions. Emerging technologies have enabled a more responsive, user friendly, and detailed approach for supporting functions such as maintenance, sparing, training, and supply chain management. This was demonstrated in a recent forum hosted at Naval Surface Warfare Center, Port Hueneme Division, “The command’s Product Support Office hosted the forum to focus on Model Based Product Support (MBPS) NAVSEA Logistics SEA 06L systems replacement” (Sashegye 2020). This forum expounded on the notion that a new and more comprehensive logistics system is necessary to meet current and future needs.

According to SEA 06L, the Navy’s current logistics IT systems that provide configuration management, provisioning, readiness modeling and technical data management support for ships and weapon systems are outdated and cannot keep pace with rapidly changing and emerging technologies. This current infrastructure greatly inhibits the enterprise to effectively and cohesively sustain the fleet (Sashegye 2020).

Programs of record are in progress that provide a range of new model-based tools to enable the functions outlined above. These include data repositories containing detailed technical data like three-dimensional computer-aided design (CAD) drawings, a single authoritative system to manage baseline and configuration control, readiness modeling and simulation, acquisition repositories to standardize contractual awards, and documentation publication and management such as technical manuals and maintenance procedures. The appetite for change has been driven by these emerging capabilities that industry shares.

a. Capabilities of MBPS

Presented in an overview presentation developed by the Model Based Product Support (MBPS) team at NAVSEA06L, the Navy is pursuing multiple programs, each with unique goals and capabilities that fit under an umbrella project called the “MBPS Digital Transformation.” “[Model Based Product Support] is a Business Capability Acquisition Category Level II (BCAT II) currently in Phase 1, with Phase 2 completion expected by the end of prototype develop period during Q2 FY21” (NSRP 2019). It is pertinent for this capstone to capture both current and planned capabilities of these systems. The capabilities are summarized in the categories below.

One primary requirement of MBPS as a domain capability is to provide the infrastructure required to provide a data repository that will interact with the authoritative sources of data. “The core of an integrated MBSE and MBPS strategy should be a central data repository that provides structured, authoritative product information, spanning from concept to end of life” (Page, Kwon, and Weinstein 2018). It is imperative that the contents of this data repository are well understood. According to SEA06L, the Navy is pursuing a repository known as Navy Product Data Management (NPDM). Navy Product Data Management will be able to perform “Configuration manage, sustain, and provide enterprise access to all components of legacy and future standards-based Navy Weapon System Technical Data Packages (TDP).” Technical data packages contain unique part-based information and “applicable technical data such as models, drawings, associated lists, specifications, standards, and performance requirements” (DoD 2009). Combat system data will reside in this repository and other MBSE tools will interface with this

system to transfer system data. In some cases, this data repository contains what is called a digital twin of that combat system. “The data repository captures the digital twin of a system, which is the digital representation of the system baseline and forms the product structure” (Page, Kwon, and Weinstein 2018). Both government and industry are pursuing the development of digital twins for both legacy systems and new systems. However, “Implementation of a DT [digital twin] may be difficult; the cost of a DT may be extremely high or cost prohibitive” (Bickford et al. 2020). This is especially difficult when retrofitting for legacy systems. The type of data required can reside in multiple locations, and that data may be disparate and lacking a common format. If done correctly, a digital twin (DT) can serve as a tool for troubleshooting, testing new software baselines, informing maintenance actions, and performing analytic studies of failure data.

Model Based Product Support supports data analytics by using the repository to perform evaluations of system failure rates, availability, and operational readiness. The Navy is pursuing a system called the Navy Common Readiness Model (NCRM). The NCRM will be able to “Analyze, report, predict, and optimize weapon system readiness and O&S cost throughout the life cycle” (NSRP 2019). The Navy plans to use this portion of MBSE to access and analyze data stored in the data repository. Model Based Product Support is especially useful because it can access post event data. For example, the data repository is updated continuously throughout a combat systems’ life cycle. When a failure occurs, that failure is recorded in the repository. The NCRM can observe all of the recorded failures for a specific part, predict the next expected failure date, and query an inventory to show if there are spare parts available. Systems engineers have also found uses leveraging analytics to inform life cycle cost during system development. “[Digital Twin] development can play an early role in identifying failure modes, symptoms, and resulting impacts, reducing long-term reliability concerns” (Bickford et al. 2020).

Model Based Product Support is designed to support data sharing with the other domains it interacts with, with critical access to the data repository is facilitated through different types of sharing technologies. One being investigated by the Navy is cloud-based data management, where data is transferred from offshore ships to land-based systems. The data can then be access on an as-needed basis utilizing secure transfer protocols. Industry

experience is being leveraged to create “[an] enterprise architecture to integrate commercial off-the-shelf and legacy products and services, cloud-based hosting, functional applications and services, and phased modernization for the shore maritime maintenance operating environment” (Rutherford 2017). Cloud-based technologies allow quick transference of data between multiple devices. This enables deployed combat systems to publish relevant data related to reliability, operations, and maintenance actions to a cloud that engineers can access. Furthermore, MBPS will act as the configuration manager of this official program of record is still in its developmental phases, but this accessibility of data expands the analytic capabilities previously mentioned.

As mentioned herein, data sharing and data storage are key enablers for MBPS. There has been a shift in the methods for storing and sharing this data. Many companies are adopting the use of cloud-based computing as mentioned earlier. In the past, data was stored locally on individual machines and files were stored behind firewalls on individual computers. There developed a need for greater flexibility in data management. This shift in data services is described as a capability called Data as a Service (DaaS). “Data as a service (DaaS) is a data management strategy that uses the cloud to deliver data storage, integration, processing, and/or analytics services via a network connection” (McDaniel 2019). The capabilities provided by cloud-computing has increased dramatically with the addition of higher bandwidth networks and applications specifically designed for large data sets. “generic cloud computing services were not initially designed for handling massive data workloads; instead, they catered to application hosting and basic data storage (as opposed to data integration, analytics, and processing)” (McDaniel 2019). Model Based Product Support will eventually need to integrate heavily with cloud-based computing. Data as a Service provides great flexibility and access to pertinent data that MBPS products will use as input to their models. As mentioned earlier, the key point for integration from MBSE to MBPS is this shared data repository. As MBPS tools evolve, there is a high probability that they will require the ability to be compatible with cloud sharing capabilities in order to facilitate data analysis.

Model Based Product Support also has applications in training, maintenance, operation, and sustainment of systems. The previously explained digital twin is being

leveraged for maintenance and operation of Navy systems. Other armed forces have found similar uses for MBPS. In the earlier definition, model-based product support was defined as a set of electronic tools and IT systems that enable product support functions. The Air Force leverages virtual reality (VR) tools to train their pilots using flight simulators that give users an interactive experience using mockups of the system cockpit. The simulators contain representative system data; they model system behavior and interface with the user to provide training. Virtual reality is also being used to train maintenance personnel, reducing risk by having trainees' practice on a virtual system. "Aviation maintenance VR permits users, right from a computer screen, to walk around or into an aircraft, to open the cowlings, to perform many line check activities, or even delve into the internal workings of any system" (Johnson 2018).

Model Based Product Support currently consists of four (4) tools, all of which perform distinct actions and satisfy specific capabilities contained herein. The tools consist of NPDM, NCRM, NDART, and the MBPS Workbench.

Overall, MBPS has been demonstrated as a useful tool with real applications toward sustainment and operation of complex systems. This capstone will focus on current naval applications within the MBPS Digital Transformation.

(1) Data Analytics

Model Based Product Support can also support data analytics by using the repository to perform evaluations of system failure rates, availability, and operational readiness. The Navy is pursuing a system called the Navy Common Readiness Model (NCRM). The NCRM will be able to "Analyze, report, predict, and optimize weapon system readiness and O&S cost throughout the life cycle" (NSRP 2019). The Navy plans to use this portion of MBSE to access and analyze data stored in the data repository. Model Based Product Support is especially useful because it can access post event data. For example, the data repository is updated continuously throughout a combat systems' life cycle. When a failure occurs, that failure is recorded in the repository. The NCRM can observe all of the recorded failures for a specific part, predict the next expected failure date, and query an inventory to show if there are spare parts available. Systems engineers have

also found uses leveraging analytics to inform life cycle cost during system development. “DT [digital twin] development can play an early role in identifying failure modes, symptoms, and resulting impacts, reducing long-term reliability concerns”(Bickford et al. 2020).

(2) Data Sharing

Access to the data repository is facilitated through different types of sharing technologies. One being investigated by the Navy is cloud-based, where data is transferred from offshore ships to land-based systems. Industry experience is being leveraged to create “[an] enterprise architecture to integrate commercial off-the-shelf and legacy products and services, cloud-based hosting, functional applications and services, and phased modernization for the shore maritime maintenance operating environment” (Rutherford 2017). Cloud-based technologies allow quick transference of data between multiple devices. This enables deployed combat systems to publish relevant data related to reliability, operations, and maintenance actions to a cloud that engineers can access. This project is still in its developmental phases, but this accessibility of data expands the analytic capabilities previously mentioned.

(3) Further Applications of MBPS

As mentioned earlier, data sharing and data storage are key enablers for MBPS. There has been a shift in the methods for storing and sharing this data. Many companies are adopting the use of cloud-based computing as mentioned earlier. In the past, data was stored locally on individual machines and files were stored behind firewalls on individual computers. There developed a need for greater flexibility in data management. This shift in data services is described as a capability called Data as a Service (DaaS). “Data as a service (DaaS) is a data management strategy that uses the cloud to deliver data storage, integration, processing, and/or analytics services via a network connection” (McDaniel 2019). The capabilities provided by cloud-computing has increased dramatically with the addition of higher bandwidth networks and applications specifically designed for large data sets. “generic cloud computing services were not initially designed for handling massive data workloads; instead, they catered to application hosting and basic data storage (as

opposed to data integration, analytics, and processing)” (McDaniel 2019). Model Based Product Support will eventually need to integrate heavily with cloud-based computing. DaaS provides great flexibility and access to pertinent data that MBPS products will use as input to their models. As mentioned earlier, the key point for integration from MBSE to MBPS is this shared data repository. Model Based Product Support tools will need to be designed and aligned with cloud sharing capabilities in order to facilitate data analysis.

Model Based Product Support also has applications in training, maintenance, operation, and sustainment of systems. The previously explained digital twin is being leveraged for maintenance and operation of Navy systems. Other armed forces have found similar uses for MBPS. In the earlier definition, model-based product support was defined as a set of electronic tools and IT systems that enable product support functions. The Air Force leverages virtual reality tools to train their pilots using flight simulators that give users an interactive experience using mockups of the system cockpit. The simulators contain representative system data; they model system behavior and interface with the user to provide training. Virtual reality is also being used to train maintenance personnel, reducing risk by having trainees’ practice on a virtual system. “Aviation maintenance VR permits users, right from a computer screen, to walk around or into an aircraft, to open the cowlings, to perform many line check activities, or even delve into the internal workings of any system” (Johnson 2018).

Overall, MBPS has been demonstrated as a useful tool with real applications toward sustainment and operation of complex systems. This capstone will focus on current naval applications within the MBPS Digital Transformation.

D. CHAPTER SUMMARY

This chapter discussed problem analysis to include stakeholders, definitions, and a literature review. Definitions were introduced to familiarize the reader with MBSE and MBPS and the environment they operate within. Policies such as the ETRF and digital IT transformation explain how DoD policies affect both modeling areas. A list of stakeholders was presented that explained their functional area, their relationship to this project, and how they are impacted by this project. The literature review familiarized the capstone

group with modeling efforts within systems engineering and product support. The literature review presented an overview of definitions and applications of MBSE and MBPS. Furthermore, the literature reviewed focused on DoD specific applications of modeling to include:

- The U.S Navy's legacy process being used at the time of this capstone.
- The capability gaps of the legacy processes.
- Future DoD-specific modeling trends in MBSE and MBPS.

III. SYSTEMS ENGINEERING APPROACH

A. SYSTEMS ENGINEERING PROCESS

In order to ensure the team could provide a functional product of high value to the stakeholders, a comprehensive and structured systems engineering process was adopted during system development. Figure 2 presents the processes as it was performed. The process can be broken into three major phases: The planning phase, execution phase, and the reporting phase.

There are two tasks that make-up the planning phase that are performed in parallel. The stakeholder analysis and literature review set the foundation of our understanding of the problems and needs, and the current environment situations. The outputs of these two tasks are outlined in the previous chapter's sections.

In the execution phase of the capstone the development and testing commenced. Model development was planned and executed using the agile framework of scrum. The scrum framework has an appending section in which the details of how the methodology was used by the capstone teams is presented. Model builds were iteratively presented and reviewed by the capstone's stakeholders, where feedback was provided and incorporated into the development's future sprints. When the model reached a level of maturity, the model ran through a simulation. The simulation constraints were explained briefly in the early chapters of this report and are explained in greater detail in a later chapter. Gaps identified during or after the simulation were incorporated as feedback into a future model development sprint.

The last phase, the reporting phase, consist of a single task. The data from the prior two phases is collected and presented in the format seen presented here in this report. The reporting phase also contains two other deliverables: A final presentation to the stakeholders and the delivery of the developed model containing annotations and simulation outputs

The realized products of this capstone were tracked and verified through a rigorous configuration management process. All elements were characterized and date-stamped for

reference, and model coherence was validated using the above practices. This configuration management approach certified the integrity of all delivered project artifacts.

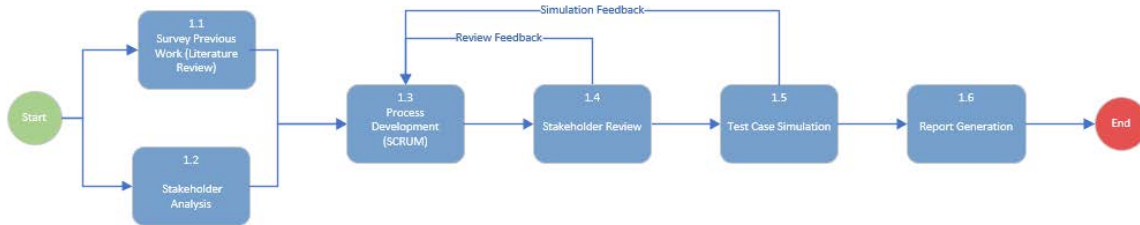


Figure 2. Capstone System Engineering Process.

B. SCRUM FRAMEWORK

This capstone has been developed and structured using the agile scrum project management framework. This agile system has allowed for replacement of the traditionally structured and algorithmic approach to project management and milestone-based planning with a more heuristic and open-ended process, allowing for greater flexibility and increased productivity (Scrum.org n.d.). The roles of product owner, scrum master, and development team make up the team members and each have a distinct role in the development process.

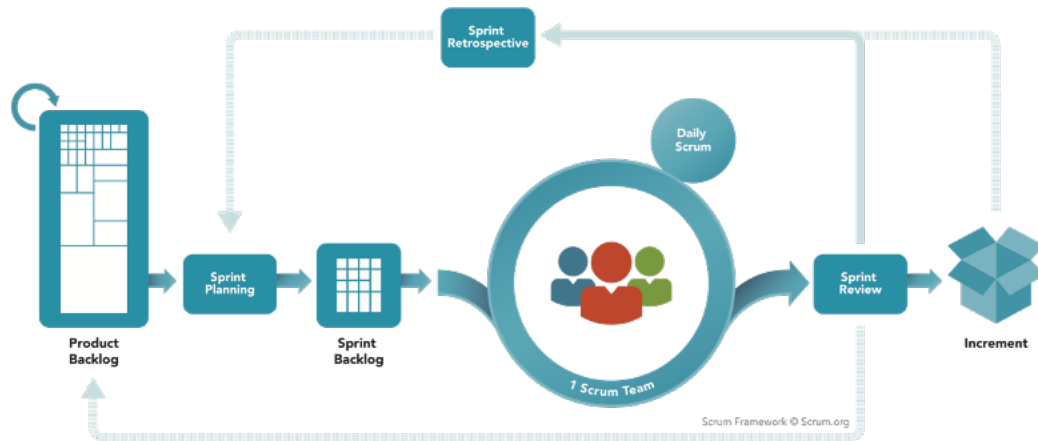
The product owner is the manager of the project and is responsible for maximizing the value of the work being performed by the development team (Scrum.org n.d.). The product owner maintains an active list of backlog items to track development progress and verify coherence with the project schedule. The product owner also defines the tasks to be completed and their order of importance, describing the details and expectations in addition to the way each element fit into the overall project structure. The communication of work and taskings between the product owner and the development is also critically important and the product owner is responsible for presenting and communicating the data in a way is understood by the development team (Scrum.org n.d.).

The scrum master helps guide the rest of the team, including the product owner, in understanding the workings of the scrum framework (Scrum.org n.d.). The scrum master actively monitored the team's progress within the scrum framework, promoting adherence to scrum practices and theory to ensure an effective workflow.

The development team consists of the performers of the tasks set by the product owner and follow the guidance of the scrum master. The development team members are given autonomy to organize and manage their tasking, allowing cross-functional collaboration between all team members to maximize output and product quality (Scrum.org n.d.).

Efforts were organized into incremental sprints containing related tasking which is divided up among team members. While team members were responsible for their assigned tasking, collaboration was encouraged. These sprints and the status of internal tasking were managed during the weekly scrum sessions led by the scrum master. Twice-a-week scrum standups were preferred by the team, vice daily standups, and consisted of a round table discussion of tasking to elicit feedback on sprint efforts and discuss any current barriers in performing tasks. Development sprints continued until the product, in this case the model, was ready to be released and reviewed by stakeholders. Feedback is solicited from stakeholders and provided input into future development sprints. This agile framework has proven effective and efficient in the development of products necessary to realize the project goals. Figure 3 shows a graphical representation of the scrum framework.

SCRUM FRAMEWORK



 Scrum.org

Figure 3. Scrum Framework. Source: Scrum.org (2020).

C. CONFIGURATION MANAGEMENT

In order to establish and maintain consistency throughout the development of the project, configuration management practices were implemented for both the documentation and model development efforts. A core output of this development effort is the SysML-based architecture of a Digital Engineering Environment which NSWC PHD can bring to realization in future efforts. Therefore, effective configuration management was critical to ensuring model coherence and applicability.

At project onset, controlled items were identified and categorized based on configuration reporting requirements. Critical project items such as the project master model and final report were assigned stringent configuration management practices with greater change approval requirements. Requirements for lower level modeling tasks were assigned less stringent reporting requirements to allow for greater agility during the development process. Initial modeling efforts were broken into tasks, which were structured chronologically into scrum sprints. Each sprint contained a set of modeling efforts which related internally. Models developed in each sprint were recorded by content

and date-stamped inside the sprint. The model identification scheme followed the following structure:

- Project Identifier
- Model Element
- Creator
- Revision number
- Date stamp

Any additional changes were sequentially date-stamped to reflect the alteration. A master model was kept in a separate folder managed by the team product owner, who was responsible for maintaining its integrity. To ensure unauthorized changes are not included with the master model, a defined approval structure for adjudicating changes was developed and adhered to. When an editor alteration was made, a date-stamped duplicate of the master model was created and placed in a history folder to ensure reversibility is available if necessary. This process has served to maintain the integrity of the product and ensure that unexpected issues or setbacks can be reverted if needed.

