A Model for Set-Based Design at the System-of-Systems Scale with Approaches for Emergent Properties

By

Jonathan Edward Page

B.S., United States Naval Academy (2002)
S.M., Massachusetts Institute of Technology (2011)
Naval Engineer, Massachusetts Institute of Technology (2011)

Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Design Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
May 2022

Author ____
Department of Mechanical Engineering
May 10, 2022

Certified by ____
Warren Seering
Weber-Shaughness Professor of Mechanical Engineering
Thesis Supervisor

Accepted by ____
Nicolas Hadjiconstantinou
Professor of Mechanical Engineering
Graduate Officer

(This Page Intentionally Left Blank)

A Model for Set-Based Design at the System-of-Systems Scale with Approaches for Emergent Properties

By
Jonathan Edward Page

Submitted to the Department of Mechanical Engineering
On May 10, 2022, in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Design Engineering

Abstract

Large projects with long time horizons and substantial capital investments challenge designers and engineers. Inevitably, the project's large scale leads to competing requirements for the design team to balance. Further, the prolonged lifetimes of these projects create uncertainty regarding the project's value retention from delivery until decommissioning or disposal.

Set-based design (SBD) offers a solution to the first issue. The method allows designers to intelligently canvass a more extensive solution space using sets and ranges of characteristics to define the design instead of specific instances. The practice of SBD continues to grow, but few examples exist of projects at scale outside of theoretical studies.

Flexibility offers a solution to the latter issue. Designing for flexibility accounts for this uncertainty by intentionally integrating options in the architecture of a project. This practice gives future managers the right, but not the obligation, to execute these options. Unfortunately, however, flexibility is often treated as an emergent property.

This research reveals a case study of an instantiation intersecting these principles in the design of a naval vessel. It uses an action research approach to develop and execute an SBD process at a large scale that integrates specific emergent properties as sets of alternatives. It shares the philosophical basis of the process, the team structure formed, the steps of the method, and the documentation created to capture the knowledge. It contributes generalizable knowledge for consideration when a team may consider establishing their SBD methods, including starting small to test the process before growing, establishing the first sets, and managing communications.

This research contributes to the development of integration by intersection by sharing how we integrated our sets and what conditions allowed their intersection. It offers a special case of practical point design within SBD called benchmarks, which acted as virtual prototypes and created reusable design knowledge for future efforts. The research could not conclude that treating emergent properties as sets suited the SBD method, but it provides insight towards answering that question in future work, especially regarding how to form those sets so that they intersect appropriately with others.

Committee Member: Richard de Neufville
Title: Professor of Engineering Systems, MIT Institute for Data, Systems, and Society

Committee Member: Daniel Frey
Title: Professor of Mechanical Engineering

Committee Chairman and Thesis Supervisor: Warren Seering
Title: Weber-Shaughness Professor of Mechanical Engineering

Table of Contents

List of Figures

List of Tables

1. Introduction

It follows then as certain as that night succeeds the day, that without a decisive naval force we can do nothing definitive, and with it, everything honorable and glorious.

- *George Washington, 1781, in a letter to Marquis de Lafayette*

This statement is as true today as it was when President Washington wrote it, but how one measures the decisiveness of our naval force changes with time. Measures like speed, quantity, precision, lethality, survivability, range, and others have evolved. Today, the US Navy finds itself in a Strategic Competition with peer adversaries that desire to upset the existing rules-based international order (Office of the Secretary of the Navy 2020; 2021b). The Navy realizes this requires multi-pronged strategies that encompass everything from technology development to tactical training and that they must execute these strategies with speed and purpose (Office of the Secretary of the Navy 2020; 2021b; Kitchener, Cooper, and Schlise 2021; Office of the Chief of Naval Operations 2018).

Maintaining this competitive edge poses intriguing challenges. Technology advances at a blistering pace, but all ships are significant capital investments with long service lives, complicating the ability to outfit all ships with the most modern equipment. Further, it is not just our technology that advances, but that of adversaries and competitors -who get a vote in the required capabilities of our fleet. Some of those competitors have capable first-rate navies and industrial capacity beyond ours and seek to challenge existing conventions reaching far beyond their territorial seas (Commander Naval Surface Forces 2021). Therefore, since technology and requirements change before, during, and after constructing a capital ship, adapting and responding to change faster than competitors is probably better than trying to out-build them.

Additionally, the challenges facing the Navy are multiplicative and non-linear. Maintaining a competitive edge would be difficult enough if the geopolitical landscape changed quickly and the Navy responded to new threats in new locations. It would be difficult enough if technology continued to change at its current rate, and we had to maintain or exceed its pace. It would be difficult enough if laws required better environmental stewardship from our designs. It would be difficult enough if the mission requirements for the Navy from Combatant Commanders continued to grow across the spectrum from peacetime, deterrence, and power projection to hostilities, and the Navy had to do its best to fulfill them all. It would be difficult enough with the Budget Control Act, flat investment accounts, and Continuing Resolutions for over a decade. It would be difficult enough to consider that vessels tend to stay in service for 20 years and more and that the requirements and use of surface vessels will change in that time frame. However, the US Navy must address *all* these with its existing fleet of fewer than 300 ships and the fleet we are investing in today. According to the Fiscal Year 2021 shipbuilding plan[1], the Navy will add between 9 and 20 ships per year to the battle force count (Office of the Chief of Naval Operations 2020). Therefore, the Navy must endeavor with each ship to deliver the right capability for present and future needs.

Adding ships to the fleet is a complex undertaking itself. There are three complementary systems of governance functioning at the same time. The first, the Joint Capabilities Integration and Development System (JCIDS), provides the foundation for all Department of Defense (DoD) systems requirements. The second, the Defense Acquisition System (DAS), provides the framework and policies for acquiring systems and services for the DoD when the JCIDS processes result in a material solution. The third, the Planning, Programming, Budgeting, and Execution (PPBE) system, makes plans for and provides resources to all programs and operations within DoD. For complex defense systems, it takes at least ten years to navigate these systems and deliver new capabilities, and even longer for our naval vessels (Dodaro 2021).

Designing a naval vessel is also a complex undertaking. It is appropriate to think of a naval vessel as a system of systems since it manifests as the integration of hull, mechanical, electrical, communications, combat, life support, habitability, navigation, and other systems; each managerially or operationally independent but functionally codependent (Walden et al. 2015). Each of these systems interfaces with the others and is tightly coupled in the design solution. Many combat and communications systems are complex enough to be independent acquisition programs within JCIDS, DAS, and PPBE. Ship classes often warrant a bespoke design with efforts exceeding one million hours. Nevertheless, even the ship classes that are not

---

[1] Fiscal Year 2021 was the last year when a complete 30-year plan was submitted. Fiscal Year 2022 did not include a complete long-range plan while analysis was still ongoing regarding the total battle force count and associated procurement plan. However, it included justification for eight battle force ships, down from 12 in the Fiscal Year 2021 plan (Office of the Chief of Naval Operations 2021).

new or unique and borrow many characteristics from an existing ship class take just as long and require as many resources because of the complicated interdependencies of the parts of a ship. The third flight of the Arleigh Burke Class destroyer, the new Constellation Class frigate, and large deck amphibious ships present recent examples of this phenomenon (Dodaro 2021).

While the ship's design has become more complex with time, the workforce to accomplish the design tasking has diminished. The engineering workforce for the government reduced significantly during the acquisition reform efforts of the 1990s (Office of the Inspector General 2000). Where there were over a thousand mission-funded engineers for the Naval Sea Systems Command in the 1970s and 1980s for the Ticonderoga Class cruiser, the Arleigh Burke Class destroyer, and the Oliver Hazard Perry Class frigate, the reform efforts in the 1990s reduced those numbers to less than 300 (Keane, Tibbitts, and Jaquith 2019). Accordingly, the government took more of an oversight role and less of a design role for ship acquisition efforts just after the turn of the Century, such as Littoral Combat Ship and DDG 1000. These circumstances contributed to the dynamics that led to the truncation of those classes before completing their original acquisition plans (Keane, Tibbitts, and Jaquith 2019). This also contributed to the loss of engineering knowledge and lessons learned from those programs. While the engineering workforce of support contractors and shipyards took on the Navy's previously executed design roles and responsibilities, the approach was necessarily project-by-project and ship-by-ship. Learning from one class to the next happened through circumstance instead of intention. Further, the project-by-project approach did not present a steady flow of design work to these organizations, so they experienced cycles of workload that did not always support the retention of a robust engineering workforce (Ricketts 2011; Office of the Secretary of the Navy 2021c).

Despite these myriad challenges, the Navy and its acquisition workforce continue the work to deliver necessary platforms and capabilities to the fleet to conduct its enduring roles of sea control, power projection, deterrence, maritime security, and sealift in support of the rules-based international order. In 2017, the Navy conducted a Capabilities-Based Assessment of its Future Surface Combatant Force, including large surface combatants, small surface combatants, and uncrewed vessels. This analysis resulted in an approved Initial Capabilities Document (ICD) in 2018 for the combatant forces. In recognition of the current and future uncertainty germane to investments of large capital ships, the Navy designated flexibility as a top priority for the future fleet. That same year, the Navy conducted a Requirements Evaluation Team (RET) to allocate appropriate requirements from the ICD to the large surface combatant, now known as DDG(X). The Chief of Naval Operations (CNO) approved the initial parameters from that study's results and asked the community to continue challenging the requirements to better understand the cost-capability trades of the design space. He also requested completion in time to award a detail design contract within five years, by 2023 (Office of the Chief of Naval Operations 2018).

Under the circumstances, those charged with continuing the requirements and design efforts chose to use set-based methods to accomplish their task. They anticipated a need to utilize this concurrent engineering approach to manage the complexity and knowledge creation of the undertaking efficiently. They recognized they needed a different process with different toolsets to bring together a diverse national team of talent from government and industry. They knew status quo ship design methods would not adequately analyze and value architectural decisions and features that intentionally incorporated adaptability and robustness in balance with other requirements efficiently and affordably.

This dissertation captures these efforts to offer generalizable knowledge about surmounting the increasingly complex nature of product design in the 21st Century using set-based design (SBD). It also conveys the team's approach to considering flexibility, which is typically considered an emergent property of a design, just like maintainability, repairability, sustainability, and survivability (Walden et al. 2015). The central research tests how one might evaluate emergent properties, especially flexibility, in an SBD process.

2. Background

Some of the topic matter deserves a brief introduction to acquaint the reader who may be unfamiliar with them. Specifically, this chapter provides foundational matter for set-based design, emergent properties in a system engineering context, and flexibility.

2.1 Set-Based Design (SBD)

> *In anything at all, perfection is finally attained not when there is no longer anything to add, but when there is no longer anything to take away...*
>
> *–Antoine de Saint-Exupery, from his memoir, Wind, Sand, and Stars*

Set-based design (SBD) provides the foundation for this dissertation. It unified the efforts of leadership and the team. The first known introduction of set-based design came from Ward's doctoral dissertation on designing a notional power train using catalog parts (Ward 1989). When researchers studied Toyota, they found success that seemed paradoxical: more design space exploration, more prototyping, loose requirements, and delaying decisions made better cars faster than the competition (Ward et al. 1995). They concluded that Toyota's design and development system contributed to its success in important ways distinct from its production system (Sobek II, Ward, and Liker 1999). Those investigators coined the term set-based concurrent engineering, which many people now refer to as SBD. Since then, the concept of SBD, as an alternative to point-based design (PBD), proliferated in research and practice (Toche, Pellerin, and Fortin 2020).

These principles first transferred to the naval engineering domain with Singer's dissertation (Singer 2003). Singer introduced the SBD method to the Navy at a Society of Naval Architects and Marine Engineers (SNAME) Ship Design Committee meeting in June 2007 (Singer et al. 2017). This introduction led to a policy memorandum from the Commander of Naval Sea Systems Command outlining high-level goals to establish relevant toolsets and capabilities to conduct SBD for early phases of ship design (Naval Sea Systems Command 2008). This policy inspired a summary article introducing SBD to the naval engineering community (Singer, Doerry, and Buckley 2009). These actions sparked several follow-on academic investigations. Frye (2010) applied the principles to a submarine design. Gray (2011) expanded the domain by testing the use of fuzzy logic systems to introduce uncertainty in the design space. Hannapel (2012) developed a new multi-disciplinary optimization algorithm inspired by SBD principles. McKenny (2013) extended the decision support framework for managing large-scale teams. The principle also inspired practical applications in the early-stage design and requirements generation of naval vessels. The Ship-to-Shore Connector program provides the first example of SBD in the US Navy (Mebane et al. 2011). The Amphibious Combat Vehicle for the US Marine Corps (Burrow et al. 2014) followed soon after, and then the Small Surface Combatant Task Force (Garner et al. 2015), which led to the Constitution-class frigate. This knowledge, and more, created a Technical and Research Bulletin to help guide naval engineers in the practice of SBD (Singer et al. 2017).

In essence, SBD is a design method that uses sets of alternatives to reason about the design space instead of iterating on point solutions. Reasoning using sets allows the designer to account for options, variations, ranges, uncertainty, and other aspects that do not exist in point solutions. The sets exist at every level of abstraction in the design structure in which a designer must consider options, variations, ranges, or uncertainty. Reasoning using a set allows the designer to consider elements of the set that are infeasible first and remove those portions from further consideration, avoiding unnecessary analyses. Subsequently, they can consider dominant solutions. Domain boundaries do not limit either consideration because of the intersections inherent in the sets. In other words, if appropriate, one domain may remove a portion of another domain's trade space if the intersection of the two domains dictates that outcome. Relatedly, dominance is a system issue and must consider impacts on intersecting sets for conceptual robustness; dominance within a domain is neither necessary nor sufficient for selection in the global design space. The SBD method converges to the final solution by systemically removing inferior alternatives from further consideration.

At its core, SBD reduces design risk by removing elements from the design space vice selecting them. In SBD, the design team removes infeasible or dominated portions of the design space. These decisions withstand scrutiny because infeasibility is highly unlikely to change with time. Therefore, the team can accommodate new information, including requirements changes, in less complicated ways. Further, one can make these types of decisions on partial information; if one domain declares a portion of the design space infeasible, that portion is infeasible for all domains. This aspect means domains can work semi-autonomously to develop and analyze their sets, enabling a dispersed team to progress. SBD minimizes rework and incurs less technical risk in the product by delaying decisions until options are proven feasible.

In contrast, PBD selects each element and characteristic near the beginning of the process, when the least amount of design information is known. This method effectively rules out thousands or millions of potentially dominant solutions and with much less justification documented. Furthermore, this method expects rework, iterating around this design point through each domain in succession to reach a converged design. In other words, it expects that one will select the wrong point at the beginning, in contrast to SBD, which endeavors to remove these points at the last responsible moment.

In their breakthrough article, Ward et al. (1995) listed the advantages they saw in the seemingly paradoxical SBD approach at Toyota:

1. Set-based concurrent engineering enables reliable, efficient communication.

   In the conventional, point-to-point search, every change that part of an organization makes may invalidate all previous decisions.

2. Set-based concurrent engineering allows for greater parallelism in the process, with more effective, early use of sub-teams.

   The conventional model makes little sense, planning the manufacturing process before product definition.

3. Set-based concurrent engineering bases the most critical, early decision on data.

   Design space exploration and prototyping create data to aid these decisions.

4. The set-based process promotes institutional learning.

   Designers keep lesson-learned books that aid future design efforts.

5. Set-based concurrent engineering allows for a search for globally optimal designs.

   Conventional methods and exploration techniques are efficient at finding local optima.

Therefore, SBD is most appropriate when a design project has: 1) a large number of design variables, 2) tight coupling among those variables, 3) conflicting requirements, 4) flexibility in those requirements allowing for trades, and 5) required learning for a solution

(Singer et al. 2017). These characteristics accurately describe the environment of early-stage naval vessel design activities.

Sobek, Ward, and Liker (1999) provided their thoughts on the basic principles of SBD. These principles have withstood modification (Ghosh and Seering 2014). They are:

1. Map the design space
    a. Define feasible regions.
    b. Explore trade-offs by designing multiple alternatives.
    c. Communicate sets of possibilities.
2. Integrate by intersection.
    a. Look for intersections of feasible sets.
    b. Impose minimum constraints.
    c. Seek conceptual robustness.
3. Establish feasibility before commitment.
    a. Narrow sets gradually while increasing detail.
    b. Stay within sets once committed.
    c. Control by managing uncertainty at process gates.

Ghosh and Seering (2014) also list several characteristics of SBD which provide a filter through which one can view set-based practices, listed here:

1. Emphasis on frequent, low-fidelity prototyping
2. Tolerance for under defined system specifications
3. More efficient communication among subsystems
4. Emphasis on documenting lessons learned and new knowledge
5. Support for decentralized leadership structure and distributed non-collocated teams
6. Supplier/subsystem exploration of optimality
7. Support for flow-up knowledge creation

Recently, independent groups conducted literature reviews surveying the state of SBD and its theoretical and practical development since 1989.

Toche, Pellerin, and Fortin (2020) sought to understand key aspects for developing models and methodologies when transitioning an organization from legacy methods to SBD. Specifically, they call for more research to:

- Develop and discuss approaches, methodologies, and frameworks that support the lean transformation in an organization
- Extend the application of SBD beyond the conceptual design phase, especially implications for detail design, prototyping, testing, and the rest of the PD lifecycle
- Address the SBD front-loading process
- Address reusable design knowledge management
- Address prototyping and testing for an informed convergence process
- Address value streams' cross-functional interactions
- Create a model that represents cross-domain communications, overlapping, narrowing, and refinement of Sets to enable institutional learning
- Explore the relationship of prototyping and testing with the overall design process

Shallcross et al. (2020) assess the state-of-the-practice in SBD organized by four tenets: robust alternative development, uncertainty resolution, delayed design decisions, and improved communication. They call for more research to:

- Expand uncertainty modeling to all stages of product development
- Expand methodologies into the field of MBSE
- Contribute more to requirements development and program management for the entire life cycle
- Create a methodology to execute SBD across the engineering and program management disciplines
- Incorporate intelligent adversary analysis
- In general, have a model that addresses all the tenets of SBD instead of narrowly focusing on one or two.

Lastly, Dullen et al. (2021) sought to understand the use of quantitative methods within SBD practices, and call for more research to:

- Development of an optimal robustness metric that evaluates how well a set of alternatives under uncertainty falls within the Pareto Frontier
- Develop a design margin metric that could be used in place of the diversity or flexibility metric
- Develop the optimal overlapping strategy for narrowing the set of alternatives

- Develop methods to effectively incorporate SME knowledge and decision-maker's preference into Bayesian Network (BN) models in SBD context
- Incorporate multi-label classification to incorporate additional metrics such as dominance and robustness
- Identify efficient methods to explore design space to improve the cost effectiveness of developing BN models
- Identify a Markov decision problem (MDP) that can be used to develop optimal modeling policies (virtual prototyping) and optimal testing policies (physical prototyping)

Each article provides an overview of SBD, an analysis of the literature, and recommendations for further research in the field. While many of these articles addressed industrial applications, the principles and characteristics apply in the public domain equally well. Outside of the articles regarding Toyota, few examples exist depicting SBD executing on the scale of a system-of-systems like a naval vessel, nor at the scale with dozens or hundreds of design team members. Further, the literature tends to cover cases of requirements generation or material selection but not simultaneously undertaking both.

SBD evokes a design language that may be unfamiliar. The first term deserving a better definition is 'set.' Dictionary definitions or those found in math textbooks point in the right direction with synonyms like collection, group, combination, and series. Shallcross et al. (2020) define *sets* as groupings of similar yet distinct design alternatives sharing at least one common design feature. Bernstein (1998) defines them as groups of design alternatives. Singer et al. (2017) define them as the range of options for a design factor[2]. Raudberget (2010) defines them as palettes of different solutions to a specific function or problem and as families of design proposals. One can see from these definitions that the community thinks about SBD in different ways.

---

[2] Singer et al. (2017) also define a design factor: a solution, parameter, characteristic, or relationship that influences the design at some level. Examples in naval engineering include length, length-to-beam ratio, producibility, and engine size.

