2008-06

Systems architecting approach to towing and salvage ship recapitalization

Southworth, George T.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

SYSTEMS ARCHITECTING APPROACH TO TOWING AND SALVAGE SHIP RECAPITALIZATION

by

George T. (Judd) Southworth

June 2008

Thesis Advisor: Clifford Whitcomb
Second Reader: Fotis Papoulias

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2008	**3. REPORT TYPE AND DATES COVERED** Master's Thesis
4. TITLE AND SUBTITLE Systems Architecture Approach to Towing and Salvage Ship Recapitalization		**5. FUNDING NUMBERS**
6. AUTHOR(S) George Southworth		
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**

13. ABSTRACT (maximum 200 words)

Four salvage ships and four ocean-going towing ships are maintained and operated by the Military Sealift Command (MSC) for the U.S. Navy. In 2019, the first T-ATF ships will reach the end of their 40-year life expectancy. The program manager for these vessels has a set of top-level performance characteristics that are deemed as desirable requirements for a new ship class, encapsulating both legacy ship class capabilities.

The DoD has shifted defense planning from the specific service requirements generated system (RGS) acquisition to the Joint Capabilities Integration and Development System (JCIDS) approach that focuses more on *how* adversaries fight rather than *whom* they are fighting. This thesis explores how to use systems architecting to incorporate the capabilities derived from strategic guidance into a Department of Defense Architecture Framework (DODAF) product. The design tool, CORE, is used to explain the architecting methodology and produce DODAF v1.5 system models for decision making and acquisition requirement generation.

14. SUBJECT TERMS Architecture, CORE, DODAF, DOD 5000.2, Enterprise Architecture, JCIDS, MSFSC, Salvage, Systems Architecting, Systems Engineering, SUPSALV, T-ATF, T-ARS, Towing			**15. NUMBER OF PAGES** 87
			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

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

SYSTEMS ARCHITECTING APPROACH TO TOWING AND SALVAGE SHIP RECAPITALIZATION

George T. Southworth
Lieutenant, United States Navy
B.S., University of Florida, 1999

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 2008

Author: George T. Southworth

Approved by: Clifford Whitcomb
Thesis Advisor

Fotis Papoulias
Second Reader

David Olwell
Chairman, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Four salvage ships and four ocean-going towing ships are maintained and operated by the Military Sealift Command (MSC) for the U.S. Navy. In 2019, the first T-ATF ships will reach the end of their 40-year life expectancy. The program manager for these vessels has a set of top-level performance characteristics that are deemed as desirable requirements for a new ship class, encapsulating both legacy ship class capabilities.

The DoD has shifted defense planning from the specific service requirements generated system (RGS) acquisition to the Joint Capabilities Integration and Development System (JCIDS) approach that focuses more on *how* adversaries fight rather than *whom* they are fighting. This thesis explores how to use systems architecting to incorporate the capabilities derived from strategic guidance into a Department of Defense Architecture Framework (DODAF) product. The design tool, CORE, is used to explain the architecting methodology and produce DODAF v1.5 system models for decision making and acquisition requirement generation.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ARS	Auxiliary Rescue Ship
ATF	Auxiliary Tug Fleet
AV	All Views
C4ISRAF	C4ISR Architecture Framework
CDD	Capability Development Document
CIO	Chief Information Officers
CIVMARS	civilian merchant mariners
CJCS	Chairman Joint Chiefs of Staffs
CPD	Capability Production Document
CRD	Capstone Requirements Document
DAS	Defense Acquisition System
DNDDAF	Department of National Defense Architecture Framework
DODAF	Department of Defense Architecture Framework
DODAF v1.5	Department of Defense Architecture Framework version1.5
DoD	Department of Defense
ESL	end of service life
ESSM	Emergency Ship Salvage Material
ESWBS	Expanded Ship Work Breakdown Structure
FGC	Functional Group Code
FEAF	Federal Enterprise Architecture Framework
GO-CO	Government Owned Contractor Operated
GO-GO	Government Owned and Government Operated
ICD	Initial Capabilities Document
INCOSE	International Council on Systems Engineering
IPT	Integrated Product Team
JCA	Joint Capability Area
JCIDS	Joint Capabilities Integration Development System
MA CHENG	Mission Area Chief Engineer
MBSE	Model Based Systems Engineering
MDSU	Mobile Diving and Salvage Unit
METL	Mission Essential Task List
MODAF	Ministry of Defense Architectural Framework
MSC	Military Sealift Command
MSFSC	Military Sealift Fleet Support Command

NAVSEA	Naval Sea Systems Command
NAERG	Naval Architecture Elements Reference Guide
NP21	Naval Power 21
OV	Operational View
PPBE	Planning, Programming, Budgeting, and Execution
QDR	Quadrennial Defense Review
RFF	request for forces
RGS	Requirements Generating System
ROC	Required Operational Capabilities
ROV	Remote Operating Vehicles
RTF	Rich Text Format
SCN	Ship New Construction
SEALOGS	Sealift Logistics Commanders
SEP	Systems Engineering Plan
SoS	system of systems
SUPSALV	Supervisor of Diving and Salvage
SUPALV, NAVSEA 00C	Supervisor of Salvage and Diving
SV	Systems View
SWBS	Ship Work Breakdown Structure
TOGAF	The Open Group Architectural Framework
TV	Technical View
TYCOM	Type Command
USD (AT&L)	Under Secretary of Defense, Acquisition, Technology and Logistics
USFF	U.S. Fleet Forces
USTRANSCOM	Navy Component Commander to the U.S. Transportation Command
WMA	Warfighting Mission Area
VOO	Vessel Of Opportunity

EXECUTIVE SUMMARY

The transformation to a capabilities-based approach was to redefine defense requirements on delivering more warfighting capability, focusing on *how* our adversaries fight, rather than *whom* the adversary may be or *where* the war might occur. The motivation for this transformation was a realization across many levels of the U.S. Department of Defense (DoD) that systems development was consistently resulting in outcomes that did not meet the needs of stakeholders, both from the single service and the joint perspective. This thesis will report the development of an integrated systems architecting and engineering process, focusing on the systems architecting activities in the earliest stages of an integrated architecture development. To achieve the development of a truly integrated system, the needs of various stakeholders — who have their own cultures, which exhibit conflicting and non-commensurate objectives — must be considered in the approach. The field of architecture deals with situations as part of its legacy, and has been defined as a promising area in which to augment the earliest stages of systems engineering's well-defined integration of the two methods. Systems architecting and engineering would be a useful process in developing the types of integrated architectures envisioned for future defense systems.

An integrated systems architecting and engineering process is an iterative development process of a system with consideration of the stakeholders' needs — at the highest hierarchical level, all the way down to individual end item components — through the projected life cycle. A top-down iterative approach is needed for the architecture development process, flowing from the system or system of systems (SoS) at the highest levels and down to the lowest levels of components. Engineering and architecting take two very different approaches to the development of systems. Engineering is a deductive method, utilizing quantifiable analysis, while architecting is an inductive process dealing with non-quantifiable processes. The distinction between architecting and engineering is what tools or tactics are used to resolve issues. The range of approaches, to both the *art* and the *science* embodied in the two approaches, is not as far removed.

The Joint Capabilities Integration Development System (JCIDS) starts the acquisition as a joint process of Strategic Direction and develops a joint perspective of warfighting. A collaborative assessment and analysis of Joint Operational Concepts identifies integrated architectures to facilitate in system design. The operation of the JCIDS process and the Enterprise Architecture Hierarchy will require the systems architect to explore the intended requirements and constraints with the stakeholders to seek a clear set of objectives to which all could agree before formulating what the optimal outcome might be. Architecture frameworks are used to provide a consistent documentation basis to describe stakeholder views. The Department of Defense Architecture Framework (DODAF) is the basis upon which the products in this thesis are derived. They provide enough information in the stakeholder's views to make technical decisions.

Both the T-ARS and T-ATF class ship are coming to the end of their service life of 40 years. Eight vessels of a single hull type have been suggested as acceptable risks to accommodate the wartime and peacetime capabilities augmented by contracted commercial tows. This thesis describes an approach for structuring the performance characteristics from the Supervisor of Diving and Salvage (SUPSALV), creating information using a framework-based approach, in this case DODAF. The goal is to provide a clear path from mission area needs to requirements to be clear and defined, thereby achieving a product to construct assets to perform present salvage and towing operations and future tasks.

CORE is the tool used to assist in creating the architecture element relationships that will provide the architectural products to support stakeholder decision making. The architecture is based on a DODAF v1.5 schema, to populate a database of element classes, enabling the architect to define relationships, list a description, show hierarchies and various other characteristics The salvage and towing systems architecture development is initiated using a "Middle Out" method, since some legacy system characteristics are known. In a "Middle Out" start the architecture model is synthesized beginning with the legacy system known characteristics and requirements. Strictly speaking, the architecting process does not typically start by defining the form of the future system as a ship, aircraft, or any other vehicle. However, in this particular case, the end item has already been named a T-ARS(X) class of ship.

ACKNOWLEDGMENTS

I thank my family for their support through my education at the Naval Postgraduate School.

Additionally, I thank Dr. Whitcomb and Dr. Papoulias for their time and mentoring throughout the writing of this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

I. MOTIVATION FOR SYSTEMS ARCHITECTURE

Our Challenge is to understand how to integrate design and production technology into an acquisition process that industry can execute. This requires a deep knowledge of systems engineering and a profound understanding of the acquisition process. System engineering is key to ensuring that each ship is configured to optimize the fleet... All this implies a need for integration of elements and capabilities [1].

A. BACKGROUND

The 2001 Quadrennial Defense Review (QDR) directed the initiation of a capabilities-based approach to defining defense requirements. The transformation to a capabilities-based approach is to deliver more warfighting capability based on how our adversaries fight, rather than on whom the adversary may be or where the war might occur [2]. The motivation for this transformation was a realization, across many levels of the U.S. Department of Defense (DoD), that systems development consistently resulted in outcome that did not meet the needs of stakeholders, both from single service and joint perspective, from end users to program managers to planners.

What happens in the Department of Defense—and it runs me up the wall—is each service comes up with their things…and how in the world do you get those four things into a single fighting force at the end? It's a train wreck…every year when you're trying to do a budget. It's just a meat grinder trying to pull things together because they didn't start coming together earlier at a lower level. And we're going to fix that. I'll be the meat grinder [3].

The method of developing systems referred to by Secretary Rumsfeld is typically known as the Requirements Generating System (RGS), and is shown on the left side of Figure 1 — a bottom up, 'stovepiped' method that only captured the originating services 'needs.' The vision of warfighting was generated by the individual service's scenarios of fighting assuming no other services would play a part. The right side of Figure 1 represents the new systems development process, called the Joint Capabilities Integration Development System (JCIDS), which starts the acquisition as a joint process of Strategic Direction and develops a joint perspective warfighting vision [2].

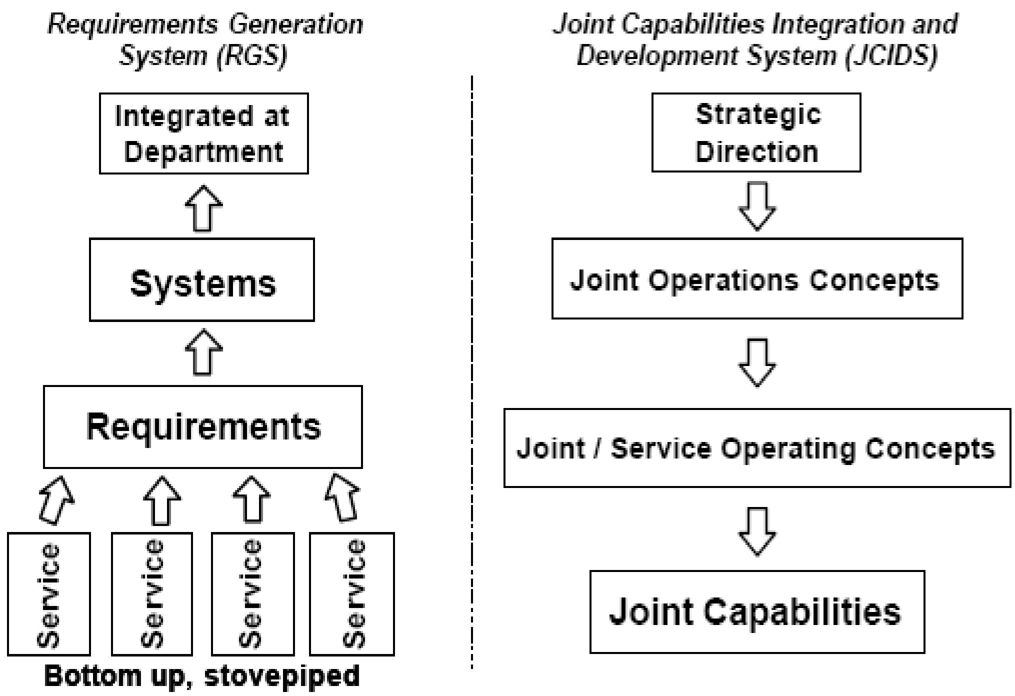

Figure 1. Capability-Based Approach From [2]

The joint vision for military strategy is shown in Figure 2 and describes the JCIDS process — an approach that utilizes the expertise of all government agencies to exploit existing capabilities and develop new capabilities to close gaps. The collaborative assessment and analysis of Joint Operational Concepts identifies integrated architectures to facilitate in system design [4]. In order to achieve the development of a truly integrated system, the needs of various stakeholders — who have their own cultures, which exhibit conflicting and non-commensurate objectives — an approach must be used that allows for consideration of this reality. The discipline of systems has traditionally been applied to such development processes, but this method is limited in the fuzzy front end of the process, especially in the explicit consideration of the incorporation of stakeholder needs and desires, and in the ability to effectively model the connection of these needs to an ability to synthesize creative thinking in an abstract, solution-neutral sense. The field of architecture deals with the situations as part of its legacy, and has been defined as a promising area in which to augment the earliest stages of systems engineering [5]. A well-defined integration of the two methods, systems architecting and engineering, would be useful process in developing the types of integrated architectures envisioned for future defense systems.

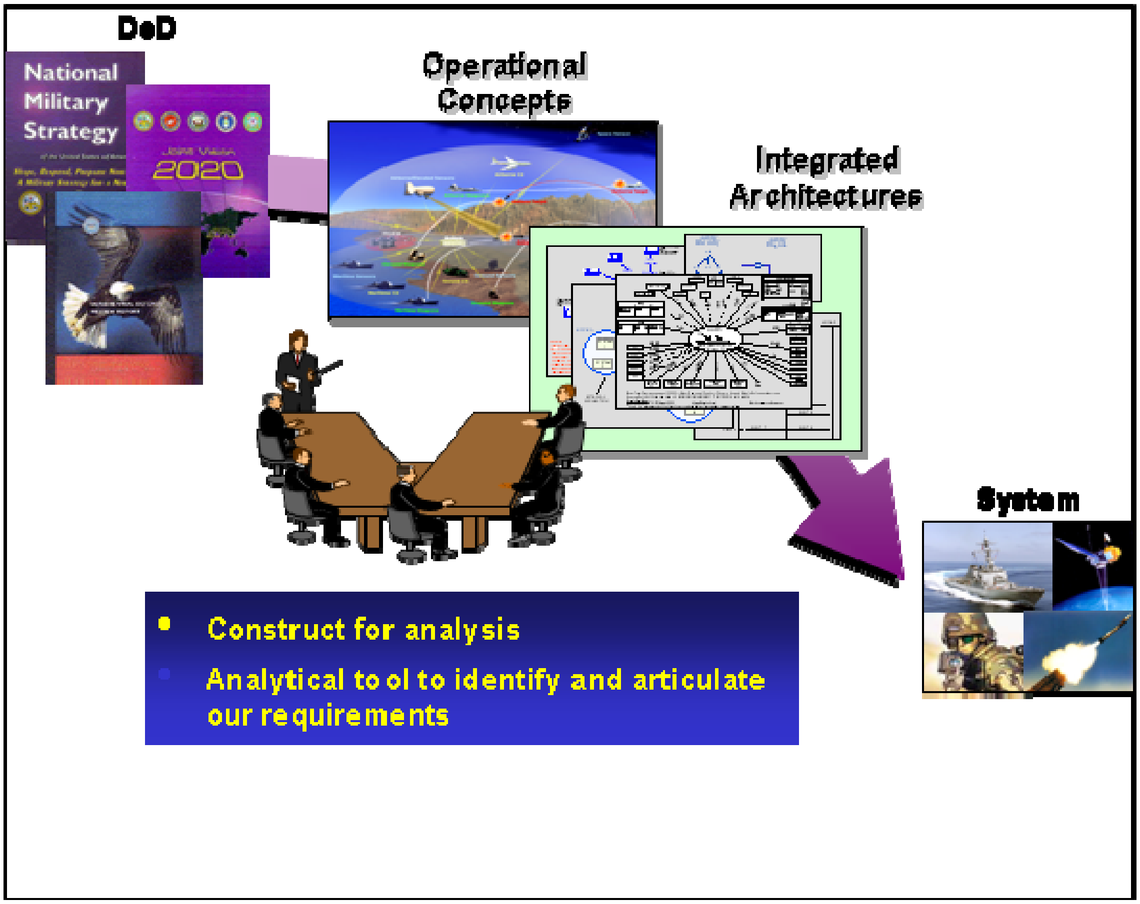

Figure 2. DOD Development Process From [6]

B. SYSTEMS ARCHITECTING AND ENGINEERING

An integrated systems architecting and engineering process is an iterative development process of a system with consideration of the stakeholders' needs — at the highest hierarchical level all the way down to individual end item components — through the projected life cycle. A system is an integrated set of elements that accomplishes a defined objective [7]. Further, a system of system (SoS) is a set of interdependent systems that are related or connected to provide a given capability [8].

Systems engineering is defined by INCOSE as "a branch of engineering whose responsibility is creating and executing an interdisciplinary process to ensure that customer and stakeholder's needs are satisfied in a high-quality, trustworthy, cost-efficient and schedule-compliant manner throughout a system's entire life cycle, from development to operation to disposal [7].” Systems engineering is generally used to describe the set of processes applied to the development of systems and spans activities, from customer need discovery, completely through to realization and eventual disposal and recycling. While this description encompasses the entire life cycle of a product development, it is perhaps too broad in that the knowledge, skills, and abilities needed to perform the process are different in the early stages, as opposed to the later stages.

Systems architecting, extending the traditional definition of architecting, deals primarily with ill-structured situations with non-measurable quantities and non-quantitative tools and guidelines, making it well-suited for the early stages of systems engineering.

> *Systems architecting and systems engineering present complementary approaches to the development of SoS. For capability-based development of unprecedented systems, the initial portions of the traditional systems engineering process have been demonstrated to show unsatisfactory results, in particular due to the complexity in transforming ill-defined capabilities into requirements useful enough to begin any engineering-based design. A capability is defined as “The ability to achieve a desired effect under specified standards and conditions through combinations of means and ways to perform a set of tasks.” In the architecting process, capabilities are represented as operational activities that are modeled and simulated in executable form as behavioral entities. While recognition of focus on capabilities-driven systems architecting has given rise to the fairly recent development of system architecting methods, frameworks, and processes, what is lacking at this time is a defined method for architecting – the development of the architecture itself* [9].