Documentation and reporting are structured in a similar manner to maintain coherence and flow while ensuring the accuracy and validity of all written work. Assigned sections of documentation were developed separately and stored in a central location with content and date stamps. The document identification scheme followed the following structure:

- Project Identifier
- Document Title
- Creator
- Revision number
- Date stamp

The master version of the report and other final documentation was managed by a single team member to avoid unauthorized edits or additions. All content was reviewed by the product owner prior to addition to the master report. A date-stamped duplicate of the master report and/or other final documentation was created each time an addition was made to document the changes. These measures have been sufficient to maintain model and document configuration integrity throughout the course of this effort.

D. CHAPTER SUMMARY

The systems engineering processes and approach described served as a structure for the development and refinement of the products created through this effort. The use of these processes ensured that the delivered final product conforms with accepted system engineering procedures and practices. The use of the agile scrum development framework allowed for rapid iteration of concepts and model elements while simultaneously supporting a flexible and adaptable development schedule to incorporate alterations or additions. Critical to the development framework was a robust configuration management scheme capable of accounting for said changes. The methods described proved effective for managing rapidly iterated model configurations. Use of these methods and practices enabled the team to efficiently model the system and recommend a process to the stakeholder.

IV. MODEL DEVELOPMENT

A. SYSTEM MODEL DEVELOPMENT PROCESS

The system model development process, shown in Figure 4, was developed to establish a standard procedure in developing conceptual system models early in a system or project's lifecycle. The model development process was created using the object-oriented systems engineering methodology (OOSEM) as a guide for the phases within the process. The process begins in the problem definition and analysis phase where the problem was defined with stakeholder concurrence and analysis to determine a recommended solution. A decision is then made based on the maturity of the solution to either integrate the existing solution set, if it is mature enough, or to develop a solution if one does not exist or is too immature. For this report is assumed that the decision has already been made that an immature solution will be pursued in the local implementation of the Digital Engineering Environment (DEE).

During the model development process, the capstone team sought to meet the project objectives by utilizing the scrum framework with an iterative development approach. Team members were individually assigned diagrams through the sprint planning process. Completed diagrams were peer reviewed for content, flow, and formatting, and then were added into the master model. Periodic stakeholder reviews, including progress reviews, were conducted to gather feedback; feedback influenced model design and development to meet the stakeholder needs. Simulations of the model using a designed scenario were also performed iteratively throughout the development process to produce model data and artifacts.

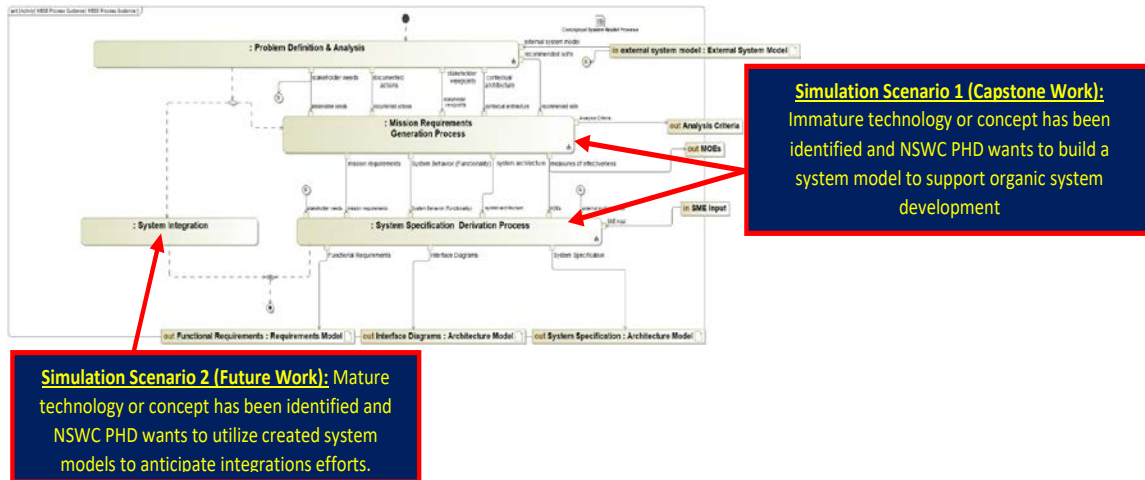


Figure 4. Overall System Model Development Process.

1. Simulation Scenarios

The basis for the selection of the simulation scenarios and the corresponding activity diagrams were determined by the project objectives. Stakeholder analysis and the sponsor command objectives played a key role in the selection of the example scenarios to represent model function. The sponsor's prime objectives for in-service engineering played a key role in the selection of the following scenarios:

Addressing new business capabilities (Simulation Scenario): A new incoming business capability has been identified; or the command performs an internal audit which identifies a desired new capability. The capability set is immature and there is not an existing system infrastructure that supports the capabilities. A system model is to be built from scratch to present conceptual information and high-level requirements of the desired solution. Post model development, the system model would be distributed to a development team for to be updated and refined as the system supporting the capability matures.

Addressing new capabilities to an existing system (Future Simulation Scenario): This scenario would focus on the addition of a capability set to an already existing system. A system model or system of systems (SoS) model exists and would be utilized to perform alternative analysis on the change prospects. Updates to the system model would happen iteratively as the change design matures and is implemented.

Activity diagrams were derived from these use cases. The pertinent activity diagrams were identified by determining the key aspects that affect the example scenarios. The activity diagrams that were modelled were:

- Problem Definition and Analysis
- Mission Requirement Generation Process
- System Requirements Generation Process
- System Integration

B. OBJECT ORIENTED SYSTEMS ENGINEERING METHOD

This capstone has utilized elements of the "Object Oriented Systems Engineering Method" (OOSEM) found within the practical guide to SysML. "[The] OOSEM is a top-down, scenario-driven process that uses SysML to support the analysis, specification, design, and verification of systems. The process leverages object-oriented concepts and other modeling techniques to help architect more flexible and extensible systems that can accommodate evolving technology and changing requirements" (Friedenthal, Moore, and Steiner 2012). The OOSEM was created in 1998 and has been further refined by an International Council on Systems Engineering (INCOSE) OOSEM working group (Friedenthal, Moore, and Steiner 2012). It is an INCOSE accepted systems engineering management process. Most of the capstone artifacts have been captured using MBSE and SysML artifacts. These artifacts include stakeholder requirements, system requirements, problem space architecture, solution spaces architecture, use cases, and parametric diagrams. Due to the large nature of model-based artifacts, this capstone chose to employ elements of OOSEM due to its applicability in both SysML development and SysML enabled management. Figure 5 shows the OOSEM steps that helped this capstone team design a tailored process for developing a conceptual system model.

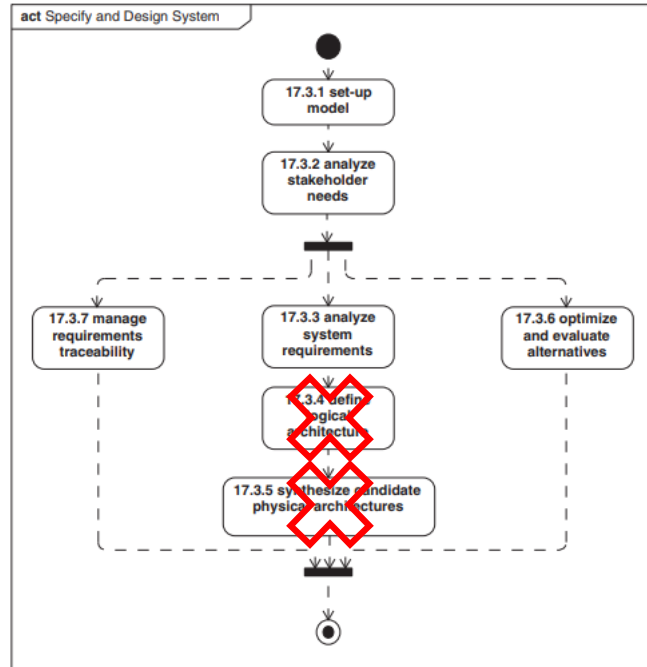


FIGURE 17.2
OOSEM *Specify and Design System* process. The action numbers refer to subsection where the action is described.

Figure 5. OOSEM Specify and Design Process. Adapted from Friedenthal, Moore, and Steiner (2012).

The SE steps shown in the figures are set-up model, analyze stakeholder needs, manage requirements traceability, analyze system requirements, optimize and evaluate alternatives, define logical architecture, and synthesize candidate physical architectures. This process was tailored to not include the optimize and evaluate alternatives, define logical architecture or synthesize physical architecture. These steps were removed as this capstone will not produce a full logical or physical system and would be up to the development team to refine the model to include the architecture definition. Instead, the focus will remain on developing a conceptual SysML model that describes the objectives laid out in the simulation scenario: The need of a MBSE environment that provides digital SE capabilities and can exchange meaningful data with other platforms within the digital transformation domain.

The model development utilized an iterative design process where incremental builds of the model were developed. These iterative builds incorporated a feedback loop to

receive stakeholder input on the developed models. Stakeholder feedback has subsequently been incorporated into each iterative design of the model.

C. PROBLEM DEFINITION AND ANALYSIS

The problem definition and analysis phase, as shown in Figure 6, is meant to support the identification of the problem and need in a data-driven way, and to devise a solution that will help satisfy the needs. The process begins with a signal that triggers the first step in the process. The trigger can be scheduled or unscheduled, as in this process could be performed with a scheduled integrated product team (IPT) annually, every 6 months, etc., or it could be spontaneous, driven by innovation within the enterprise or based on direction provided by enterprise leadership.

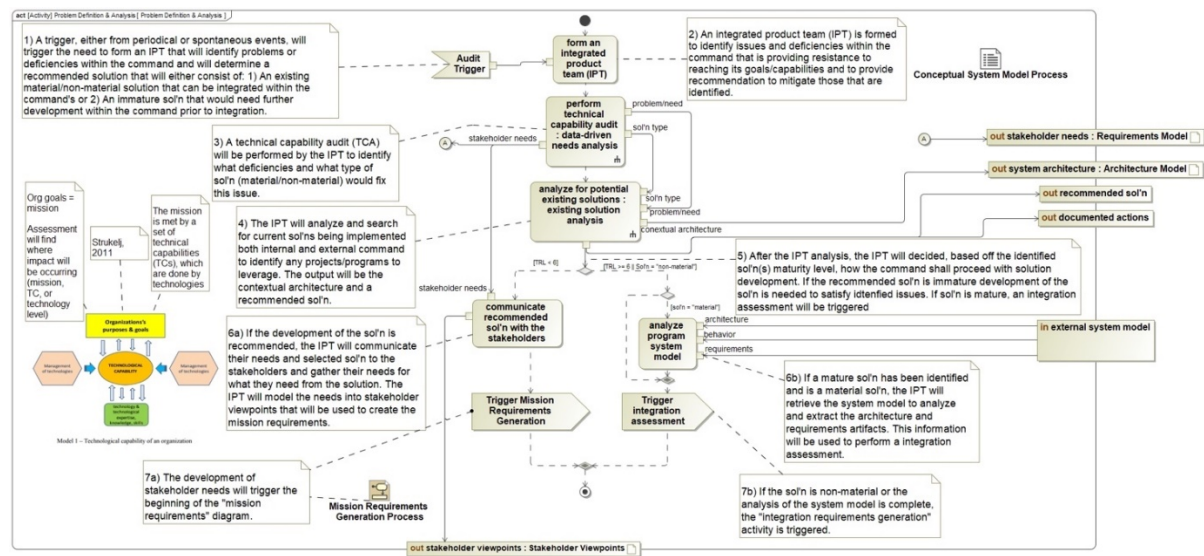


Figure 6. Problem Definition and Analysis Activity Diagram.

Once the IPT is formed, their first responsibility would be to perform the technical capability audit (TCA). The TCA is the process of analyzing technical capabilities within an organization in a data-driven way to identify potential problems and solutions to those problems (Mohammadi, Elyasi, and Kiasari 2014, 5–8). Technical capability in this context is defined as an organization’s ability to utilize technologies in a way that is most useful to the organization’s goals and mission. Technologies in this case refer to the machines and

processes that the people of an organization utilize to perform their daily activities. Technology capabilities are influenced by technological innovation and changes in organizational goals or missions (Strukelj and Dolinsek 2011).

The IPT develops a set of quantitative and qualitative indicators in which they can disburse to the workers of an organization to receive feedback. The indicators that form the TCA have four different aspects: hard, human, knowledge, and organizing and managing of technical capabilities. Hard aspects are the physical equipment, are the tools currently available to the workforce meet their needs. Human aspects relate to the skill set of the workforce and answers the question, “Does the workforce have the right skill set to perform this technical capability?” Knowledge aspects pertain to the understanding of the technological capability and is enough information known about it to make it a worthy investment. Lastly, organizing and managing of the technical capability is an aspect that focuses on how well an organization is structured, or funded, to develop new technical capabilities and the quality of the technical capability management process (Mohammadi, Elyasi and Kiasari 2014, 8–12). Feedback to the IPT from the workforce on the indicators can support the identification of problem areas where a solution is needed in order to satisfy the technical capability (Mohammadi, Elyasi and Kiasari 2014, 13–14).

For this example scenario, it is assumed that the TCA has already occurred and the problem has been identified to be a lack of hard aspects that is causing the greatest deficiency in achieving a MBSE technical capability at the organization. Upon completion of steps 3 and 4, as shown in Figure 6, the IPT should have completed the development of the stakeholder analysis, viewpoints and contextual architecture presentation views. Example of the stakeholder analysis is presented in Figure 7, while examples of viewpoints and contextual architecture are shown in the following section in Figures 12 and 13, respectively. For the purposes of this capstone, a formal stakeholder analysis was not performed and the Unified Architecture Framework (UAF) provided guidance on stakeholder composition, concerns, purpose and presentation methods (Object Management Group [OMG] 2020).

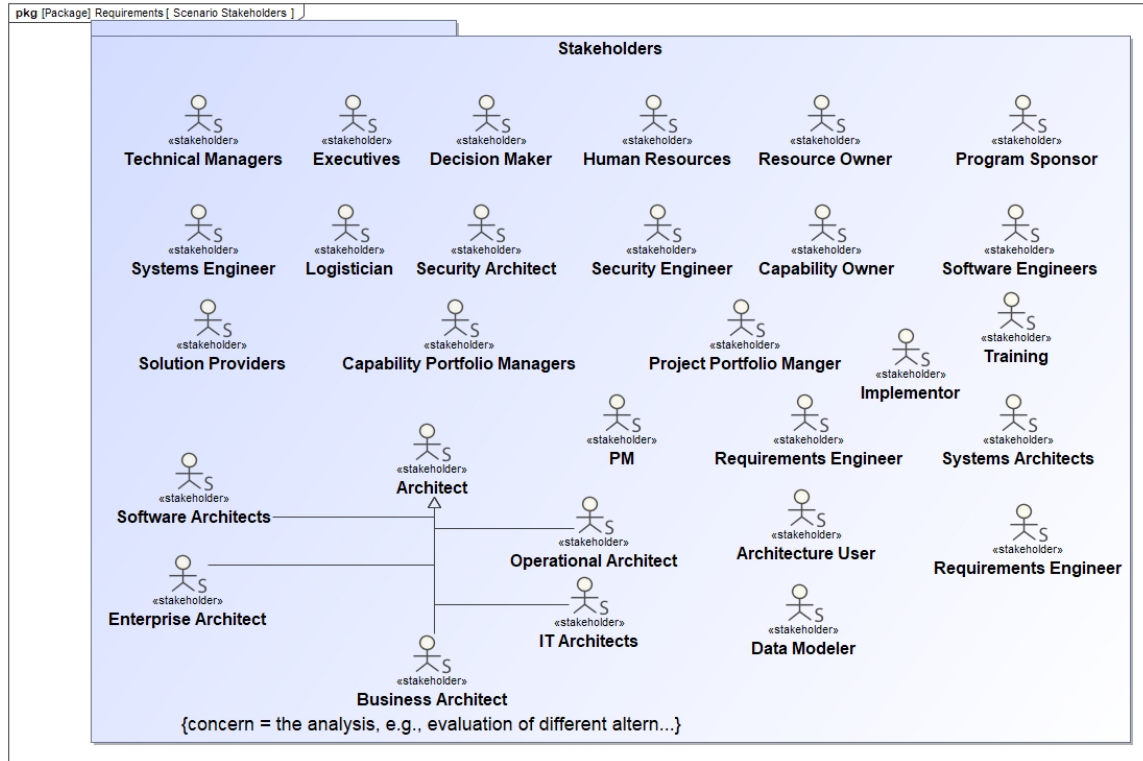


Figure 7. Analysis of the Stakeholders.

Based on the maturity of the solution, the acquiring organization decided whether the system model developing process needs to begin at the beginning or a system model or artifacts exist, and the solution will be integrated into the model. For the capstone’s scenario it is assumed that a new model describing the solution and the process will signal for the mission requirements generation process to begin. If the solution needs to be integrated into an already existing system model or SoS model, the model is consumed by the solution’s model project and any updates and refinements are made as the solution matures.

D. MISSION REQUIREMENTS GENERATION

The mission requirements activity diagram in Figure 8 provides a set of top-level requirements and traceability to the other aspects of the system model to support the management of requirements throughout the completion of the process. This diagram will

merge node has no effect on the control flow of the process the IPT goes through from the initialization.

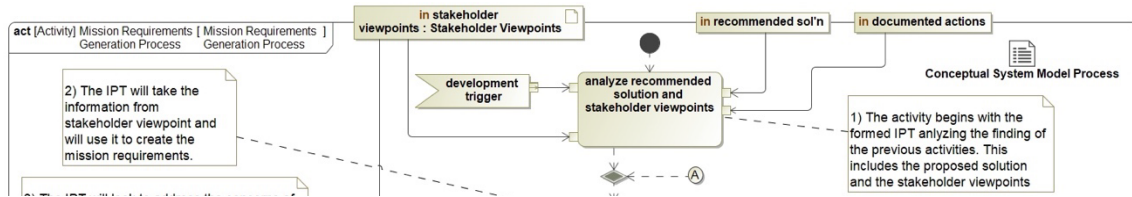


Figure 9. Mission Requirements Activity Diagram Expanded.

In Figure 10, the control flow continues into the development of mission requirements. Mission requirements are built from the understanding of the problem and stakeholder needs that were established in the previous phase. From the development of mission requirements, the control flow then goes into a SysML fork where the IPT would perform three data collection tasks simultaneously. To exit the fork node the IPT must generate a block definition diagram (BDD) for system context, retrieve and capture measures of effectiveness, and decompose the machine within the context of the BDD.

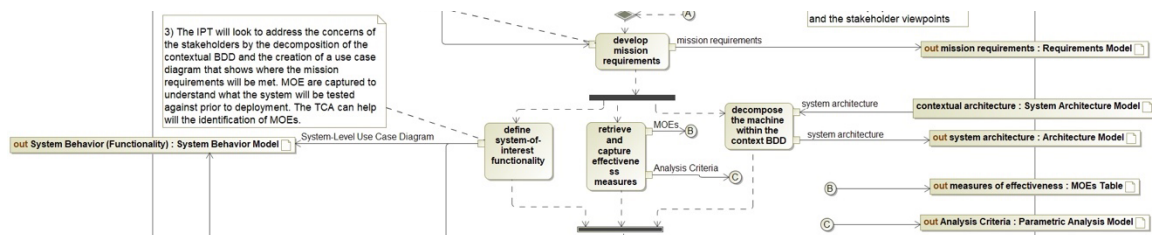


Figure 10. Mission Requirements Activity Diagram Expanded 2.