The definition of sets for this dissertation shares views with each of the four preceding references. A set represents a useful decomposition of a group or point into subsets and characteristics for our purposes. This description starts to distinguish that sets are not passive groupings of alternatives collected after creating the alternatives. Rather, it intentionally invokes the creation of sets to reason about the design and the knowledge gap(s) confronting the design team. This definition carefully includes that a set may contain both subsets and characteristics. If appropriate, this aspect allows the representation of aspects beyond physical instantiations, such as temporal or functional characteristics. Characteristics in this sense are the same as design factors. Chapter Five expands on this definition further and discusses the decomposition of groups and points into partitions. The mathematical set notation uses a standard that lists elements of a set into braces, such as $A = \{1, 2, 3, 4\}$. For this research, the notation would be $DESIGN\_SPACE = \{ship_1, ship_2, ship_3, \ldots, ship_i\}$. In plain English, this states that the variable *design_space* is defined as the set of ships in consideration. Through most of the text, the simplified notation '{ships}' represents the set of all ships in consideration, and this representation applies to all {sets} discussed, such as {propulsion} for all propulsion architectures still in consideration or {topside} for all topside designs. When appropriate, the text also uses [vector] notation to refer to a specific instance within a {set}.

2.2 Engineering Systems and Emergent Properties

> *Some problems are so complex that you have to be highly intelligent and well-informed just to be undecided about them.*
>
> \- *Laurence J. Peter*

De Weck et al. (2012) provide an overview of socio-technical systems and their evolution over several decades, separated into epochs. An insight they provide addresses the evolution of requirements with time. In the early days of inventions and artifacts, the focus and need pertained to the primary function, for instance, the light bulb providing light. Evolution in thought, the complexity of artifacts, and the interconnectedness of a growing world changed how society thought about design artifacts. Requirements expanded (automobiles needed windshields, wipers, headlights, brakes on every wheel, and much more), and so did the understanding of value. Characteristics like quality, safety, and reliability became as important as the primary functions. In other words, engineers and designers reasoned about *emergent properties*. Other terms for these properties include "ilities," quality attributes, or non-functional requirements. De Weck et al. (2012) define these:

> *The ilities are desired properties of systems, such as flexibility or maintainability (usually but not always ending in "ility"), that often manifest themselves after a system has been put to its initial use. These properties are not the primary functional requirements of a system's performance, but typically concern wider system impacts with respect to time and stakeholders than are embodied in those primary functional requirements. The ilities do not include factors that are always present, including size and weight (even if these are described using a word that ends in "ility").*

Practitioners also add that emergent properties are system-level properties not attributable to individual parts and depend on the structure of the whole system and interactions with the environment (Walden et al. 2015). They also characterize society's knowledge of these emergent properties as growing with time, including how to analyze them. Early scientific journals studied knowledge on a few emergent properties: quality, reliability, safety, flexibility, robustness, and durability. Time revealed many more: potentially over 200 (Adams 2015). Engineers must decide which properties are relevant to their project. This thesis reveals those important to this research and design project in subsequent chapters, especially Chapter 7.

The engineering and research communities continue providing a growing body of literature on emergent properties, especially on methods to evaluate them as part of the design (making them less emergent!). For example, the Systems Engineering Handbook addresses twelve[3] directly in its Specialty Engineering chapter (Walden et al. 2015). Likewise, the Department of Defense (DoD) maintains several standards and handbooks regarding emergent properties [see Table 5-1 of (Office of the Under Secretary of Defense for Research and Engineering 2022)]. However, Adams (2015) suggests that the engineering community lacks an agreed-upon formal definition, a list (of which he provides three that total 218 elements), and a

---

[3] Affordability, (electromagnetic) compatibility, interoperability, sustainability, manufacturability/producibility, reliability, availability, maintainability, resilience, safety, security, and usability.

taxonomy for emergent properties (which he also proposes). Adams (2015) also concludes that emergent properties have the following essential characteristics:

1. Define a property or quality that the system should have.
2. Can be subjective, relative, and interacting.
3. Describe how well the systems must operate.
4. Are associated with the entire system.

Some emergent properties are better defined than others. Indeed, those that have the longest histories have extensive definitions, metrics, and criteria regarding how to evaluate them. For instance, the engineering community has rigorously defined safety, quality, and reliability. Similarly, DoD has rigorously defined important emergent properties like survivability and affordability. Flexibility provides an interesting case because its definition has changed with time. Initial research and publications on flexibility regarded it as the physical ability of materials and structures to bend under an oblique load, while the version synonymous with adaptation arrived in the epoch of engineering systems (De Weck, Roos, and Magee 2012). This research pursues this younger version of flexibility.

2.3 Flexibility

> *Intelligence is the handmaiden of flexibility and change.*
>
> \- *Vernor Vinge, A Fire Upon the Deep*

The Navy's use of *flexibility* has many synonyms, including modularity, adaptability, changeability, modifiability, and upgradeability. Each of them speaks to change in the face of a stimulus. Others propose alternate taxonomies to semantically organize these nuanced change-related terms differently (Ross, Rhodes, and Hastings 2008; Ross and Rhodes 2015; Schank et al. 2016), but the team chose flexibility as its umbrella term under which other change categories fell.

The decision to use flexibility as the dominant change-related term stemmed from its existing use in the Navy. The Navy has experimented with forms of flexibility for several decades. In the 1970s, the Navy initiated a program called SEAMOD under Captain James Jolliff and Mr. Jack Abbott "to identify areas in which modularity could benefit the Navy." (Broome, Nelson, and Tootle 1982) These efforts transformed into Variable Payload Ships studies and included interface standards called the Ship Systems Engineering Standards (SSES). Elements of these efforts made it into the next major class of ship the Navy delivered: the Arleigh Burke-class destroyers, which were designed in the 1980s and delivered starting in 1991. The DoD started publishing policies and standards on open systems architectures in the 1990s. In the early 2000s, the Navy took modularity and interfaces further with the Littoral Combat Ship. Figure 1 provides a summary graphic of these modularity-related efforts.

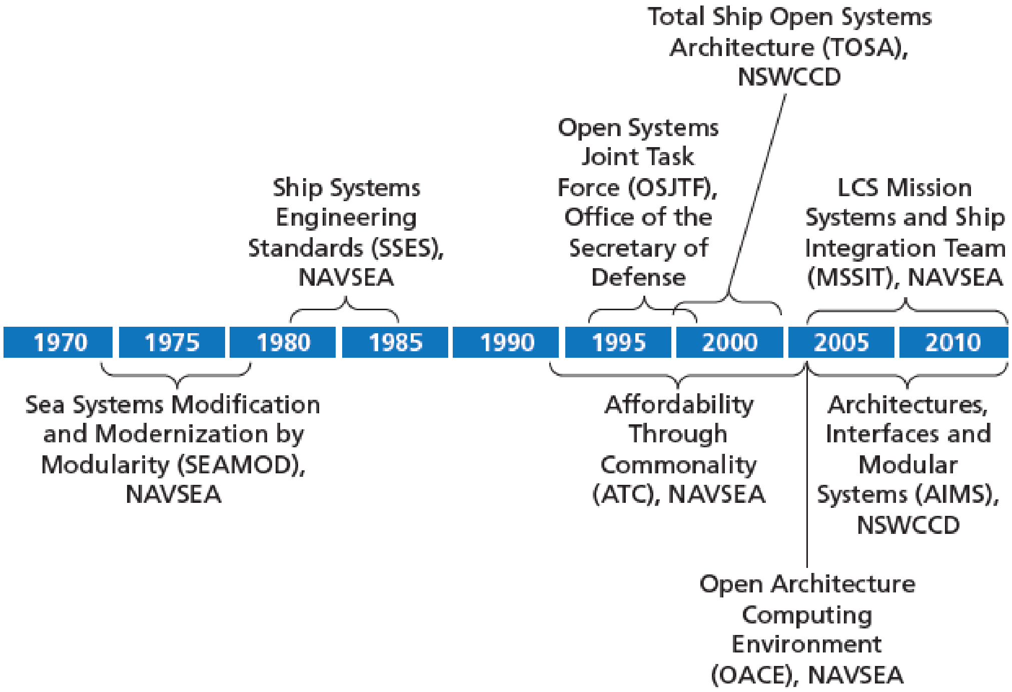


Figure 1: Modularity-Related Navy Programs, Figure 2.1 from (Schank et al. 2016)

Soon after these efforts, one finds ‘flexible ships’ entering the Navy’s lexicon (Page 2011; Sturtevant et al. 2014). Several factors drove the Navy to look beyond modularity as a necessary emergent property of their vessels, especially the metrics related to modernization for its vessels. These factors led to a Flexible Ship Integrated Project Team (IPT) standing up at the direction of the Director of Surface Warfare on the CNO’s staff and the Program Executive Officer for Ships. The IPT included members from around the naval engineering community, including shipyards, vendors, contractor support, and government personnel. It continues today and is helping new construction programs.

The System Engineering Handbook defines *flexibility* as the ability to bend or restructure (Walden et al. 2015). It further clarifies that through the sub-property *reorganization*, the system can restructure itself in the face of a threat, and through the sub-property *repairability* the system can repair itself following a disruption. These are in the *resilience* section of specialty engineering. Ross and Rhodes (2015) provide a deeper understanding of the semantics of the various related “change ilities.” They discuss flexibility having both polysemy, multiple meanings that are semantically related, and synonymy, multiple words having similar meanings. They also suggest the specificity of distinguishing adaptability from flexibility based on whether the change stimulus is internal or external to the system of interest, respectively (Ross and Rhodes 2015). This research used the semantic approach of de Neufville and Scholtes (2011) and de Weck et al. (2012). They use flexibility to summarize its synonyms, such as adaptability, reconfigurability, and modularity.

Flexibility addresses uncertainty; and only has value with uncertain conditions. Uncertainty creates a central issue for understanding and providing value with large-scale and long-lasting projects like naval vessels: what to build. Forecasts are always wrong (de Neufville and Scholtes 2011), sometimes minimally but sometimes substantially. Therefore, predicting outcomes for efforts like naval vessels is difficult but necessary. As an example, the designers of the Arleigh Burke-class should have had difficulty predicting the Navy would build their design for 40 years and operate it for another 35 years. They could not predict the proliferation of the internet and cybersecurity and vulnerabilities that seem so commonsense today. Similarly, several other threats advanced in the life of that design, including long-range ballistic missiles, hypersonic missiles, and uncrewed systems. The long service lives and capital investments of naval vessels dictate the appropriate approach to this uncertainty is providing options in the platforms to enable appropriate responses to future uncertainty.

Wang and de Neufville (2006) provide an essential distinction between options *in* a platform versus options *on* a platform. Management always reserves the right to exercise options *on* real assets like naval vessels; they can extend service lives, abandon programs, defer programs, switch combat systems, expand capabilities on a platform, and more. The critical question is whether management consciously embeds these options *in* the structure of the vessels, contracts, and sustainment practices. Put simply, options *in* a project should enable efficient and affordable execution of options *on* the project.

Real Options Analysis (ROA) provides a framework to think about these options. ROA applies financial option theory to real assets and is well-documented in the literature (Dixit and Pindyck 1994; Copeland and Antikarov 2003; Mun 2006; Kodukula and Papudesu 2006; de Neufville and Scholtes 2011). Projects with inlays and outlays use the method in straightforward ways, but public projects like naval vessels cannot follow the strict ROA approach since the Navy has no profit motive (Page 2011). Still, thinking about these projects using the principles of ROA influences design activities in valuable ways. Specifically, real options thinking causes the engineer to consider the uncertainty and options to address that uncertainty at system and subsystem levels. Real options thinking allows for analyses within a general framework to account for uncertainty, including well into the future as one is making architectural decisions. It considers and accounts for the inherent value of flexibility and the ability of a system to respond to changes over time, especially with the long service lives of naval vessels. Real options thinking recognizes that managers inherently retain the flexibility to make good decisions, and their ability to make better decisions increases with time as more information becomes available (de Neufville and Scholtes 2011). It also recognizes that options represent the manager's *right* to decide without the *obligation* to do so. Unlike Net Present Value and ROA, the valuation of options *in* a project follows no set format or formula. Chapter seven reveals the framework conceived and used to apply real options thinking and imbed flexibility -in- the architecture of a naval vessel.

3. Research Summary

If you can't explain it simply, you don't understand it well enough.

-Unknown

The original question of this research involved the treatment of emergent properties, especially flexibility, in a set-based design method. The question hypothesized that one could make emergent properties into {sets} of their own, negotiated on equal footing with more traditional functional requirements of a design. Answering this question, of course, required a functioning set-based design process, which needed creation. As a result, equivalent (maybe more) learning occurred during the creation of the process to answer the original question. There were few examples of set-based methods of this scale and scope, including how to *start* such a method. Therefore, this research first presents a case study of the stand-up of set-based design for a complex system-of-systems project.

This research uses a particular method of ethnographic studies called participatory action research, which is beginning to be used in the engineering domain and is appropriate in this case. Ethnography is the study of people in their environment, usually derived from direct observation or interviews, and is used extensively in the social sciences. Action research is a subset within ethnography in which the researcher observes process and change of a group. In participatory action research, the researcher is actively engaged in the process, often as a consultant, instead of passively observing the subjects of the study.

In this research, I was a leader, consultant, participant, and observer of the design process for the next generation of large naval combatants, the DDG(X). I learned through the basic iterative process of reflecting, planning, and acting. My leadership role was the Deputy Ship Design Manager for the design team, the second-in-command to the project's Senior Ship design Manger, or Chief Engineer. I was responsible for standing up the team, creating the process, tracking resources, and myriad other things in this role. My consultant role was the "action" part of action research, where I could insert appropriate steps in processes or change elements of the team and process after observing their performance. It also manifested in mentoring and training team members if and when they struggled with this new SBD method. My leadership role held the team accountable to deliver results, while my consultant role could coach them to help them achieve. Separately and distinctly, I participated in the process in my role in the Flexibility domain, first as that domain leader, then as a consultant to them. Flexibility is a critical feature of the ship, enabling its efficient and affordable response to uncertainty as that uncertainty resolves. Flexibility also inspired the original research question, motivating the connection to these efforts in one capacity or another throughout my tenure on the design team. This role helped me observe and experience activity at the domain level instead of strictly from leadership. Lastly, my observer role was that of the researcher, reflecting daily on the team's activities and process and if improvement was desired or required.

These various roles allowed for testing methods and procedures that helped the team conduct their work and the design progress and contribute to identified gaps in SBD knowledge. In separate papers published in 2020, researchers reviewed hundreds of articles looking for existing SBD guidance, rules, and theory (Toche, Pellerin, and Fortin 2020; Shallcross et al. 2020). Both research teams reached similar conclusions that the design community lacks holistic models that test the existing theories and benefits of SBD. They encouraged more work on developing theories and models for SBD as a practice. This dissertation creates a model that includes risks, considerations, and findings of one process of SBD.

This thesis presents the SBD process employed to inform requirements and advance design knowledge on a complex project: a US Navy combatant ship. It shares the planning and actions that established the processes and team structure. It also shares the reflections, observations, lessons learned, and knowledge created throughout the endeavor. While the summary articles reveal a body of literary contributions studying SBD over the last three decades, the field lacks examples of the theories applied on a large scale. This thesis offers one such example, including insights on the aspects of the project that can generalize to other product development projects of any scale. It also explores three other related topics of particular interest: the integration portion of the process, benchmark designs, and methods for considering emergent properties during the process.

Following this summary, Chapter Four reveals the example of SBD in a complex system. It offers insights into several of the identified knowledge gaps from the literature, including reusable design knowledge and its management, the use of prototyping and testing within the framework, and interactions among the team's various technical domain specialties. The chapter shares how we instituted a system that practically executed the theoretical processes of SBD, including the refinement of Sets and cross-domain communications. It reveals the team structure, procedures, and policies implemented in the early design of a combatant ship. It is evaluated together and constitutes a model with certain generalizable attributes and lessons learned. It also reveals some risks of transitioning to set-based concurrent engineering from a legacy process and how we addressed those risks. In addition to offering an assessable model, this research provides three deeper discussions, ideas, and lessons on integration by intersection, prototyping, and the treatment of emergent properties in a set-based method.

After presenting the model, Chapter Five further describes a method of design integration in SBD. It starts with offering thoughts on how to think about {sets} and their interactions to build upon previous work but recognize the multi-dimensionality of a complex design problem such as a Naval vessel. This chapter shares the perceived risks of the transition to this type of process, the risks we anticipated during the process, and how we managed those risks in generalizable ways. It provides more details than the preceding chapter on how we handled the reduction of {sets} including critical communication practices, documentation, and flow of information. It also offers lessons from developing, refining, and executing this part of the process. It gives insight into team dynamics and interactions of this part of the process. This information lays the foundation for a unique set of integrated information, the {benchmarks}.

Chapter Six describes the concept and use of {benchmarks}; a unique type of set in our approach. It shares the reasoning behind creating {benchmarks}, which are fully integrated sets of {sets} that make one point in the design space. It then discusses the value of creating these representative points in the space during a set-based process before establishing the baseline that moves forward in the more extensive product development process. This chapter also discusses creating reusable design knowledge using excursions from the benchmarks. It also includes a discussion on prototyping and testing and specifically the {benchmarks} acting as virtual prototypes to gain knowledge and conduct testing. Also, {benchmarks} and excursions are one way to test and gain knowledge on emergent properties, but there are other methods of exploration.

Finally, Chapter Seven explores the treatment of emergent properties in SBD to address the original research question. It discusses the primary methods by which one could explicitly consider emergent properties as part of the design and its process instead of resulting from the many other decisions an engineer or designer makes. It reveals how the team analyzed several emergent properties, especially flexibility, survivability, and affordability.

Chapter Eight and Nine conclude with other insights gained, future work suggested, and a summary of significant results and findings.

4. An Instantiation of Set-Based Design (SBD) in a Complex Project

One thing is for sure. We have to do something. We have to do the best we know how at the moment…; If it doesn't turn out right, we can modify it as we go along.

- *Franklin D. Roosevelt counseling Frances Perkins*

This chapter reveals the model created to execute an SBD method for a complex system-of-systems-scale project. It flows chronologically so that the reader can follow along through the planning, acting, and reflecting stages along the way and generalize important learning points when appropriate. It explains the initial considerations of leadership to 1) start the process, 2) scale it, 3) create a philosophical basis, 4) create the process, structure, and formats, 5) organize the people brought together to execute the process, 6) create the various forms of documentation used to communicate and capture knowledge and decisions for posterity, and 7) train the team. It also explains how the process operated as it grew.

The process began in 2019 after the Navy finished the Requirements Evaluation Team (RET) effort and briefed its results in December 2018 to Navy leadership. That effort led to the Chief of Naval Operations (CNO) approval of Initial Parameters in January 2019 and an allocation of a modest budget to stand up a design team. The directions to the team included exploring cost-capability trades within the requirements space and conducting early-stage design work that supported architectural decisions for the vessel. Those decisions and related analyses informed first Top-Level Requirements, a version of requirements for communications with industry partners, and then a draft Capabilities Development Document, the formal representation of requirements for the vessel that would drive design, verification, validation, and testing.

In the summer of 2018, the Navy's Chief Naval Architect assigned a Senior Ship Design Manager (SSDM), a role much like a chief engineer, to lead the upcoming design effort for the vessel. The RET was beginning, and she knew it would be an appropriate time to assign the senior leader and allow him to witness the analyses that led to the requirements he would be responsible for meeting with his design. The selected SSDM had almost two decades of ship design experience in both industry and the government and, at the time, was the SSDM for the DDG 1000 class. He brought relevant expertise regarding how to run a large design team on a complex project with new technology. Further, he had ideas on what to keep and what to change to utilize the lessons he learned during that class's design, construction, and commissioning. As a result, an important trait was his desire for active involvement throughout the design process instead of taking a review-and-approve approach. The full breadth of positive and negative lessons from his experience is beyond the scope of this study, but let it suffice to say that those lessons greatly influenced the structure and processes that he and his deputy created, as shared throughout this chapter. Later that summer, the Navy assigned the author as the Deputy Ship Design Manager. At the time, he was on the Massachusetts Institute of Technology faculty, teaching students the tenets of naval engineering, including SBD, real options, and new sustainment practices like model-based product support. He also had experience in ship design during a tour in the program office for DDG 1000, where he had worked closely with the SSDM and his deputies to resolve the daily engineering challenges that presented themselves to us. Further, he had experience in ship delivery after accepting USNS Lewis B. Puller on behalf of the Navy, experience in maintenance and sustainment from a tour at a Regional Maintenance Center, and operational experience from a tour serving on a destroyer. We were the nucleus of the design team and part of the leadership team, so through the rest of the paper, "we" refers collectively to the SSDM and DSDM unless specified as another group.