For capability-based acquisition, the system architecting and engineering process is utilized for the iteration and decomposition of Operational Concepts to capabilities, then capabilities to system requirements, and further decomposition to end items. Figure 3 shows a hierarchal view of how the system engineering plan is carried through each level, translating needed capabilities to functional systems and systems of systems (SoS). A recent USD (AT&L) (Under Secretary of Defense, Acquisition, Technology and Logistics) memorandum establishes systems engineering policy and mandates systems engineering for all programs. This memorandum is included in the latest revision to DoD Instruction 5000.2. An extract from the memorandum follows:

> *Systems Engineering (SE). All programs responding to a capabilities or requirements document, regardless of acquisition category, shall apply a robust SE approach that balances total system performance and total ownership costs within the family-of-systems, systems-of-systems context. Programs shall develop a Systems Engineering Plan (SEP) for MDA (Milestone Decision Authority) approval in conjunction with each Milestone review, and integrated with the Acquisition Strategy. This plan shall describe the program's overall technical approach, including processes, resources, metrics, and applicable performance incentives. It shall also detail the timing, conduct, and success criteria of technical reviews* [10].

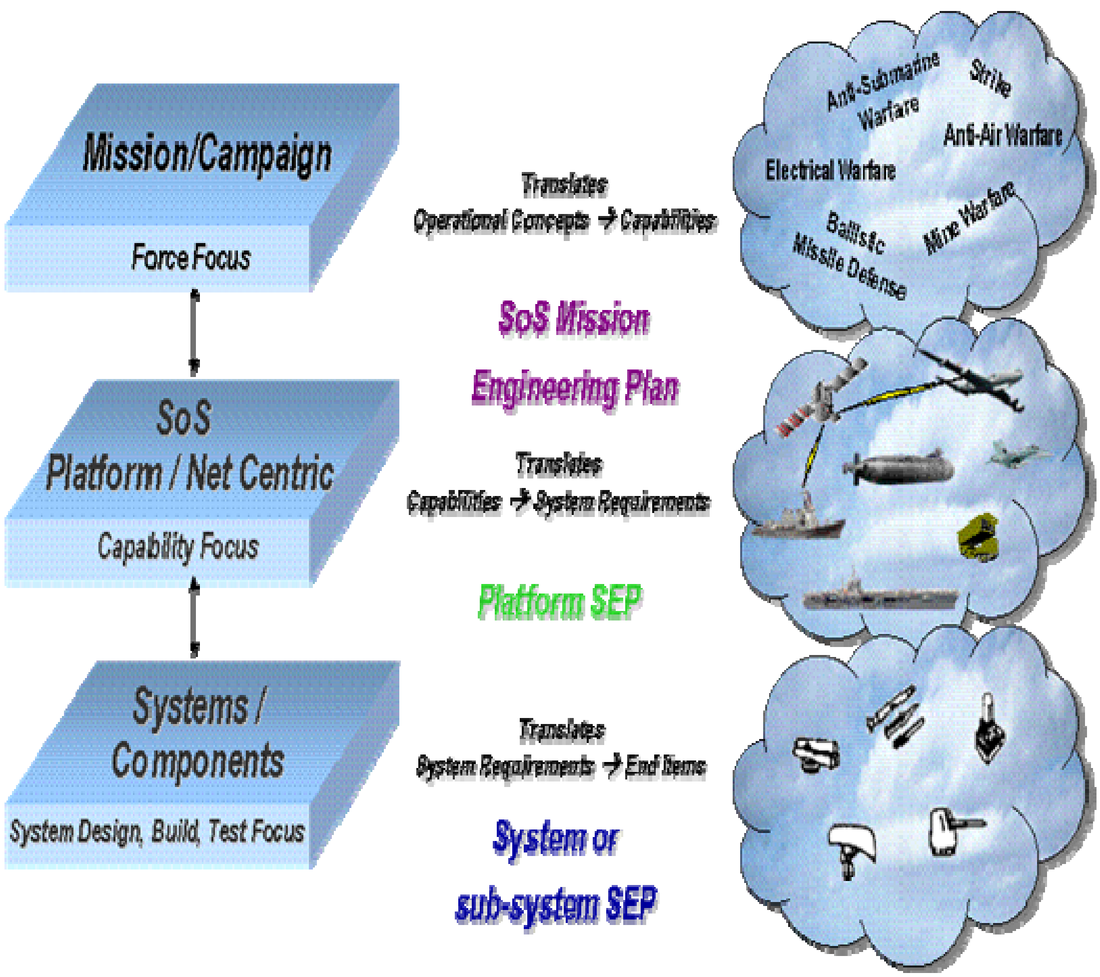

Figure 3. System Engineering Hierarchy From [11]

A more detailed description of the hierarchy shown in Figure 3 is contained in the Department of the Navy Enterprise Architecture, Figure 4. At the highest level, this hierarchy has four mission areas that categorize portfolios or SoS. These portfolios are then subdivided as capability areas that are mapped to the Naval Power 21 (NP21) pillars to provide a crosswalk to Navy Budgeting categories. The Warfighting Mission Area (WMA) is managed by the Chairman Joint Chiefs of Staffs (CJCS), who manages the portfolios through the JCIDS process and provides input to the Planning, Programming, Budgeting, and Execution (PPBE) system and Defense Acquisition System (DAS) processes. Each sub-portfolio contains Joint Capability Area (JCA), which are mapped to the Navy Required Operational Capabilities (ROC). The ASN RDA Chief Systems Engineer has assigned a Mission Area Chief Engineer (MA CHENG) to each JCA for configuration control. Figure 4 illustrates how systems and SoS are tied to the DoD Mission Areas; the context of this thesis concerns architecting the systems tied to JCAs [12].

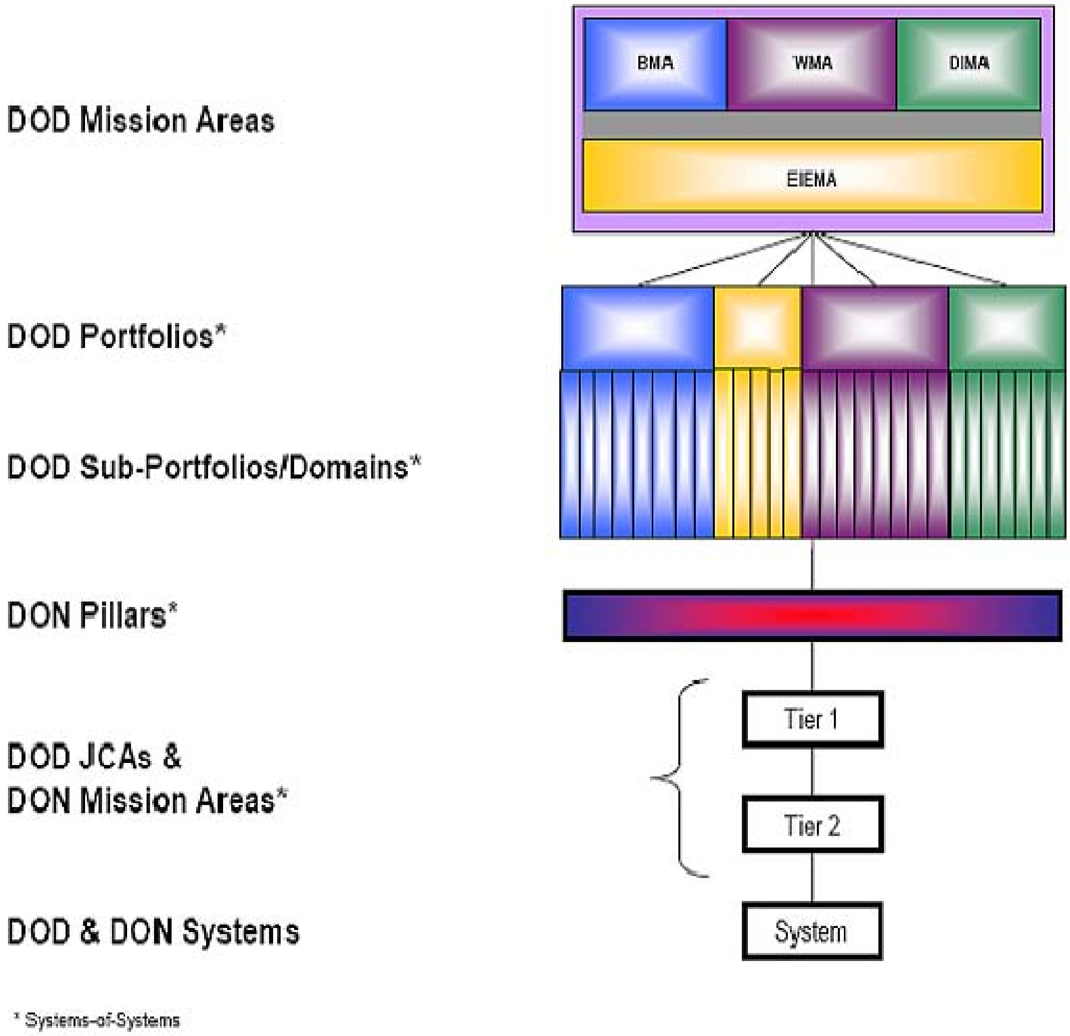

Figure 4. Department of Navy Enterprise Architecture Hierarchy From [12]

In regard to the architecting process to achieve this SoS outcome — beyond a very short summary of several systems engineering approaches in the DODAF — there is no well-defined method detailed on how to create an architecture that addresses overall capabilities, concepts of operations, and requirements to formulate design considerations for engineering a system or SoS in an enterprise architecture framework.

C. SUMMARY

This thesis will report the development of an integrated systems architecting and engineering process, focusing on the systems architecting activities in the earliest stages of an integrated architecture development. The method will be applied to an architecture that directly addresses the recapitalization of a future towing and salvage ship. The overall context of the SoS integrated architecture is taken down to a system level, through an example applied to a specific ship platform, to accomplish towing and salvage capabilities. The process of identifying a capability gap or need, defining architecture and engineering requirement specifications, analyzing system functions, and allocation to system physical components is completed, including development of all architecture elements and the creation of architecture framework products.

II. SYSTEM ARCHITECTURE AND THE ARCHITECTURING PROCESS

Architecting, the planning and building of structures, is as old as human societies— and as modern as the exploration of the solar systems [5].

A INTRODUCTION

Considering the operation of the JCIDS process and the Enterprise Architecture Hierarchy, a top-down iterative approach is needed for the architecture development process — flowing from the system or system of systems (SoS) at the highest levels, all the way down to the lowest level of components. This chapter describes the background and key aspects surrounding the need for the development of an architecture (architecting), its relationship to the integration with systems engineering, and the various characteristics of what constitutes an architecture.

B. CONSIDERATION OF SYSTEMS ENGINEERING AND ARCHITECTING

Engineering and architecting take two very different approaches to the development of systems [7]. In traditional systems engineering terminology, early stage tasks have been often referred as "requirements capture," "requirements generation," "requirements definition," and "architectural design" in an attempt to define the overall early-stage design objective. Engineering is a deductive method, utilizing quantifiable analysis to deal primarily with physical and scientific situations with measurable quantities and concepts. Architecting, on the other hand, is an inductive process dealing with non-quantifiable processes, taking into consideration the desires and needs of stakeholders who express themselves in their native language using abstract, subjective expressions. The distinction between architecting and engineering is what tools or tactics are used to resolve issues. The range of approaches to both the *art* and the *science* embodied in the two approaches are not as far removed. For instance, in systems engineering, the requirements may be ill-structured due to the customer's optimal goal not being fully realized. The architecting approach would be to jointly explore the intended requirements and constraints with the stakeholders to seek a clear set of objectives to which all could agree before formulating what the optimal outcome might be. Figure 5 summarizes the fundamental differences.

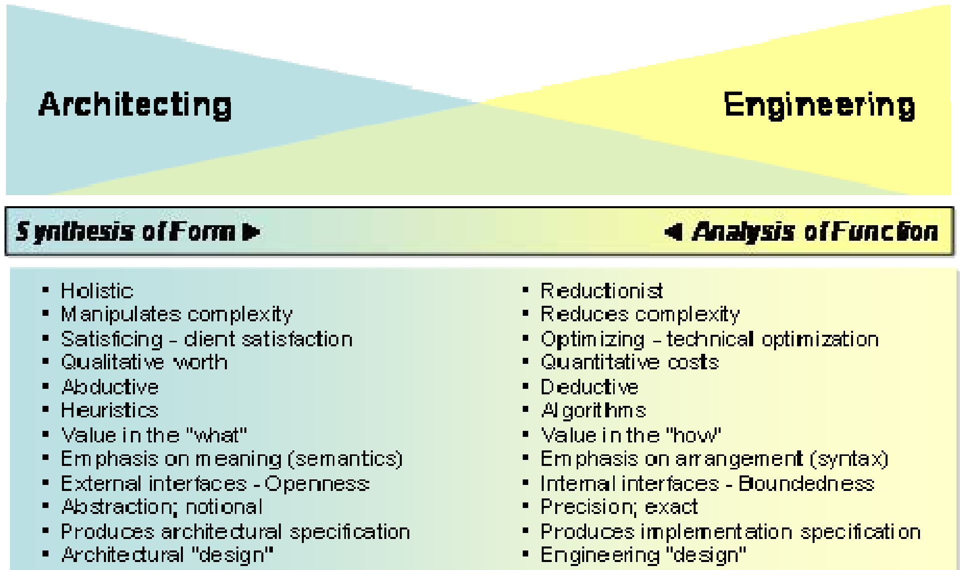

Figure 5. Comparison of Architecting and Engineering. From[13].

C. SYSTEMS ARCHITECTING APPROACH

An architecture exists for the purpose of achieving a well-defined system in the application domain to achieve the eventual system developed in the solution domain that can be used to meet desired capabilities over a specific time frame or set of time frames [5]. An architecture includes elements and interconnections; how they interrelate is the key to an effective design. Boundaries are necessary to contain the architecture to its context and help to define how it interacts with the external environment. Architecting is a process consisting of methods, tools, and people implemented to develop the system architecture.

The architecting process is one of participative discovery. The architect continually works with all stakeholders to define the architecture, continuously working to resolve ambiguity, reduce complexity and focus creativity, taking the initial capability need (or “idea”), from the abstract to the concrete through a series of ever-evolving models of continually improved fidelity. The architect uses architecting principles, model-based systems principles and activities to define, develop, integrate, and evolve the operational and system models. For DoD, architecting is guided by the JCIDS process, entering into acquisition in the earliest part of the concept refinement stage. Architecting the capabilities is a dynamic part of the early-stage process and differs in that they should not be considered as initial requirements.

Systems architecting combines working with the stakeholders in the conceptualization process to identify requirements for the development community in pursuit of eventual technical optimization [5]. The continually maturing of the stakeholder needs, into the eventual requirements specifications, demands a well-refined set of documents and models for the later stages. To support a well-defined set of deliverables for the architecting process, frameworks have been defined, along with supporting views and products.

D. ARCHITECTURE FRAMEWORKS

Every system has an architecture that fulfills a mission. Architecture frameworks are used to provide a consistent documentation basis to describe an architecture. Zachman (1987) was one of the first to define architecture in terms that can apply to any object, not just civil engineering. The intent of the Zachman framework is to establish a common vocabulary, defining complex enterprise systems. The Department of Defense developed the C4ISR Architecture Framework (C4ISRAF) based on Zachman's philosophy. The C4ISRAF was eventually expanded in scope and re-titled the Department of Defense Architecture Framework (DODAF) for use in development of all DoD system architectures. In the commercial community, The Open Group Architectural Framework (TOGAF) uses open systems building blocks for mission-critical business applications. In response to the Clinger-Cohen Act (1996), the Federal Chief Information Officers (CIO) Council developed the Federal Enterprise Architecture Framework (FEAF) to guide in sharing of federal information. Other architectural frameworks have been established in NATO and other countries, such as UK Ministry of Defense Architectural Framework (MODAF) and the Department of National Defense Architecture Framework (DNDDAF), for procuring and integrating their respective defense systems architectures [14].

At its core, the Zachman Framework for Enterprise Architecture (Figure 6) recognizes that the stakeholders see a system from their own perspective, and that they have different views of the end product. Similarly, any architecture should develop these views to enhance stakeholder understanding of the system or business model.

- Strategists and Decision Authority view of the system is used to evaluate doctrine to conceptualize a capability or capability gap.
- Planner view of the capability concept is an analysis through the Joint Capabilities Integration and Development System (JCIDS) process. Although iterative, and not in a hierarchical design within concept definition, a capability need is defined.
- Designer/Architect view is used to produce capability architecture for Builders to develop and engineer.
- Operator views are used for capability employment.