The IPT will look to address the concerns of the stakeholders by the decomposition of the contextual BDD and the creation of the use case diagram that shows where the mission requirements will be met. The measures of effectiveness are captured to understand what the system of interest (SOI) will be tested against prior to deployment and implementation. The developed indicator from the TCA performed in the problem definition and analysis phase can be utilized to further strengthen the measures of

effectiveness. From this block the output of the BDD system context diagram is generated. This artifact is used to initialize the system specification diagram.

The last block within the fork requires the IPT to decompose the SOI within the context of the BDD. The object flow needed to complete this task is derived from the contextual architecture of the problem definition and analysis activity diagram.

Figure 11 continues with the last section of the process. With the satisfaction of the three preceding taskings, the IPT control flow moves to join the control flows. The IPT will now be capable of defining the relationships between the solution contextual architecture and the mission requirements. As this development matures, the object flow output of a high-level system architecture transfers to the system specification derivation activity diagram.

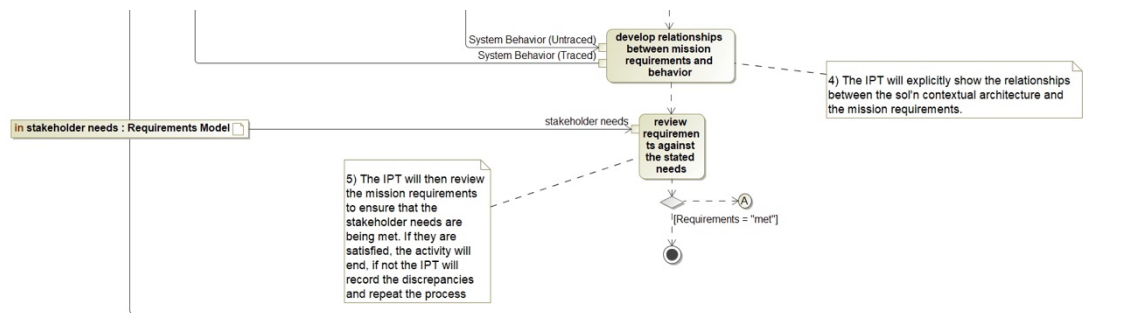


Figure 11. Problem Definition and Analysis Expanded.

The final logical control of the mission requirements activity diagram is to ensure that the stakeholders needs are being achieved. If gaps in requirements are identified, then the control flow allows for a repeat of the process flow for the IPT. The exit criteria for the mission requirements activity diagram is for the IPT to review the stakeholder requirements against the generated mission requirements. If the stakeholder requirements are sufficiently satisfied the control flow exits the mission requirements activity diagram.

Mission requirements definition and refinement is an integral phase of the overall capability achievement of MBSE and/or MBPS within the digital engineering environment. The established object and control flows that this capstone project illustrates

during the mission requirements activity diagram through the generated artifacts demonstrate the importance for an IPT to decompose and address the overall stakeholder need(s). The traceability aspect that OOSEM provides to the overall intent of the mission requirements diagram allows for further exploration of validation and verification that the system and component requirements satisfy the stakeholder requirements.

This part of the process was verified by the development of the input and output artifacts to ensure the required system model data was being produced, the following sections will discuss a selected number of these artifacts and will provide a short description pertaining to the artifacts importance to the overall presentation of the system model data.

1. Simulation Results

The process above describes an overall method for the second iteration of system model development. The process includes further refinement of the architecture facet of the system model, and it introduces the behavior and requirement viewpoints. In order to validate this method, scenario one was used as a use case and the system specification process was executed. The assumptions prior to moving into the process are that all required input artifacts have been completed from the previous activity diagrams. These input artifacts are displayed below as shown from the system context of the DEE and MBPS, where DEE is the system of interest and MBPS is the identified external system.

a. Mission Requirements Generation Inputs

The first activity within the process requires the integrated product team (IPT) to revisit the information provided from the problem definition and analysis process. Other than the recommended solution, the IPT will be using the information provided in the stakeholder viewpoints as a guide to developing the different presentation views within the model. The stakeholder viewpoints represent different stakeholder perspectives and helps capture subsets of the model that are of interest to the stakeholder (Friedenthal, Moore, and Steiner 2012). Shown in Figure 12 is the actual resources viewpoint. This viewpoint is of interest to a few different stakeholders, including the solution provider, business architect,

human resources, and the systems engineer. Viewpoints capture stakeholder concerns and their preferred methods of presentation (OMG 2020).

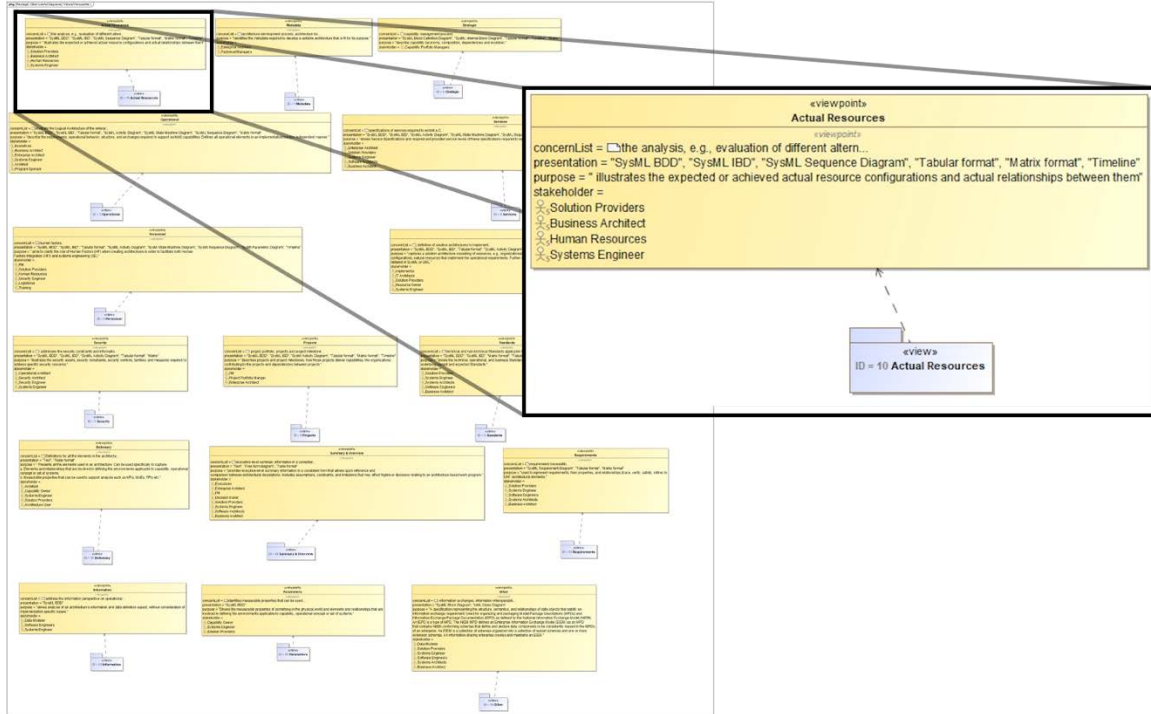


Figure 12. Stakeholder Viewpoint (Example).

Figure 13 displays another input from the problem definition and analysis phase that is used to help support the decomposition of the SOI. This artifact will define what is being decomposed, but the majority of the information needed to support the development would come from other programmatic artifacts, like a concept of operations (CONOPS), that would give the modeler a better idea of the necessary sub-systems or components needed to support the requirements for the system of interest.

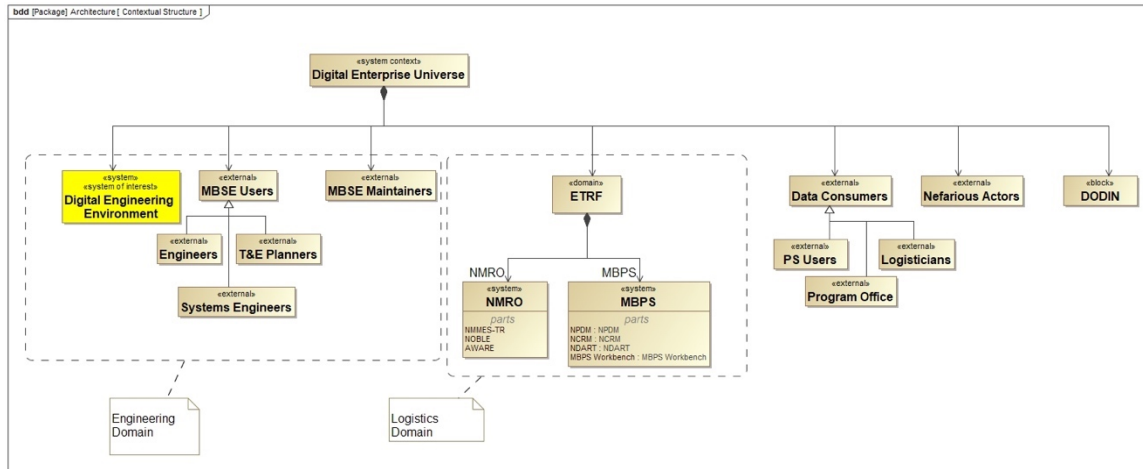


Figure 13. Contextual Architecture.

Figure 14 represents an example of a set of stakeholder needs in diagram form. A diagram was chosen for this artifact, but a table is also an acceptable way for the same information to be displayed with the SysML syntax. The stakeholder requirements should always be the alignment mechanism during the development of systems and system models. SysML toolsets provide the platform for modelers to show stakeholders that their needs are being met and can provide traceable relationships to the modeled needs to ensure the designs are, in fact, meeting the modeled needs. An example of a requirement traceability matrix (RTM) is shown in Figure 15.

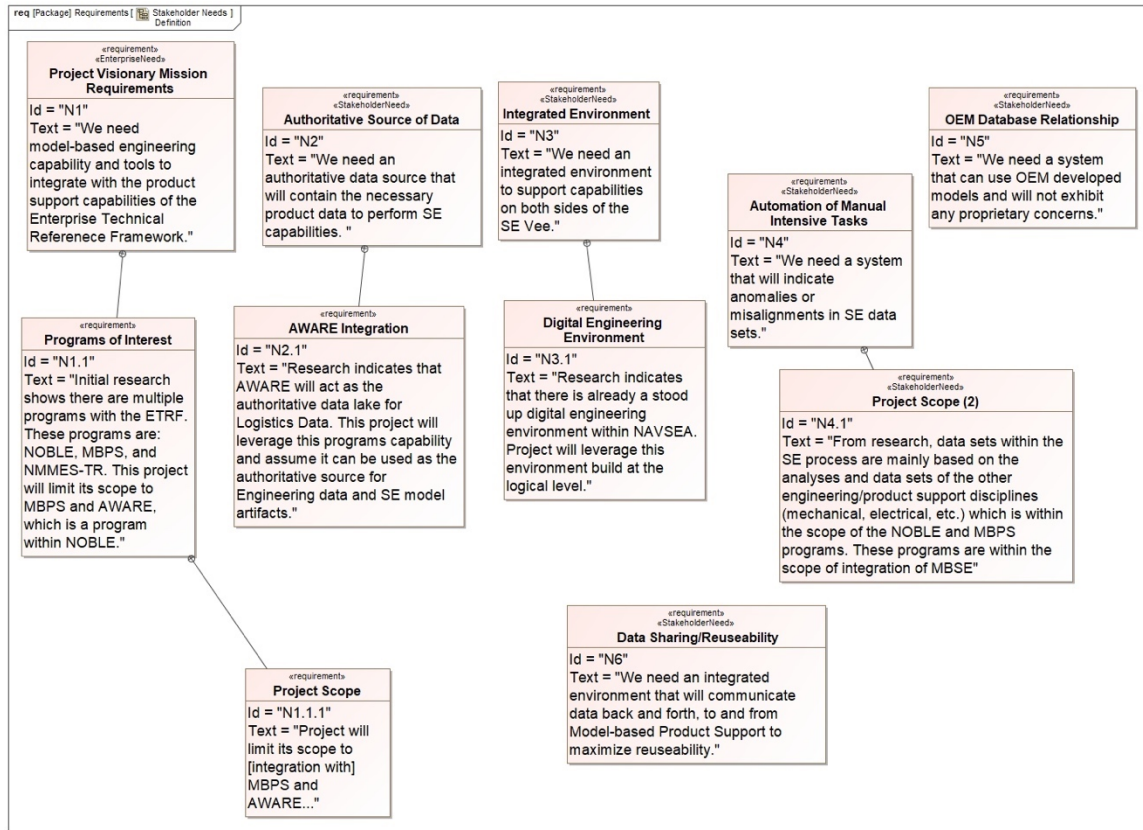


Figure 14. Stakeholder Needs Model.

Mission Requirements	Stakeholders
12 Mission Statement	Architect, Architecture User, Business Architect, Capability Owner, Capability Portfolio Man, Data Modeler, Decision Maker, Enterprise Architect, Executives, Human Resources, Implementor, IT Architects, Logistician, Operational Architect, PM, Program Sponsor, Project Portfolio Manage, Requirements Engineer, Resource Owner, Security Architect, Security Engineer, Software Architects, Software Engineers, Solution Providers, Systems Architects, Systems Engineer, Technical Managers, Training
12.1 System Digital Twin	
12.1.2 System Architecture	
12.1.3 System Requirements	
12.1.4 T&E Activities	
12.1.5 Model Configuration Management	
12.1.6 System Design	
12.1.7 Element Traceability	
12.2 System Digital Thread	
12.2.1 Communication Pathways with NPDM	
12.2.2 Communication Pathway with OEM Systems	
12.2.3 Communication Pathways with AWARE	
12.2.4 Communication Pathway with Identified Users	
12.2.5 Communication Pathway with DEE Internal System(s)	
12.2.6 SE Reports and Documentation	

Figure 15. Example of a Requirement Traceability Matrix.

b. Mission Requirements Generation Outputs

Measures of effectiveness (MOE) are captured in the model as shown in Figure 16. “[Measures of effectiveness] are mission-level performance requirements that reflect value to the customer and other stakeholders. They are derived from the stakeholder needs analysis that includes causal analysis and mission performance analysis” (Friedenthal, Moore, and Steiner 2012). The MOEs help refine the black box behavior of the system of interest by showing which properties and metrics are used to evaluate the system. For example, MOE 12 “required storage space” implies that the system must have a capability of storing data and that the size of the storage is important to the system final capability. The MOEs are also used in the mission requirements diagram to evaluate recommended system solutions.































#	Name	Applied Stereotype
1	 bandwidth consumption	 moe [Property]
2	 development cost	 moe [Property]
3	 lifecycle sustainment cost	 moe [Property]
4	 past performance	 moe [Property]
5	 required storage space	 moe [Property]
6	 security factors	 moe [Property]
7	 software architecture	 moe [Property]
8	 software maturity	 moe [Property]
9	 staff profile	 moe [Property]
10	 staff turnover rate	 moe [Property]
11	 standard hardware compatibiliy	 moe [Property]
12	 system availability	 moe [Property]
13	 system redundancy	 moe [Property]
14	 user capacity	 moe [Property]
15	 XML support	 moe [Property]

Figure 16. Example MOE Table.

Another function of the MOEs within the system model can be to create a criterion to which the program can base its decision making. Shown in Figure 17 is a parametric diagram that provides an example of how parametric diagrams can be used in the design selection process. Contracting firms may submit bids to design the system laid out in Figure

17. The organization that sent out the request could use engineering analysis criteria in the parametric diagram, based off the modeled MOEs, to establish a plan for evaluating each submission. By placing a value on each MOE based on how well the contractor met that MOE, the evaluators will determine an overall score based on the selection criteria.

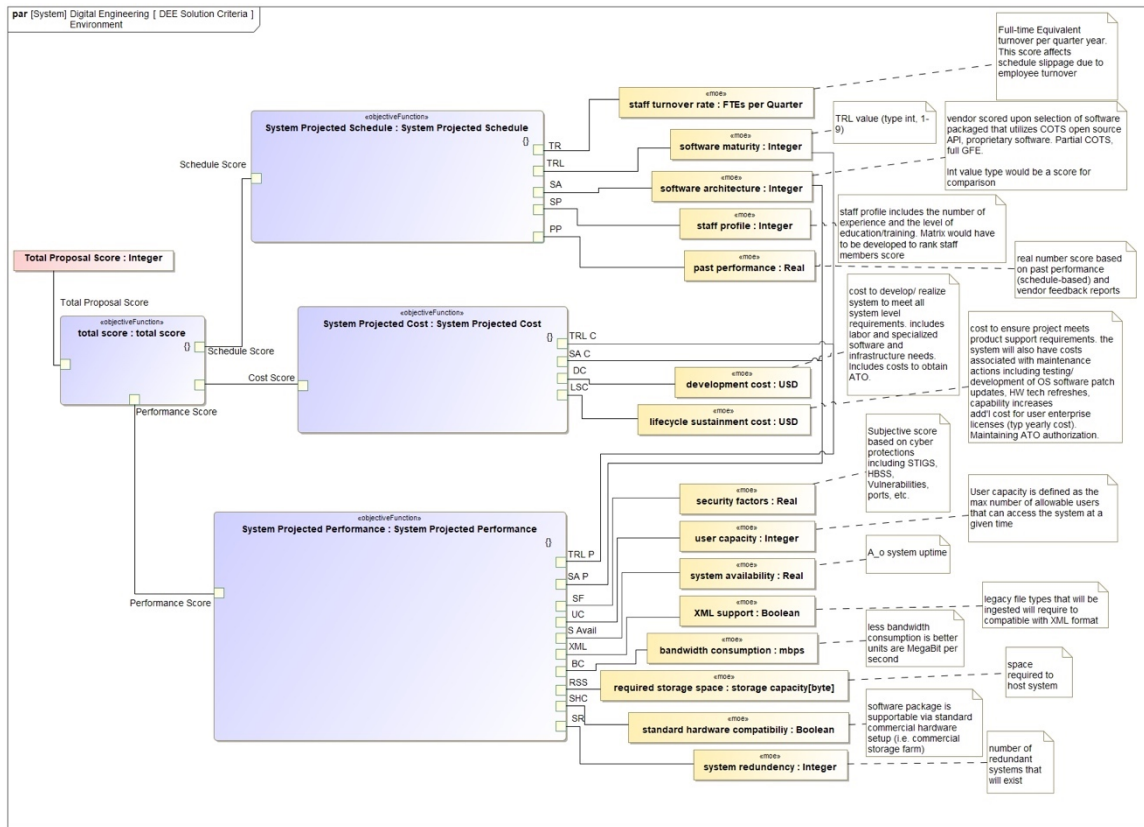


Figure 17. Engineering Analysis Criteria/Selection Criteria.

As mission requirements have been developed within the model, the system modeler will look to begin the decomposition of the system architecture, based off the understanding of what is required of the system. The program or project is still in the very early stages in this scenario and there may be little information. Our simulation scenario from the overall process description is based off a set of known, but immature, concepts and capabilities. As shown in Figure 18, like-capabilities were grouped inside the capabilities boundary and assigned to the different capability areas, they could be called sub-systems, within our system of interest. These capabilities would support the

development of the top-level objectives, to create model and document artifacts that reflect the system of interest.