Together, we had seen the cascading effects of decisions made early in the design process before enough information was known, so we knew we wanted a method that encouraged making decisions at a more responsible moment – often much later in the process. We also witnessed the value of diversity of thought and solution space and knew we wanted to establish procedures that encouraged this concept. We also both understood the complexity and inter-relatedness of a ship design. Further, we shared the perspective that these early stages of the process influenced the requirements the most. Consequently, we knew more analysis was necessary to inform requirements best while also moving forward with design products. These aspects informed our decision to create an SBD process that could scale in two ways. First, the process would scale to the system-of-systems level required of a combatant ship design from previous documented efforts of different scales. Second, the process would scale in scope and complexity from the early stages, with fewer team members, higher abstraction in the design space, and low fidelity analyses aiding the design decisions. We aimed for process stability from then to the later stages with more team members, lower levels of abstraction in the design space, and higher fidelity analyses supporting design decisions.

The startup of a design process naturally created several leadership considerations. For instance, we had to decide on the team organization and structure, whom we would ask to be on the team, in what roles, and consider how the team would grow. This growth included from an initial government-only effort to one with design agent support, and then finally to our vision of a national team that included industry partners. These aspects are covered more in Section 4.2. We also had to develop the meeting structure, meeting timing, meeting periodicity, and other administrative elements defining how the team would perform. Sections 4.3 and 4.5 cover the parts of these relevant to SBD. Lastly, we knew the importance of establishing a philosophical basis to align the team, especially considering that past practices for many engineers who would receive an invitation to the team may be point-based instead of set-based. Thus, our first task was to document this design philosophy.

4.1 Philosophical guiding principles for our design process

Make it simple but significant.

-Don Draper, a fictional character on Mad Men

We gave as much thought to the philosophy as to the administration and operation of the team. After all, a ship design is essentially the product of requirements, practices, standards, and the design philosophy. Accordingly, we created a Design Guidance Memorandum (DGM) and issued several that encompassed the Design Philosophy. **Error! Reference source not found.** displays the initial tranche of DGMs. We began thinking through these as the RET finished in November of 2018. We deemed this an appropriate amount of guidance to start the set-based process to create a robust design. However, the guiding documents and philosophy grew as the process played out and we learned.

Fundamentally, our philosophy was to manage the risk we deemed inherent to the ship and the ship design process while delivering the attributes and characteristics of an affordable and flexible combatant that would serve the Navy for the next 60 to 80 years - and do so rapidly. Admittedly, Table 1 reveals that the process was imperfect in quickly providing this guidance to support a timely design process. We started drafting this guidance as we stood up the processes and gathered the team members, essentially building this design machine as we were running it.

We managed that risk by frequently reflecting on progress and adjusting elements of the machine when and where appropriate, in line with the action research approach. The first formal guidance took almost six months to edit and release, but we provided informal versions either in hard copy or through training and mentoring, a topic of Section 4.5.

Table 1: Initial Tranche of Design Guidance Memoranda

Number	Title	Approval Date
DGM 001	Minutes and Design Decision Memo Routing Process	19 Sep 19[4]
DGM 002	Set-Based Design Method Guidance	23 Oct 19[4]
DGM 003	Defining Non-Developmental	17 Jul 2019
DGM 004	Digital Engineering Strategy	Pending
DGM 005	Initial Ship Specifications	08 Aug 19
DGM 006	Hull Form Model Test Objectives	04 Jun 20
DGM 007	Flexibility Framework	Pending

One of the first risks we addressed was knowledge capture and configuration control. We anticipated creating a considerable amount of knowledge and making many decisions. We both had experiences in design when the question inevitably arose, “What idiot made that decision?” The person asking this (somewhat rhetorical) question does not honestly believe the decision-maker was unintelligent but more likely has new information unavailable to the original decision-maker that calls the decision into question. With confidence that our effort would yield similar scenarios, we wanted to ensure that every decision was as transparent as we could make it. In part, we tried to prevent the rhetorical question. More so, we desired a better understanding of the scope of rework required when that inevitability arose for future engineers and designers (e.g., conduct a particular analysis again or change another related characteristic or attribute within the design space that was dependent on the decision in question). Therefore, we created DGM 001 to provide the process and format for capturing data and decisions for posterity. That DGM closely couples with DGM 002, which communicates the process for creating and presenting the data to reach the captured decisions. Appendix A shares a sanitized version of this guidance.

---

[4] Original release date. There have been subsequent updates to the guidance.

The choice to use SBD presented risks. A risk of rework always exists in complex design and construction. In this case, we might rescind or reverse a decision later due to new information or higher fidelity analysis. This rework would cause the team to start over in a particular portion of the design space. Additionally, there is a risk that such a reversal could have cascading effects on other portions of the design. Our philosophy postulated that SBD would reduce this risk if executed well by delaying decisions until the last responsible moment. That moment could arrive when there is reasonable confidence by the SSDM and team that no new information would reverse a decision. It could arrive when a decision in another domain removes portions of the design space. It could simply arrive when time has run out, and an engineer must select the "best" balance from what remains.

Philosophically, we also believed SBD helped reduce risk by controlling constraints and delaying their effects on decisions and the system. Requirements, specifications, standards, designer biases, laws of physics, available technology, and available knowledge all provide constraints or boundary conditions to the design space. Selecting any part of a design solution also introduces risk because it adds a new constraint to the system. All decisions not yet made must conform to this decision or else induce rework regarding that decision and its resultant products. When executed well, SBD delays the potential effects of these constraints on the design space and a robust design solution by creating a large, viable design space that seeks to keep options open. With so many constraints put on a modern combatant, the designers struggle to address all the constraints simultaneously without sacrificing something unknowingly. Traditionally, designers eventually find solutions through many iterations.

In contrast, SBD helps manage the constraints through the transparency of each constraint's effect on the design space. SBD, therefore, encourages potential alternate outcomes. For instance, the design team may relax a constraint to keep valuable portions of the design space open. Alternatively, they might decide to conduct research and development to open a part of the design space not available with existing technology. Further, they could consciously decide to relinquish a portion of the design space in one domain to allow a dominant portion of a different domain to remain viable when conducting necessary trades. In other words, the transparency and these alternate outcomes help a designer understand, "if something has to give, what is the *right* thing to give?"

Additionally, we identified and addressed the risk of using a process unfamiliar to many on the team (including the two of us!). These risks included: team members not understanding the process, team members continuing to use the same processes, methods, and tools as before and trying to declare it SBD, lack of clarity for team members on what constitutes a {set}, and team members artificially limiting their portion of the design space.

The approach we took to address these risks is in DGM 002. It provided both the philosophy and the resulting process to the team. It promulgated the policy that only the SSDM (or his delegate) could remove a portion of the design space from further consideration. Still, any team member was empowered to retain what they deemed a valuable portion of their space. The DGM included the requirement to document assumptions within the process and have them approved by the SSDM to avoid the risk of artificially limiting a region of the design space (more on this in Sections 4.4, 4.5, and 4.6). It established the guiding principle to explore the various domains and {sets} using a method such as Pugh's method of controlled convergence, where {sets} ought to be no bigger than necessary to analyze an appropriate level of abstraction. Then, after reducing the {set} through a decision, it expands at the next opportunity at a lower level of abstraction and a higher level of fidelity and continues this expansion and contraction until the {set} reduces to its baseline instantiation for integration with the rest of the decisions in other domains and {sets}.

Information ConvergenceFigure 2 provides a graphic representation of this principle that we used to train the team and affectionately became known as the "upside-down Christmas tree." This contrasts with Ward, whose labeled interval program uses every known physical instantiation of a potential solution from parts catalogs (Ward 1989).

Also, to help drive results, we required the team to think through the metrics and criteria they would use in their analyses to drive to the next decision to avoid the risks of needless investigations that did not answer the decision point or extended studies that did not reach conclusions on time. The DGM also included a glossary for the new terms that represented this new way of thinking. It established the concept of a Set Review with the desired outcome of the Reviews to be these decisions and approvals. Section 4.3 covers the process and associated terms created in DGM 002 in more detail.

While the DGMs helped us manage many anticipated risks with policies, they did not address all risks or answer all questions. One underlying question that we had to address was: what are the right {sets}, and especially, what are the right *first* {sets}? The answer is case-dependent. The answer may be different from one Navy ship to the next and even one combatant to the next and is undoubtedly more differentiated the more two projects diverge from each other in terms of requirements and solutions. However, although the specific answers will differ from project to project, generalizable questions provide some insight for thought experiments that can help find those particular answers. The answer is not that the Work Breakdown Structure of the design in question drives the {sets}. Instead, knowledge gaps are a vital concept that determines both the first{sets} and subsequent {sets}. We used requirements, known risks, critical path analysis, external sources, and our judgment and experience with previous projects to derive knowledge gaps and thus our first {sets}.

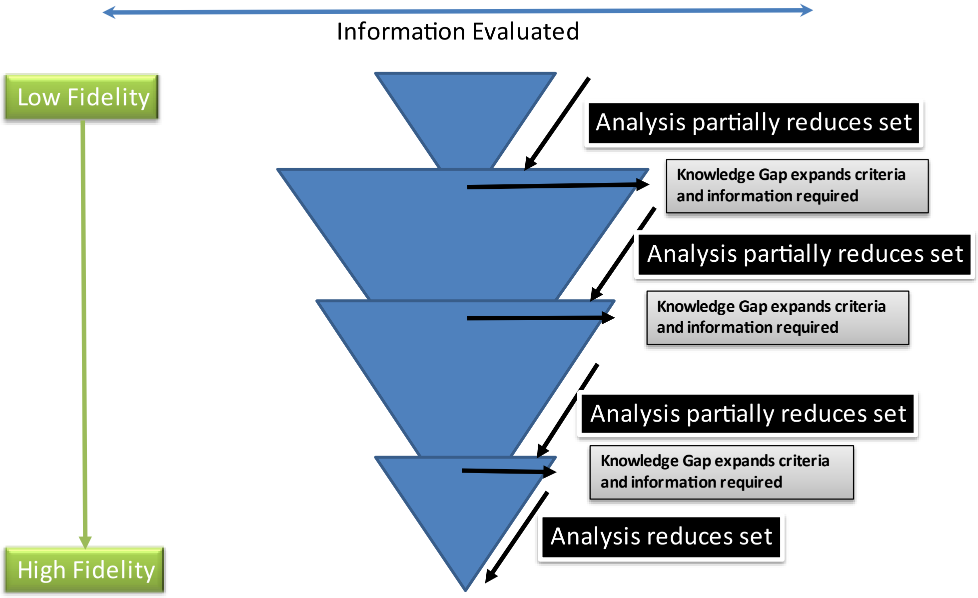


Figure 2: Information Convergence

Knowledge gaps are a vital consideration in our process, similar to the emphasis from Cloft, Kennedy, and Kennedy (2019). The knowledge gaps drove the analyses and process from the beginning. In the beginning, a “clean sheet” represented most but not all the design options. Requirements added definition to this clean sheet, revealing architectures around which decisions must be made and, in some cases, deciding material solutions. For instance, the DDG(X) has a requirement to deliver the DDG 51 Flight III combat system but retain service life allowance for the combat system to expand and change if needed. So, material solutions for the combat system and its components are immediately known, but there are still architectural decisions regarding the placement of sensors, weapons, and processing equipment. These placements represent some of the many knowledge gaps. For instance, the team must determine how far forward and aft to place sensors, how far to separate them, and whether to orient them in the axial planes of the ship like the Ticonderoga Class cruiser or at angles like on the Arleigh Burke-class destroyers. Further, analyses need to quantify how much the system may need to change and what architectural preparations can enable those changes (more on that in Chapter 7). There are many more knowledge gaps and decisions in the combat systems domain and throughout the rest of the ship design.

Thus, the question remains: which knowledge gaps should we close *first*? We knew there were thousands of design variables requiring analysis and decisions and that the design of a ship is strongly coupled. However, no tools like a Design Structure Matrix (DSM) exist for a combatant ship. A DSM could provide insight into the order in which those dependencies logically resolve. Alternately, a DSM could reveal highly coupled design loops with tight dependencies between sets of variables. But because we lacked this tool, our approach was to keep the decision space simple and high abstraction to gain quick wins and shape the "clean sheet" into something with definition. For instance, we knew that at some point, we needed to choose the prime movers for the engineering plant (some combination of diesel and gas turbine engines). That decision allows auxiliary equipment selection and engineering space sizing, which flows into sizing the ship overall. However, we anticipated the dependency of that decision on a higher abstraction decision regarding the overall architecture of the propulsion plant: mechanical like previous large combatant classes, integrated propulsion like the Zumwalt and Lewis and Clark classes, or a hybrid like the Constellation class frigates or America class Amphibious Landing Ship.

By applying a similar mindset to our knowledge of what constitutes a ship design, we decided on the first {sets} and knowledge gaps for the team to address:

1. {hull form} began by exploring three basic shaping choices that had implications on {survivability}. {hull form} is also on the critical path because of model testing.
2. {survivability} began with certain {susceptibility} studies and gathering attributes for {vulnerability}. This is a risk area for every naval vessel.
3. {propulsion} began with the architecture: mechanical, hybrid, or integrated power. {propulsion} was also a risk area with the potential to be on the critical path.
4. {power distribution} began with its architecture, i.e., ring bus, zonal, and a few characteristics like voltage and power quality standards.
5. {power generation} began with the type (gas turbine vs. diesel vs. combinations).
6. {boat handling} started with the launch location of the ship's boats: side, astern, or both. This was a risk area from previous projects.
7. {warfare system} began and was used for tracking various characteristics of the delivered combat system (the Flight III combat system) and alternate future combat system characteristics that could be envisioned and added, such as directed energy weapons, new sensors, and new computing.
8. {topside} began by exploring rough measures of deckhouse size, shape, and location.
9. {flexibility} began by establishing a framework for analysis and developing {uncertainty} to start the framework. It was a critical enabler of the ship and the design. No one had previously holistically and quantifiably analyzed it, so we felt strongly that it needed to be one of the first {sets} and that it should be its own {set}. Chapter 7 covers more on this approach.
10. {affordability} had eleven distinct studies informed by other sets and included cost specialists that helped quantify some of the cost-capability trades.
11. {ships} was a set that represented balanced concepts. We used them for validation and integration of other {sets}. Chapter 6 discusses this more.

With this philosophy to communicate and the first knowledge gaps and {sets} selected, we next set our minds to organizing our team and what people we would seek to fill that organization. There is no better way to build a process than through its use, so we felt we had conducted sufficient planning to enter that phase. Ultimately, the success of this philosophy, the guidance we provided, the team we established, and the process we set up is unknown for several more years until the ship completes the later stages of design work, including detail design, and enters construction. Metrics like change avoided, rework minimized, downstream efficiency, and others will help define the benefits of our philosophy and approach using SBD.

4.2 People and organization

Coming together is a beginning, staying together is progress, and working together is success.

-Henry Ford

To execute this new design method well, we needed a diverse group of talented, open-minded, and eager members; the success of the process depended upon the team that would be executing it. Fortunately, the Navy and our support contract design agents had quality people available to help start this endeavor: a crucial transferrable point. The fact that these members were available is notable because they were, in a sense, a representative sampling of the naval engineering workforce. In the beginning, there was little intentional selection or planning of the workforce that would support this effort, especially with all the unknowns prevalent at the beginning of an endeavor like the clean-sheet design of a new ship class. Further, several design efforts were already taking place, requiring talented engineers and experts of similar caliber. The SSDM and DSDM were the only two Naval Sea Systems Command (NAVSEA) headquarters personnel. Therefore, we requested and negotiated for the rest of our team from other organizations like Naval Surface Warfare Centers (NSWC) and support contract companies. The team was diverse, from hydrodynamicists with doctorate degrees and decades of experience to general engineers who graduated college less than five years ago. Some of the team had done concept design; some had done preliminary design; some had done contract design; some had even done detail design; few, if any, had done all. Some knew SBD, some had used it to some extent in previous projects, and some heard about it and learned it for the first time. Luckily, in this mix of representative personnel was a senior advisor who had led an SBD process before (Mebane et al. 2011) and provided a knowledgeable sounding board to the management team.

As part of the philosophical setup, we decided which elements of the design were most important or logically belonged together and used that to create a unique work breakdown structure and organizational structure to match. We called each element in the first tier of this hierarchy a functional area, with a corresponding Functional Area Lead (FAL) responsible for the work in that portion of the design space. Initially, we created this unique structure down to the second tier. Figure 3 shares this initial structure: a hub-and-spoke structure with the SSDM and DSDM at the center. We intentionally kept the number of functional areas low to simplify the management efforts. We also recognized that the functional areas for Design Integration and Systems Engineering were “hub” activities that necessitated operations and communications across the other functional areas instead of just with the Design Management Team. Figure 3 shows Producibility in a lighter color because we could not start that set due to resource constraints and a lack of suitable engagement and contract vehicles with our industry partners.

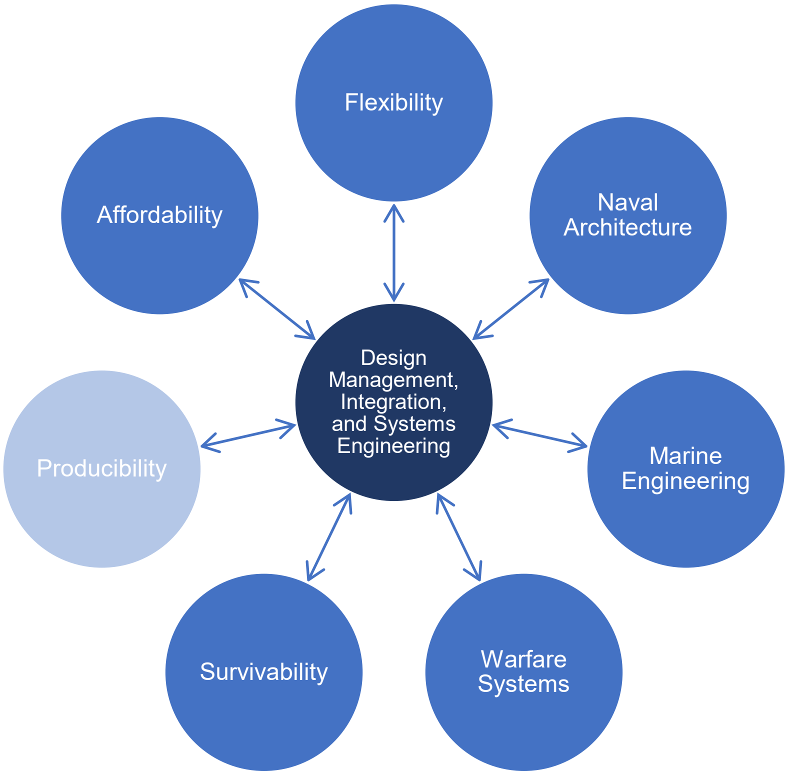


Figure 3: Initial Design Team Organization

We split the rest of the organizational choices evenly between traditional engineering disciplines for a naval vessel’s design and essential focus areas for the design, which we wanted on equal footing. Naval architecture, marine engineering, and warfare systems are the traditional and expected functional areas for a naval vessel’s design. Each aligns mainly with an NSWC that has expertise in those domains: Carderock for Naval Architecture, Philadelphia for Marine Engineering, and Dahlgren for Combat Systems. Therefore, many of the initial core team members belong to those organizations. A reader with attention to detail will notice that the functional area is W*arfare Systems,* while NSWC Dahlgren is the Navy's center of excellence for *Combat Systems*. We intentionally named the functional area *Warfare Systems* in recognition that the nature of warfare now and in the future is connected. Therefore, a combat system is less effective without an accompanying Command, Control, Communications, Computers, and Intelligence (C4I) infrastructure that extends the capabilities of the combat system beyond its host vessel.