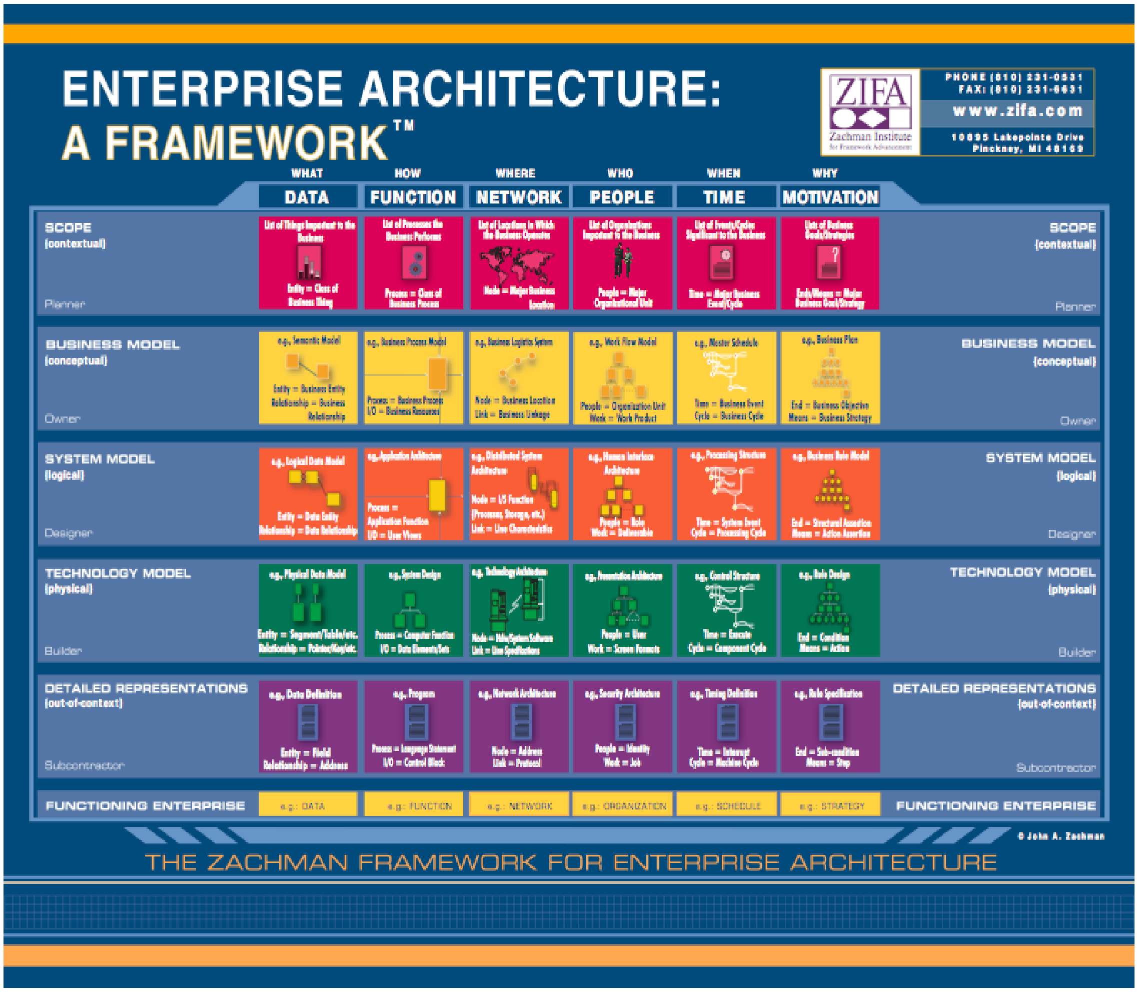

Figure 6. Zachman Framework for Enterprise Architecture From[15]

The architecting process must be configured to create and illustrate these views and include the ability to allow for an iterative process of discovery in meeting emerging capability needs. Implementation of this capability-based development process is the front end to the system acquisition and development process, and is an attempt to provide a sound basis for beginning a systems engineering approach to development by keeping the early-stage process focused on the problem space and not the solution space (Figure 7). The architect can refine the initial concept of the system to have a better definition of how it fulfills the mission. As each iteration is developed, the conceptual model of the architectural description will depict how the architect should design a solution space from the problem space. The role of the architect is to understand the scope and context of each stakeholder's concerns. Each concern establishes viewpoints that need to be modeled. As the architecture evolves into a design, the actual view is different with each stakeholder. The architecture should capture the thoughts of the stakeholders as they pertain to the element described in its environmental use. Concurrently, the design should consider the internal components and their interfaces and arrangements (Work Breakdown Structures) to fulfill the contractor's needs to build the system while balancing the interests of the stakeholders.

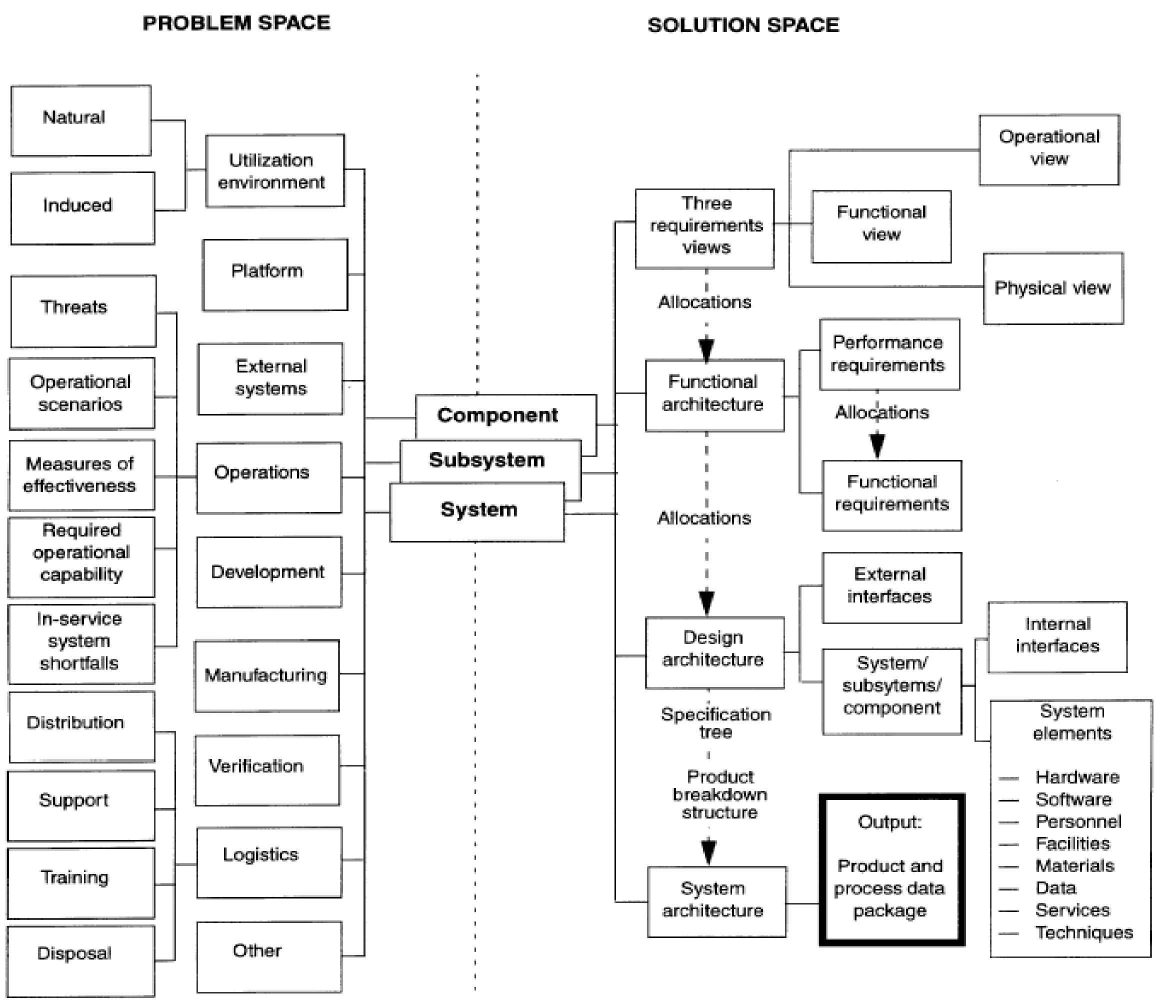

Figure 7. Problem and Solution Space for Systems Architecting and Engineering From [16].

The architecture for a system should be described in a language native to different stakeholders. The stakeholders' views are concerns that identify the architectural description. As stated before, the stakeholders' viewpoints are different, dependent on the objective and resources of the stakeholder. A model provides a rationale for the architecture. Figure 8 is the IEEE 1471 conceptual framework to describe an architecture [16].

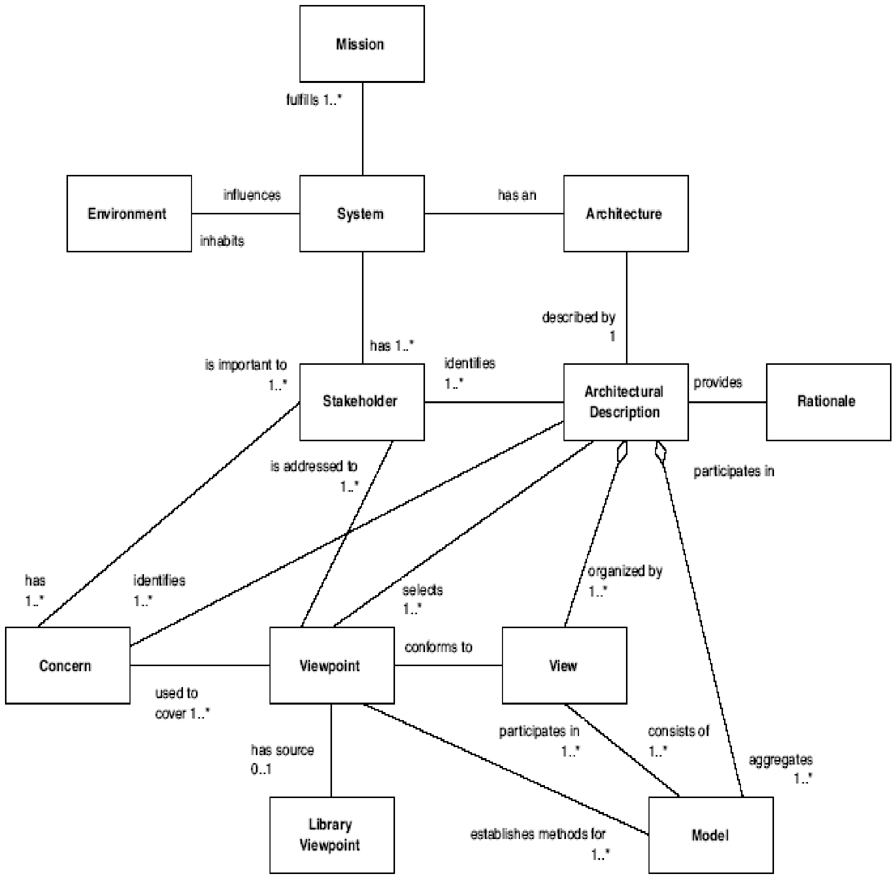

Figure 8. Conceptual model of the architectural description. From[16]

This diagram describes the relationship among all aspects that constitute architecture. To summarize what is meant by the model, some key relationships are described. Every system fulfills a mission, while at the same time that system has an architecture, which is in turn described by an architecture description, which provides a rationale as well as an identified set of stakeholders. The architecture description is organized by a set of views that conform to stakeholder viewpoints. Finally, views are consistent with models, which are used to create various aspects of the system. There is an underlying basis of any system — the architecture — and the views are critically important to make sure stakeholders can see their needs are being met.

E. ARCHITECTURE VIEWS

The DoD 5000.2 states that each integrated architecture shall have three views: operational, systems, and technical standards.

> *…as defined in the current Architectural Framework guidance and have direct relationships to DoD Component-developed functional area integrated architectures. The Joint Staff (or Principal Staff Assistant (PSA) for business areas) shall lead development of the operational view, in collaboration with the Services, Agencies, and Combatant Commanders, to describe the joint capabilities that the user seeks and how to employ them. The USD (AT&L) (or PSA for business areas) shall lead development of the systems view, in collaboration with the Services, Agencies, and Combatant Commanders, to characterize available technology and systems functionality. The systems view shall identify the kinds of systems and integration needed to achieve the desired operational capability. The DoD Chief Information Officer (CIO) shall lead the development and facilitate the implementation of the Global Information Grid Integrated Architecture, which shall underpin all mission area and capability architectures. The Military Departments and Defense Agencies shall participate in the identification of the appropriate technical view consisting of standards that define and clarify the individual systems technology and integration requirement [17].*

Figure 9 shows the relationships of the views. The purpose to develop a broad-based "big-picture" aspect of multiple views of the system as the architect documents key factors to apply insight for design, with all views being derived from one single architecture description. Each view encapsulates a complete representation of the particular perspective and is consistent with respect to other views.

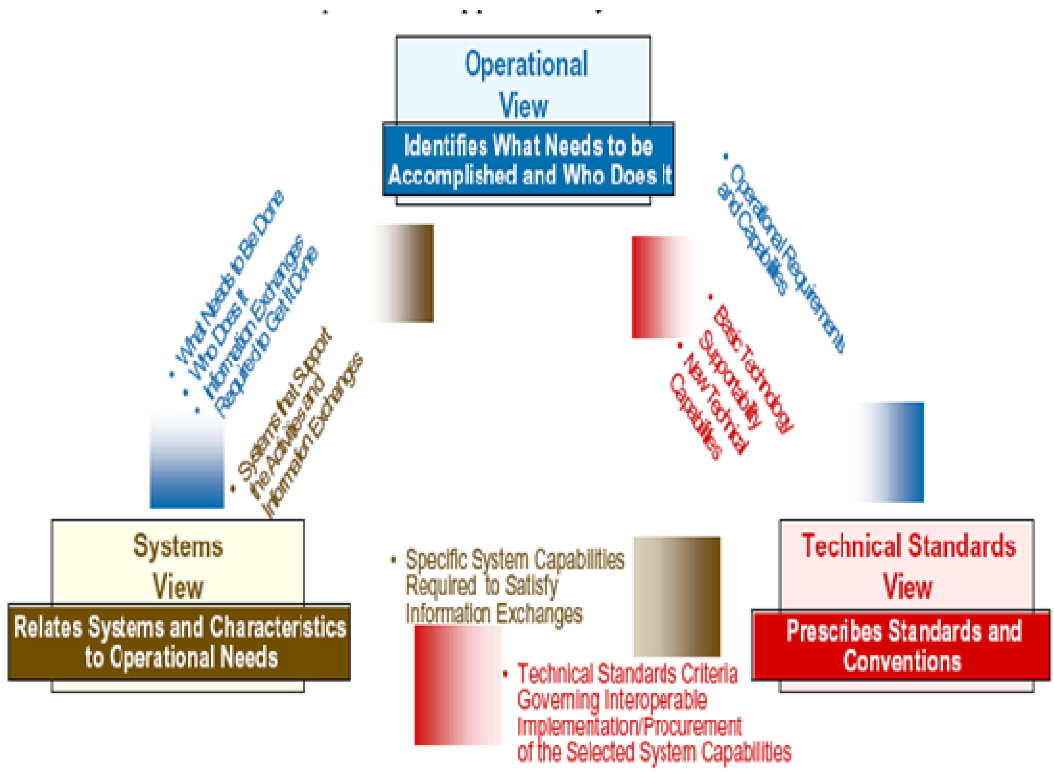

Figure 9. Linkage among views. From[18].

The Operational View (OV) is essentially the user or operator view; describing tasks, operational elements, and mission needs. Driven by document or emerging concepts, the OV shows how the system actually operates. The OV identifies conceptual processes for business and intended uses for the architecture being developed. It defines the 'what and who' of a system. and the frequency and type of information to be exchanged to fulfill the mission.

A Systems View (SV) is one that focuses on the infrastructure and connections among system described in the OV. The DoD warfighting and business functions are related to the operational needs to identify specific system capabilities. The SV identifies specific technologies and systems; these technologies and systems can be emerging or mature. For example, this view is what a homebuilder would use to emphasize electrical, water, sewer, and entertainment/communication systems.

The Technical View (TV) elaborates on the policies and standards set forth to govern the system being architected. TV delineates the technical implementation criteria to which the system or SOS should comply. The DODAF explains TV as a way to promote efficiency and interoperability [18].

The view that illustrates all three views at once is called an All Views (AV), which reflect the entire architecture as a "Big Picture." AVs capture the scope, purpose, intended users, and environment for which the system or SoS will be designed. This would include the subject area and time frame for the architecture.

As these views are derived from a common set of architectural relationships, an integrated approach is in place for development of a systems requirements governed by technical standards — as long as the basis set of element relationships is consistently refined.

F. ARCHITECTURE PRODUCTS

The DODAF uses architecture products in describing the architectural information. Table 1 shows a list of DODAF views identified for a standardized approach in identifying an integrated architecture. Additional products could be made, but should follow these same guidelines. The first column gives the applicable view, the second column provides the alphanumeric reference identifier, the third column is the formal name, and the last column is a general description of the product. The list does not imply a specific order that one must use to generate products. The architectural products used are specific to the situation, scope, and objectives to be met.

The actual framework for which these products are derived must provide enough information — in the stakeholder's views —to make technical decisions. The descriptive methodology must identify capability needs, subsequently relating them to systems development. The use of architectures has been specifically addressed in the JCIDS process as Mandatory Appendices in the Initial Capabilities Document (ICD), Capability Development Document (CDD), Capability Production Document (CPD), and Capstone Requirements Document (CRD) [8]. Table 2 lists the DODAF's recommendations for architecture products supporting decision making for a number of processes.

Applicable View	Framework Product	Framework Product Name	General Description
All Views	AV-1	Overview and Summary Information	Scope, purpose, intended users, environment depicted, analytical findings
All Views	AV-2	Integrated Dictionary	Architecture data repository with definitions of all terms used in all products
Operational	OV-1	High-Level Operational Concept Graphic	High-level graphical/textual description of operational concept
Operational	OV-2	Operational Node Connectivity Description	Operational nodes, connectivity, and information exchange needlines between nodes
Operational	OV-3	Operational Information Exchange Matrix	Information exchanged between nodes and the relevant attributes of that exchange
Operational	OV-4	Organizational Relationships Chart	Organizational, role, or other relationships among organizations
Operational	OV-5	Operational Activity Model	Capabilities, operational activities, relationships among activities, inputs, and outputs; overlays can show cost, performing nodes, or other pertinent information
Operational	OV-6a	Operational Rules Model	One of three products used to describe operational activity—identifies business rules that constrain operation
Operational	OV-6b	Operational State Transition Description	One of three products used to describe operational activity—identifies business process responses to events
Operational	OV-6c	Operational Event-Trace Description	One of three products used to describe operational activity—traces actions in a scenario or sequence of events
Operational	OV-7	Logical Data Model	Documentation of the system data requirements and structural business process rules of the Operational View
Systems	SV-1	Systems Interface Description	Identification of systems nodes, systems, and system items and their interconnections, within and betw een nodes
Systems	SV-2	Systems Communications Description	Systems nodes, systems, and system items, and their related communications lay-downs
Systems	SV-3	Systems-Systems Matrix	Relationships among systems in a given architecture; can be designed to show relationships of interest, e.g., system-type interfaces, planned vs. existing interfaces, etc.
Systems	SV-4	Systems Functionality Description	Functions performed by systems and the system data flows among system functions
Systems	SV-5	Operational Activ ity to Systems Function Traceability Matrix	Mapping of systems back to capabilities or of system functions back to operational activities
Systems	SV-6	Systems Data Exchange Matrix	Provides details of system data elements being exchanged between systems and the attributes of that exchange
Systems	SV-7	Systems Performance Parameters Matrix	Performance characteristics of Systems View elements for the appropriate time frame(s)
Systems	SV-8	Systems Evolution Description	Planned incremental steps toward migrating a suite of systems to a more efficient suite, or toward evolving a current system to a future implementation
Systems	SV-9	Systems Technology Forecast	Emerging technologies and software/hardware products that are expected to be available in a given set of time frames and that will affect future development of the architecture
Systems	SV-10a	Systems Rules Model	One of three products used to describe system functionality—identifies constraints that are imposed on systems functionality due to some aspect of systems design or implementation
Systems	SV-10b	Systems State Transition Description	One of three products used to describe system functionality—identifies responses of a system to events
Systems	SV-10c	Systems Event-Trace Description	One of three products used to describe system functionality—identifies system-specific refinements of critical sequences of events described in the Operational View
Systems	SV-11	Physical Schema	Physical implementation of the Logical Data Model entities, e.g., message formats, file structures, physical schema
Technical	TV-1	Technical Standards Profile	Listing of standards that apply to Systems View elements in a given architecture
Technical	TV-2	Technical Standards Forecast	Description of emerging standards and potential impact on current Systems View elements, within a set of time frames