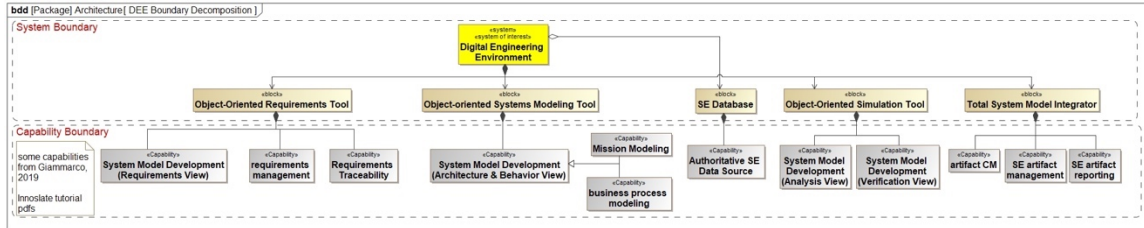


Figure 18. Boundary Decomposition.

Figure 19 shows the final output and focus of the mission requirements process. The complete list of mission requirements is captured in the model and the proceeding processes use this diagram as an input at the start of the next process, system specification process.

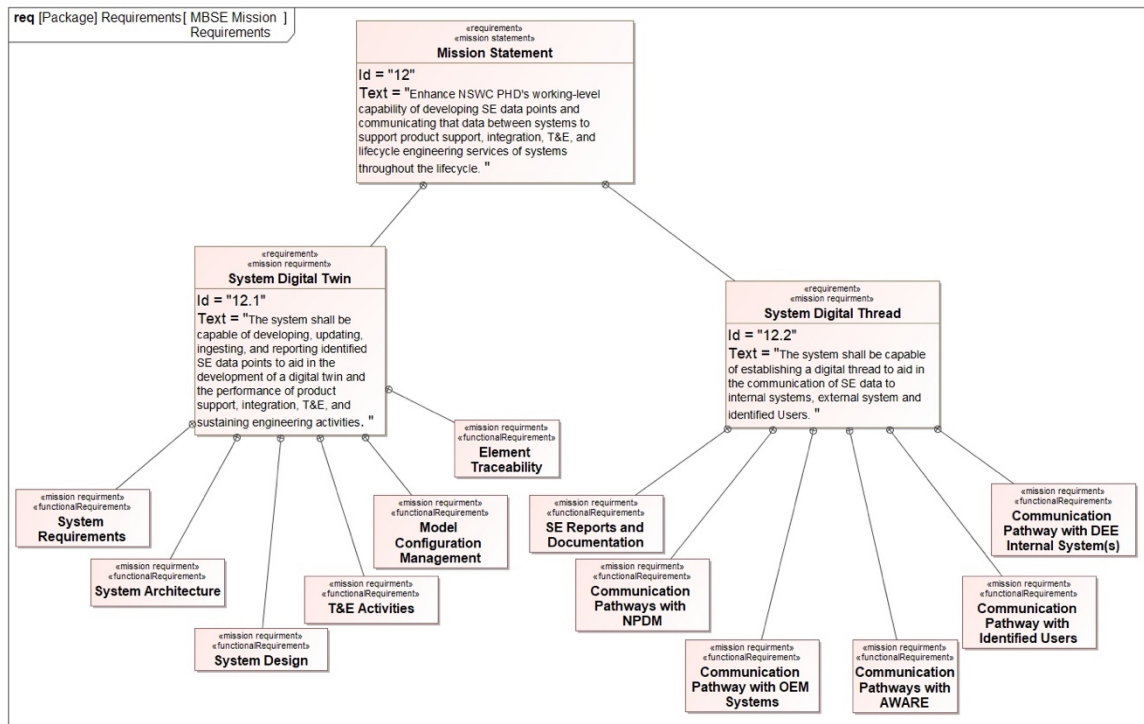


Figure 19. Mission Requirements: Requirements Model.

With an understanding of the capabilities and requirements, the system modeler can begin brainstorming system use cases that will be later refined to describe behavior or be selected as a test case for system verification. Use case diagrams present the basic functionality of the system and its relation to performers or requirements. Figure 20 is the developed use case from the capstone’s scenario simulation.

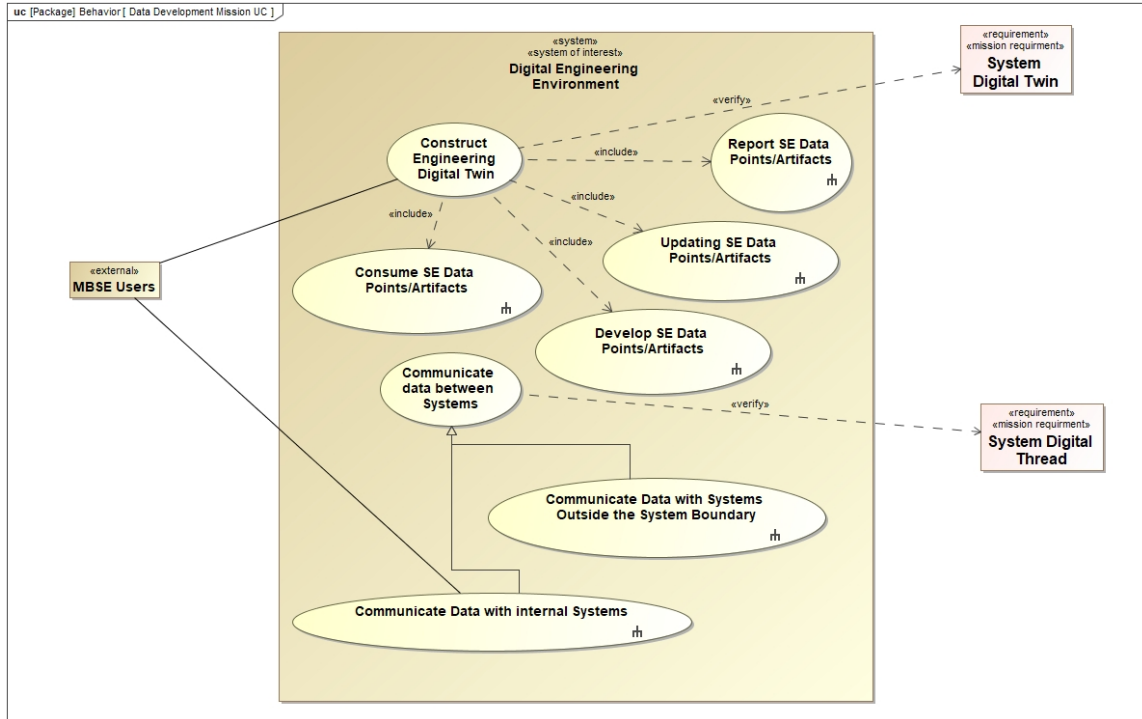


Figure 20. DEE Use Case Diagram

E. SYSTEM SPECIFICATION DERIVATION

The second activity that follows the mission requirements generation is the system specification derivation process. The purpose of the system specification diagram is to further mature the system model viewpoints, allowing for the further development of more mature requirements and a system specification. This activity is necessary to build an understanding of how the system of interest will behave within the context of external and internal systems. Some constraints imposed on this activity flow down as inputs created during the mission requirements generation. These constraints include mission

requirements and a block definition diagram (BDD) system context diagrams of the machine. The output of the activity is a system specification, an encapsulation of the SysML elements that are allocated to or share relationships with the system of interest. With a clear definition of system behavior and function, a modeler and stakeholder can use the process to develop a list of functional requirements that describe what the system of interest is required to do. However, this diagram does not specify how the system of interest will perform its functions. This process occurs earlier in the life cycle in the conceptual system design phase. The machine specified in the diagram is the system of interest for which the functional requirements are being generated. An overview of the activity is shown in Figure 21.

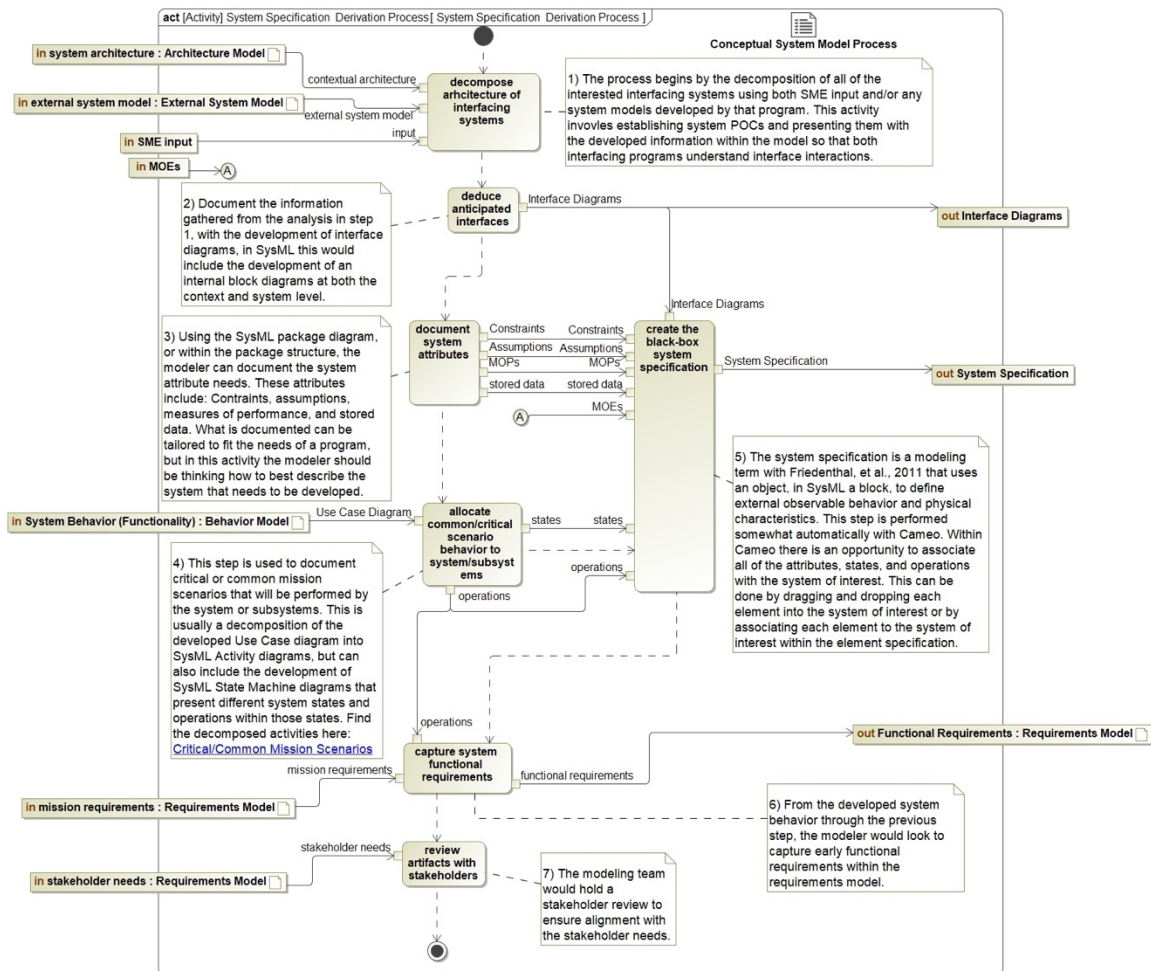


Figure 21. System Specification: Derivation Activity Diagram

The diagram inputs in Figure 21, shown on the border of the diagram are: mission requirements, system architecture, external system model, MOEs, subject matter expert (SME) input, system behavior (functionality), and stakeholder needs. Many of these artifacts were developed in the previous phases and will not be discussed further. Artifacts consumed in this process that were not developed in a previous phase will be discussed in the simulation results for the system specification derivation phase.

The system specification derivation begins with the first phase of decomposing architecture seen in Figure 22. The process begins at the initial node shown as a black circle at the top of the diagram.

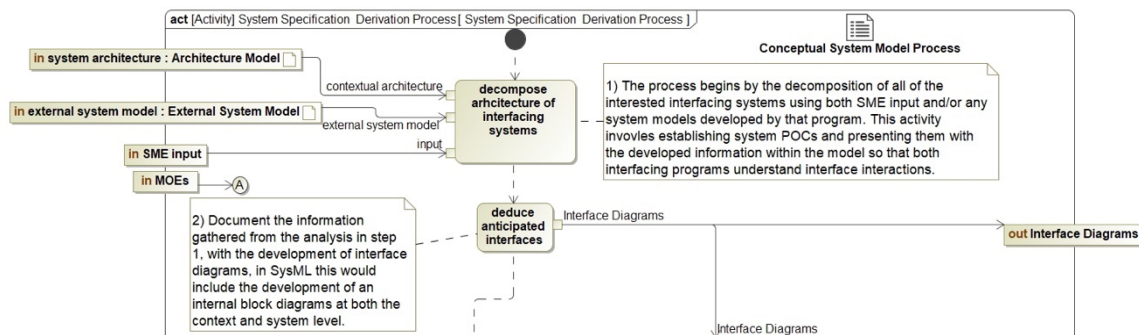


Figure 22. System Specification: Decompose Architecture.

The first action of the process is to decompose the architecture of interfacing systems. The action focuses on defining touch points between the system of interest and the external systems it will interact with. There are three inputs that facilitate this process: the architecture model, the external system model, and SME input. This first action involves searching for SMEs of external systems that can provide detailed interface diagrams and/or system models. The identified SMEs will also be presented with developed information for the SOI. In this manner, both groups will be able to identify potential points of integration and the types of data that will need to flow between the two. Once complete the next action is initiated, to deduce anticipated interfaces. In this action, the two groups will use the information found in the previous step and create an interface diagram within SysML. Internal block diagrams (IBD) of the external systems and system

of interest will assist in defining interfaces. Subsystems and subfunctions can help identify exact interface requirements between the systems. A model artifact is created, and the action outputs a developed interface diagram.

Working through the rest of the diagram, the next actions support the development of the black box specification of the system of interest shown in Figure 23.

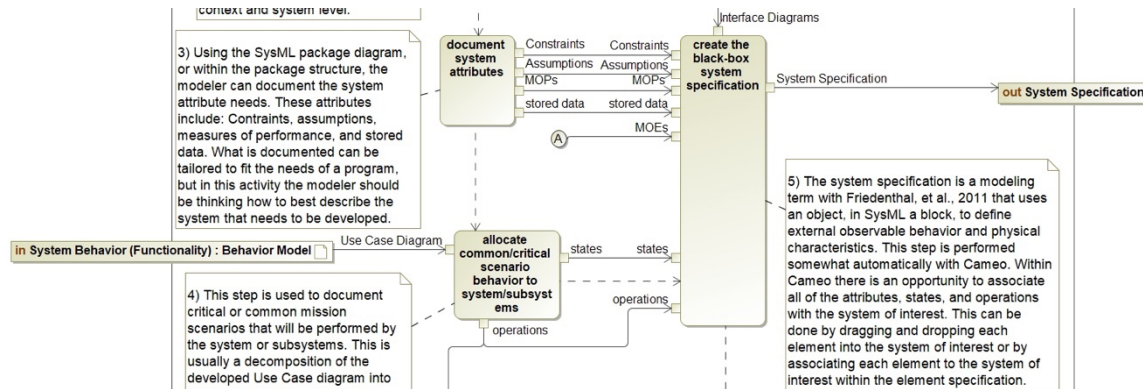


Figure 23. System Specification: Black Box System Specification.

In order to accomplish this, system attribute needs are documented. This includes defining constraints, assumptions, measures of performance, measures of effectiveness and data requirements. By defining the attributes of the system, the black box specification can be refined to fit the constraints and needs of the system. In addition to system attributes, behavior models are created to show high level behavior based on system needs. This is accomplished by creating common mission scenarios for the system of interest and designating critical/common behaviors or functions that system is expected to perform. These functions lead to the creation of behavior diagrams show interactions between subsystems previously identified in the IBD. Using all these inputs and constraints, the black box specification is developed. This can be captured as a BDD that lists model properties including constraints, parts or subsystems, properties or system functions, references, and value blocks tied model such as associated MOEs.

Lastly, the functional requirements are generated with the last two actions in the process in the functional requirements phase shown in Figure 24. The functional

requirements are the main desired output artifacts of this process. and all other actions have led to its final production. This artifact is the focus of what the process is trying to create.

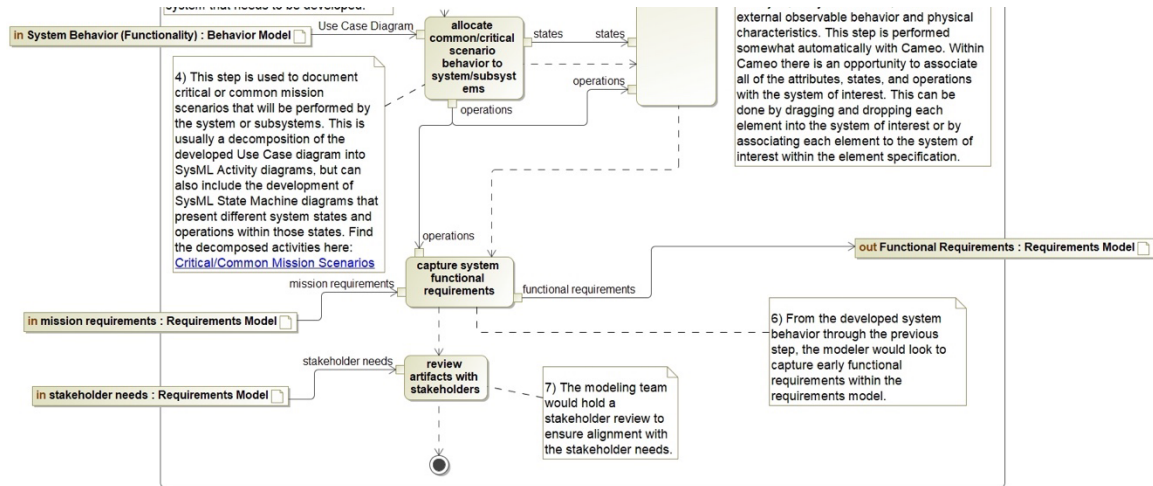


Figure 24. System Specification: Functional Requirements.

Using all the information from the previous steps, the functional requirements are drafted and tied to mission requirements. The mission requirement feed directly into this action to ensure that the functional requirements are derived and traced back to higher level mission requirements. A detailed list of functional requirements is generated and captured either in a requirements diagram or table. These requirements are then reviewed with stakeholder in order to receive concurrence on the final product. This review also ensures that the stakeholder needs are accurately addressed and traced to the functional requirements.

1. Simulation Results

The process above describes an overall method for developing the system specification and decomposing top-level requirements into functional system requirements. It is assumed that all required input artifacts have been completed from the previous activity diagrams prior to moving into the system specification process.

a. System Specification Process Inputs

Artifacts developed in the mission requirements generation phase and presented in the previous section are fed into the system specification process from the mission requirements generation. Mission requirements are used in the system specification process to refine and constrain system behavior and is ultimately traced directly to the functional requirements output. The system operational behavior is derived from the basic functionality expressed in the use case diagram and allocated to systems and sub-systems. As shown in Figure 24, functionality is traced to a mission requirement, enabling the support of system verification later. This analysis ensures that the system function requirements, which are generated from the behavior diagrams, are also traced back to a mission requirement.

The stakeholder needs in Figure 14 are compared against the developed functional requirements of the system of interest. This is the last step in the diagram and is performed to ensure that the functional requirements align and address the previously created stakeholder needs. The mission requirements and stakeholder needs are reviewed with the stakeholder prior to finishing the process.