Additionally, Flexibility, Affordability, and Survivability were meaningful emergent properties. Therefore, we assigned them as functional areas and formed teams around those concepts that would focus on those aspects of the design throughout the ship. However, we did not have them at the hub like Systems Engineering and Design Integration. Chapter 7 discusses this choice in more detail.

These Functional Areas do not always have a direct correlation to the initial {sets} revealed in Section 4.1. For instance, the Marine Engineering area is responsible for {propulsion}, {power distribution}, {power generation}, {boat handling} and others, while the Affordability area held responsibility over {affordability}. In the beginning, we intentionally decided NOT to fill the role of Design Integration Manager, even though the {ships} and {topside} sets fell into that functional area. Instead, the individuals in charge of those {sets} reported directly to management. Over time, all functional areas assumed responsibility for multiple {sets} as the set space expanded.

The hub-and-spoke organization addressed some of the people and process risks we anticipated. One risk is how to fit this process into the larger organization of which we were a part at Naval Sea Systems Command (NAVSEA). NAVSEA already has people and processes in place and is organized a certain way. Our team would need to interact with them in at least partially familiar ways and aligned with "company policy." Another risk was balancing the competing interests of the various functional areas and {sets} and avoiding giving preference to any of them when an optimal decision for one sub-optimizes another. Kossiakoff at al. (2020) provide a suitable graphical depiction of this phenomenon, replicated in Figure 4. One more risk we addressed from the beginning was communication. We recognized that with this new process that has a new language, the team members might not be able to effectively communicate and work with each other based on their experience (or lack thereof) with SBD. The last risk we addressed was team expansion. A large surface combatant design will involve hundreds of people working simultaneously, but we started with a few dozen. We needed to ensure the team's structure accommodated this magnitude of growth in personnel and scope.

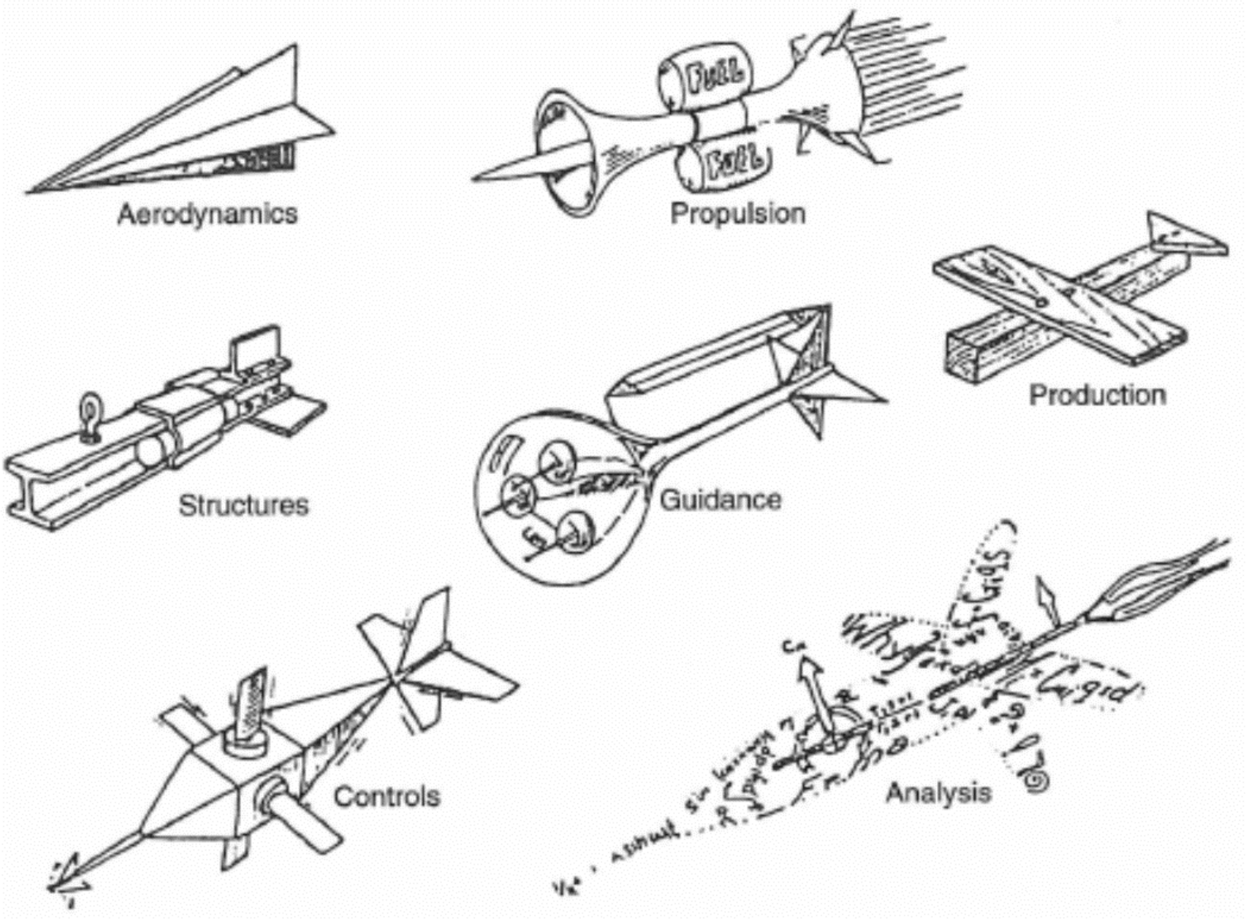


Figure 4: The ideal missile design from the viewpoint of various specialists (Kossiakoff et al. 2020)

The hub-and-spoke organization made it easier to fit the process into the larger NAVSEA organization. NAVSEA is a matrixed organization with specialty disciplines like contracts, finance, and engineering available to complement the multiple programs in Program Executive Offices. The engineering staff splits further into two broad categories: those that are system-specific, called technical domain managers, and those that are platform-specific, called chief system engineers. The SSDM is a chief system engineer for their ship that integrates technical authority efforts for the platform, in our case, the DDG(X). Firefighting, signatures, and propulsors are three examples of system-specific experts who provide deep technical expertise for those systems to all appropriate ships in the Navy across the lifecycle. The Navy calls both types of experts a Technical Warrant Holder (TWH) because they are given a warrant, designating them the expert in a particular domain for the Navy. The warrant's pedigree originates from the Secretary of the Navy (Office of the Secretary of the Navy 2021a) and law (10 U.S.C. §8669b). Then, the policy naturally follows that technical domain managers have authority in their domain, and platform TWHs have control over everything not designated to another warrant (Naval Sea Systems Command 2014). Each TWH has a pyramid of support personnel that helps them execute their warrant. The fact that technical domain managers aligned well with our functional areas allowed easy integration between our hub-and-spoke organization and the competency-aligned organization of NAVSEA. Over one hundred TWHs exist in the NAVSEA Enterprise, but the SSDM is the only one assigned exclusively to DDG(X).

We expected each functional area to interact with their relevant TWHs regularly as part of their workups to a Set Review. We ensured that our functional area leads knew their relevant TWH(s) by listing them in DGM 001 and encouraged active engagement with them. When available and appropriate, we asked the technical domain managers to participate in reviews with the SSDM to ensure relevant information flowed between these recognized experts transparently and decisions could converge. Many of the technical domain managers actively involved in our processes appreciated the information flow and our requests for their participation in the design decision process from the beginning. Lastly, in some instances, like with boat handling systems, the SSDM delegated decision authority to the domain manager TWH because of the trust established through their participation.

Further, the hub-and-spoke approach addresses who is first among equals by designating the SSDM to act as the hub, determining what is allowed to dominate within competing requirements and solutions across the functional areas. For instance, if the propulsion team desires a larger engine but the naval architecture team is trying to keep the size of the main engine rooms smaller to keep the overall length of the ship to a minimum, the SSDM gets the deciding vote if he allows a larger engine because its value is greater than the loss in value or additional cost of a larger overall ship. If a chief system engineer like the SSDM and a technical domain manager do not agree on a solution set or decision, a process exists to resolve the dispute. An observation from these interactions is that many of the TWHs follow the adage that "reasonable people, equally informed, seldom disagree," and they tend to resolve the differences in information and reach an agreement.

The hub-and-spoke organization we started with allowed us to manage communications and ensure consistent communications in a common language across the team. We managed the communications efficiently because they all generally happened through the design management team. We did not allow much communication between the various functional areas at first. We wanted team members to practice communicating in {sets} with us before integrating and sharing information with each other. This approach worked because the team was small at first, but this practice had to change as the team, and the scope, grew. Typically, a design team arranges a design site where several members collocate and have a high frequency of communications and the opportunity for accidental (but highly valuable) touchpoints and interactions. Alas, many things prevented this, from the small team size (not worth the cost) to training to funding shortfalls to COVID and the pandemic that kept the team and most of the world at home through most of 2020 and 2021.

Lastly, the hub-and-spoke organization helped us think through how the team would expand. It provided a basic structure that we felt enabled team expansion through the Preliminary Design and Contract Design phases. Over time, the spokes provided structure for those outside the hub to create a wheel, where information in the form of {sets} and findings and decisions could flow. Figure 5 depicts the basic structure after this growth. The team kept its basic structure. New {sets}, new team members, and new tiers in the work breakdown structure continue to fit within the original team structure. The evolution from Figure 3 to Figure 5 began in early 2020, about one year into the process. The team would grow to about five times its original size that year, and the number of {sets} and associated analyses expanded.

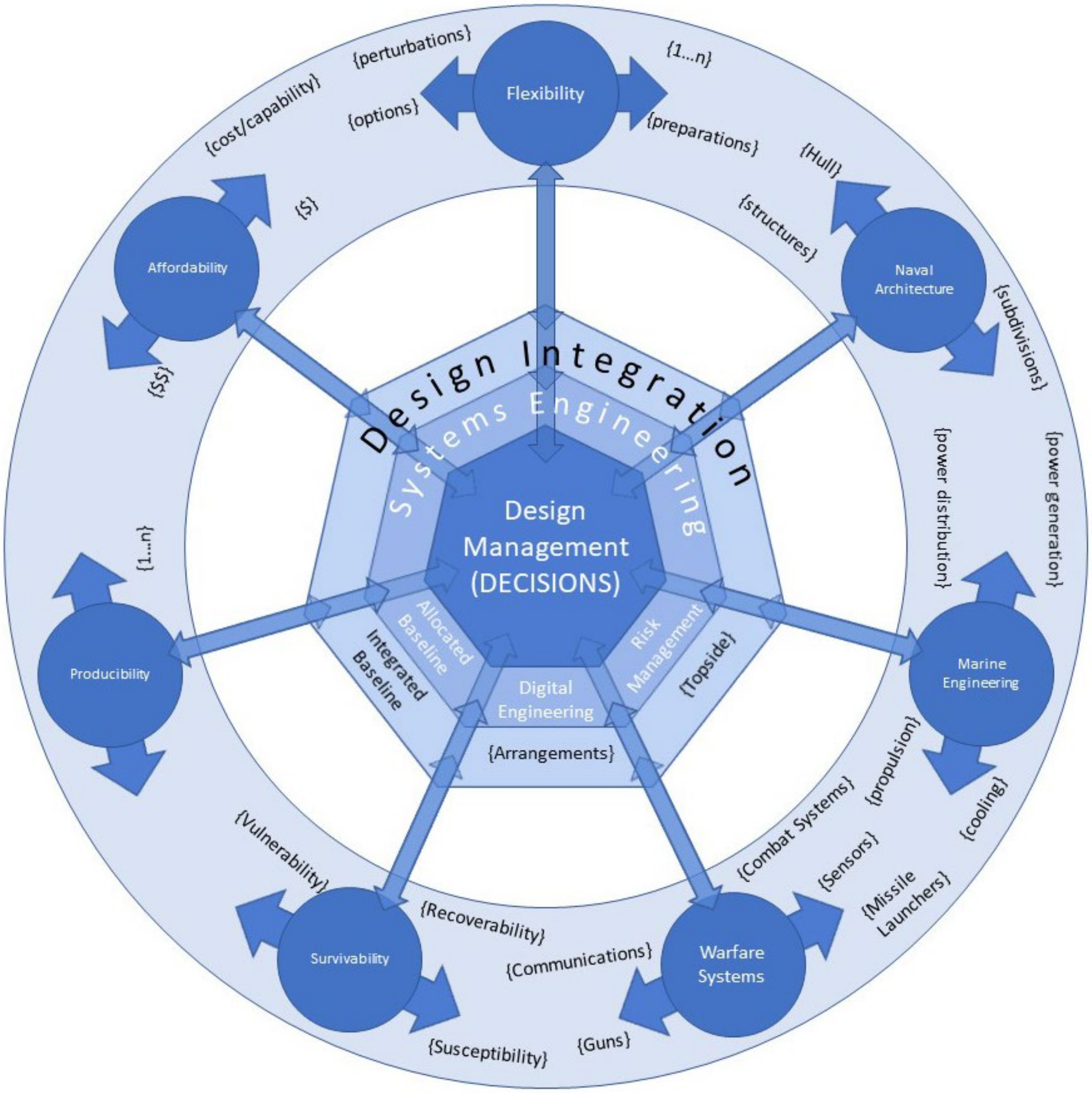


Figure 5: Evolved and scaled design team organization with a sampling of {sets}

This team evolution involved several noteworthy changes. First, the hub expanded and separated. The hub still represents all activities that operate across all domains and {sets}. It provides guidance and tools to the rest of the team and captures the information created by the rest of the team. Systems Engineering and Design Integration became separate areas that had some activities related to and some distinct from Design Management. However, they were still "hub" activities that acted across all functional domains. Figure 5 intentionally represents that some information flows to and from Design Integration, other flows to and from Systems

Engineering, and some flows to and from Design Management. The Figure does not graphically represent information flow between hub activities. This absence kept the Figure simple. We treat Systems Engineering and Design Integration as functional areas with their own leads, meetings, and tasking. Also, those two domains have large overlaps and communications within the hub to support each other.

The Lead System Engineer took full control of the systems engineering domain with responsibilities for risk, configuration control, model-based systems engineering, the integrated data environment, digital strategy, and others. He also assumed responsibility for the {requirements} set.

We hired someone for the Design Integration Manager position at this time, too. He took responsibility for baseline creation and management, cross-cutting {sets} like {topside}, {ships}, and {arrangements}, and overall integration activities. Among those, notably, we handed the SBD process over to the Design Integration Manager and his SBD Lead. The SBD Lead acted as the gatekeeper to Set Reviews (a concept covered in Section 4.3), handled all formatting, structure, content, and agendas, and reviewed each presentation for adequacy before coming to a Set Review. The SSDM or DSDM still chaired the Set Reviews to maintain appropriate outcomes management and associated documentation. The SSDM still retains decision authority. Second, the spokes/rim of the new structure changed fundamentally. While we retained the same functional areas, the set teams started to share {sets} and information with each other. The outer ring encloses these to represent that these characteristics, variables, and {sets} are communicated or negotiated with other set teams (Singer, Doerry, and Buckley 2009) in contrast with internal variables for the use of only one set team.

Part of the evolution of the design team involved the addition of a formal program office in April 2021. Before this point, a dual-hatted program manager for a separate shipbuilding effort had executed program management duties. Therefore, we accounted for their oversight and interactions with the design team and external stakeholders from the beginning. Typically, their interaction with the design team is through the SSDM, but there are exceptions where they may interact directly with the functional area leads or receive briefings from them. Overall, the team's structure was also resilient to the addition of the program office. Certain decisions now fell under the purview of the Program Manager instead of residing with the SSDM. This change occasionally resulted in an extra meeting to take the decision brief to the Program Manager after the SSDM approved something. In this sense, our SBD process also proved resilient to time, scaling, scope, and changes.

4.3 A set-based design (SBD) process

If you can't describe what you are doing as a process, you don't know what you're doing.

- *W. Edwards Deming*

The process we used to conduct SBD was a necessary complement to the people we led. The processes accommodated concurrent engineering for each of the functional areas, domains, and {sets}. They enabled the basic SBD principles of mapping the design space, integrating by intersection, establishing feasibility before commitment (Sobek II, Ward, and Liker 1999), and delaying decisions until the last responsible moment. They also integrated the seven characteristics of set-based product development. Notably, they scaled, both in the sense of working for a system of systems and in the sense that they worked while the team was small and still worked when the team and scope grew.

We used the Set Review as the primary mechanism to conduct SBD within our processes. At a high level of abstraction, the principle is simple: regularly, we conduct a structured review of the information with the SSDM that leads to a decision. In SBD, those decisions remove infeasible or dominated regions of the design space instead of selecting "the best." Each Set Review was an experiment with a learning opportunity in the academic sense. We reflected each time on what we learned about the design of DDG(X), if we learned anything about the *process* of designing DDG(X), if we had provided the proper guidance, and why the Set Review did or did not have the desired outcome.

Put simply; a Set Review is a meeting. We provided a structure and flow to this meeting to provide consistency in information presentation that the SSDM could use to make decisions or the rest of the group could use to inform their {sets} and decision spaces. Each Set Review has the same agenda and similar desired outcomes. The standard allotment of time for a Review is one hour, but the complexity of the knowledge gap or analysis may extend the review time. At first, the Set Review schedule was ad hoc. Sometimes days went between reviews but sometimes weeks or months, as Figure 6 reveals. We felt this was reasonable since we, too, were adjusting to the process, and the group of {sets} and the team were both small and manageable at the time.

Over time, we shifted to a standard schedule that primarily dedicated every Thursday to Set Reviews. We also became more disciplined in scheduling {sets} for reviews, ensuring that there were interim progress reviews for extended analyses.

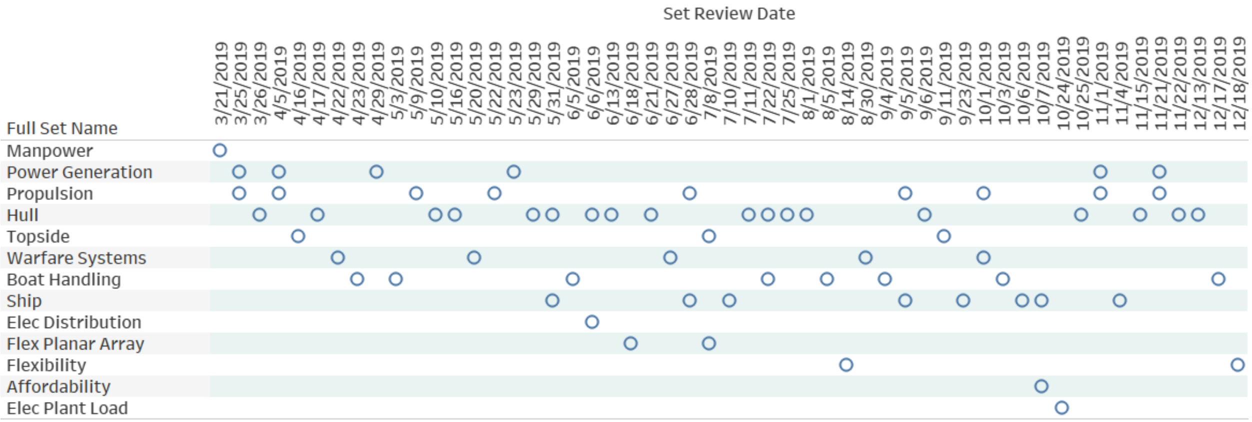


Figure 6: Set Review schedule through the first year

The agenda for a Set Review contained eight basic parts:

1) Purpose – this part should address two elements: first, the purpose of the study, and second, the purpose of the review.
2) Approach – this part communicates how the team would close the knowledge gap, including analysis methods and tools.
3) Assumptions – this section documents all the assumptions necessary to complete the analysis.
4) Criteria – this part shares the team's criteria or metrics to develop findings, conclusions, decisions, or make recommendations.
5) Analysis and Results – this section reveals the methods and data, including any sensitivity analysis or excursions performed.
6) Outcomes – this section summarizes the data that leads to the desired result and requests approval of that outcome.
7) Uncertainty – this part captured the remaining uncertainty in the {set}.
8) Next Steps – the last part made sure the team had a planned path forward based on the meeting outcome.