Table 1. List of architecture products. From[19]

APPLICABLE ARCHITECTURE PRODUCTS

RECOMMENDED USES OF ARCHITECTURE:	All View		Operational View (OV)							Systems View (SV)											Tech Stds View	
	1	2	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	10	11	1	2
Planning, Programming, Budgeting Execution Process																						
CapabilityBased Analysis for IT Investment Decisions	●	●	●	●	●	●	●	●	⊙	●	⊙	●	●	●	●	●	●	●	●		●	⊙
Modernization Planning and Technology Insertion/Evolution	●	●	⊙	●	⊙	⊙	●	⊙		●	⊙	⊙	⊙	●	⊙	●	●	●			●	●
Portfolio Management	●	●		●			●	⊙		●			⊙	●		⊙	●				⊙	
Joint Capabilities Integration and Development System																						
JCIDS Analysis (FAA, FNA, FSA)	●	●	●	●	⊙	⊙	●	●		●	⊙		⊙	●							⊙	
ICD/CDD/CPD/CRD	●	●	●	●	●		●	●		●	⊙	⊙	●	●	●	⊙	⊙	⊙	●		●	⊙
Analysis of Alternatives AoA)	●	●	●	●	⊙		●	●		●	⊙	⊙	●	●	⊙	⊙	⊙	⊙			⊙	⊙
Acquisition Process																						
Acquisition Strategy	●	●	●	●	⊙		●	⊙		●	⊙			●							●	
C4ISP	●	●	●	●	●		●	●		●			⊙		●	⊙					●	⊙
System Design and Development	●	●		●	●		●	●	⊙	●	●	●	●	●	●	●	●	⊙	⊙	⊙	●	⊙
Interoperability and Supportability of NSS and IT Systems	●	●	●	●	●	⊙	●	⊙	⊙	●	⊙	●	●		●	●	⊙	⊙	⊙	⊙	●	⊙
Integrated Test & Evaluation	●	●		●	●	⊙	●	●	⊙	●	●	●	●	●	●	●			●	⊙	●	
Operations (Assessment, Planning, Execution, ...)																						
Operations Planning & Execution	●	●	●	●	●	●	●	●	⊙	●	●	⊙	⊙	●	⊙	⊙					⊙	
CONOPS & TTP	●	●	●	●	●	●	●	●		●	⊙	⊙	⊙	⊙							⊙	
Communications Plans	●	●	●	●		⊙	⊙			●	●						⊙	⊙			●	⊙
Exercise Planning & Execution	●	●	●	●	●	●	●	●		●	●	⊙	⊙	⊙	⊙						⊙	
Organizational Design	●	●	●	●	●	●	●	⊙	⊙	⊙	⊙			⊙								
BPR/FPI	●	●	⊙	●	●	●	●	●	⊙													

● = Product is highly applicable

⊙ = Product is often or partially applicable

(dark shading) = Product is specifically addressed in policy

(light shading) = Product is required for an integrated architecture

blank = Product is usually not applicable

Table 2. Architecture Products by use. From[19]

G. SUMMARY

The architecture consists of basis description of the elements and their relationships. The architecture framework defines a set of views that show the system in the stakeholders' native language. The views are all derived from the elements and relationships, so they accurately represent the system. This conceptual model of the element relationships, along with the series of views, provides the necessary information to begin what is typically the beginning of the traditional systems engineering process.

III. SALVAGE AND TOWING BACKGROUND

Navy diving - performance under pressure.
— navydiver.org

A. INTRODUCTION

This chapter will give a brief explanation of how the U.S. Navy Salvage and Towing community is organized. It will also present a brief history of salvage and towing ship classes, describing their utility to the military. Finally, this chapter emphasizes a motivation for recapitalizing the current fleet of salvage and towing vessels.

B. U.S. NAVY SALVAGE COMMUNITY

The Salvage Facilities Act (10 U.S.C. 7361-637) authorizes the Secretary of the Navy to have a Salvage program. It allows for the maintenance of a national salvage capability for use in peacetime, war, or a national emergency [20]. Salvage forces have unique tasks that require specialized equipment and highly trained personnel. The collaboration with the Military Sealift Command (MSC) enables the Navy to provide salvage and towing capabilities. The 'triad' of U.S. Navy Salvage forces integrates Mobile Diving and Salvage Unit (MDSU), MSC, and Supervisor of Salvage and Diving (SUPALV, NAVSEA 00C). They serve as the core for removing hazards of navigation (in foreign and domestic coastal waters), repairing and towing damaged vessels, recovering sensitive items (such as aircraft *black boxes*), and recovering other high-value objects from the ocean depths. These are a few examples of the tasks of these organic forces within the Department of the Navy and the requests from federal and civilian entities through the request for forces (RFF) process to the prospective service command. Although each functional element of the salvage triad works jointly for both salvage and towing, each has a separate organizational structure [21].

SUPSALV is an agency (Code 00C) of Naval Sea Systems Command (NAVSEA); which is not in the operational chain of command. For salvage operations directed by the Chief of Naval Operations, SUPSALV is tasked as operational control.

Furthermore, SUPSALV is the technical authority for all U.S. Navy diving and authorizes any equipment used. In the instance that personnel or equipment assets are not available, SUPSALV maintains and exercises all diving and salvage contracts. For deep ocean recovery and emergent ship salvage material, SUPSALV maintains and exercises government-owned, contractor-operated (GO-CO) equipment and capabilities to provide recovery of objects to a depth of at least 20,000 feet, ship salvage, pollution control, and underwater ship husbandry.

MDSU One and MDSU Two's mission is to provide a combat-ready, deployable detachment to conduct harbor clearance, salvage, underwater search and recovery, and underwater emergency repairs. They are equipped with diving and salvage equipment that is air-mobile and scalable to mission objectives. The detachment can be as small as five divers for small operations, such as side scan sonar search operations, to larger groups for larger salvage operations. Both MDSUs are components of the Naval Expeditionary Combat Command. For the purposes of this paper, the mission describes only aspects of diving, salvage, and towing as part of the triad discussed [21].

The mission of Military Sealift Command (MSC) is to provide combat logistics to sustain U.S. forces worldwide, during peacetime and in war, for as long as operational requirements dictate. Administratively, MSC is a Navy echelon III command under U.S. Fleet Forces (USFF), providing more than 40 government-owned/government-operated ships. Operationally, MSC is the Navy Component Commander to the U.S. Transportation Command (USTRANSCOM), supporting mission objectives through Sealift Logistics Commanders (SEALOGS). An Echelon IV command named Military Sealift Fleet Support Command (MSFSC) was formed in October 2006 to man, train, equip, and maintain Government-owned and Government -operated (GO-GO) ships. The T-ARS and T-ATF ships are a part of the Type Command (TYCOM) under the MSFSC and are manned by civilian merchant mariners (CIVMARS). Performing these type-commander responsibilities makes MSFSC the only subordinate command under MSC with global responsibilities [22].

C. SALVAGE AND TOWING SHIP HISTORY

“Ships are built for a variety of purposes, but all must meet certain fundamental requirements. They must have reserve buoyancy to enable them to carry their designed loads and resist damage, stability to resist environmental forces or damage, and strength to withstand the stresses imposed on their structure by their own weight, cargo stores, and the sea” [23]. U.S. Navy Salvage and Towing ships are required to be robust enough to endure the stress of:

- combat salvage
- rescue towing
- fire fighting
- emergent repair
- ocean towing
- heavy lift

Figure 10. ATF Navajo Class. From[24]

Fleet tugs are used to tow ships, barges and targets for gunnery exercises. They are also used as platforms for salvage and diving work, as participants in naval exercises, to conduct search and rescue missions, to aid in the cleanup of oil spills and ocean accidents, and to provide firefighting assistance. The Navy designed the first seagoing 'tug,' the NAVAJO (ATF-64) class in 1939, to support logistic requirements for World War II. The German U-boats were inflicting damage to both military and commercial vessels to the point of clogging major waterways in Europe. The vessels served the Navy well for over fifty years. Their long-range seaworthiness made them valued as ocean-going tugs and for combat towing operations. During combat operations, the ATF class ships would also perform some firefighting and salvage assistance. Wooden-hulled Rescue vessels (ATR class) were utilized in submarine-infested waters for firefighting support and light towing, but did not have an ocean-going towing capability. The steel-hulled BOLSTER (ARS-38) class vessels were designed as salvage ships but incorporated a powered reel for towing, which made them interchangeable with the ATF for ocean-going towing. This made for an excellent design and served the U.S. Navy well until the mid 1990s [25]. The newer SAFEGUARD (ARS-50) class salvage ships were an improved design to accommodate heavier salvage work and increased bollard pull.

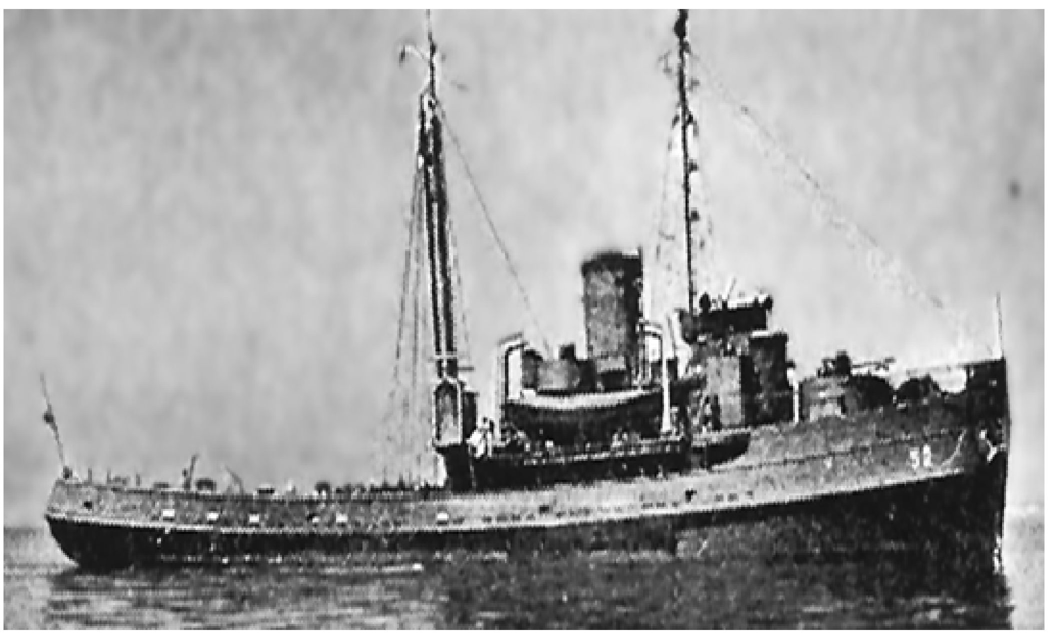

Figure 11. ATR-52 – Rescue Tug From[24]

Although the Navy recognizes several types of towing, the purpose of this study is limited to ocean towing, rescue towing, and salvage towing. Ocean towing is defined as point-to-point (from one harbor to another) towing with no refuge enroute. This includes the safe towing of defueled, nuclear-powered ships. Rescue towing is the saving of a stricken or inoperable ship at sea and towing it to a safe harbor. Salvage towing involves the immediate towing of a vessel after a salvage operation. Combat Salvage and towing missions involve service in hostile areas where vessels are damaged, afire, disabled, or stranded due to enemy fire [26].

Figure 12. ARS-38 Bolster Class From [24]

Salvage ships can perform as towing platforms, but their primary missions are to perform combat salvage, emergency repair, and firefighting. Many of the attributes that make salvage ships good salvage and towing platforms also make them good platforms for performing engineering operations. Specifically, tugs that can perform open ocean tows are often equipped with heavy lift cranes, have large expanses of deck area for temporary installation of specialized equipment, and are designed to keep station or moor over a site of interest [27]. An emerging capability is a Vessel Of Opportunity (VOO) for Remote Operating Vehicles (ROV) and Side Scan Sonar systems to explore the ocean's depths for search and recovery operations.

D. TOWING AND SALVAGE TODAY

Figure 13. T-ATF Powhatan class From [24]

Presently, the MSFSC has a force of four T-ATF-166 (Powhatan class) ocean-towing ships and four T-ARS-50 (Safeguard class) Salvage ships. The vessels are defined as Government-Owned Contractor-Operated (GO-CO) and are located throughout the world. Both ship characteristics are located in Table 3 (unknown data marked as UNK).

Both classes of ships can perform ocean-going towing services such as tow ships, barges and targets for gunnery exercises. The ARS class towing system incorporates a double drum, automatic towing winch and traction winch. The two main drums hold 3,000 feet of 2.25-inch wire rope. The deck area aft has a cap rail and two tow bows can be installed. Retractable stern and side tow rollers are also available on the ARS class. The bollard pull of 65.5 tons is adequate to pull a NIMITZ class aircraft carrier [27]. The primary mission for the ATF class is towing; it has a bollard pull of 75 tons. Its towing system is a SMATCO holding 2,500 feet of 2.25-inch wire rope. It also has a 15-inch Lake Shore Inc. traction winch [28].

	ARS-50 class	ATF-166 class
Hull		
Length, Overall (ft)	255	226
Breadth, molded maximum (ft)	52	42
Depth, molded, at side to main deck amidships (ft)	23	UNK
Height, maximum to DWL (ft)	115	UNK
Displacement, light (approximate without margin) (Tons)	2160	1387
Displacement, full load (approximate without margin) (Tons)	3282	2260
Draft, full load (ft)	17.5	15.5
Machinery		
Number of propulsion shafts	2	2
Design full power ahead, shaft horsepower (hp)	4,200	7,200
Endurance at 8 knots (ARS,13 knots (ATF))(in miles)	8000	10000
Speed, 100 percent power, free route (knots)	14.5	14.5
Bowthruster (hp)	500	300
Number of diesel generators, 60 Hz 750kW, 450v, 3phase	3	UNK
Accommodations		
Ship's crew	26 civ., 4 mil.	16 civ., 4 mil.
Transients		
Provisions and Stores		
Dry (days)	75	UNK
Chilled (days)	30	UNK
Frozen (days)	75	UNK
Liquid Load		
Fuel (gallons)	123,492	UNK
Potable water (gallons)	30,745	UNK
Ballast (gallons)	87,847	UNK
Bollard pull (tons-force)	65.5	75

Table 3. Ships characteristics From[26],[28]

Figure 14. T-ARS Safeguard class. From [24]

The main mission of the ARS class is diving and salvage. The diver's life-support system is integrated into the ship and is capable of supporting up to six divers through its consoles. It also has a fixed decompression chamber. The system has primary 300psi medium pressure air and 3,000psi high pressure secondary air. At the forward part of the fantail, there are diver's davits for lowering and raising a diver's stage. Additionally, a 300psi tunneling manifold is available on the port side of the fantail for tunneling under large objects on the ocean floor [26]. A detachment of MDSU sailors augment the crew for deployment and emergent salvage operations. MDSU sailors also supplement the ATF MSFSC crew for missions. The ATF class ship does not have any diver's life-support systems integrated; it must be brought onboard by the MDSU detachment. Although MDSU detachments have been embarking ARS class ships for only the past two years, they have successfully completed missions onboard ATF classes for more than a decade.

Off-ship firefighting can be accomplished by both classes. Both classes have three fire monitors and are capable to dispense (ARS - 1000gpm, ATF - 2200gpm) seawater or aqueous film-forming foam (AFFF) [26] [28]. Fire hoses can also be attached to manifolds when coming alongside the vessel on fire.

For heavy lifts of sunken material, the T-ARS can exert 300 tons of lifting force via stern and bow rollers [26]. The T-ATF has a maximum lift of 100 tons over stern rollers [28]. The aft boom on the T-ARS forms a compensating system with its vang and topping tackle, which allows for simultaneous control of slewing and topping, ensuring load stability up to 40 tons. The forward boom on the T-ARS also has this capability up to 7.5 tons when rigged in the aft position [26]. The T-ATF vessel is equipped with a 10-ton capacity crane [29]. The T-ARS has an installed de-beaching capability. A pull of 160 tons can be exerted on a stranded vessel utilizing tow wires, propulsion engines, and two legs of beach gear. Additionally, a retraction force of up to 360 tons can be rigged with six legs of de-beaching gear [26].

E. MOTIVATION FOR TOWING AND SALVAGE MODULE

Both classes are coming to the end of their service life of 40 years; to allow for the budgeting process, plans to recapitalize the MSFSC fleet of ocean-going tugs and salvage ships must begin soon. The course of action described in this thesis is to architect the capabilities (and future capabilities) of these vessels into a single hull design. This course of action is to save Ship New Construction (SCN) funds and have the capabilities of both vessels on one platform. Although many factions of architecture (such as hull, propulsion, electrical loading, etc) for ship design are needed, this thesis will cover only those that are in the view of achieving the capabilities for towing and salvage.

The USFF analysis suggested the combined warfighting and peacetime requirements of nine vessels to accomplish both towing and salvage tasks; an acceptable risk would be eight. CNA analysis suggests seven ships, augmenting with contracted commercial tows as needed. Currently, the warfighting requirement dominates peacetime demand [30].

To accommodate a shipbuilding plan for eight ships of a single hull type, the assumption is made to establish a timeline to begin lead ship construction in FY16. The notion is to replace the legacy ships (both ARS and ATF) near the end of their service life (ESL) of 40 years. The breakdown of replacing these ships by hull number is shown in Table 4. Pre-construction activities should begin in FY10, starting with establishing an Integrated Product Team (IPT), an acquisition strategy, and formalizing ship requirements. In following years, a notional timeline is as follows [30] :

- FY12 – Perform concept and preliminary design and program documentation
- FY13 – Develop notional design, cost estimates, and contract documentation; release RFP
- FY14 – Award competitive Phase I design contracts and oversee design phase
- FY15 – Complete design phase; execute source selection for Detail Design and Construction phase
- FY16 – Award contract for Phase II Detail Design & Construction

RECAPITALIZATION OF SHIPS IN OUTYEARS															
		FY14	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**
BATTLEFORCE INVENTORY															
	ATF	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	ARS	4	4	4	4	4	4	4	4	4	4	4	4	4	4
PROCUREMENTS															
	ATF/ARS			1		1	1	1	1	1	1	1			
DELIVERIES															
	ATF/ARS						1	1	1	1	1	1	1	1	
RETIREMENTS															
	ATF						-1	-1	-1	-1					
	ARS										-1	-1	-1	-1	
SHIP AGE BY YEAR AT RETIREMENT															
		FY14	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**
T-ATF (ESL = 40 YRS)	**HULL #**														
CATAWBA	168	34	35	36	37	38	39	40	41	42	43	44	45	46	47
NAVAJO	169	34	35	36	37	38	39	40	41	42	43	44	45	46	47
SIOUX	171	33	34	35	36	37	38	39	40	41	42	43	44	45	46
APACHE	172	33	34	35	36	37	38	39	40	41	42	43	44	45	46
T-ARS (ESL = 40 YRS)	**HULL #**														
SAFEGUARD	50	29	30	31	32	33	34	35	36	37	38	39	40	41	42
GRASP	51	29	30	31	32	33	34	35	36	37	38	39	40	41	42
SALVOR	52	28	29	30	31	32	33	34	35	36	37	38	39	40	41
GRAPPLE	53	28	29	30	31	32	33	34	35	36	37	38	39	40	41

Table 4. Recapitalization of T-ATF/ARS. Modified From [30]

F. SUMMARY

Recapitalizing the Navy's fleet with a single hull design requires an architecting approach that is mapped to the capabilities of the vessels already in service, along with future capabilities anticipated by the salvage and towing community. The architecture high-level system is already defined as being a ship with mission needs and stakeholder concerns. The architecting process synthesizes a design for acquisition of this new ship class. The explanation of the architecting process, and applying it with an architecting tool, is described in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. THE ARCHITECTING PROCESS AND APPLICATION

Architecture is a science, arising out of many other sciences, and adorned with much and varied learning: by the help of which a judgment is formed of those works which are the result of other arts.