The BDD in Figure 25 shows the subsystems and properties of the overall external system MBPS. The MBPS system is decomposed into four subsystems: NPDM, NCRM, NDART, and MBPS workbench. Each subsystem contains parts, properties and data values. This detailed view of the external system assists in identifying potential integration points with the SOI.

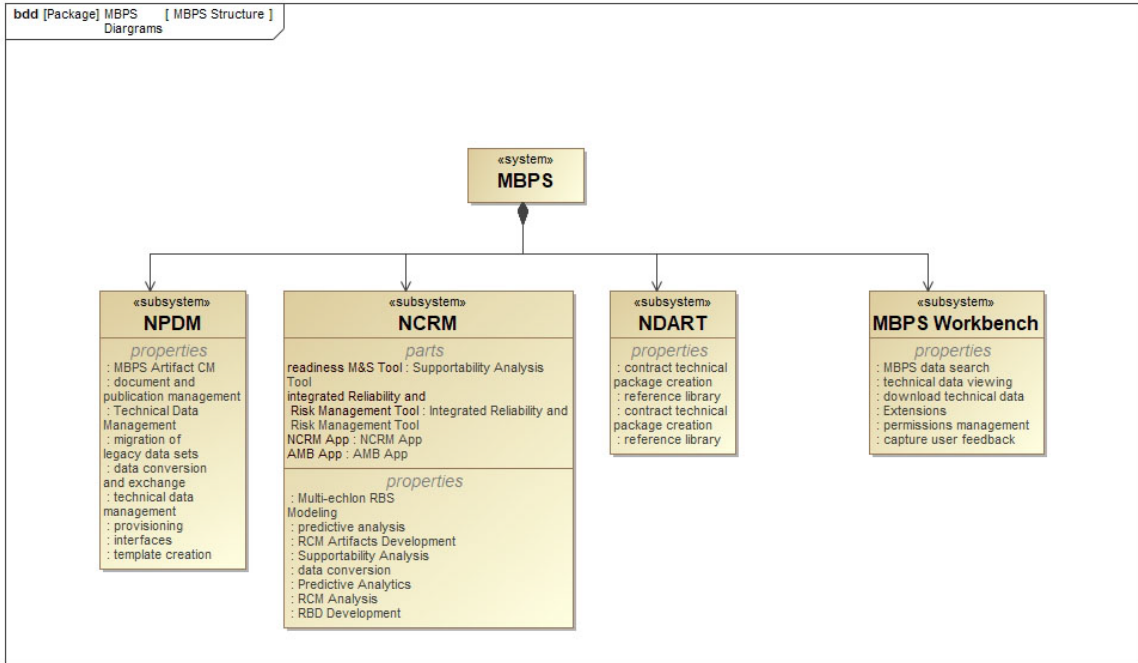


Figure 25. External System (MBPS) Model.

Figure 26 displays a free form diagram (FFD) of the six common/critical mission scenarios (functional behaviors) the black box is designed to perform. The FFD contextually allows for the presentation of various behaviors along all structured nested diagrams for exhibition. Each mission scenario has at least one decomposed diagram for further depth and relational exploration. For example, the scenario for communicate data with internal systems has three nested diagrams tied to its structure. Those diagrams are a sequencing diagram for the internal systems automatic updates, an interaction diagram for the internal systems manual update, and an interaction diagram for the internal systems save new data. One of these behavior diagrams, "updating se data points/artifacts" can be seen in Figure 27. Each functional behavior has a developed diagram as an artifact in the mode.

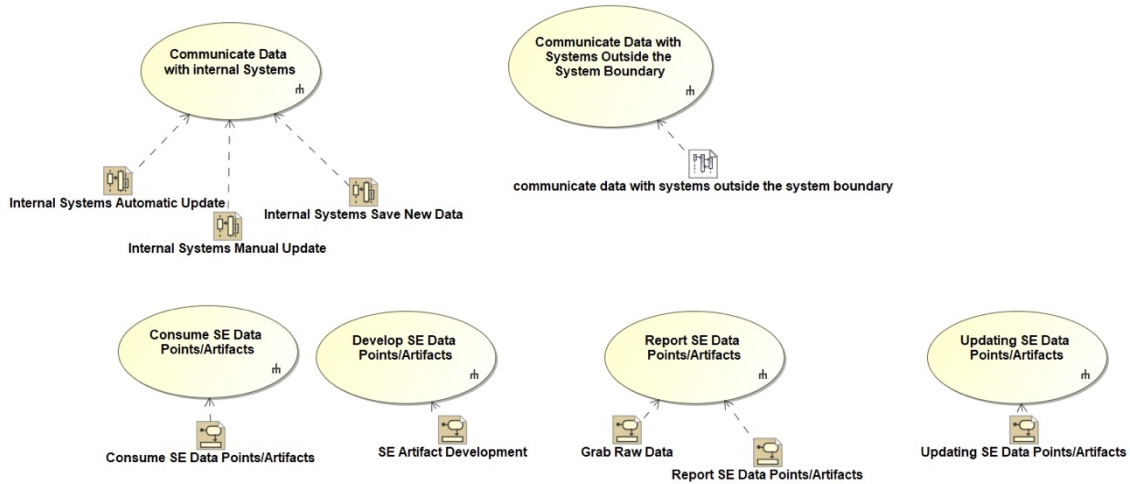


Figure 26. Mission Free Form Diagram: Critical/Common Mission Scenarios.

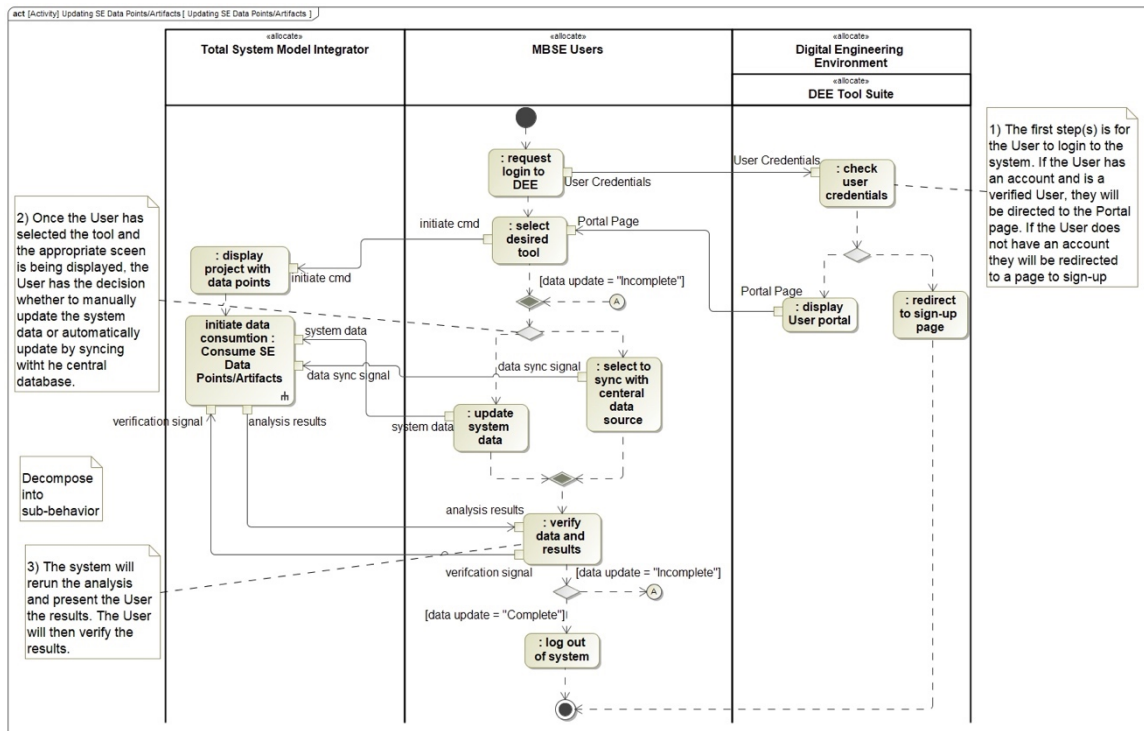


Figure 27. Behavior Diagram: Updating SE Data Points and Artifacts.

b. Model Output Artifacts

The interface diagram is shown in Figure 28. This diagram describes various interfaces between the system of interest and external systems. In this case it is showing

the system of interest (DEE) and how it interfaces with the three external systems: DODIN, MBPS, and SE Database. The diagram also shows allocated subsystems where different elements, including classes and blocks, are passed.

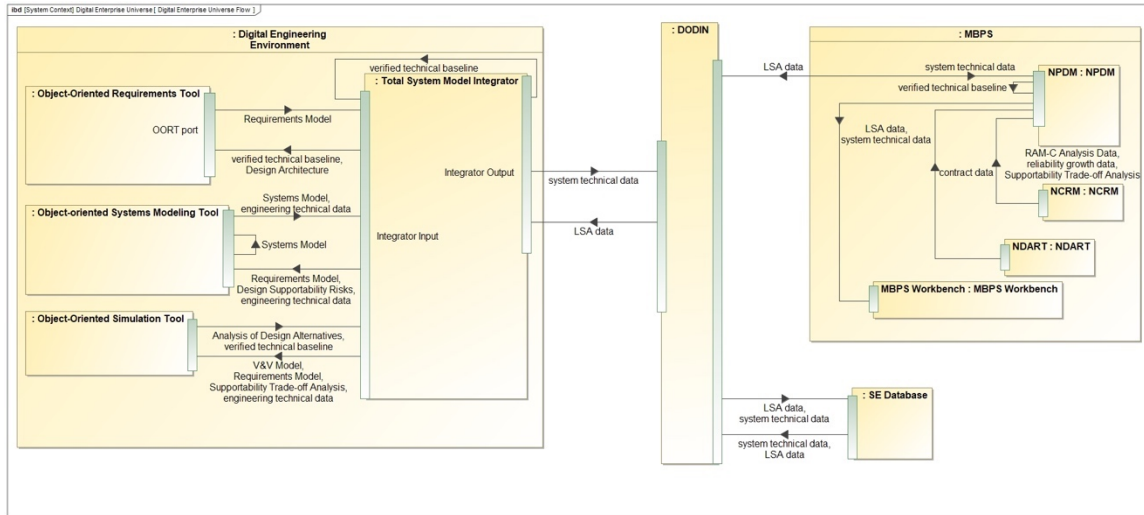


Figure 28. Interface Diagram.

The culmination of all the collected system data is shown in Figure 29 as the system specification. The system specification is an overview of the data elements contained within the SOI system model. The system specification displays the architecture information, allocated behavior, stored data elements, constraints, MOEs, MOPs, parametric information, and other related data items captured with the system model development process.

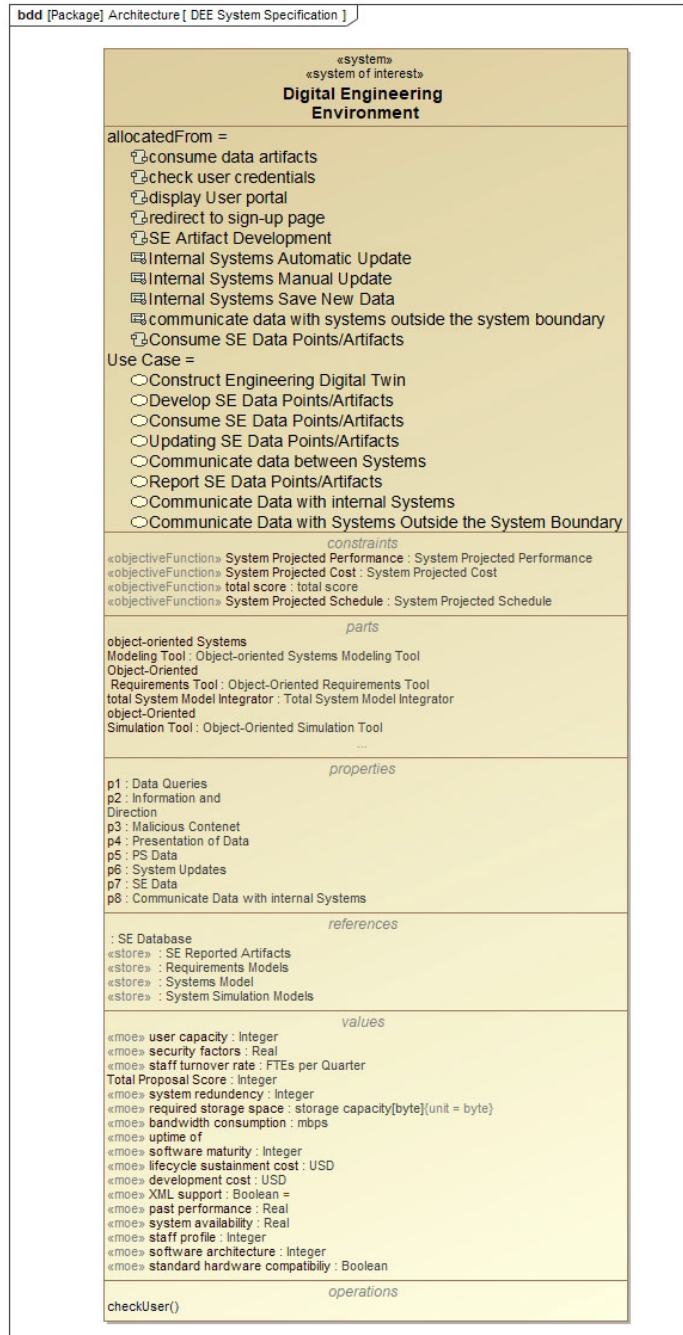


Figure 29. DEE System Specification.

The final artifact produced by this process is a list of functional requirements as shown in Figure 30. The functional requirements describe how the system of interest needs to perform. When developed through the described process, these requirements can be directly traced back to mission requirements and are validated against stakeholder needs.

#	△ Name	Text	Satisfied By	Applied Stereotype
1	12 Mission Statement	Enhance NSWC PHD's working-level capability of developing SE data points and communicating that data between systems to support product support, integration, T&E, and lifecycle engineering services of systems throughout the lifecycle.		Requirement [Class] mission statement [Element]
2	12.1 System Digital Twin	The system shall be capable of developing, updating, ingesting, and reporting identified SE data points to aid in the development of a digital twin and the performance of product support, integration, T&E, and sustaining engineering activities.		Requirement [Class] mission requirement [Element]
3	12.1.2 System Architecture	The system shall be capable of developing a system architecture model to include tables and diagrams that can help describe a system's contextual, logical, and physical architecture.		mission requirement [Element] functionalRequirement [Class]
4	12.1.3 System Requirements	The system shall be capable of developing a requirements model to include the development of requirements tables and diagrams.		mission requirement [Element] functionalRequirement [Class]
5	12.1.4 T&E Activities	The system shall be capable of developing and storing models, diagrams, and/or documents that support a system's test & evaluation activities.		mission requirement [Element] functionalRequirement [Class]
6	12.1.5 Model Configuration Management	The system shall be capable of performing configuration management of the models created within the system.		mission requirement [Element] functionalRequirement [Class]
7	12.1.6 System Design	The system shall be capable of developing and storing system design models to include three-dimensional and two-dimensional computer aided models and drawings with any level of detail, from conceptual to commercial-level as described in MIL-STD-3100B.		mission requirement [Element] functionalRequirement [Class]
8	12.1.7 Element Traceability	The system shall be capable of developing and storing relationships between requirements and all other facets of the system model to support the verification of system design.		mission requirement [Element] functionalRequirement [Class]
9	12.2 System Digital Thread	The system shall be capable of establishing a digital thread to aid in the communication of SE data to internal systems, external system and identified Users.		Requirement [Class] mission requirement [Element]
10	12.2.1 Communication Pathways with NPDM	The system shall be capable of exchange data between the Navy Product Data Management system of Model Based Product Support that will support the capabilities of all systems within Model Based Product Support.		mission requirement [Element] functionalRequirement [Class]
11	12.2.2 Communication Pathway with OEM Systems	The system shall be capable of storing data incoming from original equipment manufacturer's to be integrated into a program's system model.		mission requirement [Element] functionalRequirement [Class]
12	12.2.3 Communication Pathways with AWARE	The system shall be capable of communicating and sharing data with the AWARE to support AWARE's capabilities.		mission requirement [Element] functionalRequirement [Class]
13	12.2.4 Communication Pathway with Identified Users	The system shall provide a graphical user interface where Users can interact with the system to perform all described functionality of the described system.		mission requirement [Element] functionalRequirement [Class]
14	12.2.5 Communication Pathway with DEE Internal System(s)	The system shall be capable of sharing data between all of the system within the Digital Engineering Environment boundary that supports the capability of each system.		mission requirement [Element] functionalRequirement [Class]
15	12.2.6 SE Reports and Documentation	The system shall be capable of developing dynamic reports that can import artifacts from a system model to be included in a template of the modeler's choosing.		mission requirement [Element] functionalRequirement [Class]
16	N1 Project Visionary Mission Requirements	We need model-based engineering capability and tools to integrate with the product support capabilities of the Enterprise Technical Reference Framework.		Requirement [Class] EnterpriseNeed [Class]
19	N2 Authoritative Source of Data	We need an authoritative data source that will contain the necessary product data to perform SE capabilities.		Requirement [Class] StakeholderNeed [Class]
21	N3 Integrated Environment	We need an integrated environment to support capabilities on both sides of the SE Vee.		Requirement [Class] StakeholderNeed [Class]
23	N4 Automation of Manual Intensive Tasks	We need a system that will indicate anomalies or misalignments in SE data sets.		Requirement [Class] StakeholderNeed [Class]
25	N5 OEM Database Relationship	We need a system that can use OEM developed models and will not exhibit any proprietary concerns.		Requirement [Class] StakeholderNeed [Class]
26	N6 Data Sharing/Reuseability	We need an integrated environment that will communicate data back and forth, to and from Model-based Product Support to maximize reuseability.		Requirement [Class] StakeholderNeed [Class]
27	N7 Digital Engineering Strategy Requirements			Requirement [Class]
28	N7.1 Strategic Goal 1	Formalize the Development, Integration, and Use of Model to Inform Enterprise and Program Decision Making.		Requirement [Class] StrategicObj [Class]
34	N7.2 Strategic Goal 2	Provide an Enduring, Authoritative Source of Truth.		Requirement [Class] StrategicObj [Class]
40	N7.3 Strategic Goal 3	Incorporate Technical Innovation to Improve the Engineering Practice		Requirement [Class] StrategicObj [Class]
46	N7.4 Strategic Goal 4	Establish a Support Infrastructure and Environments to Perform Activities, Collaborate, and Communicate Across Stakeholders		Requirement [Class] StrategicObj [Class]
51	N7.5 Strategic Goal 5	Transform the Culture and Workforce to Adopt and Support Digital Engineering Across the Lifecycle		Requirement [Class] StrategicObj [Class]

Figure 30. System Specification: Function Requirements.

F. MODEL SUMMARY

Three process diagrams were reviewed each following actions are performed sequentially which result in having documents/artifacts created that provide the necessary information to address an incoming capability. At the conclusion of these processes, the problem has been defined and analyzed, mission requirements are generated, and functional requirements are developed. All the artifacts provide a concrete strategy of what is needed to provide the command a strategy to address an incoming capability or what is known as scenario one. The stakeholders will be able to use these artifacts to clearly define

a solution that details the necessary actions/steps to prepare the command for integrating a new capability.