An exception to this Set Review format is with the first review of a new {set}. In this review, the intent is to have assumptions and criteria approved to support the analysis that closes the knowledge gap. Therefore, we abridge the Set Review agenda by removing steps five, six, and seven to accommodate this focused purpose.

Generally, the format of the Set Review reflected our decision-making process and what we thought we would need to make defendable, hopefully irreversible, decisions to remove portions of the design space. Our thought process to define a Set Review started at the Outcome section and worked backward and forward. We consciously thought through what we thought we would need for that defensible decision, to which the first clear answer is – data. We also knew that we should capture the boundary conditions creating that data for our sake and posterity, which generated the Approach, Assumptions, and Criteria sections of the format. We thought those sections flowed best in the order presented here. Next, we included the Purpose section at the front to provide a solid structure to the meeting and a foundation for the conversation. That purpose section would force the presenter and their team to think through why they were taking the rest of the team's time and drive towards that purpose throughout the brief. The study aims to present the strategic view of the knowledge gap we are resolving. The review is more tactical and could include making recommendations, requesting decisions, reporting findings, or even simply providing an interim progress review. Then, we added Next Steps at the end to close out the structure. Having to brief this would force the presenter and their team to think through follow-on activities (assuming a successful outcome). These seven elements could have been enough, and sometimes were, for a Set Review.

For this SBD method and a ship design in which flexibility was a cornerstone characteristic, we wanted the team to consider uncertainty intentionally and often. Therefore, we added that part after Outcomes and before closing the Set Review. We thought placing it seventh in the order made logical sense because of the volume of uncertainty present at this stage. We did not want team members to become overly burdened in the Approach through Analysis steps by trying to account for the uncertainty. This approach provided the balance of conducting analyses to close knowledge gaps and generate reportable results, and progress with understanding if the results would be resilient to future information and scrutiny. This section could include an assumption made during the analysis that requires further resolution, a knowledge gap that drives the subsequent analysis, a requirement or parameter within the {set} that did not require an assumption for the current analysis, a parameter that had an assumption but requires a range, or an interface through the flexibility team regarding causes to or effects from another domain.

We created a configuration control method to keep track of artifacts across the various processes and Set Reviews. We assigned each {set} a three-letter identifier. For instance, {hull} became {HUL} and {propulsion} became {PRO}. However, whether we used the full name or three-letter code, we encountered "rules" issues regarding when a {set} becomes a {set}. We had not yet exactly defined what a {set} was and was not, so it was unclear to the team when to declare something a new {set} or if it was an analysis of an existing {set} or a {subset} of an existing {set}. This absence led to the propagation of the number of {sets}, but not in a harmful or hindering way.

Traditional naval engineering sets like {hull} and {propulsion} classified their sets more easily because of decades of systems engineering efforts that already defined how physical artifacts accomplish functions and requirements in those domains. However, the classification was less defined and trickier for emergent property sets like {flexibility} or sets that naturally crossed functional boundaries like {arrangements}. Also, some other design artifacts took on codes, like {sets}, even though they were not for analysis but rather for tracking purposes only. For instance, we created {assumptions}, {findings}, and {analysis}. These sets enabled the team to track all approved assumptions together so that if the susceptibility domain needed to understand or use similar assumptions or analysis results from the hull domain, they could pull them from those {sets}. That way, each team did not have to duplicate germane assumptions. One last special set in our process was [ship], which represented an integrated and balanced collection of {sets} and {assumptions}.

4.4 Documentation of outcomes

Good design is obvious. Great design is transparent.

-Joe Sparano

A successful Set Review culminated in a Design Decision Memorandum (DDM). The DDM provides the documentation of the work and the outcome and is a critical artifact to support the team and our SBD process. It captures the data that represent the execution of the principles of SBD. However, not every Set Review culminated in a DDM. Sometimes, there was no DDM because we did not feel that the data supported the desired decision, so we gave that team more work to do outside the Set Review to further develop the {set} or conduct more (or different) analysis. Other times, there was no DDM because the Set Review was intentionally an in-process review to show preliminary results and receive feedback from the SSDM.

The DDM contains three parts: a standard form, the presentation given at the Set Review, and the minutes from the Set Review. First, we created a form to capture a summary of the event and the signatures of relevant stakeholders: Appendix B: Sample Design Decision Memorandum Form shows an example. Each DDM had a disposition from at least the SSDM; the program office and resource sponsor provided concurrence of approvals on a limited subset of the DDMs. The form was in an electronic format (including signatures) and loaded into our Integrated Data Environment (IDE) to execute the routing process online. This aspect was critical for a distributed design team, especially during the global pandemic. Eventually, we required the team to make the DDM and its attachments into a portfolio. This way, the entire package was together in one file for review instead of searching through the IDE and the workflow for at least three separate files.

The second standard part of a DDM was the presentation given to the SSDM for the decision, only altered if directed by the SSDM during the Set Review. For instance, the presenter may have suggested removing a portion of their design space above a specific value. The SSDM may have approved a different value based on his knowledge or interpretation of the data and in line with the risks he perceived. By including the presentation, the reviewers could reference the material from the meeting before signing. We felt this added value and, in some cases, was necessary. For instance, the reviewer may not have been at the meeting to hear the presentation and listen to the data presented and relevant questions and answers. Alternatively, if enough time elapses between the Set Review and the review of the DDM, memories may prove imperfect. This aspect was especially true once the design team executed several Set Reviews each week. We felt an additional value to requiring this was in retaining the data with the decision for posterity. The DDM form captured the decision; the presentation showed what data and assumptions led to that decision.

The third and final standard portion of the DDM is the Set Review’s meeting minutes. They capture any alterations and actions along with any conversation during the Set Review. Including them ensured the inclusion of genuinely all information relevant to a decision. The presentation includes the data in graphs and words and often generates conversations, questions, answers, actions, and clarifications needed to the information on the slides. Many of those aspects were germane to the decision, so we felt the DDM was incomplete without them. Notably, a subset of these meeting minutes is often a chat conversation captured verbatim. As with many teams during the pandemic, we used a meeting forum that allowed screen sharing, video, and chat for the team. These features enabled questions and answers using the chat function instead of interrupting the presenter, which allowed them to work on the answers at an appropriate time. It also allowed others to answer the question on behalf of the presenter, creating relevant information in parallel with the presenter. While we cannot capture the audible meeting minutes verbatim using transcription services through our tool, we capture these conversations from the chat window verbatim as part of the minutes.

In this way, the DDMs captured reusable design knowledge for posterity. The reuse could come in many forms. One reuse scenario is when the design effort incurs rework. In this case, the DDM helps scope the rework. It can help determine which analyses the team may need to conduct again and what other domains may be affected. It reveals what assumptions may come into question. In this sense, it informs the magnitude of the rework loop, whether it is local or more pervasive and impactful. Another reuse scenario is when another ship design effort occurs. In this case, the DDM captures knowledge that could benefit another ship class. The value depends on the nature of the study and assumptions and how specific they were to DDG(X). If the study and assumptions are general enough to translate to another ship design effort, our efforts could remove knowledge gaps for future ship designs. Should a question arise from another ship design team in the future, a review of the relevant DDMs for the domain in question could prove fruitful. The DDMs may prove helpful, for instance, in the power and propulsion domain, where we continue to create more knowledge regarding integrated power systems - and their controls - at this scale.

4.5 Training and mentoring the team, collective learning

Tell me and I forget. Teach me and I may remember. Involve me and I learn.

- *Benjamin Franklin*

With a philosophy, a team, and processes in place, we knew the next step was to bring them all together to ensure that the team executing our vision understood our philosophy and the processes. We accomplished this using a few different methods. First, we created training briefs and shared them electronically. We held training meetings in-person and virtually (at that time, we held virtual meetings because the team was distributed and face-to-face was not always possible; much of this took place in 2019 before the COVID-19 pandemic). We also held workshops and larger gatherings. Those gatherings brought the team together when we could and shared as much as possible as efficiently as possible. We anticipated that the new philosophy, method, and language might feel foreign to many on the design team, so we created many opportunities to train and mentor everyone. We also took every opportunity to assess how well the process was performing and adjust as needed.

Workshops were one method we used to introduce the material. We initially envisioned them as introducing the material for the flexibility domain because it had little historical precedent in the Navy and lacked strong domain expertise. During the build-up of material to support the flexibility workshops, though, we realized that these could serve two purposes by introducing both the philosophical basis and processes of SBD and how the flexibility domain would fit into the SBD construct. We conducted two workshops, one in April 2019 and one in June 2019. The first engaged the larger stakeholder community, while the second one was within a focused domain. The second workshop provided an opportunity to update the material and approach we used. We had an intent to conduct more workshops, but scheduling, resources, and other factors external to our control limited those possibilities.

In addition to the workshops, we created general training materials for consumption. We provided the materials to new team members as we generated them or as the members joined our team. Admittedly, we learned that we could have been better about reviewing the documents and guidance we created with the team and ensuring they understood the implications of the directives and DGMs. We assumed that the team read these documents when ready and should have done better to disseminate, train, and discuss policies instead. For instance, even some leadership team members did not fully understand the SBD philosophy and process we had established, which meant they could not help the SSDM and DSDM enforce the principles with the rest of the team. Another example is that we realized we may not have been clear enough on the objectives of the Set Review: to attain a DDM with a decision, recommendation, finding, assumption, or scope approval. As a result, some teams brought forward a Set Review in the wrong format and presented information and analysis before we approved the approach or assumptions.

In addition to general training materials, we learned and adjusted as we employed the process and realized that other design guidance was necessary. This realization created the second tranche of DGMs, shared in Table 2: Second Tranche of Design Guidance Memoranda . Notably, one of them involves team training. That DGM codifies the requirements for team members to conduct annual training such as cybersecurity and review several documents to familiarize themselves with the project and our processes. It delineates between indoctrination for new members and training for existing members and provides checklists for each. Hence, it responded to the lesson we learned: ensure consistent training and understanding of, first, how we chose to organize and run the team, second, how our instantiation of SBD operated, and third, the operational constraints such as schedule.

Table 2: Second Tranche of Design Guidance Memoranda

Number	Title	Approval Date
DGM 008	Design Margin Management	Pending
DGM 009	Critical Systems	20 Mar 20[5]
DGM 010	SWAP-C Reservation Guidance	07 May 20
DGM 011	Critical Systems Testing	20 Mar 20[5,6]
DGM 012	Indoctrination and Team Training	02 Jun 21
DGM 013	Naval Combatant Design Specification Review	30 Nov 20
DGM 014	Design Budgets	Pending

Much of our training was training by doing: on-the-job training. We spent a good portion of our time training and mentoring individual functional area leads and their team members during the initial Set Reviews and, in many cases, in separate meetings afterward. Each of the Set Reviews was an opportunity to learn, and at the beginning, the team learned a tremendous amount. After about one year, the Set Review schedule started to pick up, and we realized that we were training and mentoring the same learning points repeatedly with the different teams.

---

[5] Original release date. There have been subsequent updates to the guidance.

[6] Subsequently canceled: no longer a DGM.

Throughout the first year, we conducted each Set Review on an ad hoc basis with only one functional area at a time. We intentionally kept the team segregated (aided by the lack of a design site) and limited inter-team communications. This separation allowed them to establish their initial sets independently and avoid using each other to limit their design trade spaces artificially. The team was small enough that we could manage this way. Then, after about a year, we changed the structure of the Set Reviews to require each domain to have a representative present at each review since we felt the domains had adequately developed their sets. Conducting virtual reviews due to the pandemic enabled this.

This new structure accelerated the learning across the team since others could learn from what went well or did not from another analyst's presentation. It happened in July 2020, represented by the dotted line in Figure 7. Figure 7 also shows there were more Set Reviews (more circles), more regular Set Reviews (every Thursday of the week), and more variety in the design space (more rows in the chart). The dashed line separates the {sets} from the first year and those we added in the second year. Mentoring still occurs today, but less frequently. The team achieved better alignment to the method and with each other. They follow the format. They improved their communications because they knew they were briefing the entire team and had to speak in a way that all would understand. The minutes from the reviews reveal conversations regarding integration began, a topic covered in more depth in Chapter 5.

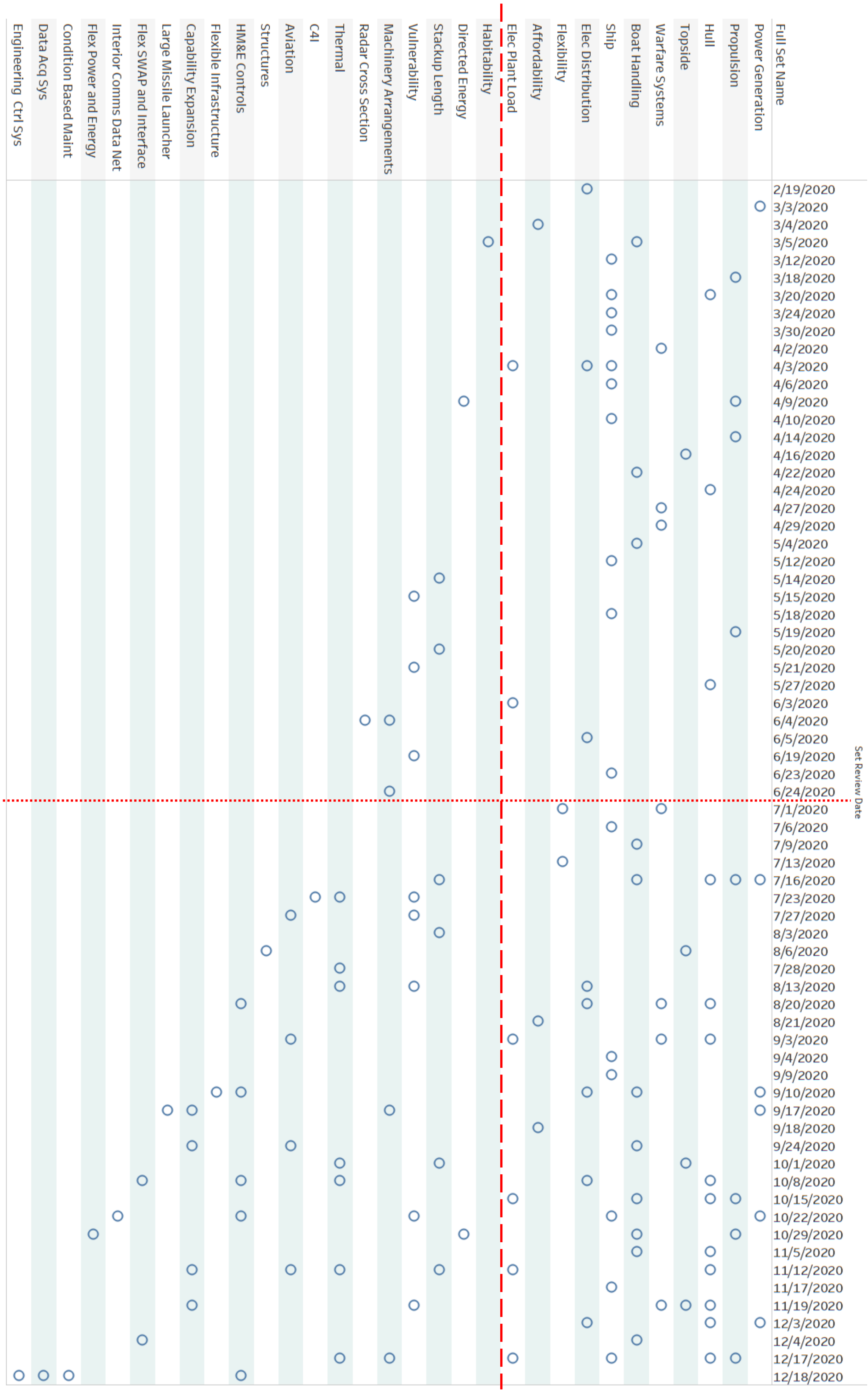


Figure 7: Set Review schedule for the second year

The on-the-job approach to the training and mentoring worked because of the size of the team. Because it was small, we could handle training and mentoring more closely with each team member at the beginning. If the team had started larger, we would have taken a different approach. We would have had to trust others to carry our message or conduct more workshops to train more people in large groups at once. In this sense, we felt fortunate that resource constraints kept the team small at first so we could conduct ourselves this way. We hypothesize that the design team would have had more difficulty achieving successful outcomes if the team had started larger. Instead, we formed a core team to train and grow. Therefore, by the time we integrated the team for Set Reviews and the process accelerated, that core team could extend the training for us since they had already achieved results themselves and could train new members toward successful outcomes. We suggest this reduced the risk for the standup of this process.

4.6 Executing the model and ruling out portions of the solution space

A process cannot be understood by stopping it. Understanding must move with the flow of the process, must join it and flow with it.

\- *Frank Herbert*

The final portion of the model executes the previous five sections; it implements the *philosophy* and *process* through the *team* that we *trained* and results in *documenting* the process and its outcomes. Generally, we executed this portion after developing the philosophy (4.1) and in parallel with the others (4.2-4.5), not strictly after their completion. The RET finished in December of 2018. Thus, the documenting, team-building, training, and executing started in 2019. Therefore, this research uses 2019 as “Year 1” and 2020 as “Year 2” for tracking purposes. The leadership team, especially the SSDM and DSDM, used the principles of action research to reflect on the entire ecosystem we established and intervened appropriately and when necessary. Figure 6 and Figure 8 revealed that we conducted the first Set Review in March 2019. {Manpower} appears to be the first set review, but this was a special review conducted by logisticians in the acting program office at the time. We commissioned the study with them several months before formalizing our processes and formats. The study created {manpower}, so we track it as a Set Review, but the first real learning opportunity began with {power generation} and {propulsion} in the Marine Engineering domain. Starting with this review, we reflected on whether all the elements of our model were aligning the way we envisioned.

{Boat handling}[7] provides a simple example of the running process and how small scope helped the team obtain several quick, successful outcomes. Figure 8 shows it and {power generation} obtained the first DDMs in our process during the same week in 2019. The meeting started with opening comments from the SSDM, where he addressed the first Set Review, at which the team struggled in framing the assumptions and approach to analysis. He also addressed his fellow TWH (a technical domain manager) for launch and recovery systems on the SBD process and why {boat handling} was one of the first {sets} in our design process. Then, he yielded the floor to the briefer, who proceeded through the prescribed format slide by slide.

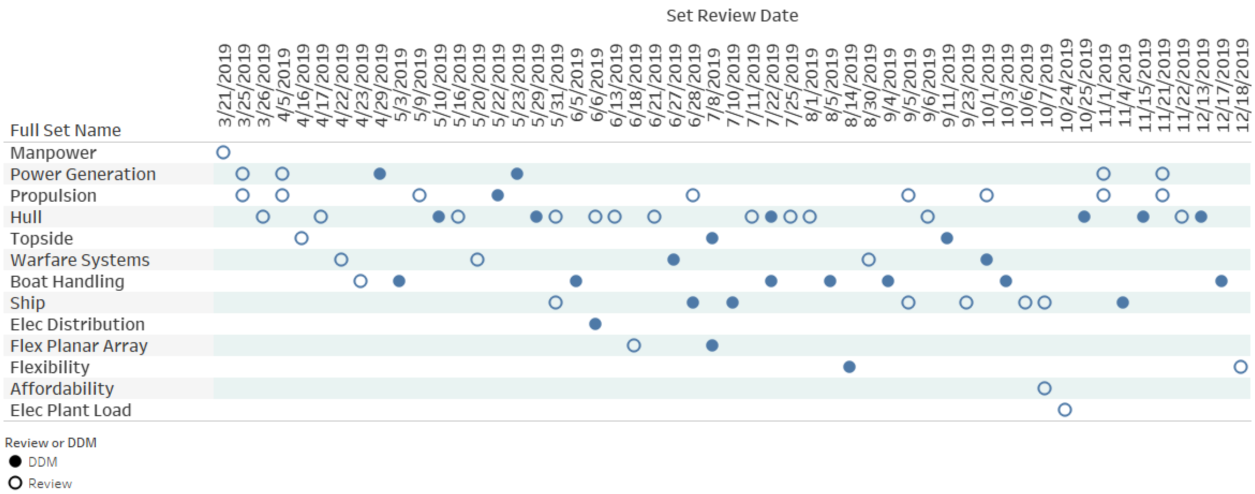


Figure 8: Set Review schedule and outcomes through the first year

The {boat handling} lead followed the Set Review format well. First, he briefed the purpose: approving assumptions, scope, and the initial design space. He included the scope: evaluate handling system types considering current Navy small boats for performance characteristics and ship impacts. Next, he presented, and we discussed, sixteen assumptions that they needed to conduct this analysis. Example assumptions included what constituted a small boat for launch and recovery from DDG(X), another for what would NOT launch and recover from the ship, and sea state characteristics used in performance evaluations. We approved nine of

---

[7] For those unfamiliar with naval vessels and their design, this domain explores the ship's boats and how to handle them, more specifically launch and recover them. This is separate and distinct from 'ship handling' which involves the rudders, appendages, and maneuvering of the ship itself.

the assumptions without change, five of them with change, rejected one, and required clarification from another TWH for one and left it open for a future Review. These ratios represent typical results from the first reviews of any {set}. Further, the discussion created through this process revealed other analysis scope and commensurate assumptions, which we resolved on the spot and approved three additional assumptions.