— Vitruvius

A. INTRODUCTION

There is a great need to describe a process to ensure that the architecture, the arrangement of elements and their relationships, is well-defined and addresses the needs of the stakeholders [19]. The intention of this chapter is to define a systems architecting process, and to illustrate the manner of how a legacy system could be architected in the context of a SoS from a given set of stakeholder information — in this case, from the U.S. Navy Diving and Salvage community.

B. ARCHITECTURE METHODOLOGY

The intention of the current JCIDS process is for the architecting process to begin with a top-down approach, starting with a capability need. Currently, the DODAF defines an outline of the architecture product development through a high-level, six-step process. The first five steps consists of defining the purpose, defining a scope, identifying key architecture characteristics, determining products to build, and then building the crucial products. The sixth step is not covered in either volume of the DODAF, but it states the architecture is to be used for its intended purpose. No specific description of a methodology is included (such as object-oriented or structured analysis), or notation to be used (such as UML or SySML). DODAF VOL II: Products Descriptions provides UML diagrams as an example of a general purpose modeling language for software systems. Figure 15 shows the six-step process, for simplification without the feedback loops [19].

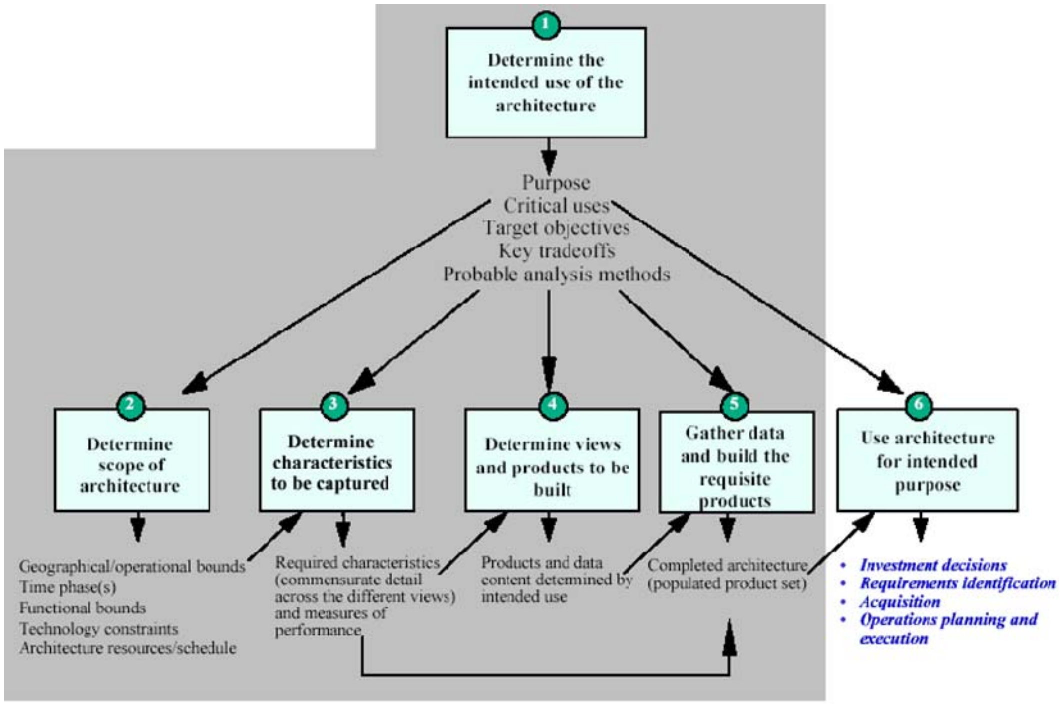

Figure 15. DODAF Six Step Process for Building an Architecture Description From[19]

These products are mapped as part of the process using SysML shown in Figure 16. The process loosely joins the diagrams and aids in verification of the system. Table 5 shows the actual products mapped; note that the blank spaces imply no mapping between the entities [31].

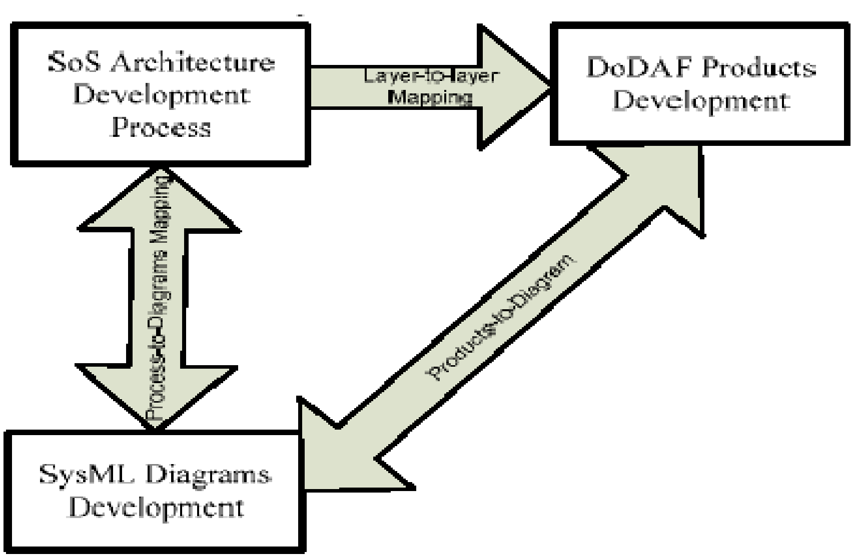

Figure 16. Mapping of DODAF Products. From [31]

SoSADP	DoDAF Product	SysML Diagrams							
		Requiremen	Structure		Behavior			EngrAnalysis (Performance)	
		Requirement Diagram	Block Diagram	Parametric Diagram	Activity Diagram	Sequence Diagram	Use Cases	Modeling & Simulation	Context Diagram
SoS Problem	Textual information (not AV-1)								
Needs Analysis									
Analyze SoS Needs	Textual information (part of AV-1)								
Develop MOEs				√					
Mission Analysis	**AV-1, OV-1**								
Determine Threats	OV-1 (High-level Operational Concept Graphic) AV-1 (Overview and Summary Information)						√		
Define Scenarios							√		√
Define Missions							√		√
Requirements Analysis	OV-6, OV-5, SV-4, SV-5								
Perform Operational Requirements Analysis	SV-4 (Systems Functionality Description)	√							
Perform Functional Analysis	SV-4 (Systems Functionality Description)	√							
Perform Non-functional Analysis		√							
Flowdown Requirements	SV-4 (Systems Functionality Description)	√						√	
Develop MOPs									
Perform Thread Analysis with Data & Messages	OV-6c (Operational Event Trace Description)				√	√			
SoS Architecture Alternatives	OV-3								
Define SoS Force Composition Options			√						√
Identify Critical Elements									
Identify Existing Systems									
Postulate Future Systems									
Perform Functional Embedding	SV-1 (Systems Interface)								
Define SoS Comm. Structures			√						√
Define SoS C2 Structures	OV-4 (Organizational Relationships Chart)		√						√
Define SoS Architecture Options	SV-1 (Systems Interface)		√						√
Define CONOPS									
Develop concepts of operations	OV-5 (Operational Activity Model)				√	√			
Develop Internal Threads with Data & Messages	SV-1 (Systems Interface)				√	√			
Cost and Risk Analysis									
Model Cost									
Identify Risk									
SoS Architecture Ranking									
Perform M&S				√				√	
Conduct Performance Analysis				√				√	
Select SoS				√				√	
Rank SoS Architecture Alternatives				√				√	

Table 5. Mapping between SoSADP, DODAF, and SySML Diagrams. From[31]

Architecture is a key aspect the systems engineering process, as the architecture is the first tangible part of the design. Systems architecting, along with systems engineering, takes high-level, capability-based need identification to define and analyze capabilities, activities, mission tasks, requirements, functions, synthesis of abstract concepts, and systems analysis of alternatives to their eventual instantiation as a physical system. The architecting methodology is part of a disciplined process that creates and documents architectures, and uses them as an integral part of the overall systems development process. It provides the framework for the description of the operational usage of the system, which in turn provides a starting point for the systems engineering product development. To define this architecture-based systems engineering process, the architecting process is first associated with the steps of classical systems engineering, especially the early phases, as shown in Table 6 [9].

Requirements Definition Process	Problem Definition and Needs Identification
	Advance System Planning
	System Feasibility Analysis
	System Operational Requirements
	Maintenance & Support Concept
	Technical Performance Parameters
	Functional Analysis & Allocation
	System Trade Off Analysis
	System Specification
	Conceptual Design Review

Table 6. Early Stage Systems Engineering Steps, Based on [32]

Requirements analysis, functional analysis, and allocation are accomplished to achieve specific capabilities based on needs. The resulting functional architecture should be a fairly stable view of the system. Design synthesis is the creation of sets of components as integrated concepts and their respective mapping from the functional architecture. System analysis is then performed to determine how to use the trade space to balance between alternative concept designs combinations of legacy and innovative new components to create an elegant, enduring, and useful physical architecture. System specifications are then used as the basis for the next major phase, the realization of the system in final form. Next, the process is expanded and categorized into a series of aspects associated with each phase: architecting and then engineering, as seen in Table 7.

Systems Architecting	Operational Need	Identify pressing need or emerging opportunity as a capability gap Develop concept of operations including existing and planned systems Define operational activities Develop operational nodes and elements Develop operational scenarios Define operational requirements
	Architecture Synthesis	Develop activity models Create functional architecture (hierarchy and flow) Define and characterize functional behavior Run simulations Characterize architecture specifications
Systems Engineering	Requirements Definition & Analysis	Develop "engineerable" requirement specifications
	Functional Analysis of Functional and Non-Functional Requirements & Allocation to Derived Elements	Create derived functional architecture (hierarchy and flow) Derive functional and non-functional requirements Develop system elements Allocate functions Model and simulate behaviors
	Design Synthesis	Create physical architecture (structure) Model and simulate feasibility "physics" Trade off alternatives Select "best" from Pareto optimal set Define system/subsystem specifications
	Test and Evaluation	Continuously validate development

Table 7. Systems Architecting & Engineering Activities High Level Outline. From[9]

The use of consistent terminology to describe the elements in an architecture is important. The ASN (RDA) Chief Systems Engineer has addressed the existing situation of non-specific, architecting-related terminology by standardizing the definition of Architectural Elements within a Naval Architectural Repository System (NARS). Each architecture should be created with common terms, thereby forming interoperability between them even if the architecting tools are different. Table 8 lists the architecture elements cross-referenced the different views as required in the CJCSM 3170. Further explanation of each architecture element is defined by the ASN (RDA) is in Table 9.

Designated by the DODAF as Critical

ARCHITECTURE ELEMENTS	AV 1[(2)]	AV 2[(2)]	OV 1[(2)]	OV 2[(2)]	OV 3	OV 4[(2)]	OV 5[(2)]	OV 6[(2)]	OV 7	SV 1[(2)]	SV 2	SV 3	SV 4[(2)]	SV 5[(2)]	SV 6[(2)]	SV 7	SV 8	SV 9	SV 10	SV 11	TV 1[(2)]	TV 2
Operational Nodes[(1)]		●		●	●	●		● (c)														
Operational Tasks[(1)]		●			●		●	● (b, c)						●								
Operational Activities[(1)]		●			●		●	● (b, c)						●								
Information Elements[(1)]		●			●		●		●	●	●		●		●					●		
Systems Nodes[(1)]		●								●	●				●							
Systems[(1)]		●								●	●	●			●	●	●		● (a,b,c)			
System Functions[(1)]		●											●	●	●				● (a,b,c)			
Triggers/Events[(1)]		●						● (b,c)											● (b,c)			
Performance Parameters[(1)]		●														●						
Technical Standards[(1)]		●																			●	●
Technology Areas[(1)]		●																●				●
Mission Capabilities[(1)]																						

● = Element Required by DODAF 1.5

(1) Standardized in the Naval Architecture Elements Reference Guide, https://ncee.navy.mil/
(2) Required by DODI 4630.8, CJCSM 3170.01C or CJCSI 6212.01D (excluding OV-6a/b)

Table 8. Architecture Usage Products. From [33]

Architecture Element	Definition
System Functions	Data transform supporting automation of activities or information element exchanges
Operational Tasks	Discrete unit of work enabling a mission to be accomplished
Technical Standards	Provides for a common set of interface and technical build-to parameters
Operational Activities	Action or process performed to accomplish an Operational Task
Information Elements	Identification of information to be passed between and among forces, organizations, or administrative structures supporting ongoing activities
Operational Nodes	Represent organizations, organization types, and occupational specialties.
System Nodes	A node with the identification and allocation of resources (e.g., platforms, units, facilities, and locations) required to implement specific roles and missions.
Systems	Organized assembly of resources and procedures united and regulated by interaction or interdependence to accomplish a set of specific requirements
Mission Capabilities	Possession of the means to use military force to achieve an intended effect with the battle space that can be measured
Triggers/Events	Specifies the cause that initiates a series of actions
Performance Parameters	Common set of performance parameters and their metric units
Technology Areas	Description of emerging technologies providing an operational capability. (Based on current state-of-the-art technologies and expected improvements.)

Table 9. Architecture Elements. From [34]

The definition of element relationships is key to defining the architecture. Table 10 shows the correct attribute needed to connect element classes to each other in CORE. The column of notes will help guide the architect to which attribute is needed. An element can use various attributes, but CORE is designed to allow only the correct target class to be selected, assisting in keeping the architecture in its proper organization.

Element Class	**Attribute**	**Targeted Class**	**Note**
Highest Level Objective			
Architecture	*achieves*	Capability	Highest level accomplishment objective
Who Said What We Need?			
Documents	*document*	Guidance	Top level authority describes a need in terms of capabilities.
Operational Domain			
Guidance	*defines*	Capability	
Capability	*achieved by*	Operational Activity	Operational Activity stated as operational scenario. Context scoped by and described based on CONOPS.
Operational Activity	*achieves*	Mission	
Mission	*achieved by*	Operational Task	Operational Tasks based on *e.g.* UJTL; Mission based on *e.g.* METL
Operational Task	*results in*	Requirement	Operational Task results in Operational Requirement, which ties Operational Domain to System Domain
Operational Node	*assigned to*	Organization	
Operational Node	*performs*	Operational Activity	
System Domain			
Operational Task	*implemented by*	Function	For “Top Down” development situation where capabilities stated first, as in unprecedented systems, function derived from tasks that are used to accomplish mission. Ties Operational Domain to System Domain.

Element Class	Attribute	Targeted Class	Note
Function	*based on*	Requirement	For unprecedented systems, Operational Requirements are determined first. At highest level, Operational Requirements lead to System Requirements, both functional and non-functional, which provides the basis for Function. For “Middle Out” or “Bottom Up” development situation, where requirements stated explicitly in Guidance, or legacy systems exist, system requirements may already available, and can be input directly from Guidance
Function	*performed by*	Component	Component is part of physical system solution.
Other Element Relationships			
Component	*consumes*	Resource	System resource (fuel, data storage, *etc*)
Function	*triggered by*	Item	Items are what is I/O through links
Requirement	*specifies*	Component	Information for engineers to use for system design.
Component	*implements*	Operational Node	Ties System Domain to Operational Domain.

Table 10. Architecting Relationships.

C. ORGANIZATION OF DODAF FRAMEWORK PRODUCTS

The DoD Systems Acquisition process requires documents such as the ICD, CDD, CPD, and CRD, which use architectural products to support decision making. These products are created in an iterative manner, and the level of detail increases throughout the development life cycle. For instance, the first generation of models consists of the highest level of the SoS. As the process continues, more detailed graphical textual products are generated. Figure 17 shows an example of some products produced and how they align to the different views supported. As the architecture is developed, elements must be able to map back to the products throughout each cycle or iteration of analysis and decomposition. A tool to assist in creating the element relationships, CORE, has been designed by Vitech Corporation, an industry provider of Model Based Systems Engineering (MBSE) tools and solutions.

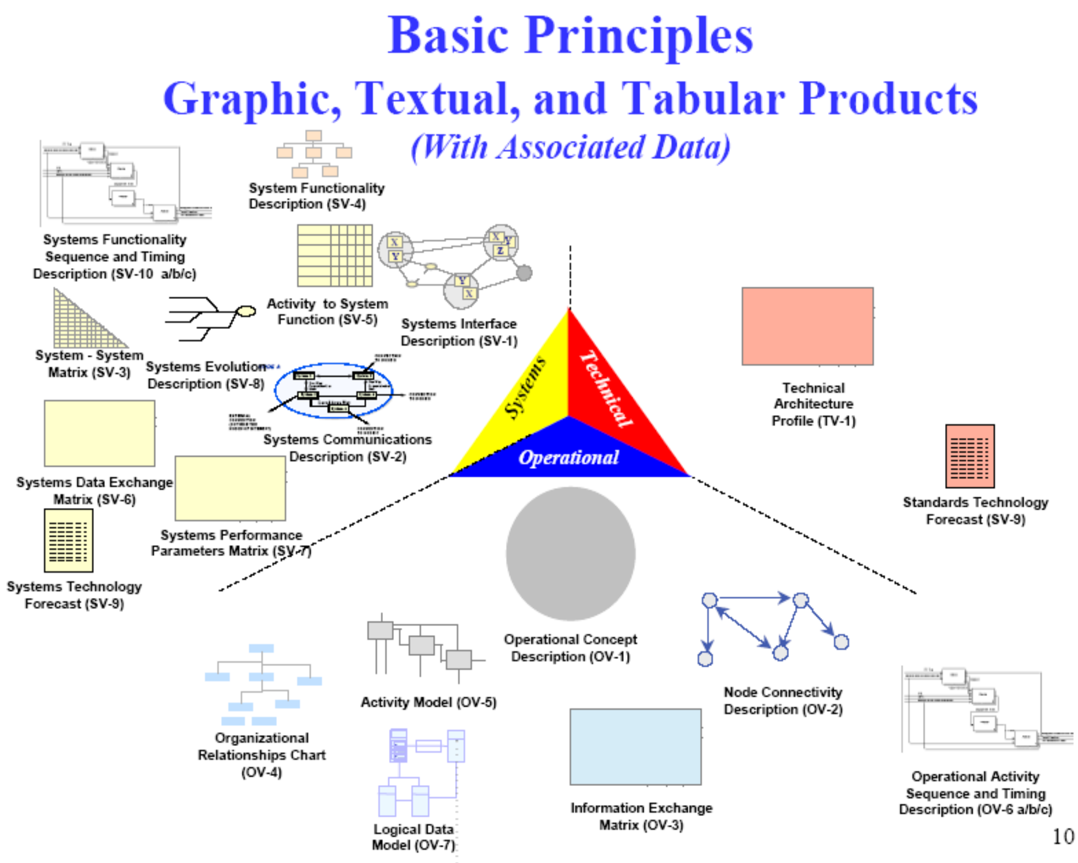

Figure 17. Illustration of Products to each view. From [34]

D. CORE ARCHITECTURE DEVELOPMENT

CORE focuses on an architecture synthesis centric approach rather than a view or document centric approach. This provides traceability from capability through requirements and analysis to testing. The CORE software suite was designed by systems engineers to satisfy diverse civilian and military customer (or stakeholder) needs.