Within each process are additional artifacts that help further document system architecture, expected behavior, parametric diagrams for analyzing the solution, and identifying interfaces between existing systems and incoming external systems. Together the models fully define the problem and an associate solution to that problem. After this point, the command will be able to start implementing the identified solution.

G. CHAPTER SUMMARY

This chapter presented three process diagrams that describe the necessary actions to produce the required artifacts for developing the conceptual system model. The processes were explained through expanded diagrams and step by step instructions of walking through each process. Input and output artifacts were developed using a simulation scenario and summarized with provided descriptions that relate their usage within the diagram. After completing all three processes the sponsoring command should have a clear understanding of the problem and a strategy ready for implementation to address that problem. As stated earlier, this project will not result in the creation of a physical system but will provide all information to allows for the creation of the solution.

PAGE INTENTIONALLY LEFT BLANK

V. MODEL FINDINGS

A. DATA EXCHANGES BETWEEN DOMAINS

Findings on the Model Based Product Support program's capabilities shows that the program is implementing the AeroSpace and Defense Industries Association of Europe and Aerospace Industries Association (ASD/AIA) S-Series standards to regulate the data necessary for their suite of capability. Shown in Figure 31, the logistics support analysis (LSA) data structure is the standard database and supports the other specifications (Aerospace and Defense Industries Association of Europe and Aerospace Industries Association (ASD/AIA) 2018).

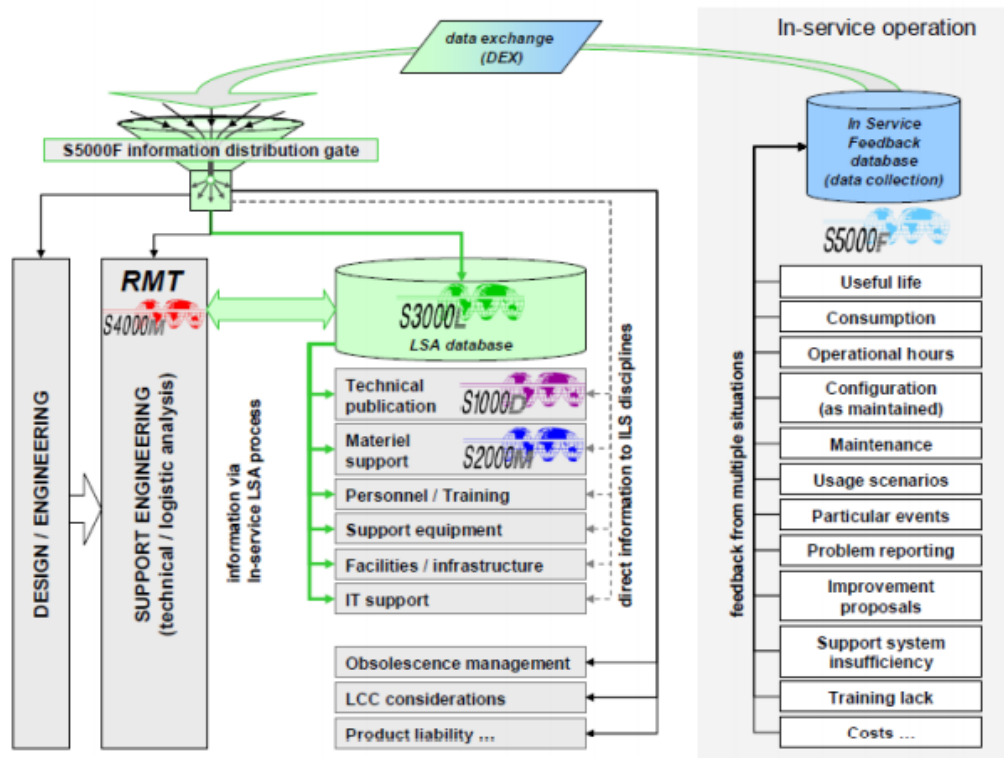


Figure 31. In-Service Data Analysis Process as an Example of S3000L Feedback.
Source: ASD/AIA (2018).

The S3000L LSA database is built over the lifecycle of the developed product and its development is supported by the import of engineering technical data. Figure 32 shows how the development of the database consumes and produces data for the development of the physical product. The LSA database is structured according to the S3000L extensible markup language (XML) schema presented in the standard. Therefore, any data exchanges between the database shall be supported by XML. Currently, some SysML tools support the importing and exporting of XML, but during the conversion some data, like SysML stereotypes, are lost or converted to its Unified Modeling Language (UML) equivalent (No Magic, Inc n.d.). For example, shown in Figure 33, user capacity is stereotyped as a measure of effectiveness (MOE) within SysML. When converted to XML, the type is changed to a UML property of the Digital Engineering Environment (DEE) class, shown in Figure 34.

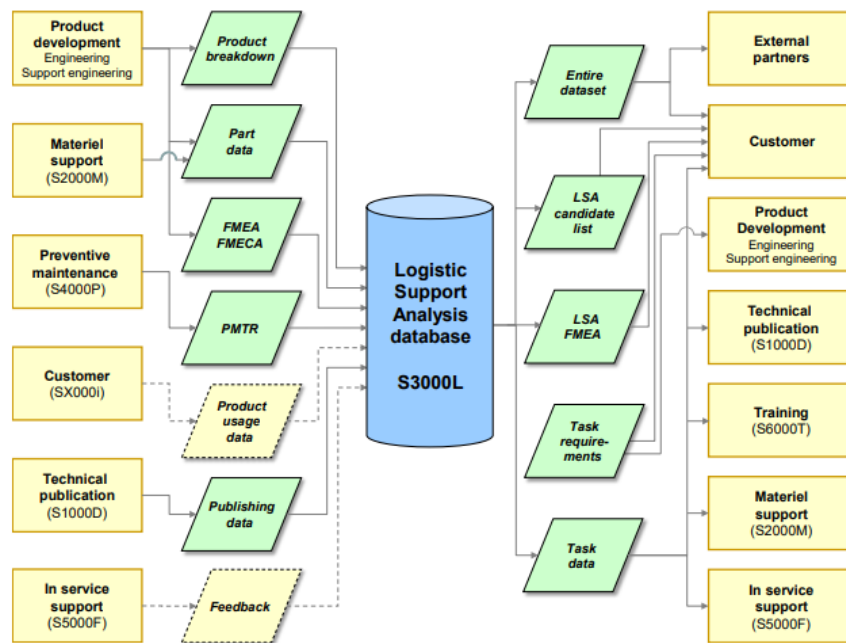


Figure 32. S3000L Data Exchanges. Source: ASD/AIA (2018).

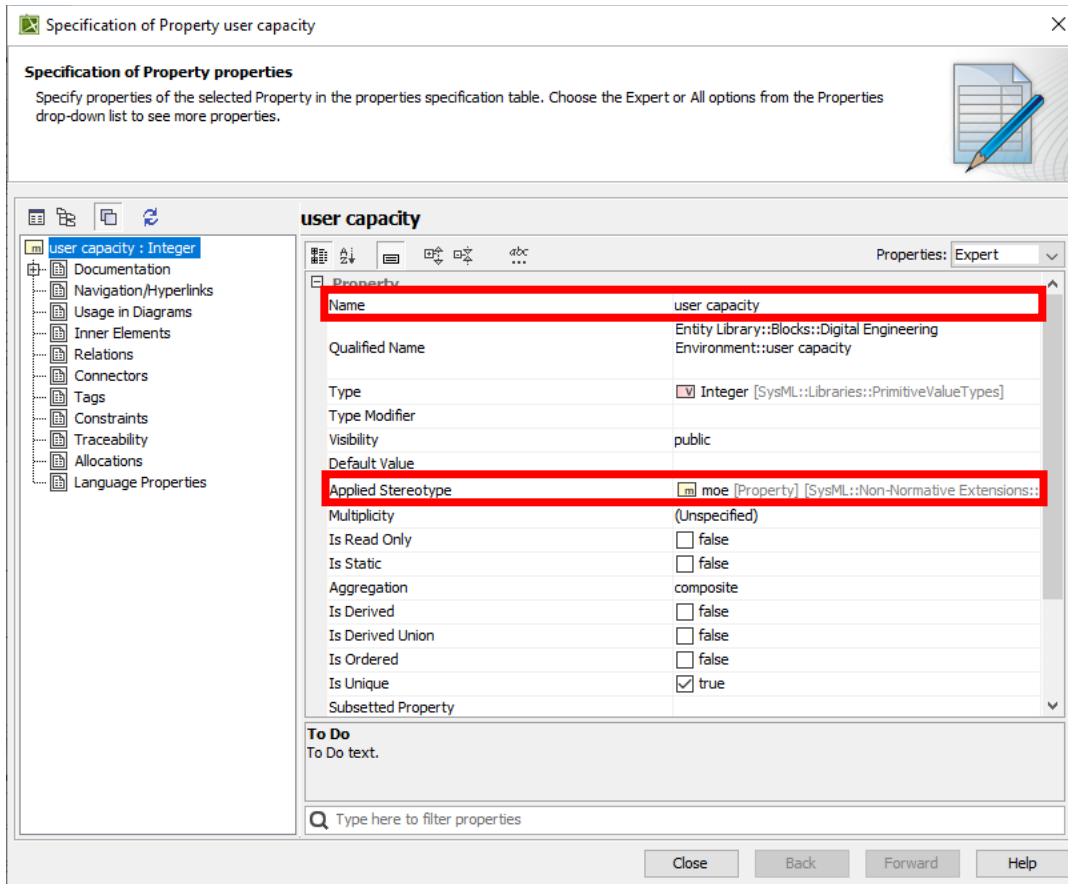


Figure 33. User Capacity MOE Specification in SysML.

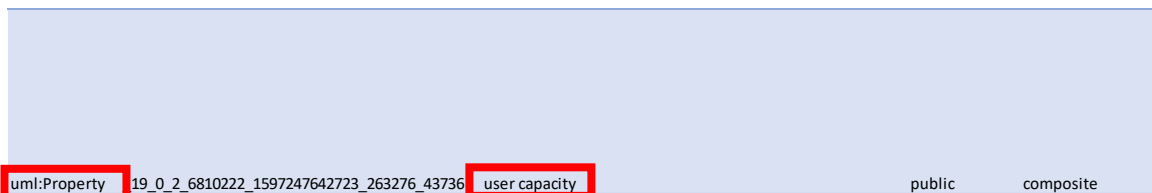


Figure 34. XML Data Table of User Capacity.

The S-Series specifications developed an importable XML file that contains the S-Series data model as UML classes. Instance elements, as shown in Figure 35, can be developed within a system model to create supporting data elements. Current XML exporting features only allow for a total model export. Due to this limitation, an isolated model containing the instances would be needed to ensure only required data is exchanged between systems. The creating of instances is currently a full manual process, which creates

a lot of work if the system model is developed using processes that utilizes SysML and tool or process-specific stereotypes. The mapping of SysML-specific data types to the S-Series UML data model could support the creation of a translator that would drastically cut down the conversion time. Further work and research are needed to develop a data map that is able to automatically convert data from a SysML system model into elements capable of being consumed and useful within the PS domain.

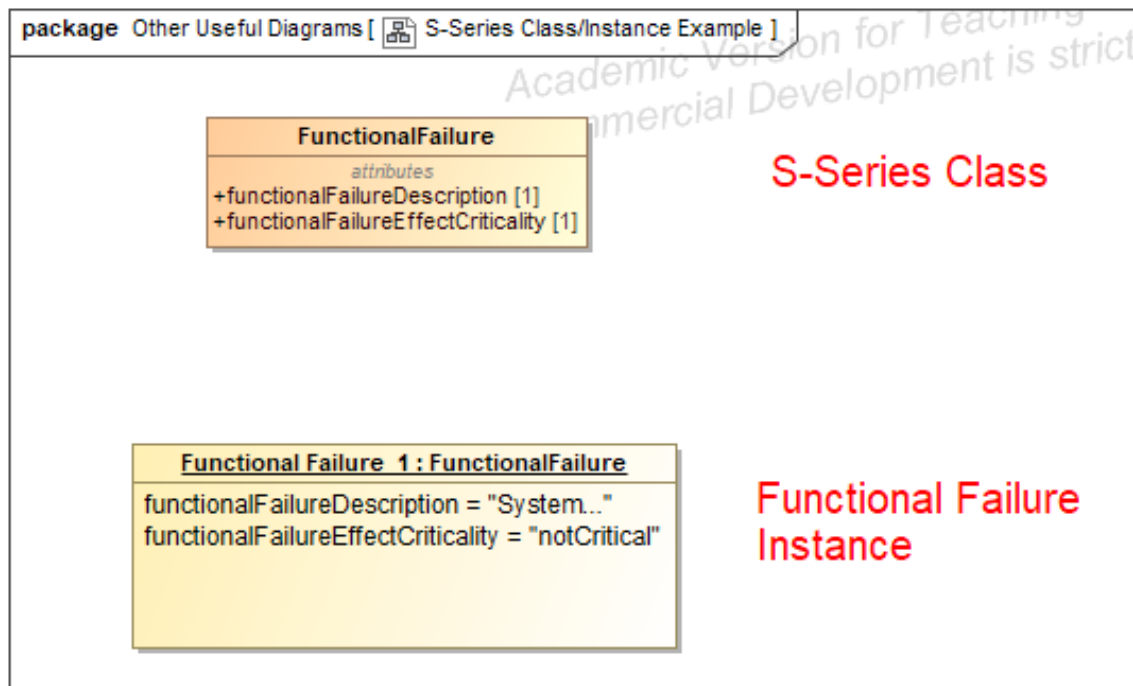


Figure 35. Example of Producing a Class Instance in a System Model with SysML.

B. USEFUL ENGINEERING ARTIFACT CREATION

1. Artifacts Supporting Logistics Support Analysis (LSA) & S3000L

The LSA database interacts with the engineering community to gather engineering technical data to support the definition of the LSA database and performance of the system LSA (Aerospace and Defense Industries Association of Europe and Aerospace Industries Association (ASD/AIA) 2014). Shown in Figure 36, the engineering data set supports the

performance of different reliability, availability, maintainability, and safety (RAM-S) analysis and reports. The data set is also stored in the database for future analysis iterations.

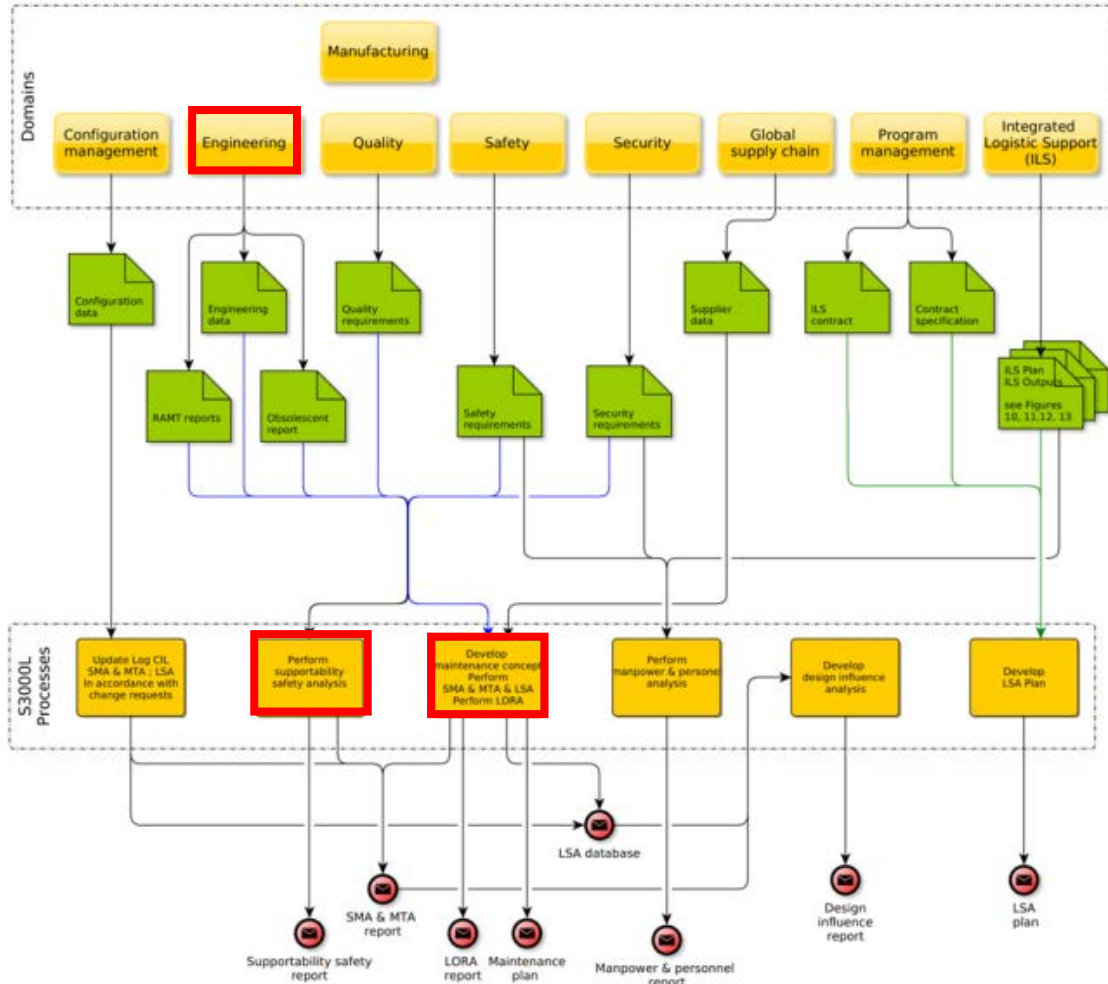


Figure 36. The Uses of Different Domain Data Sets for S3000L Processes. Source: ASD/AIA (2014).

A program’s system model is not going to contain the entirety of the required data sets. However, the data can be useful early in a program’s lifecycle, when engineering drawings or three-dimensional models do not yet exist. For example, early level-of-repair analyzes are derived from the supportability failure modes and effects analysis (FMEA), which is derived from engineering inputs as shown in Figure 37 (ASD/AIA 2014). When done correctly, a system model can be configured to output the elements required for these

inputs, as shown in Figure 38 and Figure 39. Iterated over all the identified failure modes, a full FMEA can be developed in a SysML tool. Similar tables and diagrams can be created for other engineering analysis to be imported into the LSA database from the system modeling tool as discussed in Section 2.

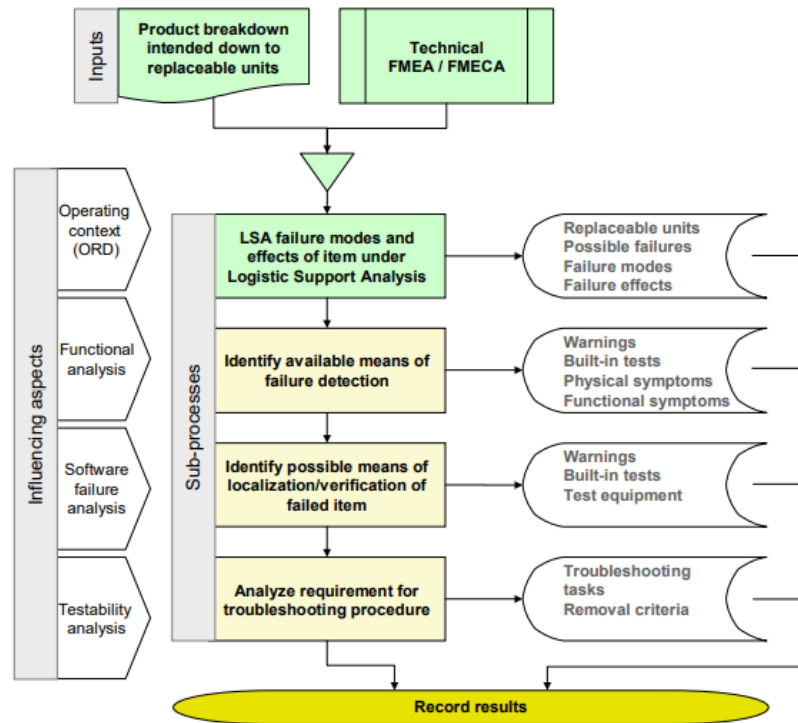


Figure 37. S3000L FMEA Development Process. Source: ASD/AIA (2014).