Next, the briefer shared their approach for developing {boat handling} including market research, generating {subsets} to {boat handling} and populating them with representative characteristics, and evaluating the set using guidance from DGM 005. He followed this by sharing the initial set space of sixteen options, including photographic examples for some of the options. After that, he highlighted seven attributes the team intended to use as primary evaluation criteria, including cost, weight, area/volume, and verified launch/recovery times. He also shared future anticipated evaluation criteria like maintainability and flexibility. He pointed out ten sources of uncertainty next, in line with the briefing format, and if the team addressed that uncertainty in any way, such as with an assumption. Even though there were no decisions for this first review, he kept that slide in the presentation and pointed out they were not requesting decisions or reductions but confirming assumptions and the approach. He ended with the next steps: expanding {boat handling} from the sixteen examples so far and where they would look to expand (e.g., USCG systems, foreign systems); developing any other necessary assumptions; conducting qualitative and quantitative analysis; and proposing the next Set Review in the following month to potentially recommend elimination of a portion of {boat handling}.

The meeting concluded after reviewing the calendar, approving the next Set Review, reviewing action items from the meeting, and closing with comments from the SSDM. Overall, we appreciated the conduct of the meeting and highlighted our appreciation for following the desired format.

This meeting and its subsequent reviews provide an excellent example of the working process. The team followed the structure and achieved the desired outcome, including three additional assumptions resulting from the review discussions. In the following review, the team updated on the status of the approach and market research results, including some data they obtained regarding the acquisition cost of the sixteen options. They also suggested the removal of one of the sixteen options. The removal suggestion took a white paper approach with TWH-validated qualitative evidence regarding why we should eliminate the particular option from further consideration. That approach highlights a feature of SBD: using both qualitative and quantitative results to remove portions of the design space from consideration. SBD accommodates a subject matter expert's (SME's) opinion in the process with appropriate justification in the documentation. In this third Set Review, the team presented the white paper, discussed its merits, and the TWHs collectively agreed to remove the option from {boat handling}.

The meeting also demonstrates the value of keeping the scope small and the decision space small and manageable when possible. While we would appreciate analyses that provide faster convergence rates to the eventual solution by removing larger swaths of the design space, we also appreciate the "quick win" element of making several less-impactful decisions in succession. {Sets} with smaller scope provide smaller successes more frequently than when the scope is larger, which naturally drives longer analysis times with greater complexity, less frequent decision points, but more impactful decisions as a result. For instance, in the marine engineering domain, at the same time we were gaining quick wins and decisions in {boat handling}, we also wrestled with {power generation} and {propulsion}, in which we explored the options for mechanical, hybrid, or integrated power architectures. This decision took over a year and had twenty Set Reviews leading up to it that addressed new assumptions or new knowledge gaps requiring adjudication to inform this more significant decision. However, when we ultimately decided, we eliminated two-thirds of the design space from further consideration. Hindsight now clarifies that the actual first decision was not on the architecture but rather on the multitude of other assumptions and analyses that preceded that significant and desired first decision.

Fortunately, future ship design efforts that consider engineering plant architectures will benefit from these lessons learned. The team created reusable design knowledge. They discovered dependencies within the domain and from other domains that this important architecture decision required. Future efforts can follow our discovery process instead of rediscovering the order to produce relevant data. Further, because the decision was at a high level of abstraction, many of the results can apply outside the ship class, subject to our assumptions. For example, some assumptions may limit the results to ships of a similar displacement range since displacement correlates to resistance which correlates to the power required to move the vessel through the water at given speeds. Other assumptions could limit the outcomes to vessels with similar service loads. Combat systems' loads contributed to our total projected load profile, but other vessels with different combat systems or different service loads may change the analyses and outcomes. They can use our process and outcomes considering each of these assumptions, changing the analyses, if necessary, with assumptions and data appropriate for their case. These could change the outcome or not, but in any case, they should make a future design team's work more efficient by using the foundational work conducted by our marine engineering team to establish the analysis within a set-based method.

Figure 7 also reveals we struggled at the beginning of the process, having several Set Reviews without DDM, including the first {boat handling} review. As revealed in Section 0, the first review for a new {set} typically covered the first half of the Set Review structure. The team proposed the study's purpose, the analytic approach, the assumptions, and the criteria for reasoning about outcomes. Since we typically gave the domain team the initial purpose of the initial sets, that section posed no issues. Typically, the approach section caused no issues, either. We spent many of the first Set Reviews working through contentious aspects of assumptions and criteria.

With assumptions, we rejected early assumptions that we felt limited the design space too much. Engineers develop different paradigms and rulesets based on their education and experiences. We found it helpful and enlightening to explicitly extract those design biases, understand them, and keep, change, or reject them with appropriate discussion and documentation. We rejected about one out of eight assumptions outright, and about half or more required rework before approval. Of note, even of the nine assumptions approved in the {boat handling} vignette above, the set team modified some of them from the previous review. Also, some teams created assumptions early in the process that we felt belonged in the "approach" section, but these were minor errors and did not hold up the process.

We noticed the team contended with generating appropriate criteria, especially in well-developed domains like naval architecture and marine engineering. More than others, those domains already contained criteria developed through decades of work. However, many of those analyses and criteria were appropriate for point designs but less so for SBD. For instance, in naval architecture, measuring a ship's stability generally requires at least a crude hull form to understand how the center of buoyancy acts in concert with the center of gravity to create stability or instability. However, in our version of SBD, we did not want the naval architects generating thousands of hull forms to test them each for intact stability. Instead, we wanted them to develop more general rules that they refined with time as we ruled out sections of the design space such as "longer than ___" or "wider than ___" or "with a length-to-beam ratio larger/smaller than ___." This method forced the team and the technical domain manager TWH for hydrodynamics to develop new methods and criteria to suit this method. Fortuitously, the methods and criteria they developed together translated to other ship classes and benefitted the Navy beyond this single ship class. The teams will use the analyses they proposed eventually, of course, but they did not serve to close the knowledge gaps in front of us at the time.

We also observed that teams often searched for criteria that would deem an area infeasible vice dominant. The nature of these two criteria made this outcome natural and expected. For infeasibility, failing to meet a single criterion eliminates that portion of the design space from further consideration with high confidence. However, for dominance, negotiations and balance must occur because dominance is more of a system issue than a {set} issue, meaning that a given solution point or solution set could dominate one criterion but not others. For example, in {power generation}, diesel engines may dominate in fuel efficiency, maintainability, and range of power outputs in the market. However, gas turbines dominate in power/weight ratios and noise aspects. Both types of engines can likely meet requirements, so eliminating portions of the design space eventually requires a balance with external factors and decisions, increasing the importance of the chief engineer/SSDM role to balance the requirements across all {sets}. One such early recommendation requested the removal of diesel engines from the design space entirely using a justification that it would not meet a particular requirement (infeasible). However, the SSDM knew that diesel engines could be incredibly valuable (dominant) for meeting other requirements. Hence, we helped commission a quick R&D project to test the viability of improving diesel system performance to meet the first requirement (infeasible to feasible). The successful test kept a large portion of the design space open for consideration instead of locking in a solution too early based on one criterion.

We observed that simpler, smaller Set Reviews were more likely to obtain successful outcomes in many cases. Keeping the scope smaller from review to review resulted in shorter and more manageable analyses that required fewer (potentially contentious) assumptions to complete. It ensured the team had less information to present during the review, which increased the chances of presenting all information and finishing on time, often with the desired outcome. Smaller scope generally led to a smaller breadth of integration concerns, also. Alternately, the smaller scope meant that the decision eliminated a smaller part of the design space from further consideration, and therefore more options were still left to explore and remove.

Two external phenomena presented challenges to the design team. First, the Navy offered a new opportunity to the author in August of Year 2, as the process began picking up pace. This move shifted the research from participatory action research toward action research, where the team continued to execute the processes and adapt them as necessary. However, the author no longer actively participated in evaluating and adapting with them. Therefore, observations and action research diminish significantly towards the end of Year 2 and discontinue in the third year as part of the transition to the new job. Therefore, all vignettes and interpretations for this research occurred in the first two years, but this dissertation reports some available third-year data. Secondly, outside factors affected the team and its activity. Specifically, resource constraints adversely affected the team's ability to make progress, causing the team to shrink instead of growing, thereby altering - but not eliminating - our ability to observe and adjust as the effort grew.

The team adapted well to these setbacks and learned something as a part of each Set Review. Figures Figure 8 and Figure 9 show one dimension of that learning. They repeat Figure 6 and Figure 7 and have the added dimension in the data that highlights when a Set Review resulted in a DDM. Table 3 summarizes some relevant metrics so that the reader does not need to count dots in the figures.

Table 3: Year over year performance metrics

	Meetings	Reviews	DDMs	Average Time to first DDM (days)	Longest Time to first DDM (days)
Year 1	53	66	26	49.2	164
Year 2	68	139	75	18.3	203
Year 3[8]	49	156	86	7.9	41

This added dimension creates the opportunity to discuss the evolution of the process between the first and second years. First, there were 15 additional meetings held throughout the second year, but now that we were more disciplined in our approach and started to combine Set Reviews, we held 73 additional reviews. Second, while more than doubling the number of Set Reviews, we also increased the number (and ratio) of successful reviews, with 75 (54%) resulting in a DDM versus 26 (39%) the first year. Third, the team's shared learning experience drove higher first-time success rates. We introduced 20 new {sets} in the second year, and eight of them (40%) had a successful outcome at their first review, versus three (25%) that had successful outcomes on the first try in the first year. Further, {Stackup Length}, {Vulnerability} and {Capability Expansion} obtained a successful outcome within one week of the original review, signaling that the issues with the original review were minor or that the meeting required more time to complete. If one counts these additional three as successful, it increases the metric to 11 out of 20 (55%) first-time successes during the second year. Finally, the increasingly successful first-time Set Reviews contributed to the metric regarding the average time it took to get the first DDM of a new {set}. During the first year, if a {set} eventually gained a DDM, it took on average almost 50 days from the first Set Review to the first DDM, whereas for {sets} started in the second year, it took approximately 18 days, on average, from the first Set Review to a subsequent DDM.

---

[8] Year 3 data is included because it is available but was not actively observed.

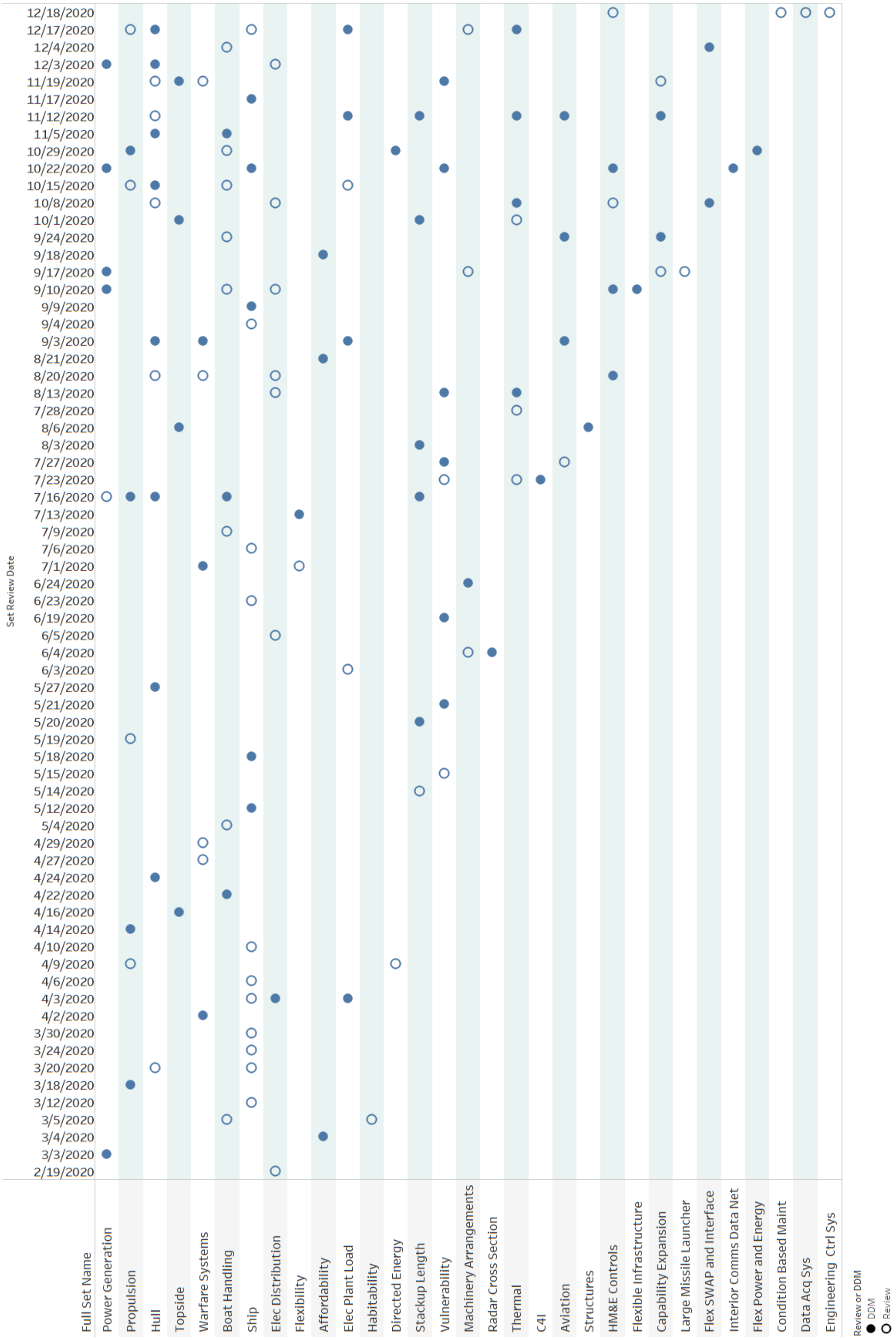


Figure 9: Set Review schedule and outcomes for the second year

While executing these Set Reviews, risks from previous sections still applied. We were frequently concerned with removing a portion of a design space that may remove options in other domains. Determining the next question or study always mattered. Choosing the right level of abstraction weighed on our minds. We understood the need to execute with haste to prove the approach, support requirements, and rule out dominated or infeasible portions of the design space. We worried we might watch the team operate as usual but think their methods are set-based. Furthermore, of course, we were concerned with setting up and intersecting {sets}.

Based on these risks and others, we proactively developed policies and procedures or reacted to unfolding events as a part of this execution phase. Previous sections related many of these already but deserve one more summary approach. Therefore, while we intended for the Set Reviews to execute the process, we realized they also provided an opportunity for feedback and mentoring and became the grounding and communication tools for the design team. At the beginning of the process, we limited team interactions to drive each domain team to develop their {sets} and {subsets} independently. Within about a year, the initial design spaces were developed enough. Around the same time, we noticed we were mentoring the individual domains on similar topics across Set Reviews and desired to reduce the repetition. Therefore, we shifted the approach around July 2020, bringing the whole team together for the Set Reviews instead of domain-by-domain and meeting-by-meeting for the leadership. This new approach accelerated progress in several dimensions because it allowed the team to learn from each other, provided the opportunity for integration by intersection (thereby reducing the risk of removing valuable portions of the design space), and provided a platform for team-wide training and mentoring when needed. This change in approach and its outcome proved invaluable because the timing was approximately coincident with the global pandemic. The Set Reviews with the entire team present broke down communications barriers and strengthened teamwork, despite most of the team working from home in lockdown. In fact, this created a new and efficient communications path with the chat function of our virtual meeting room. The Set Review framework enabled a distributed workforce to conduct business effectively.

We also had new risks and questions to address in this portion of the model. We needed to address when and how a {set} was "done" in this part. One interpretation from Figure 8 and Figure 9 concludes we have not "finished" a {set} yet. Each of the original {sets} from Figure 6 and Section 4.1 still actively conducted reviews towards the end of the second year (and into the third year, as revealed in Appendix C: Full Set Review Data Extraction Through Year 3. However, these basic figures do not reveal that each review and DDM is on a new knowledge gap or refinement of information with higher fidelity. Further, the team members determined what to call the {set} and leadership provided admittedly sparse guidance beyond the initial {sets}. Therefore, the figures also reveal an inconsistent approach to naming conventions and labels for Set Reviews. Each domain team proliferated {sets} differently. For some, each new study or design characteristic became its own {set}. For instance, {boat handling} created the subsets {side launched}, {stern launched}, and {side and stern launched}. Other teams added richness and fidelity to an existing {set}. For example, {hull} maintained the set of hull characteristics in increasing levels of fidelity as the naval architecture team conducted more analyses from stability to seakeeping to resistance. Also, even if a {set} was a {subset} within a domain, some teams referred to the highest abstraction of {set} instead of the {subset}. For instance, we never conducted a {stern launch} review but rather had a {boat handling} review where we explored {stern launch} options of that design space. Thus, certain {subsets} completed, like {stern launch}, but the sets of higher abstraction, like {boat handling}, remain open while the team refines analyses and converges the trade space to the initial baseline one {subset} at a time. These varying approaches highlight the lack of guidance for these aspects of the process. It is uncertain whether guidance of this nature is important for the process. While lack of consistency affects metrics and analyses like these, Table 3 suggests that convention consistency may be less important to successful process outcomes since the team consistently had more reviews in fewer meetings and more DDMs from those reviews.

Further, we carefully monitored for any potential changes needed to enable the scaling of the process. We were concerned that the processes and documentation could be burdensome and stall the process as the size of the team grew, the number of {sets} expanded, and the fidelity of analyses increased. We continue observing the process for these answers. Table 3 shows the process and documentation are robust to this growth since the team produced more DDM from more Set Reviews conducted over fewer meetings. Despite the challenges and constraints faced during this period, the design team and the process performed admirably in creating these noteworthy results.

Another interesting aspect of our effort was the concurrent development of requirements and design artifacts. Ward (1990) offers that this ought to be the approach for complex projects to resolve the requirements paradox: rigid requirements at the beginning are risky due to new information that could change them, but flexible requirements are not requirements at all. Kennedy et al. (2014) also suggest this as a preferred approach to reduce rework loops. We agree with these authors but do not find many analogs in literature to emulate, leading to everything that encompasses this chapter.

Part of what makes this difficult is simultaneously focusing on the material aspects and functional, behavioral, temporal, data flow, and other architectural elements that help inform the requirements. There is not always a good translation between the two. We had a workforce primarily experienced with the methods where requirements developed independent of architecture. Those developing requirements passed them ‘over the wall’ to the engineers and designers to make something that meets them without solid feedback loops. So, this was a welcome challenge, but another one without many analogs. It is exciting to see the Navy moving in this direction, in any case. In a sense, we seemed to be merging the fields of naval engineering and operations research, which is a new discipline to ask many naval engineers to have. It is possible, but probably not in their school curricula, and picked up strictly via on-the-job training. There is potentially an approach where a specialist group with this skill set can translate between the material and functional forms to help more efficiently conduct these simultaneous activities. However, we did not have an opportunity to test such a method this time around.