CORE is built around a central integrated design repository. It includes a comprehensive behavior modeling notation to understand the dynamics of a design. CORE also has the ability to perform product simulation [35].

CORE is a MBSE tool designed to integrate architectural and engineering activities while developing operational and system models. Documentation, such as the DODAF views, are derived from the basis architecture produced. The architecture is created based on a DODAF schema, Figure 18, which explicitly shows the attributes and relationships established. The schema has two distinct domains; Operational and System. The Operational Architecture Domain captures originating concepts, capabilities, and supporting operational analysis to exploit, whereas the System Architecture Domain expresses the requirements, functions, and components comprising the physical design [36].

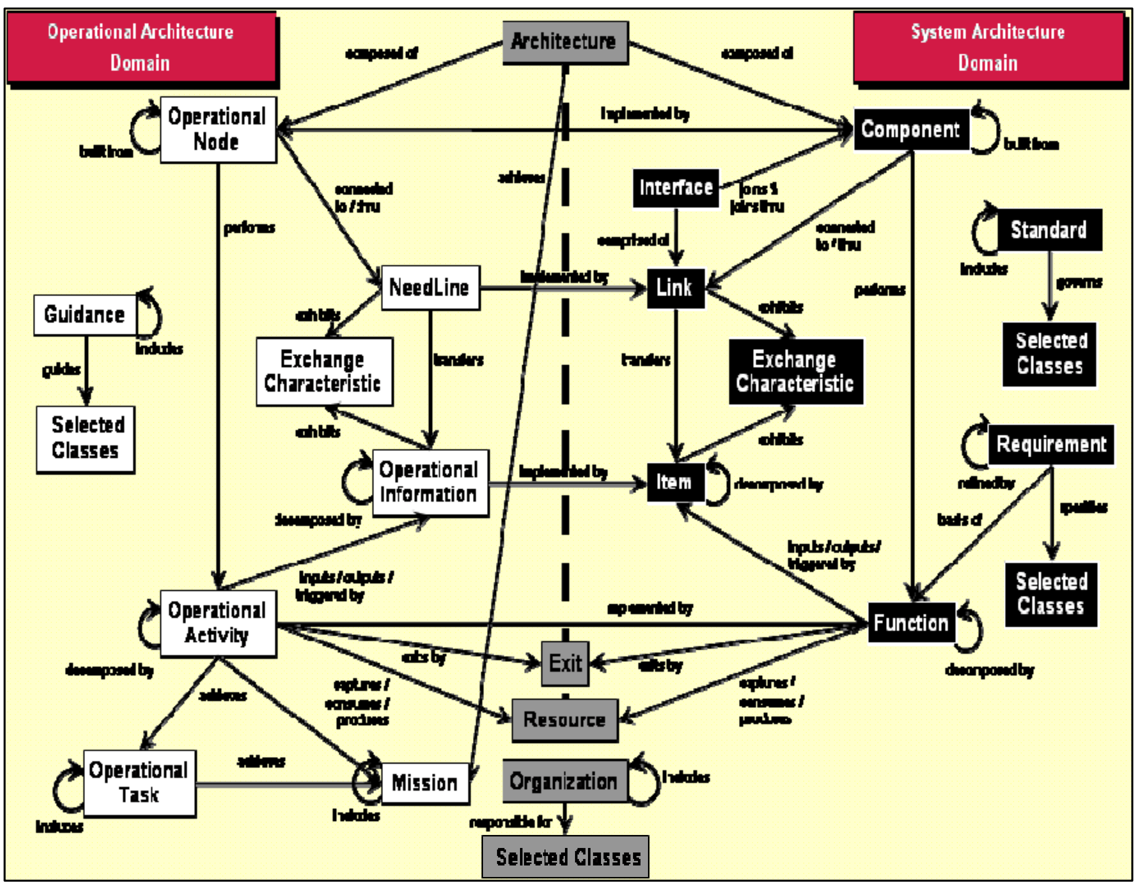

Figure 18. CORE's DODAF (v1.5) Schema. Modified From [37]

E. CORE'S OPERATIONAL AND SYSTEM ARCHITECTURAL DOMAIN RELATIONSHIPS

VITECH has implemented the architecture elements in a DODAF v1.5 schema to populate a database of element classes to enable the architect to define relationships, list a description, and show hierarchies and various other characteristics. The two domains, Operational and System, are portioned such that the architecture is defined considering two views that are linked. In CORE, architecture is composed of Operational Nodes and Components. Operational Nodes identify the operational environment or external aspects of the operational domain and components are the physical system elements that map to the operational nodes. The Operational Nodes are implemented by the components, and, conversely, the components implement the operational nodes. In addition to this mapping between operational and system domains, there are three other major mappings, summarized in Table 11 [36].

Operational Node	Component
Need Line	Link
Operational Information	Item
Operational Activity	Function

Table 11. Operational to Systems Traceability Based on[36]

F. T-ARS(X) ARCHITECTURE DEVELOPMENT EXAMPLE

Just as in the case for general architectures, there is no particular beginning entry point necessary to start the architecting process. The architecting process for this example has a large portion of legacy architecture that essentially acts as context for the architecting of the desired new towing capability. The process for systems architecture development then is initiated using a “Middle Out” method, as some legacy system characteristics are known. In a “Middle Out” start, the architecture model is synthesized beginning with the legacy system’s known characteristics and requirements. The element relationships form the basis of the architecture structure, and, since their formation can be accomplished in just about any order, this still results in a complete set of element relationships. In this case of “Middle Out,” as the requirements are iterated, mapping back to capabilities, capability gaps can be exploited to resolve legacy system faults as the architecture becomes more fully developed. The top-down aspect still must exist as mapped to the JCA and ROC, as needed, since the highest level context must still be included. The system architecture model has the opportunity to expand to accept new capabilities, should they arise, upon further study. Each iteration will bring the system closer to a design to carry on to the solution domain for further systems engineering. The documentation for the operational capability needs for this does not exist as a set of JCIDS products. Nonetheless, the architecture products must be developed. The starting point for a system acquisition, once the JCIDS process is complete, is the ICD that will be used as a starting point for this example. Mandatory Appendices for an Initial Capabilities Document (ICD) [4] require only one integrated architecture product — the OV-1. This view illustrates the intended use of the architecture for a quick, high-level description. The value added is a summary level description of organizations and/or roles, mission, and context for the architecture. It will highlight main operational nodes and provide a description between the architecture, the actors, and the environment. CORE does not provide a function to produce an OV-1; instead, a textual description is imported into the program for linking products. Figure 19 is an example of an OV-1 for the T-ARS(X) developed in Microsoft PowerPoint. CORE is utilized to develop examples of Operational and System Views mandatory for the Capability Development Document (CDD). The OV-1 in Figure 19 is translated into CORE to begin the architecting process.

Figure 19. T-ARS(X) OV-1. Modified from [38]

G. TOWING AND SALVAGE ARCHITECTURE INITIATION

The future Towing and Salvage Platform Structure architecture is envisioned to be a single-hull replacement for the present T-ARS and T-ARS class ships, making the top-level components of an architecture comprised of legacy systems already fulfilling some capabilities. Strictly speaking, the architecting process does not start by defining the form of the future system as a ship, aircraft, or any other vehicle; however, the end item here has already been named a T-ARS(X) class of ship.

In general, system physical hierarchical relationships can be developed in any predefined way, one of which is shown in Figure 20, which depicts a generic representation of a hierarchy of elements within a system. Each physical element functions to implement a task to achieve a mission. Note: Although CORE uses the term "component," this particular element can represent any of the physical elements in the example hierarchy.

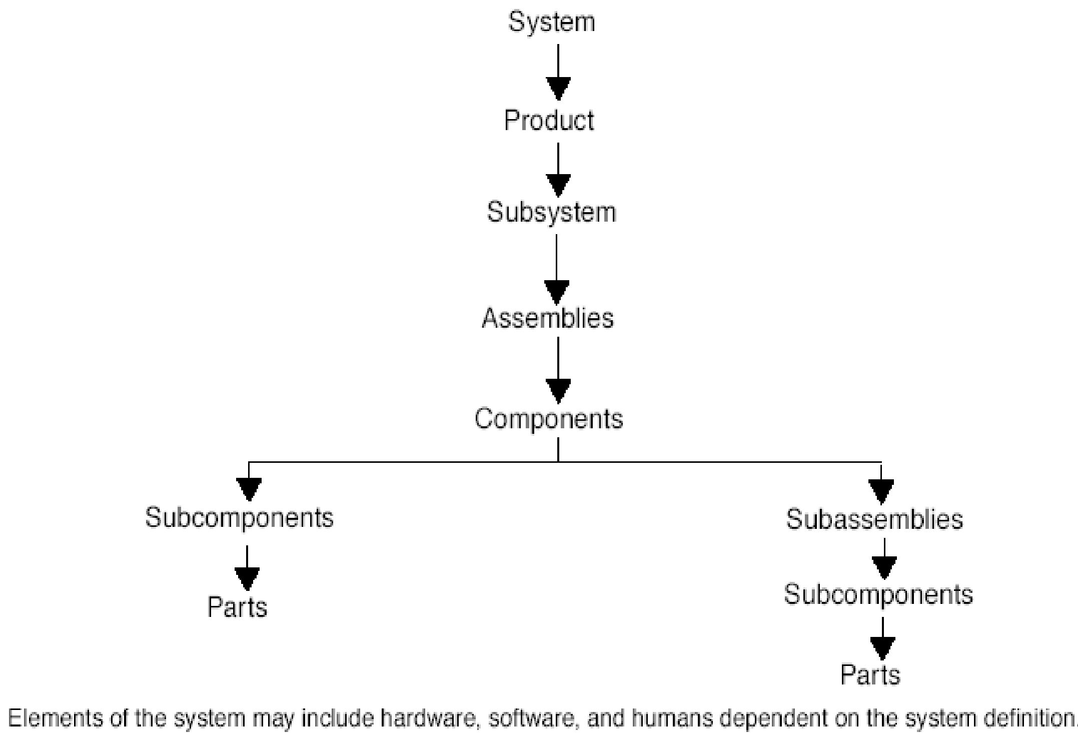

Figure 20. Hierarchy of elements within a system From [39]

The initialization of the architecting process in this case starts at the Component element class and how each attribute is exhibited by Performance characteristics identified by SUPSALV. The Expanded Ship Work Breakdown Structure (ESWBS) (Figure 21) will serve the basis for accounting for a complete ship decomposition hierarchy, since this is the standard physical element language used by naval architecture for combatant ship design.

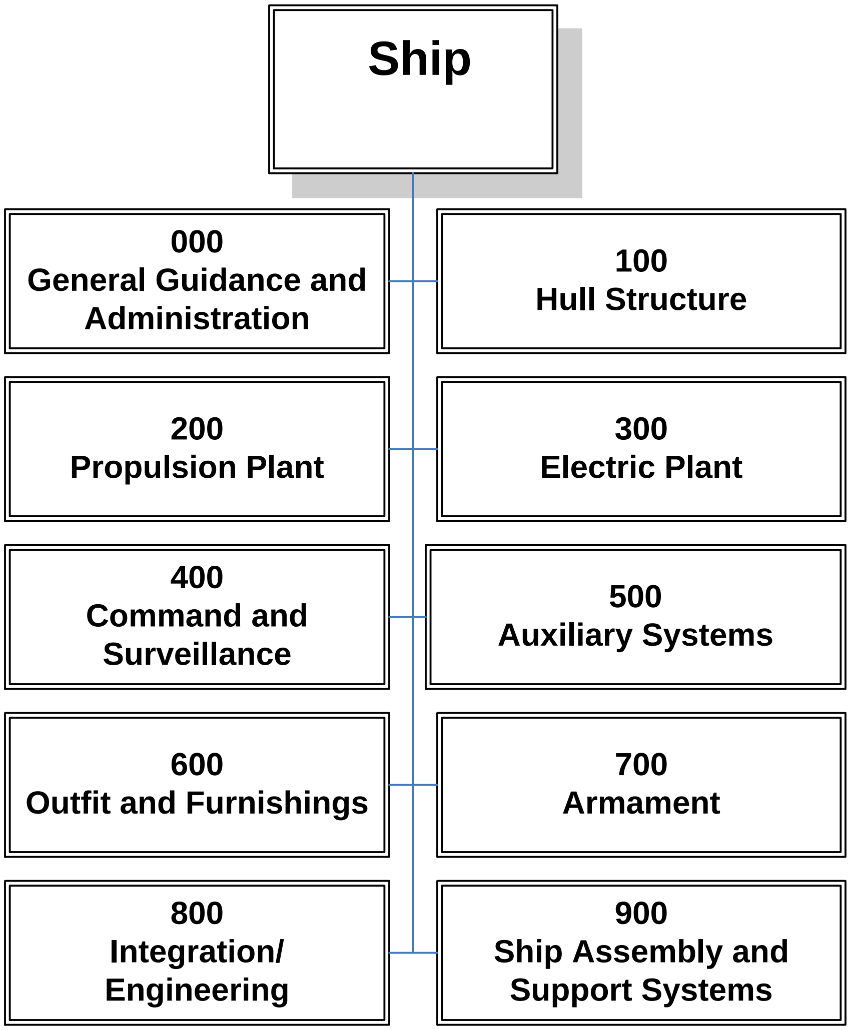

Figure 21. Expanded Ship Work Breakdown Structure

The ESWBS is based on the Ship Work Breakdown Structure (SWBS) to define and categorize to boundaries in ship's systems and SoS. The expanded level of detail is designed to enhance preliminary and detail designs for a baseline. As further development continues, the ESWBS system becomes the first five digits in the Functional Group Code (FGC). The FGC is provided to the shipbuilder as a functionally oriented breakdown sequence of a system. The ESWBS Structure shown in Figure 21 is the ten major SWBS sub-groupings serving as the second level of component class for this architecture. Subsequently, the groups provide other functions as well. The groups, as a whole, classify total weight-related cost for ship Condition A (Light ship without margin). Groups 100 through 700 equate to hardware cost and weight. Combining Groups 100-700 with 800 and 900 will equal construction cost [40].

Each component is entered into CORE by clicking on the Component file and entering it in the pop up New Component window. Component element Relationships are developed by double clicking on the element in the element window and populating the asPropertySheet window that pops up. The Relationships window portion lists each relationship that can be identified with the element selected. Each of the ESWBS groups mentioned was identified, in addition to a few of the elements (as defined by [40]) in CORE to begin the first iteration in the architecture. After all elements were identified in the asPropertySheet, the asER (Element Relationship) illustrate the relationships of the component element highlighted (Figure 22).

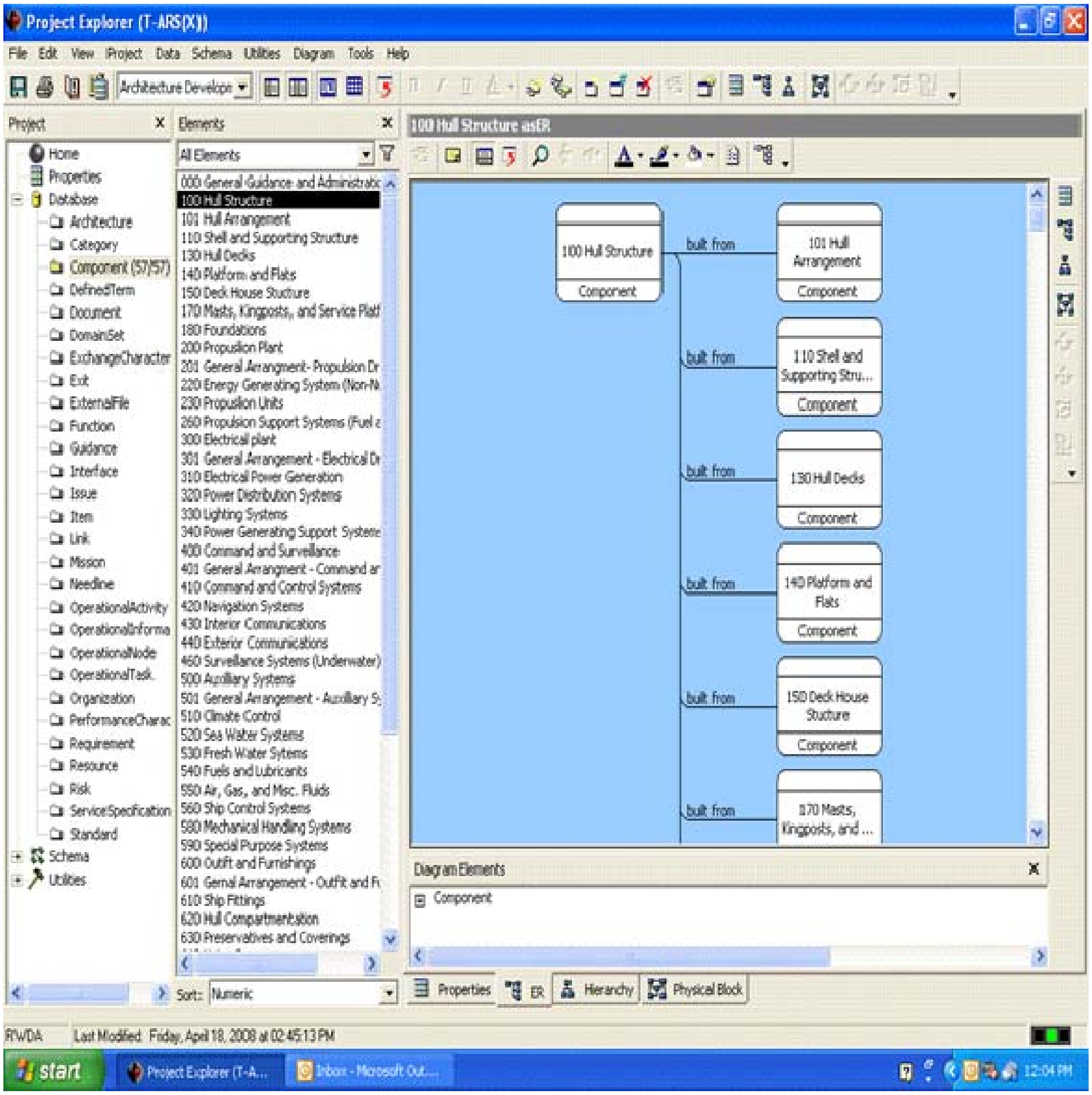

Figure 22. CORE snapshot of Component construction

The next step performed was to enter the Requirements defined by SUPSALV [41]. They are listed in Tables 12 and 13. Each performance characteristic (as a requirement) is entered in the same manner as the Component elements were defined. These elements are linked to the Component elements via the asPropertySheet in the linking them as 'exhibited by' attributes. These Characteristics are not inclusive to all parameters needed to design a complete vessel; an iterative development populating each of the classes needs to be performed. To accomplish this, each performance characteristic is established as the top level requirement in Baseline 1. Following Figure 24, the spiral development for ship design (skipping over those spokes not intended for Towing and Salvage) will designate the hierarchy of these requirements. After a series of requirements decompositions are performed, the actual level of the specified requirements will be identified.