#	Name	Slot	Client Dependency	Dependency
1	hardwarecomponent_1		Dependency[hardwarecomponent_1 -> Total System Model]	Total System Model Integrator
2	FailureEffect_1	<ul style="list-style-type: none"> ⊗ = TechnicalFailure_1 ⊗ failureModeEffect = "System..." 	Dependency[FailureEffect_1 -> TechnicalFailure_1]	TechnicalFailure_1 : FailureMode
3	TechnicalFailure_1	<ul style="list-style-type: none"> ⊗ = hardwarecomponent_1 ⊗ failureModeDetectionAbilityDescription = "System..." ⊗ failureModeFailureRate = "0.00001" ⊗ failureModeIdentifier = "FFA1" 	Dependency[TechnicalFailure_1 -> hardwarecomponent_1]	hardwarecomponent_1 : FailureAnalysisItem

Figure 38. Example of Functional Design FMEA within the System Modeling Tool.

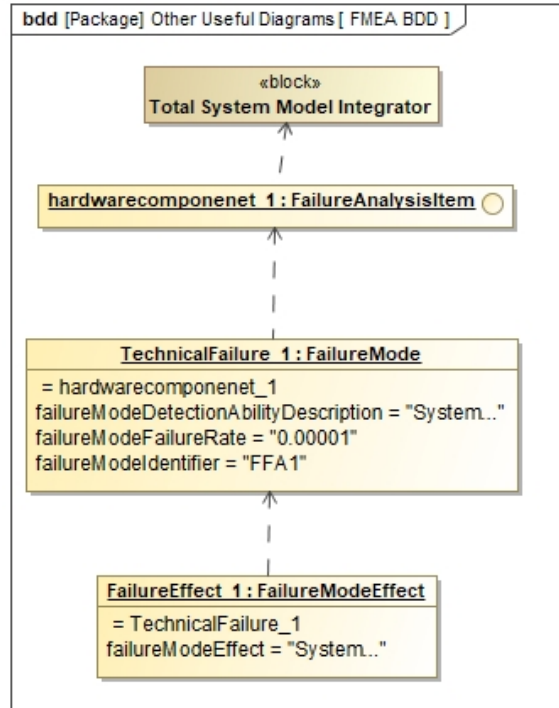


Figure 39. FMEA BDD within System Modeling Tool.

2. Model-based Documentation Generation

Organizations will still require and benefit from creating documents throughout the systems engineering process. A model-based documentation generation process can be utilized to extract model information and integrate it with current documentation templates to be supplemented with text, as shown in Figure 40. Currently, SysML tools allows for the automatic generation of reports based on an uploaded template. Once the template (*.docx file) has been configured with the correct dynamic code identifying where to find the correct model information, the user can generate reports based on that template. Shown in Figure 40, the capstone team developed a model-based document from a concept of operations (CONOPS) template using the velocity template language (VTL) to constrain which information is to be presented (Department of Veteran Affairs n.d.). Using the stakeholder viewpoints developed early in the system model process, the modeler can present important stakeholder information in ways that is familiar and understood by the stakeholder without the need of understanding how to use and navigate through a new tool. For a command wanting to implement model-based systems engineering (MBSE), it is

recommended to build a library of VTL configured documents that enable the production of model-based documentation. To accomplish this, it is also recommended that a standard modeling format or a modeling style-guide be developed to enable the reuse of the model-based documents.

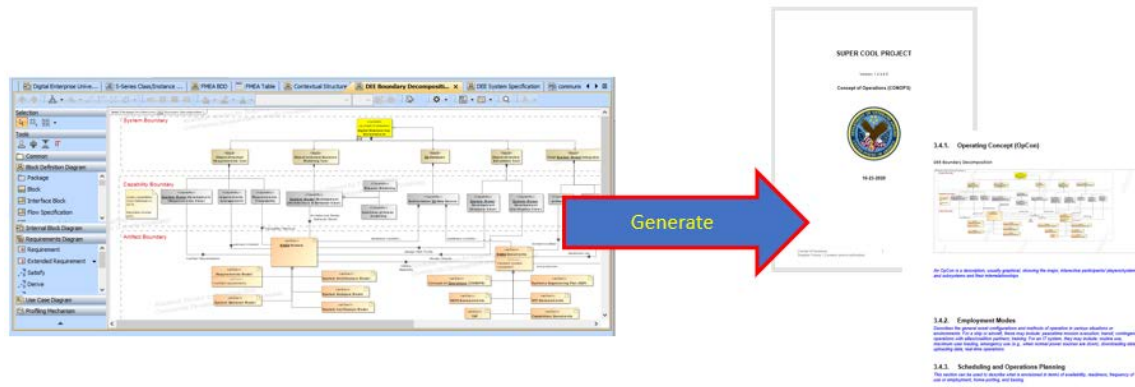


Figure 40. High-Level Concept of Generating Model-Based Documents from the System Model.

C. CHAPTER SUMMARY

This chapter discussed pertinent findings related to the interactions between the model-based systems engineering framework and exterior environments. Utilization of the XML schema, as defined by the S-series specification, allows for an MBSE elements to be exported for use in alternate applications. Three instances of export use were discussed, beginning with the prospect of a direct interface between the model-based product support and digital engineering domains that can be structured to facilitate express data exchange. Second, the export of data and information from model-based systems engineering diagrams can be translated to a structure of artifacts that support S3000L LSA database entries. The creation of and/or modification to data elements would be enabled by XML data transfers. Lastly, the MBSE framework can be coupled with document templates to construct documentation utilized by traditional SE methods, such as the development of a CONOPS document using a predefined template.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. A SUMMARY OF PROJECT OBJECTIVES

This capstone object was to develop a formal process using systems engineering (SE) methodologies to develop a conceptual system model and compile a report that explaining our development efforts, findings and conclusions from simulations and research, and recommendations for future work. This chapter summarizes the major findings that support the project objectives. It also includes insights that emerged and recommendations for future work.

B. DEFINING, DEVELOPING, AND IMPORTANCE OF THE SYSTEM MODEL

The process proposed utilizes a tailored approached based on the object-oriented system engineering methodology (OOSEM) and the systems modeling language (SysML) to capture system modeling data into a system model. A proposed conceptual system data model is shown in Figure 41. The center of the system model is the system of interest (SOI). The SOI of the system model acts as a piece of the digital twin, containing the architecture, behavior, requirements, and verification and validation models.

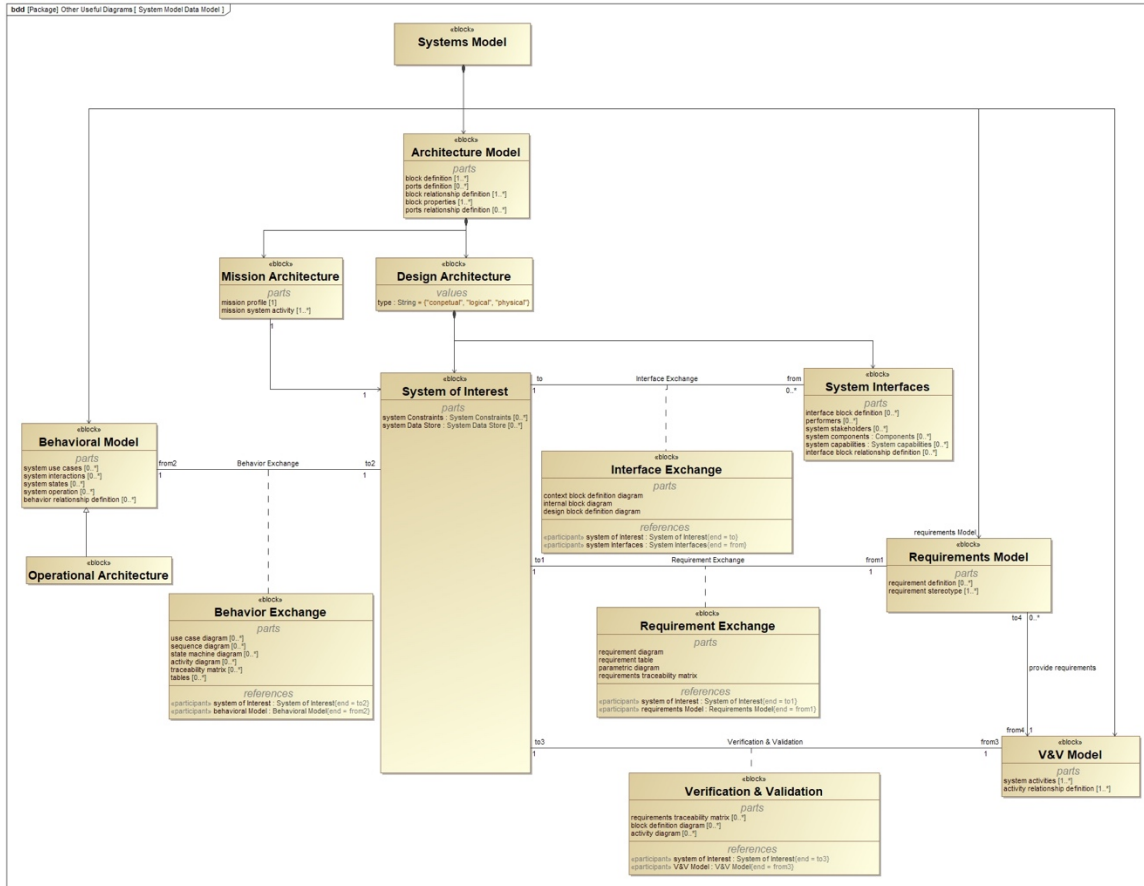


Figure 41. Proposed Conceptual System Data Model.

Over time, it would be expected that the attributes within the data model would remain constant but the level of detail of the presented information would change. For example, shown in Figure 42, activities of systems and subsystems are created at the conceptual level. Once more information about the desired capabilities of the SOI are known, the modeler can provide a logical definition to how the conceptual behavior is performed. In the selected scenario, the capability of one of the subsystems is the ability to communicate data developed within the environment to external databases. From the modeler’s understanding of the current conceptual system architecture, contextual system of systems (SoS) architecture and public information of system-to-system data exchanges a logical definition allocated to the system architecture can be formed. It would be up to the development team to further define these interactions at the physical level once the physical architecture is defined. As to the example, this would include the addition of

computer coding that demonstrates how each interaction is performed. A block containing the coding information within SysML would be allocated to the signals displayed on the logical sequence diagram shown in Figure 42. Some SysML tools can auto generate a model from code developed outside of the tool, where inner model elements can then be related to different elements within the developed system model (Dassault Systems n.d.).

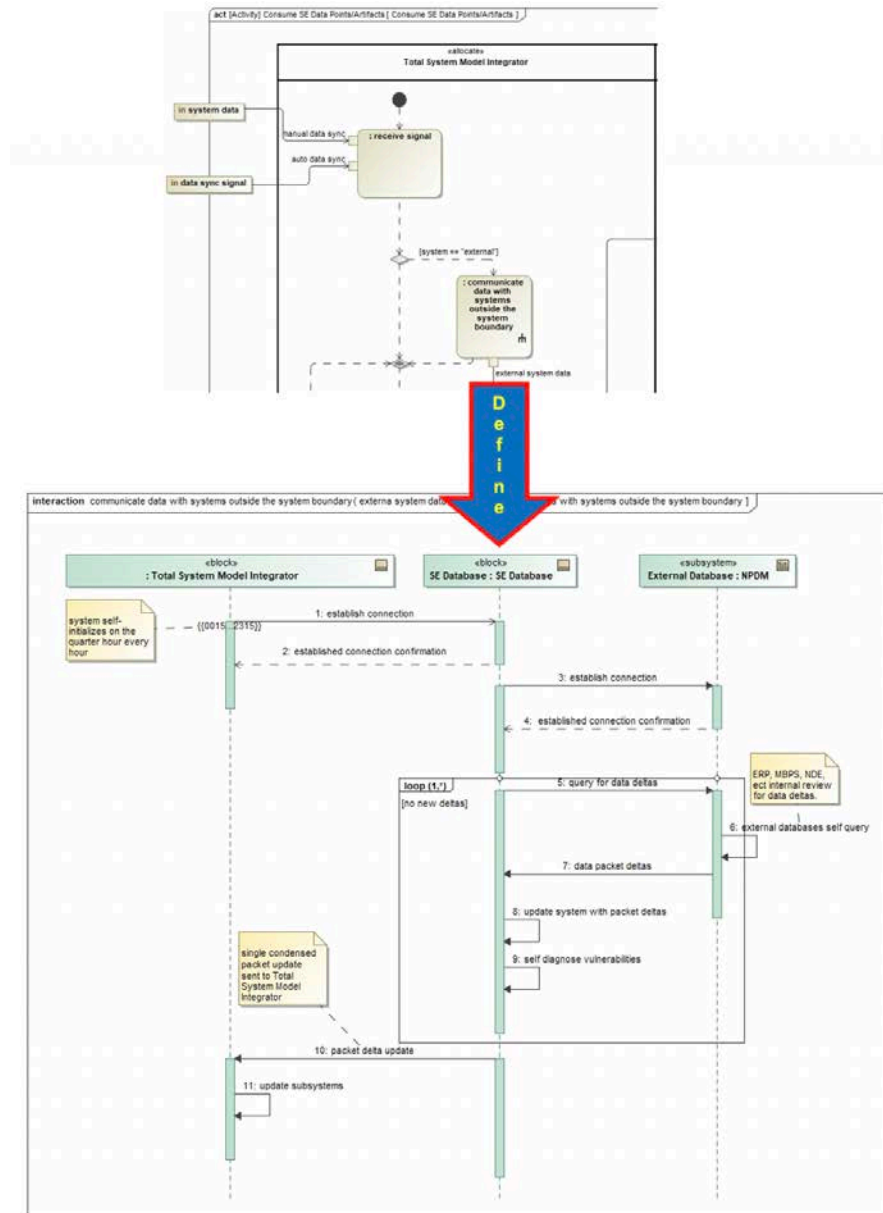


Figure 42. Transformation of a Conceptual Action into a Logical Sequence of Signals.

Establishing relationships and traceability between elements within the systems model during design and development is critical for the reusability of the systems model throughout the rest of the system's lifecycle (Friedenthal, Moore, and Steiner, 2012, 349–352). System models developed using SysML can be used throughout system sustainment to support different changes to the system, including changes to design, mission, and maintenance procedures. For this to occur, a strong interoperability with the information technology (IT) systems in the PS domain is required.

Data captured within the system model has the capability of being transformed into a presentation graphic that could be shown to stakeholders to display the data in a way that is understandable. This capability of presenting model-critical data to decision makers is critical to ensure the design meets expectations (DoD 2018). Generating documents using models does not necessarily mean that the developed templates used within an organization are useless. Demonstrated in this capstone, SysML tools can utilize an organization's templates, as built, configure it to enable the document to collect model presentation artifacts, and embed them with the specified document area. Further developed could lead to auto generation of required programmatic documentation from the system model.

C. CONCLUSIONS

As systems are becoming more complex and more constrained, processes are going to have to become more streamlined. The MBSE stakeholders at the Naval Surface Warfare Center Port Hueneme Division (NSWC PHD) assigned the capstone group with the objectives to provide methods that would bring MBSE concepts to the command. From early research it was determined that MBSE is early in its conceptualization with few processes being implemented across the Naval enterprise. The capstone provided a proposed workflow that was designed to be independent of a single system modeling tool and capable of developing a conceptual system model and a partial logical system model. The capstone used SysML to capture and present the modeling data, but the verbiage inside the workflow was presented in a way that another modeling language (UML, LML, etc.) could be used.

The stakeholders at NSW PHD were also interested in learning about how an MBSE environment would integrate with another currently occurring digital transformation, the logistics IT (LOG IT) transformation. Data sharing is a major concern and an objective of the implementation of MBSE. With the current toolset and understanding of the systems within the LOG IT, out of the box data configurations would need to be translated in a suitable format in order to be usefully communicated across the domain. The MBPS program has established that their program would be setting up an LSA database based on the S3000L specification and an XML schema. Current importing and exporting capabilities in SysML limit the amount of data that can be converted and will convert all unmapped sources of data to its UML equivalent. The loss of data is not satisfactory, but information and artifacts useful to other domains could be created using instance elements within SysML and the UML classes that were developed by the S-Series specification authors. The data needed to be communicated can be exported to an isolated model, converted into an XML file, and consumed by the external MBPS system to develop analysis artifacts within its system.

Model generated documents can be utilized by programs to develop programmatic documentation from their model. A template of a CONOPS was discovered by the capstone team through the public domain, configured using VTL, and uploaded to the selected SysML tool to generate a report with the developed system model artifacts from the process simulation. Any template can be configured and uploaded if it is a supported format and could be a very useful tool to present system model data to different stakeholders.

D. RECOMMENDATIONS

It has been identified that the artifacts and findings developed from this capstone are not as mature as they could be. The developed process had completed a single simulation developed for this capstone to present potential outputs, but more research and implementation is needed to verify and validate this existing process. The process's implementation in pilot programs can help identify any unaccounted-for gaps and allowing for updates.

The process also does not consider the data developed during more detailed design efforts, including a majority of the logical and the physical architecture. The introduction of computer aided drawings (CAD), computer aided manufacturing (CAM), computer aided software engineering (CASE), finite element analysis (FEA), and other computer aided simulation artifacts could help support further definition of the system model but further research on this implementation is needed.

The conversion of instances supported in SysML to XML were mostly manual and since the XML data format is in place to be the format of choice for existing systems, it would be of interest to look for ways to automate the data conversion and transmission. The process outlined in this capstone for conversion can support this automated process. Development of a standard system data model completed with data mappings to the S3000L XML data structure is the logical next step to automating the process. It is theorized then plug-in software or middleware could be developed that supports and automates XML conversion.