As the process played out, the first knowledge gaps and {sets} proved resilient. Many of them were the right first question to ask and progressed the knowledge of the dominant design space. The result of this scenario was that once we made the higher abstraction decision and the knowledge gap closed, the team naturally moved to a lower level of abstraction and used the same methodology (requirements, known risks, critical path analysis, external sources, expert judgment, and experience with previous projects) to determine what the knowledge gaps were. Another possible outcome was that the analysis result naturally revealed the next knowledge gap to investigate. For example, choosing a generic location for a boat leads to adding fidelity to that location, such as placing on the weather decks or integrating it into the ship’s structure.

Narrowing that knowledge gap leads to higher fidelity information such as height above the waterline and longitudinal locations.

In some cases, though, the knowledge gap could not be resolved, and the process of trying to close the gap revealed a different knowledge gap to resolve first, as with the {power generation} and {propulsion} vignette previously mentioned. The beauty of this outcome is two-fold: first, all the work on the first knowledge gap is not lost, nor does it need to be restarted; it simply needs to wait for the new knowledge gap to resolve (or for one to make a temporary assumption), and second, that the process still created knowledge: to paraphrase former Secretary of Defense Donald Rumsfeld, we turned an unknown-unknown into a known-unknown through the discovery of the new knowledge gap. In this way, many new {sets} and analyses spawned from these original decisions. The leadership team or one of the domain teams still jump-starts some {sets}, such as {arrangements}, instead of resulting from one of the original {sets}. This outcome is acceptable and expected for a project of this magnitude and complexity.

5. Integration by Intersection

Now that the model and its execution are known, this Chapter focuses on a critical principle of set-based design (SBD): integration by intersection. Much of the literature that educates on SBD gives little treatment to the mechanics and mathematics of this integration (Ghosh and Seering 2014; D. Singer et al. 2017; Gray 2017). Instead, they provide graphics and descriptions that relate that integration by intersection happens, and the reader must intuit how it applies. Figure 10 presents an often-cited example from Bernstein, and derivations of it abound. Accordingly, the literature reviews also identify this tenet as a knowledge gap worthy of study (Toche, Pellerin, and Fortin 2020). This Chapter offers a contribution to the development of this principle from what we experienced in practice.

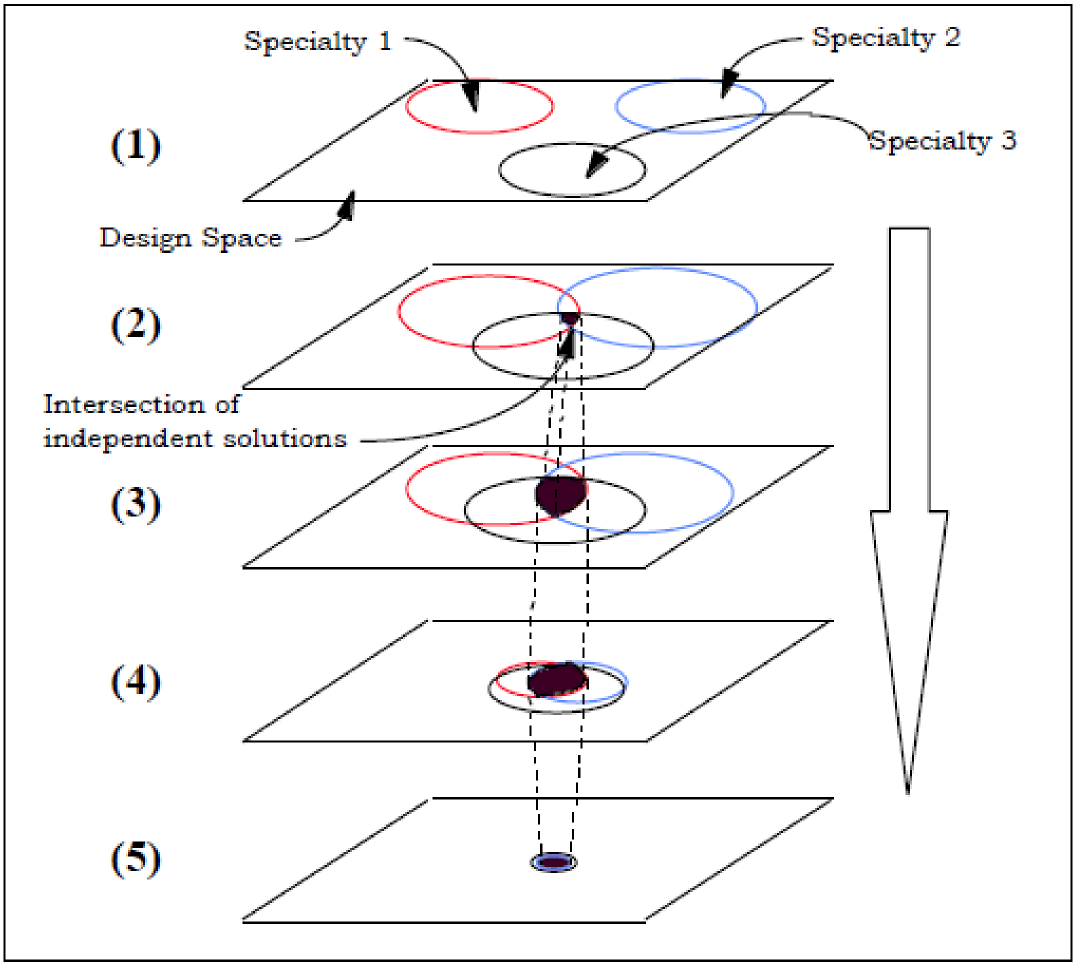


Figure 10: Bernstein's view of set-based concurrent engineering (Bernstein 1998)

First, this Chapter's discussion warrants a short review of set theory and notation. Specifically:

a) List elements of a set in braces.

$$DESIGN\_SPACE = \{ship_1,\ ship_2, ship_3, \ldots, ship_n\} = \{ships\}$$

This means that the variable *DESIGN_SPACE* is a set we call {ships} that contains a set of ship concepts from 1 to n. An individual ship concept is represented by [ship].

b) A vertical bar or colon stands for "such that."

$$l_{ship} = \{l_{hull} |\ l_{hull}\ is\ an\ integer, and\ 180m < l_{hull} < 200m\}$$

This means that the length of the ship is the set of {hull lengths} that are integers between 180 meters and 200 meters.

c) Denote membership in a set with ∈; a set containing an element as ∋; and lack of membership with ∉.

$$188m \in l_{ship}\ or, in\ reverse, l_{ship} \ni 188m$$

This means 188 meters is a member of the set that defines {ship length} or, in reverse, that the set of {ship length} contains 188 meters as one of its elements.

$$150m \notin \mathrm{l_{ship}}$$

This means that 150 meters is not a member of the set that defines ship length.

d) Denote subsets with ⊂ and supersets with ⊃.

$$shortlength = \{180m, 181m, 182m, 183m\} \subset l_{ship}$$

This means that the variable '*shortlength*' with the listed values is a subset of ship length because every element of '*shortlength*' is in l_{ship}.

$$l_{ship} \supset shortlength$$

This means that l_{ship} is a superset of the variable '*shortlength*' because it contains the values of '*shortlength*' and other variables not in '*shortlength.*'

e) Defining sets allow algebraic operations, specifically unions, intersections, complements, and Cartesian products.

$$A \cup B$$

Represents the *union* of A and B: the set of all things that are members of A, B, or both A and B.

$$A \cap B$$

Represents the *intersection* of A and B: the set of all things that are members of both A and B.

$$A - B \text{ or } A/B$$

Represents the *complement* of B to A: the set of all elements that are members of A but not members of B.

$$A \times B$$

Represents the *Cartesian product* of A and B: the association of every element of A with every element of B. For instance:

if A= {engine type} = {diesel, gas turbine} and B= *shortlength*

then A x B = {(diesel,180), (diesel,181), (diesel,182), (diesel,183),
(gas turbine,180), (gas turbine,181), (gas turbine,182), (gas turbine,183)}.

f) Then, one can apply the commutative, associative, and distributive properties to these operations appropriately.

g) *Partitioning* a {set} divides it into pairwise disjointed {subsets}. This means elements in one partition cannot exist in another partition of the same {set} and that the union of all partitions constitutes the {set}.
For a simple example, we may partition *shortlength* by even and odd numbers, and create $\{\text{shortlength}_{\text{even}}\}$= (180,182) and $\{\text{shortlength}_{\text{odd}}\}$= (181,183). There are no common members of $shortlength_{even}$ and $shortlength_{odd}$, and their union equals *shortlength*.

With this knowledge of set notation, we can begin to reason about integration by first understanding sets. To understand sets, we ought to start by addressing points. SBD calls for a new way of thinking about points: we should rarely think of them as points. Instead, we ought to think of points as either members of a {subset}, representations of a range of possibilities, or as intersections of {sets} of lower abstraction. Our design process observed each of these.

First, we can think of points in a design space as members of {subsets}. Figure 11 provides a graphic representing this principle. In fairness to their graphic's original intent, Shallcross et al. showed the value of SBD approaches with a wider range of options, including those on the Pareto, over point-based design (PBD) approaches, which consider fewer alternatives. Therefore, the legend distinguishes between the two. Nonetheless, each of the symbols represents points in the design space, and, appropriately, {subsets} in the space contain all of the alternatives from PBD. One could create several views from the data set, organizing {ships} by {subsets} like {hull type}, {power generation architecture}, {warfare system}, and other discriminating factors and {subsets}. Each view would potentially take a different shape as the points moved according to the different value-to-cost measures, and each {subset} likely takes a different shape, also. Also, these types of sets may not distinguish themselves as clearly as in Figure 11, depending on axis choices and how the population of points corresponds to those choices. Neither {subsets} nor characteristics of a {set} may group so cleanly.

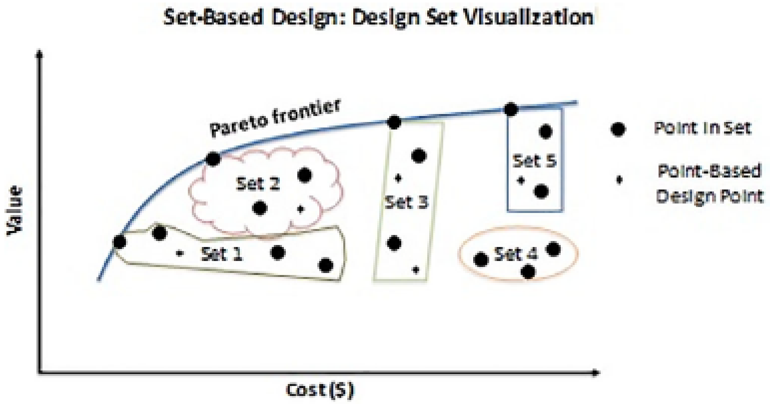


Figure 11: SBD design set visualization (Shallcross et al. 2020)

Alternatively, we can think of points as representations of a range of possibilities. For instance, the four points in "Set 2" of Figure 11 represent the entire space encompassed by the cloud shape of the set. Creating a {subset} of representative points in the {set} simplifies computations and analysis if they represent the rest of the set in a meaningful way. If the {set} makes it through a decision gate, the team can add fidelity and expand the {set} space, as we did in our process (recall the "upside-down Christmas tree" of Figure 2), which could fill in more of that 'point cloud.' Further, points represent ranges even at the lowest level of abstraction when considering tolerances and variation; one or more characteristics that define that point will have ranges to them, and those ranges may be significant. The steel plates of a hull provide an example of this phenomenon. At this lowest abstraction, the characteristics captured may be elements like steel type (AH36, DH36, HT36, HY80, HY100, …) and thickness (5mm, 8mm, 12mm, …), both of which are subject to some level of variation because of imperfect chemistries and processes. Using points instead of ranges in these cases once again simplifies the analyses with some finite risk that the point will not adequately represent the range. The variation may not be significant in the concept design stage but carry more significance in later design stages, especially detail design.

Lastly, design points can represent intersections of {sets} of lower abstraction. In this way, each point is a set of characteristics that define the point. This is precisely the case with our {ships}. Each "point" in {ships} is actually a [vector] that contains "points" and ranges from partitioned {subsets}. Figure 12 shows this principle. In this sense, it may be more useful to think of a point in a design space as a specific instance of architecture selections and characteristics that define that point instead, like a [vector]. For instance, with the Requirements Evaluation Team (RET), that team created thousands of “point” designs. However, each of those points represented balanced ship concepts, each of which had engines, a hull form, a combat system, auxiliaries, and other defining elements and characteristics. So, those thousands of points in a design space were millions of data points represented at a high level of abstraction as a balanced ship concept. Each “point” design consisted of dozens of variables converged to a balanced solution after hundreds of calculations.

$$ship_a \equiv \begin{pmatrix} hull_i \\ warfare\ system_j \\ propulsion_k \\ power\ distribution_l \\ power\ generation_m \\ topside_n \\ \dots \\ flexibility_x \\ survivability_y \\ element_z \\ cost \\ characteristic_b \end{pmatrix}$$

Figure 12: Set Intersection Representation

The principle of points being pointers to sets applies to continually lower levels of abstraction in the design space. In our case, {ships} is the highest-level abstraction: the item we ultimately select, design, and acquire. Within {ships}, [$ship_a$] presents an intersection of the constituent {sets}; a collection of selections from {hull}, {warfare system}, {propulsion}, and others at the next lower level of abstraction. Correspondingly, that selection from {warfare system}, [$warfare\ system_j$], is an intersection of a specific collection of its {subsets} such as {guns}, {missile launchers}, {radars}, and {satellite communications}. In concept design, this principle continues to a level appropriate to rule out portions of the design space until the process ends with establishing of the baseline. In later design stages, this principle could continue to even lower levels of abstraction to aid in material and component selections, if appropriate.

One other aspect of Figure 12 deserves attention: the characteristic ‘cost.’ {Sets} have both {subsets} and characteristics. The {subsets} allow us to divide the space into useful organizations for decisions and analysis, and the characteristics provide the means to make decisions within the spaces through metrics, comparisons, intersections, and analyses. One can represent both elements and characteristics as {sets}, and they each have their use. Therefore, we end up with sets of {sets}, each with both elements and characteristics. It all has structure, so we know what is in and out of a {set} or [vector] and how to reason about it. When reasoning about sets, we must distinguish between these set elements and characteristics. An element of a {set} represents an architecture, a system, a subsystem, or something else carrying out a function. The elements divide the space into useful {subsets}, and in this sense, elements and sets are synonymous, as are subsets and sub-elements. A characteristic defines the {set} in a useful way and is typically the result of calculations or represents physical characteristics and metrics and measures of the set. Basically, characteristics have units associated with them, like length, mass, or dollars. In this case, the *cost* is a characteristic of the ship resulting from roll-up costs calculated from the aggregate of the other elements and their subsets. We can partition characteristics if useful or necessary, such as breaking up the cost into the basic cost of construction and other costs. Characteristics have variations stemming from uncertainty.

These aspects present an obvious principle worth stating. A given element of a superset (a {set} of higher abstraction) can only contain one element of a {subset}, but an element of a {subset} can be a member of multiple {supersets}. For instance, in our case, this translates to [$ship_a$] can only have [$hull_i$] defining it, it cannot have [$hull_i$] and [$hull_{i+1}$] define it, but [$hull_i$] can define [$ship_a$], [$ship_{a+1}$], and [$ship_{a+2}$], while [$hull_{i+1}$] defines [$ship_{a+3}$], [$ship_{a+4}$], and [$ship_{a+5}$]. This requires careful partitioning of the {subsets} and may require creation of {sub-subsets}. Explicitly stating this principle allows us to transition from discussions of {sets} to their intersection.

Intersections occurred between {subsets}, not within them. A proper collection of {subsets} partitions the {set} it in a way that creates Ø for intersections within. For instance, within {propulsion}, a propeller is either fixed pitch (FPP) or controllable pitch (CPP). However, variations in the {subsets} for {FPP} and {CPP} may have characteristics like propulsive efficiency. At this level of abstraction, the {sets} intersect to share portions of the design space, which leads to keeping or removing these intersections depending on their valuation. Of course, the intersection of these subsets around propulsive efficiency could create Ø, which could present this characteristic as a determining factor in removing or keeping {FPP} or {CPP} in {propulsion}.

Further, intersections occurred between characteristics defining {sets}. For example, at face value, it is difficult to determine the meaning of {power generation} ∩ {hull}. However, {power generation} may partition to a point where we get the characteristic $length_{prime\text{-}mover}$, while {hull} may partition to a point where we get the characteristic $length_{main\text{-}engine\text{-}room\text{-}1}$. These characteristics of the {sets}, often carried up from their {subsets}, make the intersection of these sets more intuitive since the dimensionality matches. This matching provides an important understanding of Figure 10 (Bernstein's view of set-based concurrent engineering): the intersecting circles are not merely nebulous groupings of things but rather characteristics of different domains that match dimensionally. If those characteristics have ranges, the intersection is the overlap of the ranges and may occur across multiple functional areas as appropriate. However, the intersection may not necessarily represent feasibility, dominance, or selection criteria. In the instance of the current example, suppose that:

$$length_{prime-mover} = \left\{integer \middle| 5m < length_{prime-mover} < 8m\right\}$$

And

$$length_{main-engine-room-1} = \left\{integer \middle| 7m < length_{main-engine-room-1} < 10m\right\}$$

In this instance, we do not want the lengths to match; we want the length of the prime mover to be less than the length of the main engine room so that it fits in the space. Therefore, in this instance, $length_{prime\text{-}mover} \cap length_{main\text{-}engine\text{-}room\text{-}1}$ offers no value as a standalone analysis or integration activity. In this sense, SBD integrates by reasoning through rules of intersection versus strictly integrating <u>by</u> intersecting. In this case, we would create a rule that defines the feasible set as $length_{prime\text{-}mover} + c < length_{main\text{-}engine\text{-}room\text{-}1}$. The constant in that definition represents a value such as a maintenance envelope or producibility factor we could incorporate. Defining the appropriate characteristics of a {set} allows the integration activities and communications to occur.

Intersections inform both contractions (reducing the design space) and expansions (exploring more design space). The reasoning and rules created by examining how {sets} intersect could lead toward contracting the design space. For instance, we may choose to remove the portion of $length_{main\text{-}engine\text{-}room\text{-}1} \leq 8$m or the portion of $length_{prime\text{-}mover} \geq 7$m if adequately justified. Alternately, reasoning about the {sets} could reveal that a particularly long prime mover has desirable performance characteristics like power output in another dimension. Suppose we desire to keep this prime mover in the trade space for power output. In that case, we may choose to expand the $length_{main\text{-}engine\text{-}room\text{-}1}$ {set} to avoid this scenario until more information is known, including if extending the length of the main engine room is feasible.

We made generalizable decisions with sets that could not generalize with points through these intersections and rules regarding intersections. SDB helps reduce risk in this way because it permits making decisions with imperfect or incomplete information. SBD permits this because selecting something (PBD) carries much more risk than removing something (SBD).

These intersections, discussions, and integration occurred during a Set Review for our design team. More specifically, they occurred during Set Reviews beginning in July of the second year, once we required all domains to have a representative empowered to speak on their behalf attend each Set Review. To reiterate, prior to this, we carefully controlled initial communications between the domain teams to prevent the risk of undue influence from others into their {sets}, thereby focusing on their set's criteria. This separation forced the design team away from the traditional practice of creating a point and iterating from it by limiting their ability to balance their solution space within the larger ship design space. They required faith that we would not let them develop a {set} that would not intersect with others. The {ships} managed this perceived risk, as Chapter 6 reveals.