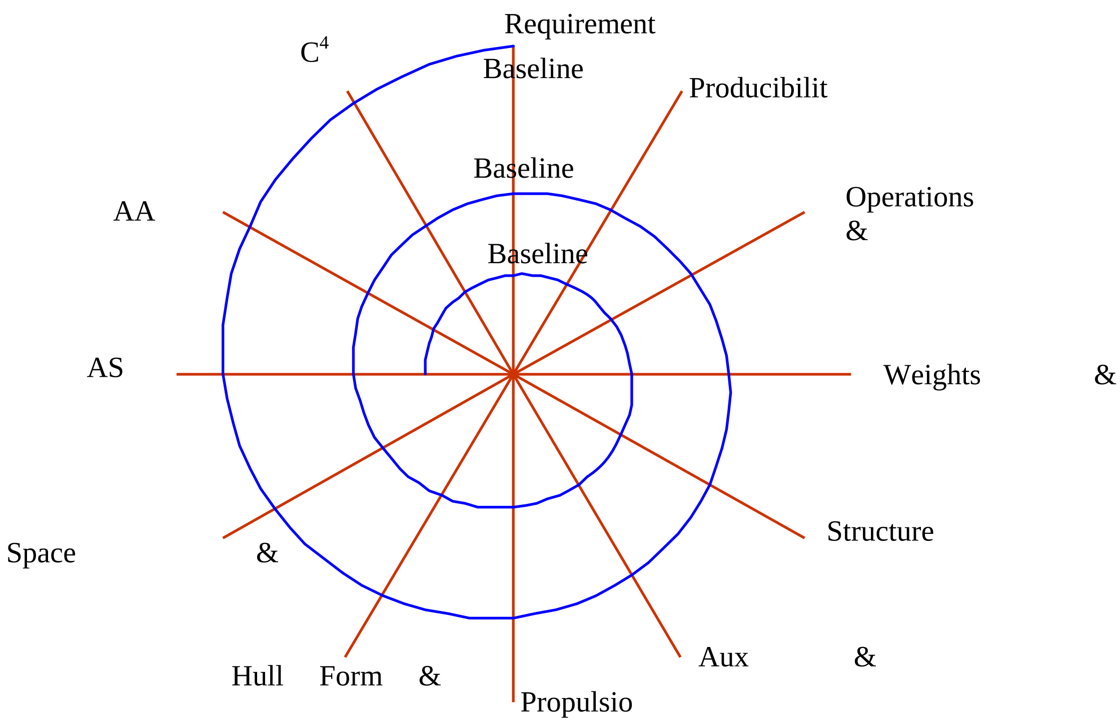

Figure 23. Iterative nature of ship design From [42]

CHARACTERISTIC	(THRESHOLD/OBJECTIVE)
Speed (knots sustained)	15/20
Bollard Pull (tonnes)	150; T=O
Navy Personnel Accommodation	42; T=O
Civilian Crew Accommodation	15; T=O
Positioning/Mooring	DP-2/DP-3. + 4 Point Moor; T=O
Endurance	8,000 nm @ 8 kt/12,000 @10 kt
Unobstructed Deck Space; AFT	3600 ft^2; T=O
Crane; Lift Capacity Min. (Amidships/Amidships + Stern)	110 Tonnes SWL
Crane; FWD	Min. 10 Tonnes SWL; T=O
Towing	- Twin Drum - 3” wire x 3,500 ft - Traction Winch - Shark Jaws - Auto-tow Pins - Portable Tow-bow
R & A Firefighting	4 monitors @ 10K + GPM Min. Each; AFFF Capable; T=O
Interoperability	Deck Loading and Ship Service Support For: - FMGS - Deep Ocean Search and Recovery - SAT-FADS - SRDRS (RCS only/TUP with ADS) - Submarine Salvage Support

Table 12. Primary Performance Characteristics. From [41]

CHARACTERISTIC	(THRESHOLD/OBJECTIVE)
Ice Classification	A1; T=O
Unobstructed Deck Space; Fwd	720 ft^2 ; T=O
Bow/Stern Roller System	Heavy-lift and Beach Gear Capable/Interoperable
Propulsion System	"Big/Little Brother" Arrangement
SNDL Recompression Chamber	Modular/Fixed
Diver Support Boat (distinct from lifeboat)	One RHIB (7m/11M)
Compressed Air	Non-DLSS; to meet ship service support function
Salvage Equipment Stowage (Below Decks)	3000 ft^2; T=O
Fabrication	- Machine Shop - Wood Shop
Deck Fastener System	TBD (ISO container; Tie-down; and/or Baxter Bolt)
Line/Wire Stowage	To Towing Mission
Portable Bulwarks	Bow & Stern
Communications	T-ARS 50 Functionality
Environmental	- Ambient Air (-20°F to 130°F) - Sea Water (28°F to 100°F)
Survivability	Commercial Salvage Standards (e.g., ABS classification)

Table 13. Secondary Performance Characteristics. From [41]

An architecture element class named "Aircraft Salvage in less than 180 fsw" was generated with the list of requirements. Each element class was populated to identify the requirements given to the components (listed from the ESWBS) and establish a Baseline 1. This entailed connecting the architecture to an OperationalNode by the 'composed of' attribute. OperationalNodes are considered as the actors in performing OperationalActivity, and are illustrated in the OV-1 as 00C, MDSU, MSFSC, and ESSM. The elements were left at a high level, but should be further explained as lower levels — such as 00C1, MDSU-2, and ESSM Port Hueneme — to fully gain context of the architecture. These lower level elements are considered ‘children’ and would have the relationship of ‘built from.’ In CORE, OperationalActivities represent Capabilities and are behavior entities. For example, Execute Salvage Operations capability will have 00C, MDSU, and MSFSC as OperationalNodes and achieve Salvage Aircraft Mission element. Figure 24 shows a rough skeleton schema for “Aircraft Salvage in less than 180 fsw.” A diagram such as this could serve as a starting point for discussion on how one requirement is needed to perform a capability. This could be traced back to a policy or procedure listed in the Guidance element class. One example would be a source such as Mission Essential Task List (METL).

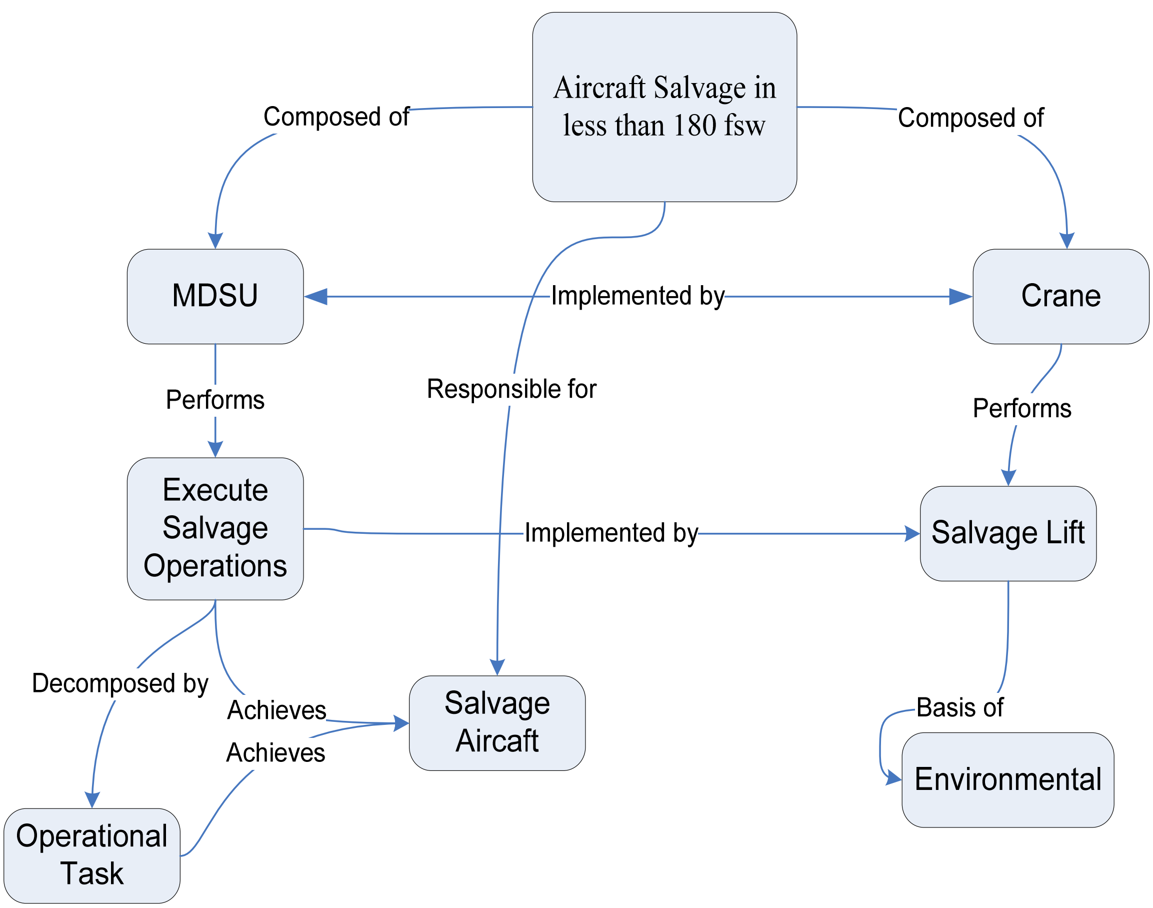

Figure 24. Rough Skeleton Schema for Aircraft Salvage in less than 180 fsw.

H. HIERARCHICAL AND ELEMENT ARCHITECTURE RELATIONSHIPS

A hierarchical relationship of any element can be composed within CORE easily by configuring them in the properties tab. Figure 25 shows a small piece of the Auxiliary Systems breakdown to find cranes within the ESWBS. The benefit to this is the naval architect who will be using the products from CORE as a basis for the next level of decomposition. As each function is decomposed to lower levels, the lower level functions can be allocated to the lower level functions. Figure 26 illustrates how the lower level functions Salvage Lift and AFT Deck Diving Operations are performed by the Component Cranes. The Element Relationships aid in showing how each element class is connected to each other and equips the architect in linking ideal relationships. This helps in giving a better visualization of the architecture and mapping. For example, Figure 27 maps requirements to the function (Salvage Lift) for a better explanation of how they relate to the capabilities of the architecture. Correct decomposition and element relationship mapping will greatly influence tradeoffs in further discussions. The decision maker has a better perspective of how the system is developed and can be mapped back to a capability need.

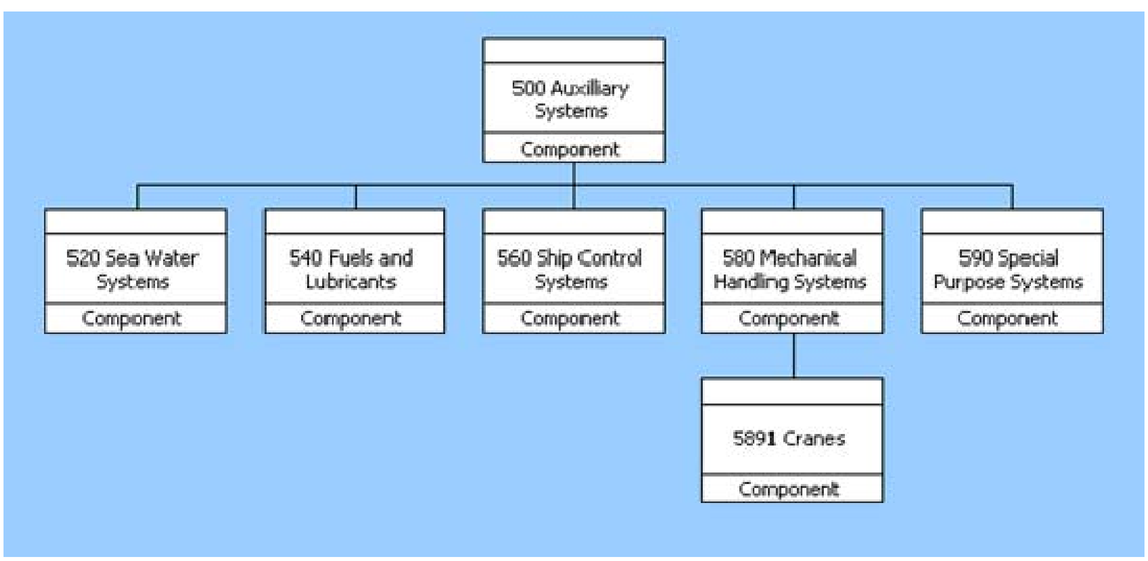

Figure 25. Component Hierarchy from ESWBS as seen in CORE.

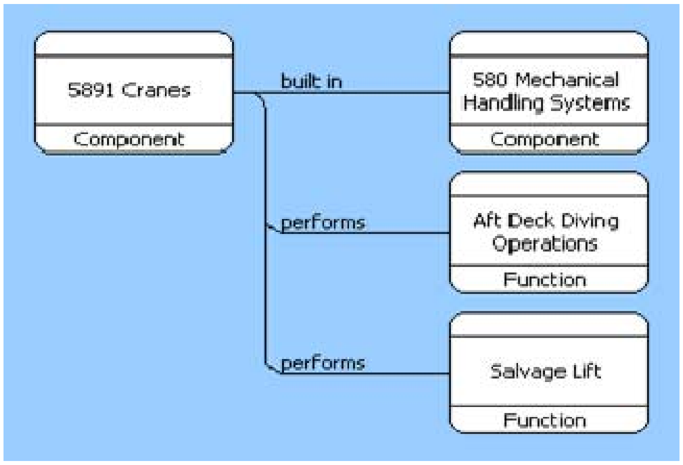

Figure 26. Element Relationship in CORE for Cranes

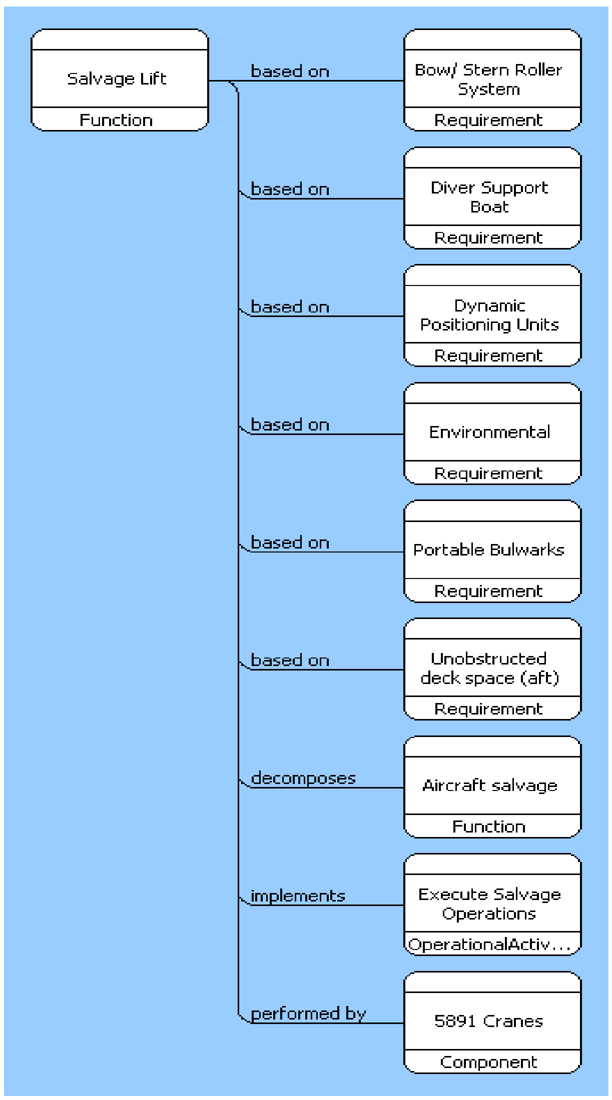

Figure 27. Element Relationship for Salvage Lift function.

I. DODAF 1.5 VIEWS FROM CORE

CORE can document the architecture product as a Rich Text Format (RTF), via what they call "scripts." Each script generates a standard DODAF diagram, a table of the repository contents, or and external file referenced by an ExternalFile element. The DoDAF v1.5 view scripts are designed to be flexible in order to support any later iteration completed by further architecting processes to produce views for the customer. Each time the script is run, it is checked for errors, ensuring traceability [35].

1. The OV-5 Activity Model for Execute Salvage Operations

One script produced by the rough architecture skeleton of "Aircraft Salvage in less than 180 fsw" is the OV-5 activity model shown in Figure 28. The "Execute Salvage Operations Hierarchy Diagram" illustrates this 'children' OperationalActivities of the user selected OperationalNode(s). This activity model graphically structures the activities in a probable activity hierarchy to enable the architect to understand the level at which function is needed.