With the increase in interest of studying system of systems (SoS), SoS engineering, and SoS modeling, researching the effect of SoS concepts have on the development of a system model could be of interest to many stakeholders. Capability gaps could be produced from emerging capabilities within the SoS, signaling a need for a solution and the start to the capstone's developed process. This fact was not considered, but its effect and further iterations of the process should include research into how the implementation of SoS modeling could affect the process. The system model development process did consider that building a new system is not always the best choice and some solutions require updates or refreshes to existing systems, but the process is currently incomplete and lacks simulation results. Further development of the process to include system changes and refreshes is recommended for future project work

APPENDIX. LITERATURE REVIEW MATRIX

	Topic 1	Topic 2	Topic 3	Topic 4	Key words searched			
	The design of existing support and logistic systems	current initiatives by the logistics digital transformation stakeholders include Naval Systems Engineering Directorate (NAVSEA 05), Acquisition and Commonality (NAVSEA 06), Logistics Maintenance and Industrial Operations (NAVSEA 04), Naval Information Warfare Systems Command (NAVWAR), Office of the Chief of Naval Operations (OPNAV), and Naval Air System Command (NAVAIR)	the use of MBSE in government and military as well as in private industry.	MBSE to MBPS Integration	MBSE, MBPS, Logistics, Digital Twin, Maintenance, Training, VR, AR, DoD, US Navy, Architecture Framework, Viewpoint, Models, Data as a Service			
	Type of Literature (Thesis, US Government, Other Government, Non Government Industry)	Title	Date	Where was it (journal, conference proceedings (committee approved), conference proceedings(no screening), book, etc)	Scholarly Level Journal (Peer Review) Book Conference Proceedings (committee approved) conference proceedings (no screening), Article	Topic Area (1, 2, 3, 4, other))	How it applies to capstone (modeling methodology, technical approach, MBSE applications, MBPS applications, others)	How it applies to capstone report ---->
x	Non Government	Integrating a Model-Based Systems Engineering and Model-Based Product Support Approach for Affordable System Sustainment	7/16/18	INCOSE Conference	Conference Proceedings (screened)	Topic Area 4	MBSE	Integration
x	Other	Operationalizing Model Based Systems Engineering through the Application of Digital Twins	4/1/17	Journal	Journal (Peer Review)	Topic Area 3 and 4	MBSE and the use of Digital Twins	Problem Analysis + System Design
	Industry	Survey of Model-Based Systems Engineering (MBSE)Methodologies	5/23/08	Report	in response to n INCOSE MBSE initiative from INCOSE (see initiative below)	Topic 3	Summary of MBSE tools	Problem Analysis
	Industry	INCOSE Model Based Systems Engineering (MBSE) Initiative	June 2019	INCOSE	Presentation	Topic 3	Background Info on MBSE	
x	US Government	NAVSEA 06L Navy Model-Based Product Support (MBPS) Project Technical Supplement	1/17/19	NAVSEA 06L	Policy	Topic 2	MBSE Initiatives/Policy	Stakeholder Analysis
	Other	Application of Model-Based Systems Engineering Concepts to Support Mission Engineering	September 2019	NPS	Scholarly (Peer Reviewed)	Topic 3	MBSE Applications	Problem Analysis
x	US Government	ENTERPRISE TECHNICAL REFERENCE FRAMEWORK ETRF WHITE PAPER	Uncertain (2018? 2019? Ask Will)	CNO, OPNAV	Policy	Topic 2	MBPS Product	Problem Analysis

	US Government	MBSE-driven Visualization of Requirements Allocation and Traceability	March 2016	NASA Report	IEEE Aerospace Conference	Topic 3	MBSE/ Technical Approach	Requirements + Value System
X	US Government	DoD Digital Modernization Strategy	7/12/19	DoD Strategy	Policy	Topic 2	Digital IT transformation	Operational Architecture + Needs Analysis
x	US Government	TECHNICAL SPECIFICATION TITLE: ALTERATIONS TO SHIPS ACCOMPLISHED BY ALTERATION INSTALLATION TEAMS NO.: TS9090-310D	2/1/04	NAVSEA TechSpec	policy	topic 2		
x	Industry	VIRTUAL REALITY, AUGMENTED REALITY, AND PLAIN OLD REALITY FOR MAINTENANCE TRAINING	october 2018	Article FAA	articles	topic 1	MBPS capabilities	Problem Analysis
X	Industry	Systems Engineering Vision 2020 (INCOSE-TP-2004-004-02)	september 2007	Report	INCOSE Technical Product	Topic 3	MBSE	
x	US Government	Navy Deploys Augmented and Virtual Reality for Many Tasks	6/26/17	article Navy	San Diego Business Journal ;	Topics 3	MBPS Capabilities	Problem Analysis
X	Industry	The Ongoing Adoption of Model Based Systems Engineering	2013	INCOSE Survey	INCOSE Survey	Topic 3	MBSE	Understanding of Industry use and different types of modeling languages
	us Government	Model Based Product Support (MBPS) Overview	7/1/19	Presentation NAVSEA06L	Presentation	Topic 2	MBPS Policy and capabilities	problem analysis

x	US Government	MIL-STD-31000 DEPARTMENT OF DEFENSE STANDARD PRACTICE TECHNICAL DATA PACKAGES	11/5/09	DOD MIL STD	Standard Practice	topic 1	Available data for MBSE to integrate with	Problem analysis Functional Architecture.
X	Industry	Guide to the Systems Engineering Body of Knowledge (SEBoK), version 2.2	10/31/19	INCOSE + IEEE	Industry best practices/ guidance	Topic 3	Systems Engineering Guidebook	Functional Architecture, AOA, System design
X	Industry	Unified Profile for DoDAF and MODAF (UPDM) version 2.1 (formal/2013-08-04)	8/1/13	OMG (Object Management Group)	OMG specification	Topic 3	DoDAF/UPDM architecture framework	physical architecture
	US Government	NAVSEA Instruction 4130.12B Configuration Management (CM) Policy and Guidance	7/21/04	NAVSEAINST 4130.12	NAVSEA Guidance	Topic 1	Current System Guidance	needs analysis
X	Other	Exploring the use of Model-Based Systems Engineering (MBSE) to develop Systems Architectures in Naval Ship Design	5/7/10	NPS	Thesis	Topic 3	MBSE for US Navy Warship Design	system design
	US Government	NAVSEA Instruction 4160.3B Technical Manual Management Program	6/21/11	NAVSEAINST 4160.3b	NAVSEA Guidance	topic 1	current system guidance	needs analysis
	US Government	MIL-DTL-24784D(NAVY) Technical Manuals: General Acquisition and development Requirements	11/3/07	MIL-DTL-24784D(NAVY)	NAVSEA Guidance	topic 1	Current System Guidance	needs analysis
	US Government	A Web-Based Architecture for Interactive Electronic Technical Manuals (IETMs)	1/1/06	report	article	topic 1	current system guidance	needs analysis
	US Government	NSWC-Corona Process Evaluation - Materials Readiness Database (MRDB)	2/1/12	report	articles	topic 1	Current System Guidance	
x	US government	CHIPS Articles: Modernizing the Fleet: Revolutionary Approach Needed	2017	report	article	topic 2	MBPS vision	problem analysis
x	industry	What is Data-as-a-Service (DaaS)? Talend	9/30/19	DaaS provider Talend	article	topic 4	MBPS applications	operational architecture
X	US Government	DoD Digital Engineering Strategy	6/1/18	DoD Strategy	Report	Topic 3	US DOD Digital Engineering guidance	understanding US DoD vision and mission with regards to Digital Engineering

1. Research Questions

The following research questions form the foundation of the literature review and were used to define the scope of the project.

Question 1: What are the current logistics support strategies, methodologies, tools, infrastructures, and processes within the Navy? (NAVSEA 06L, 05, NAVWAR PMW 150, and OPNAV N96 and N41)

Question 2: What are the current capabilities of MBSE and where and how is it being applied? (industry, government, research, and simulations)

Question 3: What tools currently exist to support a model-based product support strategy and what are its applications?

2. Types of Literature Reviewed

The types of literature reviewed include journals both peer-reviewed and non-peer reviewed, scholarly reports, conference proceedings and presentations, Navy policies and related projects, INCOSE standards, DoD standards, and articles. For a complete list of all references, see the literature review matrix.

3. Topics of Review

This literature review focuses on the U.S. Navy's digital IT transition. The items that will be investigated include: Currently used legacy product support operations and the transition to utilize MBSE and model-based logistics IT, including MBPS products. With the U.S. Navy's initial operating capability for MBPS in fiscal year (FY) 2021, the use of integrating MBSE with MPBS will be investigated. The three (3) topics of this review include: 1) Current U.S. Navy product support methods and tools, 2) the use of MBSE for military and private sector applications, and 3) the U.S. Navy's transition to a MBPS capability.

Other specialized MBSE modeling languages include EAST-ADL (Architecture Description Language), designed for use in the automotive electrical and electronic architecture sectors, Arcadia, used for general systems, and Modelica, an open source and open access standards-based product. Other organizations have created their own "home-grown" MBSE modeling tools with their own views to meet their particular needs.

In 2014, the Object Management Group (OMG), owners of the SysML architecture framework, put out a survey to better understand which sectors of industry use MBSE and

which language. The survey asked, “To what extent are the following modeling languages used for system architecture modeling as part of your MBSE effort?” Their findings showed that SysML, UML, and Simulink were ranked top 3, respectively. The survey allowed for a response on a scale of 1 (for never) to 5 (for almost always). SysML averaged at 3.69, UML averaged 3.16, and Simulink averaged 2.93 (Cloutier and Bone 2015).

As part of the 2014 OMG survey we see that six (6) primary sectors are using some sort of MBSE framework. Those sectors were led by defense industry at 49.5% of respondents using MBSE, followed by aircraft companies at 28.6% usage, space systems industry at 27.1% usage, automotive industry with 17.7% usage, the IT sector with 15.6% usage, and lastly the medical industry with 9.4% usage. Responses from industry partners that did not fit into any of the six (6) categories were categorized as other, and 25% of those respondents utilize some sort of MBSE product (Cloutier and Bone 2015).

The DoD has its own MBSE business enterprise modeling language called DoD Architecture Framework (DoDAF) that was designed to meet the specific business and operational needs of the DoD for MBSE use. The DoDAF systems view language resides within the Unified Profile for DoDAF and MODAF (UPDM) and was designed with two compliance levels. Level 0 DoDAF supports a UML-based profile, and the Level 1 DoDAF supports a UML and SysML profile. DoDAF was set to add capabilities unique to DoD needs, such as classification levels and information security markings for the artifacts (Object Management Group 2013). All DoDAF artifacts, called viewpoints within the framework, fall into one of eight (8) categories. Those eight (8) viewpoints include the following (Object Management Group [OMG] 2013):

- All Viewpoint
- Capability Viewpoint
- Data and Information Viewpoint
- Operational Viewpoint
- Project Viewpoint

- Services Viewpoint
- Standards Viewpoint
- Systems Viewpoint

Model-based systems engineering is a subset of digital engineering that utilizes a host of models and viewpoints to assist with the systems engineering processes. The purpose of the MBSE approach is to have an all-encompassing spectrum of models that represent all aspects of systems. The MBSE languages define how the information within the models is presented, and while all languages present that information in slightly different ways, the application is very similar between the languages. MBSE allows for the generation and management of system requirements, architecture, design, analysis, verification and validation of a system or set of systems, from the conceptual design through the development and system support phases (Cloutier and Hutchison 2019). MBSE shifts away from the traditional systems engineering approach, which emphasizes producing and leveraging control documentation to define a system. Instead, MBSE utilizes a multi-layered visual model approach to system conceptualization, starting at the highest level of system design, that provides greater system definition in lower layers (Cloutier and Hutchison 2019).

There are challenges that exist with the use of MBSE. The first challenge is that MBSE is a relatively new concept that has not fully matured yet. The concept of MBSE has existed since 1997, however many organizations did not adopt the use of MBSE until 2007 with the introduction of MBSE via INCOSE's Systems Engineering Vision 2020 report (INCOSE 2007). Another issue with the shift to MBSE is the resistance to change within the workforce. The use of MBSE is not yet widespread, and requires users and organizations to invest in methods, tools, and training, and a greater commitment to deploy this capability to their programs (Cloutier and Hutchison 2019).

The MBSE process uses a host of different models and views, often referred to as artifacts, that are used for analyzing, designing, and verifying complex systems that may include hardware, software, information, personnel, procedures, and facilities. Those

artifacts can be depicted in different ways depending on which MBSE modeling language is utilized. Per the INCOSE and IEEE Systems Engineering Book of Knowledge (SEBoK), the four (4) primary modeling languages used in industry today include: (1) Integration Definition (IDEF), (2) SysML, (3) Unified Modeling Language (UML), and (4) Simulink (Cloutier and Hutchison 2019).

Model-based systems engineering provides a great improvement to the document-based systems engineering process, specifically providing enhanced communication, requirements traceability, and improved decision making. The feasibility of using MBSE for U.S. Navy systems exists all the way up to designing an entire U.S. Navy warship (Tepper 2010). Furthermore, the nature of MBSE's digital footprint allows for very complex systems, such as those used by U.S. Navy and other military service branches, to thrive. One example of added capability is that changing a specification or design element within the MBSE framework allows for seamless change within all the models, as the MBSE program will update all models from top to bottom. On the contrary, making a high-level change while utilizing traditional systems engineering methods would require a great deal of work to ensure all instances have been updated.

LIST OF REFERENCES

- Accenture. 2019. "Enterprise Technical Reference Framework." Unpublished Whitepaper, April 7, 2020.
- AeroSpace and Defense Industries Association of Europe and Aerospace Industries Association. International guide for the use of the S-Series Integrated Logistics Support (ILS) specifications. Issue No. 1.2. July 2018. http://www.sx000i.org/docs/SX000i_Issue_1_2.pdf.
- . International procedure specification for Logistics Support Analysis LSA. Issue No. 1.1. July 2014. <http://www.s3000l.org/docs/S3000L-Issue%201.1.pdf>.
- Bickford, Jason, Douglas Van Bossuyt, Paul Beery, and Anthony Pollman et al. 2020. Operationalizing Model Based Systems Engineering through the Application of Digital Twins. <https://cle.nps.edu/access/content/attachment/capstone1923s/Messages/26a015c-07b7-4461-9adf-6e7609a0f8d3/Bickford%20SE%20Journal%20Digital%20Twinn.pdf>
- Bill Johnson. 2018. "Virtual Reality, Augmented Reality, and Plain Old Reality for Maintenance Training." *Aircraft Maintenance Technology* 29 (7): 44–47.
- Cloutier, Robert J, and Nicole Hutchison. "Guide to the Systems Engineering Body of Knowledge (SEBoK), version 2.2" October 31, 2019.
- Cloutier, Robert, and Mary Bone. 2015. "The Ongoing Adoption of Model Based Systems Engineering." *IIE Annual Conference Proceedings*, January 2015, 2799–2807. Proquest.
- Dassault Systems. n.d. "Code Engineering." Accessed October 25, 2020. <https://docs.nomagic.com/display/MD190/Code+Engineering>.
- Department of Defense. 2009. *Department of Defense Standard Practice Technical Data Packages. MIL-STD-31000*. Washington, DC. <https://quicksearch.dla.mil/Transient/24FF9F4A893A48CE98318A80DEA55196.pdf>.
- Department of the Navy. 2020. *United States Navy & Marine Corps Digital Systems Engineering Transformation Strategy*. Washington, DC. Office of the Deputy Assistant Secretary of the Navy Research, Development, Test and Evaluation. <https://go.usa.gov/xfQpx>.
- Friedenthal, S, A Moore, and R Steiner. (2012). *A Practical Guide to SysML the Systems Modeling Language*. 2nd ed. San Francisco: Morgan Kaufmann.

- Giachetti, Ron. 2020. SEBok: Guide to the Systems Engineering Body of Knowledge . May 15. Accessed June 12, 2020. https://www.sebokwiki.org/wiki/Digital_Engineering.
- INCOSE Technical Operations. 2007. *Systems Engineering Vision 2020, version 2.03. Report No. INCOSE-TP-2004-004-02.*
- McDaniel, Stacey. 2019. “What Is Data-as-a-Service (DaaS)? | Talend.” Talend Real-Time Open Source Data Integration Software. September 30, 2019. <https://www.talend.com/resources/what-is-data-as-a-service/>.
- MITRE. 2015 “Architectural Frameworks, Models, and Views.” April 10, 2015. <https://www.mitre.org/publications/systems-engineering-guide/se-lifecycle-building-blocks/system-architecture/architectural-frameworks-models-and-views>.
- Mohammadi, Mehdi, Mahdi Elyasi, and Mostafa Mohseni Kiasari. 2014. “Developing a Model for Technological Capability Assessment Case of Automotive Parts Manufacturing in Iran.” *International Journal of Innovation and Technology Management* 11 (2) (Winter): 1-19. 10.1142/S021987701450014X.
- National Shipbuilding Research Program. 2019. “Model Based Product Support (MBPS) Overview.” July 18, 2019. https://www.nsrp.org/wp-content/uploads/2019/08/05-MBPS_Overivew_June-2019-Updated_v5.pdf.
- No Magic, Inc. n.d. “Cameo XSD Import: Adding XML Data Models to your MagicDraw Model Repository.” Accessed October 25, 2020. https://www.nomaci.com/files/brochures/Cameo_XSD_Import_Plugin_brochure.pdf.
- Object Management Group (OMG). 2013. “Unified Profile for DoDAF and MODAF (UPDM) Version 2.1.” UPDM. Object Management Group (OMG), August 2013. <https://www.omg.org/spec/UPDM/2.1/PDF>.
- . 2020 “Unified Architecture Framework (UAF) Domain Metamodel.” April 2020. <https://www.omg.org/spec/UAF/1.1/DMM/PDF>.
- . n.d. “What Is SysML?” What is SysML? | OMG SysML. Accessed April 23, 2020. <https://www.omgsysml.org/what-is-sysml.htm>.
- Page, Kimberly, Steve Kwon, and Kevin Weinstein. 2018. “Integrating a Model-Based Systems Engineering and Model-Based Product Support Approach for Affordable System Sustainment.” *INCOSE International Symposium* 28 (1): 1412-1419. <https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2334-5837.2018.00557.x>
- Rutherford, Heather. 2017. “CHIPS Articles: Modernizing the Fleet: Revolutionary Approach Needed.” October 2017. <https://www.doncio.navy.mil/chips/ArticleDetails.aspx?ID=9788>.

- Sashegye, Mark. 2020. "Model Based Product Support Heralds in New-Age Logistics and Engineering." DVIDS. March 31, 2020. <https://www.dvidshub.net/news/366283/model-based-product-support-heralds-new-age-logistics-and-engineering>.
- Scrum.org. n.d. "What is a Product Owner?" Accessed April 18, 2020. <https://www.scrum.org/resources/what-is-a-product-owner>.
- . n.d. "What is a Scrum Development Team?" Accessed April 18, 2020. <https://www.scrum.org/resources/what-is-a-scrum-development-team>.
- . n.d. "What is a Scrum Master?" Accessed April 18, 2020. <https://www.scrum.org/resources/what-is-a-scrum-master>.
- . n.d. "What is Scrum?" Accessed April 18, 2020. <https://www.scrum.org/resources/what-is-scrum>.
- Strukelj, P., and S. Dolinsek. 2011. "3-Level Modeling of Organization's Technological Capability," *IEEE*, 1–10.
- Tepper, Nadia A. May 2010. "Exploring the Use of Model-Based Systems Engineering (MBSE) to Develop Systems Architectures in Naval Ship Design." Master's thesis, Naval Postgraduate School. <http://hdl.handle.net/10945/24368>
- US DoD, Department of Defense Digital Engineering Strategy. June 2018. <https://fas.org/man/eprint/digeng-2018.pdf>.
- US DoD, DoD Digital Modernization Strategy: DoD Information Resource Management Strategic Plan FY 19-23 § (n.d.), 2019. https://ac.cto.mil/wp-content/uploads/2019/06/2018-Digital-Engineering-Strategy_Approved_PrintVersion.pdf

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California