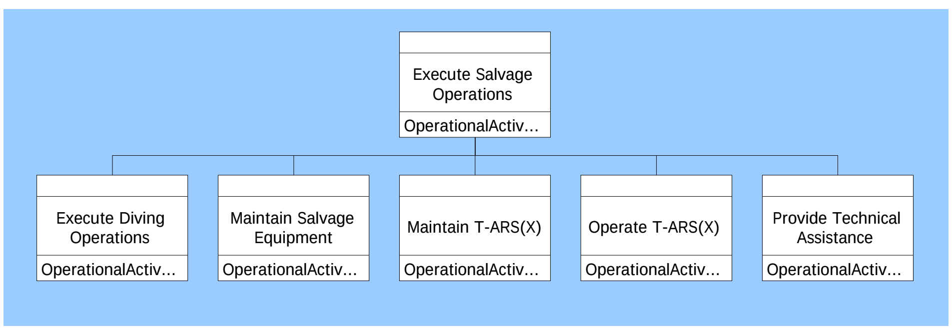

Figure 28. Execute Salvage Operations Hierarchy Diagram

Figure 29 is the IDEF0 diagram depicting which OperationalNodes perform OperationalActivity. Note: There is overlap within the activities, which are intended to demonstrate those actions performed by humans. The term 'function' refers to those actions performed by systems.

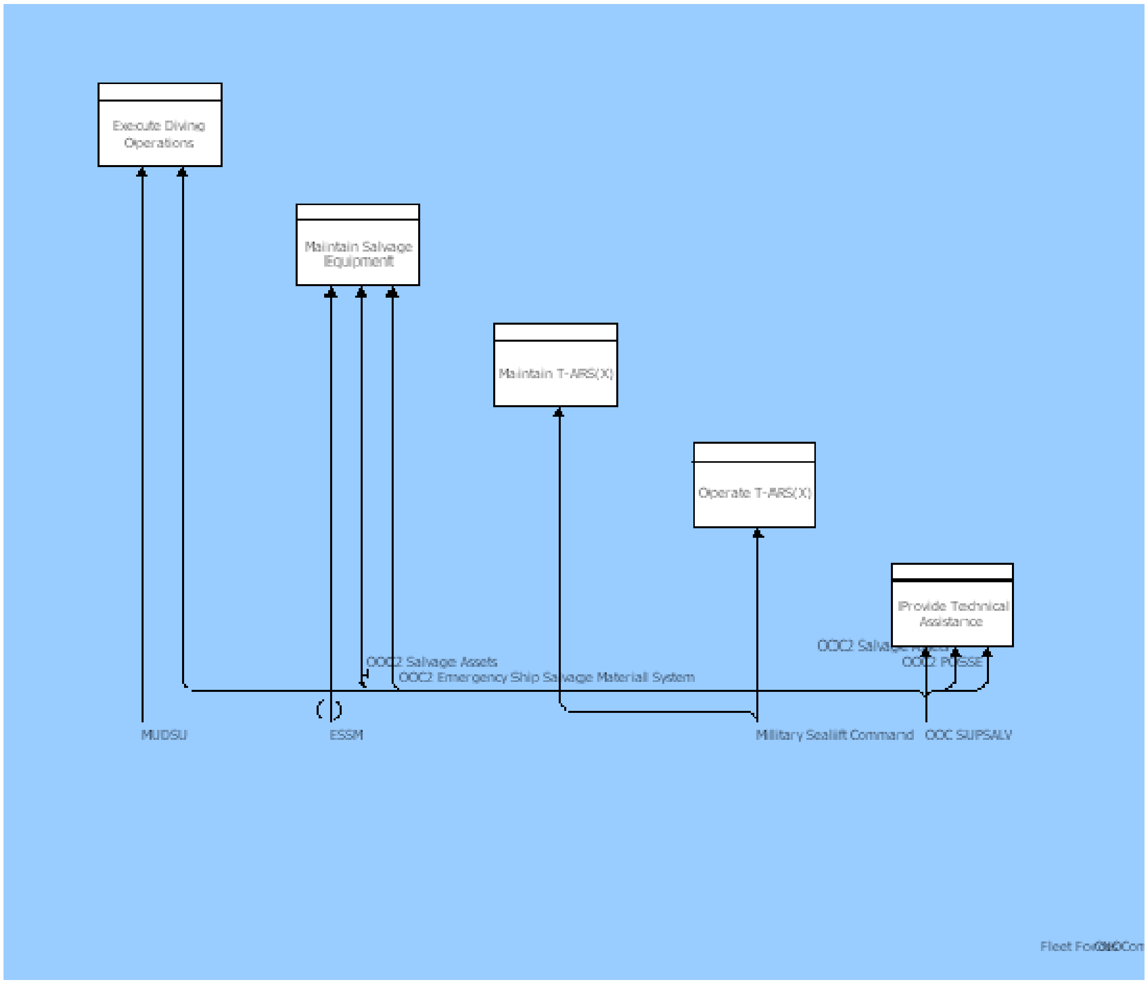

Figure 29. Execute Salvage Operations IDEF0 Diagram

2. The SV-4 Activity Model for Aircraft Salvage

The SV-4 view documents the user-selected function and its children. Figure 30 is the Aircraft Salvage Hierarchy diagram. Figure 31 goes on to show the IDEF0 diagram; connecting the components to the functions within 'Aircraft Salvage.' A SV-4 view is the OV-5 for systems; documenting system data flows between functions. Adequate functional decomposition within these views will ensure sufficient level of detail for system design.

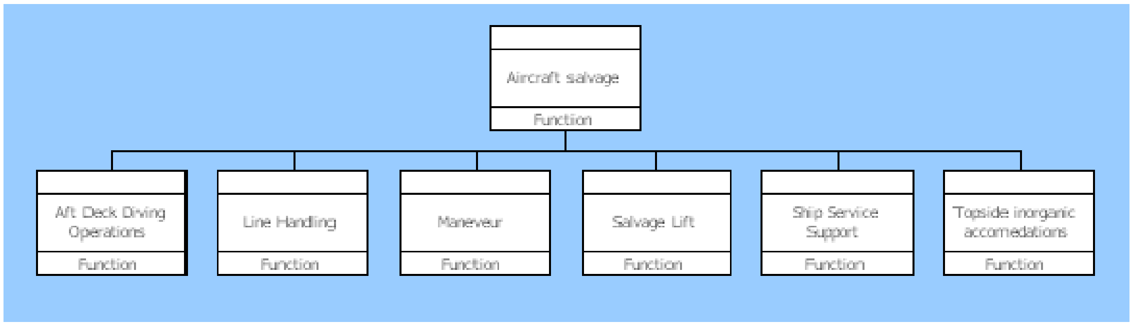

Figure 30. Aircraft Salvage Hierarchy Diagram

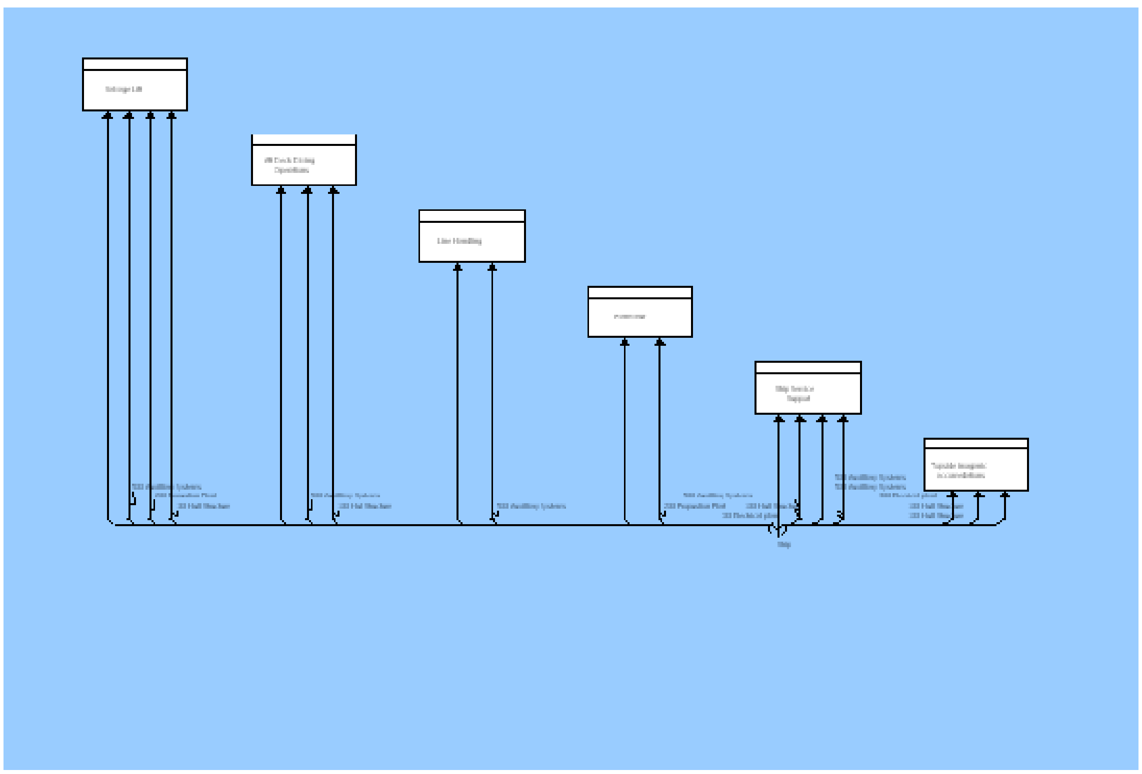

Figure 31. Aircraft salvage IDEF0 Diagram

J. SUMMARY

This chapter outlined a systems architecting process, and illustrated the manner of how a legacy system is architected in the context of a SoS from a given set of stakeholder information — in this case, from the U.S. Navy Diving and Salvage community. The process is integrated with the front end of the traditional systems engineering process. The architecting process is partitioned from the systems engineering process, primarily due to the differences in the approach to the problem, which, in the architecting stage, requires inductive reasoning, abstract conceptual thought, and a way of dealing with stakeholder needs that do not play to the typical analytical strengths of engineers. The process described is deployed in the context of the integrated architecture development philosophy in the DoD JCIDS process, and the DODAF.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

DoD has recently changed to a philosophy of developing integrated architectures to meet capability needs. The JCIDS process provides the context to identify capabilities and capability gaps to shift toward capability-based planning. The DODAF and Naval Architecture Elements Reference Guide (NAERG) provide a basis for defining standard architectures for DoD systems and SoS. As the Navy changes its approach to developing systems — from a bottom up, stove-piped design to a joint service strategic vision — a systems and SoS architecting process definition and implementation is needed.

What has been illustrated is a systems architecting approach that is traceable from military strategy to system requirements in a way that produces architecture that can be created, defined, communicated, recreated, rerun, and moved around as needed to explore the design possibility space in an interactive manner focused on stakeholder needs. A computational software tool, CORE, has been used to accomplish this. CORE is a software tool that can be used to link element relationships to give the full visualization required for stakeholders to make decisions. In CORE, this can be revised for investigation and added throughout the acquisition process. The DODAF views are generated within CORE, and Scripts are developed for system and subsystem specifications in a RTF for presentation, review, and analysis of the architecture.

The Recapitalization of a Towing and Salvage vessel was used as an example in a 'middle out' process. The beginning elements known were that the architecture was going to be a ship and a list of requirements. The architecture was iterated using the CORE DODAF 1.5 schema to produce example scripts that could be used in DODAF 1.5 views. This process provides a useful method to allow architecting of the diving and salvage fleet that can meet the capabilities needed not only by the Navy Supervisor of Diving and Salvage in an operational and strategic sense, but also in the context of the needs of the joint service architecture for accomplishing warfighting and business goals.

B. RECOMMENDATIONS

The process outlined and implemented on the diving and salvage platform provided the basis for the creation of an architecture and the related products. The development of the overall diving and salvage architecture, and the demonstration of the use of the architecture for strategic and operational decision making should be accomplished. This includes:

- Developing an architecture that is populated to the extent that elements include enough description to demonstrate the use for operational and strategic planning, including the need for trade offs for acquiring new platforms or contracting of outside resources
- Exercising the architecture in a dynamic sense to create options for planning and design
- Integrate the architecting process and respective tools (*e.g.,* CORE) with ship design tools (*e.g.,* ASSET, POSSE) in order to allow a more quantitative, physics-based analysis of ship platform development, and connect the traditional ship design process to the stakeholder need.
- Expand the process scope to core warfighting capabilities and business capabilities, such as combatant ships, aircraft, ground vehicles, and system command organizations.

Continuing the development of the architecting process will facilitate the ability to operate and plan the implementation of a Navy diving and salvage capability, but will also lead to implementations in other Navy ship and system development organizations.

LIST OF REFERENCES

[1] D. C. Winter, Secretary of the Navy, Speech at Sea Air Space Exposition, Washington, D.C., April 2007.

[2] S. K. Walker, Capability-Based Planning How it is Intended to Work and Challenges to Its Successful Implementation, U.S. Army War College, 2005.

[3] D. Rumsfeld, Secretary of Defense, "Transformation Trends-5 August Issue," available from (http://www.oft.osd.mil), May 2008.

[4] "Joint Capabilities Integration Development System" CJCSI 3170.01E (United States Department of Defense), 2005.

[5] Maier and Rechtin, *The Art of Systems Architecting*, CRC Press, 2002.

[6] Clifford Whitcomb, (May 2008) Microsoft PowerPoint Presentation, SE3100 Module 2, System Architectures and Types.

[7] International Council on Systems Engineering (INCOSE) webpage, [http://www.incose.org/practice/whatissystemseng.aspx], March 2008.

[8] Operation of The Joint Capabilities Integration and Development System, CJCSM 3170.01C, May1 2007.

[9] C. Whitcomb and R. Vaidyanathan, "Architecture Development for an Unmanned Vehicle Sentry System of Systems," 23rd Bristol International Unmanned Air Vehicle Systems (UAVS) Conference, Bristol, UK, April 2008.

[10] Policy for Systems Engineering Memorandum, Feb 20, 2004.

[11] Systems of Systems (SoS) Mission Engineering Plan (MEP) Preparation Guide, Version 0.7 (Draft), Assistant Secretary of the Navy (RDA) Chief Systems Engineer, September 2007.

[12] Department of the Navy Enterprise Architecture Hierarchy Version 0.3 (Working Draft), Assistant Secretary of the Navy (RDA) Chief Systems Engineer, March 2008.

[13] Brad Mercer, "Thoughts on Architecting, and How to Improve the Practice," Microsoft PowerPoint Presentation to the Naval Postgraduate School Systems Engineering Colloquium, *MITRE Corporation*, 2008.

[14] L. Urbaczewski and S Mrdalj, "A Comparison of Enterprise Architecture Frameworks," *Issues in Information Systems*, Vol. VII, No. 2, 2006.

[15] "Zachman Institute for Framework Advancement (ZIFA) (http://www.zifa.com)" May 2008.

[16] IEEE 1220: for Practical Systems Engineering," *Computer*, vol. 39, 2006.

[17] Preparation Guide. DoD 5000.2 webpage, [https://akss.dau.mil/dag/DoD5002/], March 2008.

[18] Department of Defense Architecture Framework, Vol. I: Definitions and Guidelines, February 9, 2004.

[19] Department of Defense Architecture Framework, Vol. II: Product Descriptions, February 9, 2004.

[20] Salvage and Recovery Program, OPNAVINST 4740.2G, Office of the Chief of Naval Operations, Washington, D.C., December 2007.

[21] Navy Salvage Operations NWP 4-12 (Draft), Naval Warfare Publication, Office of the Chief of Naval Operations, Washington, D.C., December 2007.

[22] Military Sealift Command Standard Operating Manual (SOM) (Draft), COMSCINST 3191.9B, Commander, Military Sealift Command, February 2008.

[23] U.S. Navy Salvage Engineer's Handbook, Volume 1 (Salvage Engineering), S0300-A8-HBK-101. Naval Sea Systems Command, Washington Navy Yard, D.C., July 1997.

[24] NavSource Naval History Homepage, (www.navsource.org/archives), May 2008.

[25] U.S. Navy Towing Manual, SL740-AA-MAN-010, Rev 3., Naval Sea Systems Command, Washington Naval Yard, D. C., July 2002.

[26] ARS-50 Class Operations Handbook, SS500-AM-MMO-010, Vol.1 Rev 1., Naval Sea Systems Command, Port Hueneme, California, December 2002.

[27] U.S. Army Towing Manual, TB-55-1900-232-10,Departmnet of The Army Headquarters, Washington D.C., 1991.

[28] Supervisor of Salvage NAVSEA 00C Homepage, (http://www.supsalv.org/00c2_t-atf166.ageId=2.), May 2008.

[29] Federation of American Scientists Homepage, (http://www.fas.org/man/dod-101/sys/ship/tatf-166.htm), May 2008.

[30] Christopher Light, (December 2007). Salvage Platform POM10 Shipbuilding Presentation. Microsoft PowerPoint Presentation, OPNAV N42.

[31] T.V. Huynh and J.S. Osmundson, An Integrated Systems Engineering Methodology for Analyzing Systems of Systems Architectures, Proceedings of the 1st Asia-Pacific Systems Engineering Conference, Singapore, March 2007.

[32] Blanchard and Fabrycky, *Systems Engineering and Analysis, 4th ed.* Prentice Hall, 2006.

[33] Jayson Durham, (December 2007) Microsoft PowerPoint Presentation, Naval Architecture Elements Guide. ASN (RDA) Chief Systems Engineer.

[34] Microsoft PowerPoint presentation, Mr. Truman Parmele at the DoD Architectures Conference, February 24, 2004.

[35] Microsoft PowerPoint presentation, CORE SE Guided Tour Vitech Corporation August 2007.

[36] CORE Architecture Definition Guide (DODAF v1.5), Vitech Corporation. August 2007.

[37] Microsoft PowerPoint presentation ADI user group meeting, An Overview of CORE and its Role within the NUWC Modeling and Simulation Architecture, November 29, 2007.

[38] Mike Herb, (January 2008) Microsoft PowerPoint Presentation, Salvage Doctrine Development. Director, Salvage Operations (00C2), Supervisor of Salvage and Diving Naval Sea Systems Command.

[39] *IEEE Standard for Application and Management of the Systems Engineering Process*, IEEE Std. 1220, 1999.

[40] Expanded Ship Work Breakdown Structure For All Ships/Combat Systems, S9040-AA-IDX-101/SWBS 5D, Vol.1 Naval Sea Systems Command, February 1985.

[41] Commander, Naval Sea Systems Command (SEA 00C) (21 February 2008) Proposed T-ARS(X) Top Level Characteristics. Letter. Washington, DC.: R. Hooper Supervisor of Salvage and Diving.

[42] Fotis Papoulias, (Spring 2008) Microsoft PowerPoint Presentation, TS4001: Lecture Summary 6, Ship Design and Impacts.. Naval Postgraduate School.

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

3. ATTN: CAPT Hooper, 00C
   Commander, Naval Sea Systems Command
   Washington Navy Yard, D.C.

4. ATTN: Mike Dean, 00CB
   Commander, Naval Sea Systems Command
   Washington Navy Yard, D.C

5. ATTN: Mike Herb, 00C2
   Commander, Naval Sea Systems Command
   Washington Navy Yard, D.C

6. ATTN: LT Robert Williams, 00C2OA
   Commander, Naval Sea Systems Command
   Washington Navy Yard, D.C

7. ATTN: LCDR Jay Young, 00C5
   Commander, Naval Sea Systems Command
   Washington Navy Yard, D.C