Integration occurred in one of two ways: through discussions among the design team at a Set Review or through the Design Integration team, who was responsible for the {ships}. We were developing a holistic model hosted in a model-based systems engineering (MBSE) environment concurrently with the rest of the design activities. Therefore, we had no strict negotiation protocol nor any digital threads tying together the various characteristics of the {sets} within the design space and aiding the integration and analysis. As a result, frequently, integration resulted organically from presentations at the Set Reviews. We empowered design team members from other domains to have a voice equal to the SSDM and DSDM during the Reviews. This led team members to speak up (or type a question in chat) during another domain's review to ask a question along the lines of, "did you consider ___?" or "can it accommodate ___?" Sometimes the team member might outright state the potential outcomes to their domain resulting from how the other domain team was reasoning about their solutions. In each case, the team's interactions and integration transpired informally compared to how the earlier parts of this Chapter communicate them. We did not have any Set Reviews with the stated intent for the marine engineering domain and the naval architecture domain teams to negotiate $length_{prime\text{-}mover}$ and $length_{main\text{-}engine\text{-}room\text{-}1}$. Rather, we made each domain responsible for thinking about their intersections and speaking up if they felt a decision might adversely affect their design space or reduce a part of the design space not formally under consideration at that review. Informally, we have evidence that these conversations took place outside of Set Reviews. We accepted and expected this outcome. That practice did not affect the outcomes of the Set Reviews since only the SSDM could remove portions of anyone's design space as a matter of policy. Instead, the conversations regarding intersections and integration outside of the Set Reviews created valid data for the decisions. They prevented a domain from requesting a decision that had not been thoroughly vetted and allowed that vetting to occur outside Set Reviews.

Outside the Set Reviews, the DDM served as an integrating document. It provided one last opportunity for a domain team and relevant TWH to comment on a decision and how it may affect their design space.

Lastly, the {ships} integrated {subsets} into a balanced solution set. Each [ship] served a purpose in closing a knowledge gap or proving the possibility of integration at the concept level. Chapter Six delves into these purposes more.

6. Benchmark Ships, Virtual Prototyping, and Set-Based Design (SBD)

Practice safe design: Use a concept.

-Petrula Vrontikis

This chapter reveals a counter-intuitive concept to SBD: the creation of point designs. It will discuss why we created them, what purpose they served, and how creating points in the design space fits the SBD paradigm.

Many SBD cases in the literature use point designs (Frye 2010; Garner et al. 2015; van Oers et al. 2018; Specking et al. 2021). However, the nature of those cases using points is different from how we used them with DDG(X). The RET provides one such case that is representative of others. The RET effort used a synthesis code and designer input to generate a Latin hypercube sample set of design alternatives across dozens of variables. First, the process generated over twenty thousand ships and their constituent characteristics for evaluation. Then, the team organized the results into sets. This organization allowed them to sort by input variables or output characteristics; anything tracked by the model. Those results, often represented by ‘point clouds’ with different colors or symbols to call out regions of similar elements or characteristics, allow the team to infer relationships in the data, as in Figure 13. Next, the teams might load the model outputs into statistical software to help them visualize the data or reveal patterns in the data. Subsequently, they reduce the trade space by reasoning about these conclusions. For instance, if the y-axis of Figure 13 is ‘cost’ and the x-axis is a technical characteristic, one might conclude that Characteristic 3 (orange) is, on average, more expensive than the alternatives. Alternately, one could conclude that Characteristic 4 (green) has higher values in the technical measure, on average, although whether this is ‘good’ or ‘bad’ depends on that technical measure. A team using these techniques would consider one or more characteristics ‘dominant’ in this respect. The team may then look at other relationships between other characteristics and find other dominant characteristics. Thus, the selection of the baseline becomes a point within the space that has many dominant characteristics across the various relationships examined. In this sense, such a team treats the entire design as emergent, only understanding the design and reasoning about it after synthesis code has created it.

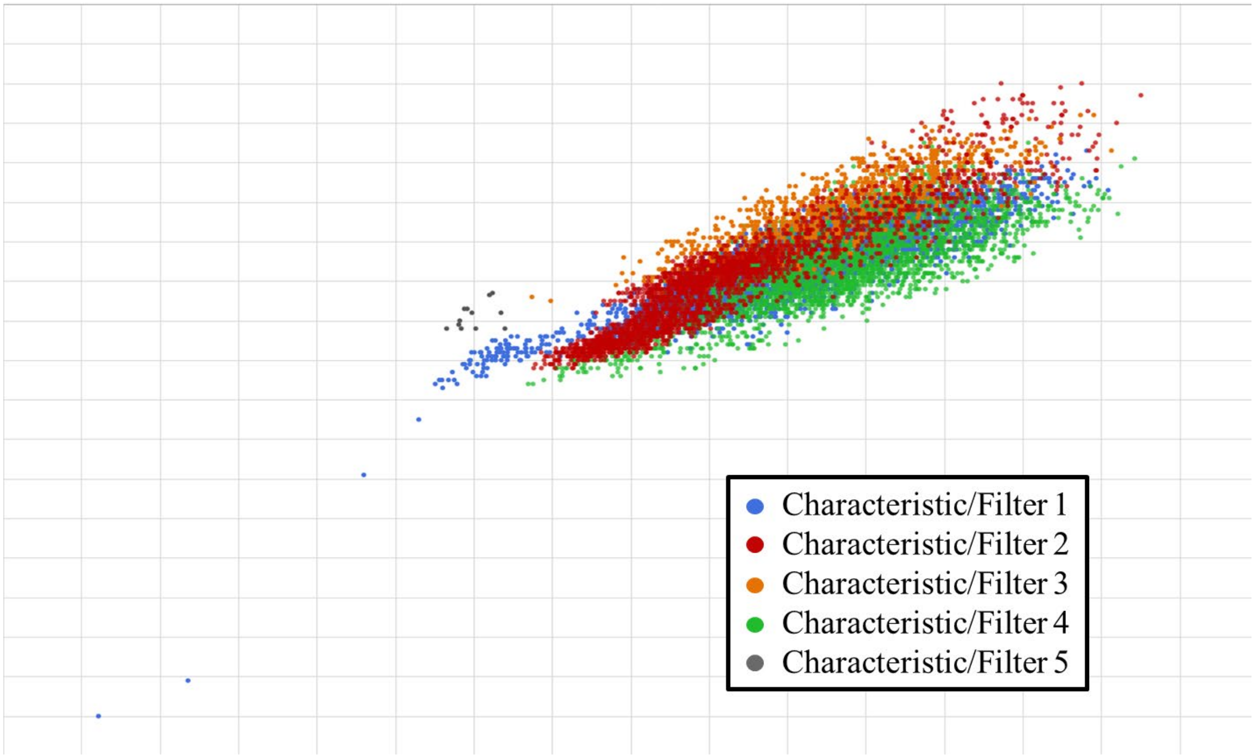
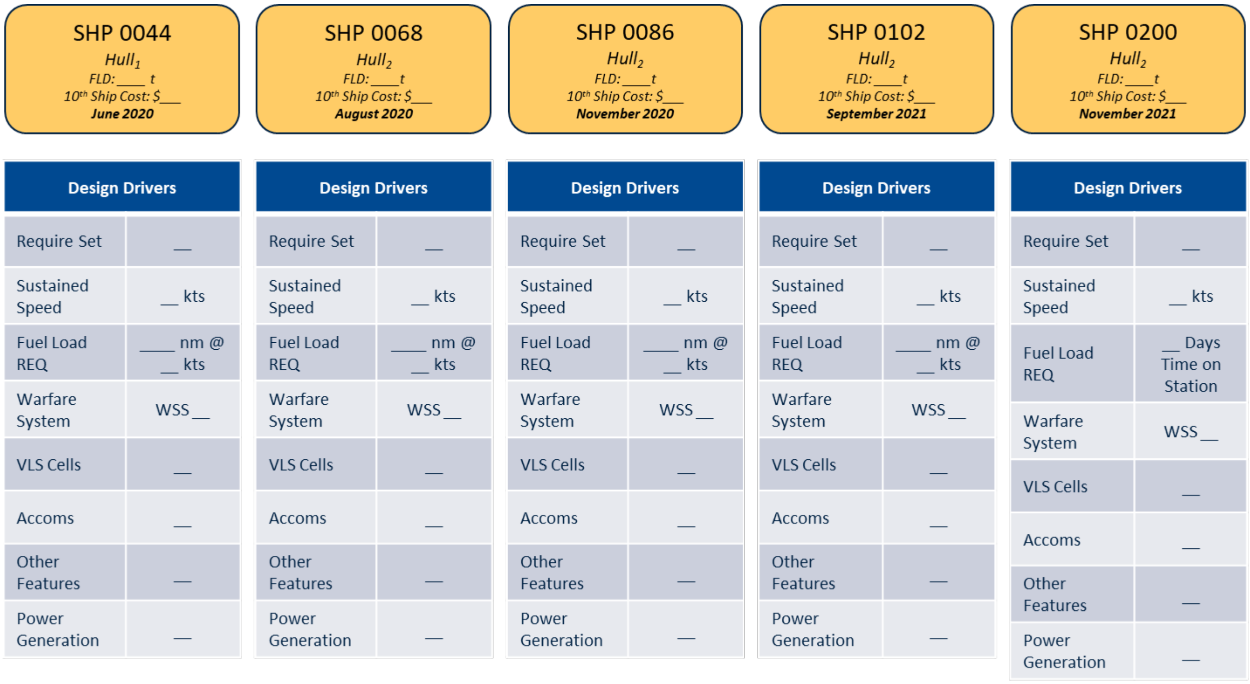


Figure 13: Point Cloud Representation of a Trade Space

This approach is close to SBD but not SBD according to agreed-upon principles. Rather, this approach explores design points with set-based reduction techniques, which is more like going from PBD to many-points-based design. Therefore, it partially meets the first principle: mapping the design space. It enables meeting the third principle, establishing feasibility before commitment, but does not guarantee its application, since it cannot guarantee conceptual robustness. In other words, changing one parameter slightly may not change the viable solution set to adjacent points in the cloud. Instead, it may change the solution set to a dominated or infeasible one. However, it does not meet the second principle; there is no integration by intersection, only integration by execution of synthesis code. This technique is also less efficient because it creates thousands (if not millions) of design points and data for evaluation, only for much of it to be thrown out. The Navy design community is fortunate for today's tools and computing power, which remove hesitation to create millions of data points. However, these methods leave less room for innovation or creativity, necessary elements of design, because the reasoning about the design space occurs after its creation, not before. The methods remove the decisions from an engineer or designer and instead delegate them to a computer program and the whims of the code. Further, the engineer evaluates the design space from synthesis results, leaving less ability to influence solutions in unique ways. Those usually involve "tricking" the existing tool or creating a new one. Tools like synthesis codes can provide value when used intelligently, but the tool can also become a crutch.

In contrast, {benchmarks} are a special subset of {ships}. Each represents a fully balanced ship design: a 'point' in the overall design space. Each [benchmark] and [ship] consists of a controlled set of {subsets} of the design space: a curated combination of sub-elements and characteristics that serves a purpose. Figure 14 shows a sampling of {benchmarks} and some of the distinguishing characteristics. Knowledge gaps determine the purpose and guide the curation of the {subsets}. Therefore, we only produce a few dozen elements of {ships} each year, and even fewer {benchmarks}, versus the thousands created in other efforts. **Error! Reference source not found.** shares a sample set of {ships} in tabular format.



SHP 0044
$Hull_1$
FLD: ____ t
10th Ship Cost: $___
June 2020

Design Drivers	
Require Set	__
Sustained Speed	__ kts
Fuel Load REQ	____ nm @ __ kts
Warfare System	WSS __
VLS Cells	__
Accoms	__
Other Features	—
Power Generation	—

SHP 0068
$Hull_2$
FLD: ____t
10th Ship Cost: $___
August 2020

Design Drivers	
Require Set	__
Sustained Speed	__ kts
Fuel Load REQ	____ nm @ __ kts
Warfare System	WSS __
VLS Cells	__
Accoms	__
Other Features	—
Power Generation	—

SHP 0086
$Hull_2$
FLD: ____t
10th Ship Cost: $___
November 2020

Design Drivers	
Require Set	__
Sustained Speed	__ kts
Fuel Load REQ	____ nm @ __ kts
Warfare System	WSS __
VLS Cells	__
Accoms	__
Other Features	—
Power Generation	—

SHP 0102
$Hull_2$
FLD: ____t
10th Ship Cost: $___
September 2021

Design Drivers	
Require Set	__
Sustained Speed	__ kts
Fuel Load REQ	____ nm @ __ kts
Warfare System	WSS __
VLS Cells	__
Accoms	__
Other Features	—
Power Generation	—

SHP 0200
$Hull_2$
FLD: ____t
10th Ship Cost: $___
November 2021

Design Drivers	
Require Set	__
Sustained Speed	__ kts
Fuel Load REQ	__ Days Time on Station
Warfare System	WSS __
VLS Cells	__
Accoms	__
Other Features	—
Power Generation	—

Figure 14: Sample View of Benchmark Ships and Characteristics

Table 4: Sample Set of Ships, Benchmarks, and Excursions

Ship #	Date	Description
SHP0004	9/5/2019	FY19 Benchmark 1
SHP0005	9/5/2019	FY19 Benchmark 2
SHP0006	9/5/2019	FY19 Benchmark 3
SHP0007-0012	5/1/2019	Affordability Study Ships
SHP0014	9/5/2019	Benchmark 4
SHP0016	9/5/2019	SHP 0006 Benchmark Excursion (speed and engineering plant)
SHP0017	10/22/2019	SHP 0004 Benchmark Excursion (speed and engineering plant)
SHP0018	10/22/2019	SHP 0005 Benchmark Excursion (flexibility)
SHP0020	10/22/2019	SHP 0005 Benchmark Excursion (endurance)
SHP0021-0022	10/22/2019	SHP 0020 Excursions (endurance)
SHP0025-0035	2/13/2020	Machinery SELECT Concepts
SHP0036, 37, 42	3/25/2020	OPNAV Initial Requirements Validations
SHP0041	5/21/2020	Diversity of Thought Independent Look
SHP0043	5/20/2020	FY20 Benchmark
SHP0044	5/20/2020	FY20 Benchmark
SHP0065-0066	6/23/2020	Vulnerability Excursions
SHP0068	7/13/2020	FY20 Benchmark

Benchmarks act as a virtual prototype to test certain hypotheses or create specific knowledge. The Navy cannot produce prototypes of our vessels at the system-of-systems scale as automakers do with their cars or vendors with their subsystems. Even scale models of our ships cause their testing to fall on the critical path. Therefore, the Navy uses tools that allow us to create virtual representations of the ship with many relevant characteristics. Their virtual prototypes inform the characteristics of the scale models we generate for further testing and validation.

{Benchmarks} provide a starting point for excursions. When we had a knowledge gap requiring a benchmark to resolve, we planned a family of ships to prototype. The benchmark of the family served as the starting point for comparison. The excursion ships changed a focused set of {subsets} or characteristics from that benchmark. Figure 15 shows how a benchmark can branch into many related excursions and studies.

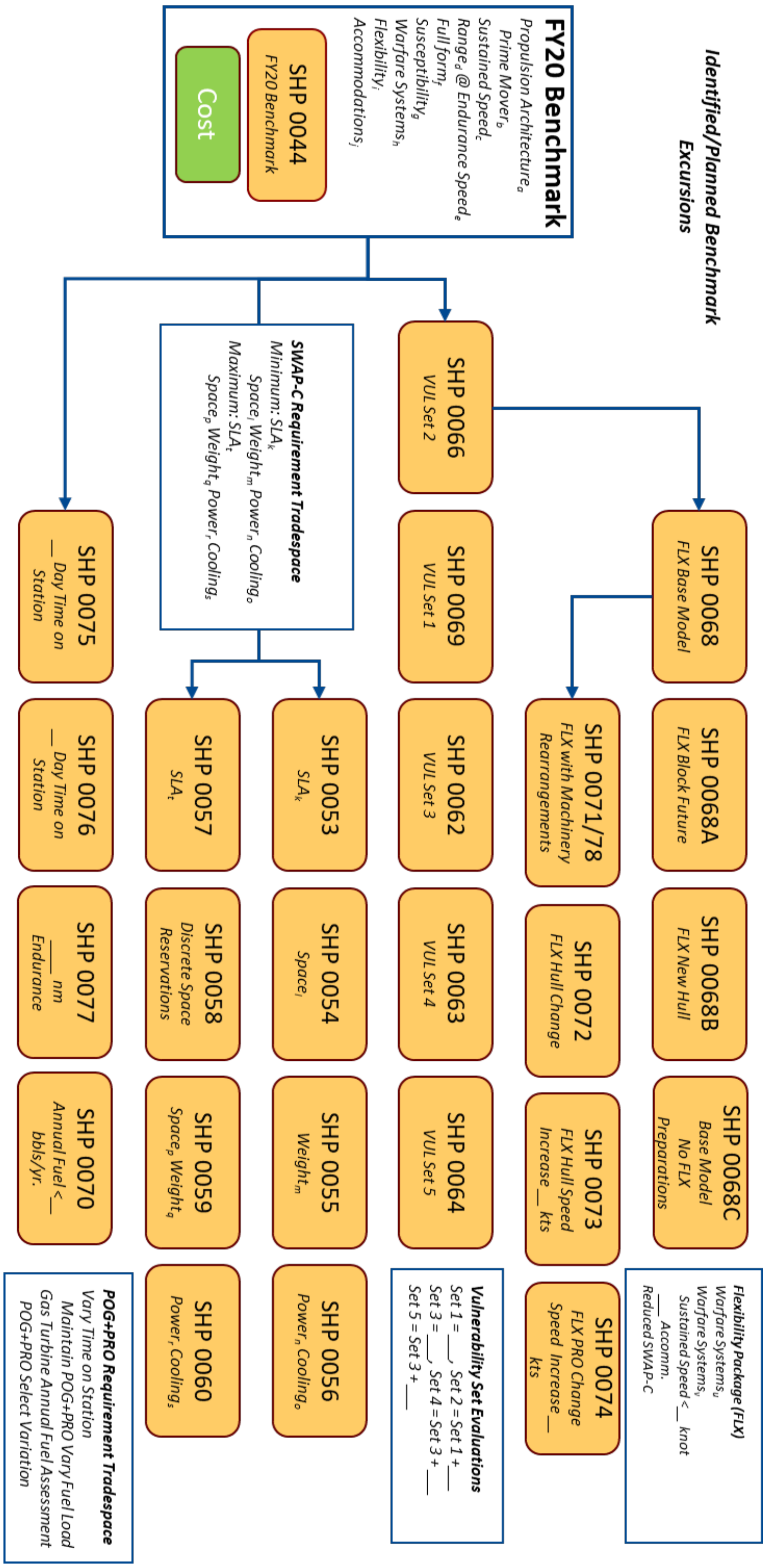


Figure 15: Sample View of Benchmark to Excursion Mapping

This approach enabled the creation of reusable design knowledge depending on how the family of ships differed. For instance, we created two benchmarks in the second year, one with a mechanical engineering plant and the other with an integrated power system. From there, we could plan excursions to test changing space, weight, power, arrangements, and cooling (SWAPC) measures, both greater than and less than the benchmarks, creating four additional members of {ships}. Creating these ships generated knowledge of how SWAPC requirements affected the ship on a global scale, including cost. In the end, we knew if SWAPC made any appreciable difference in the cost of a ship and if that difference was more, less, or the same depending on the engineering plant. We could do this separate from the many-points-based design approach because the intent was to leave the other design parameters the same. Therefore, we knew that the conclusions derived from these benchmark-excursion activities were 'all else equal' results that isolated the design aspects that we needed to investigate. Therefore, we could draw conclusions like, "An __% increase in SWAPC causes a __% increase in cost," or perhaps derive two results, one for the increase in a mechanical ship and another for an integrated power ship. The Latin hypercube sampling method does not accommodate this kind of learning since it intentionally varies all elements and characteristics to explore the design space fully.

The team also created virtual prototypes tied to requirements sets from the sponsor. These prototypes addressed several topics. First, they served as a check for the sponsor. We created a [ship] that met a set of requirements, thereby proving the outcome feasible. With the report came the cost so that the sponsor stayed informed of how a notional set of requirements affected cost. The sponsor understood that the [ship] represented what the ship -could- be, not what the ship -would- be. They also understood the nature of excursions and prototype studies and requested some themselves, demonstrating buy-in to our process. Second, these prototypes addressed the risk that some team members and stakeholders perceived that the process might not "work out." In other words, some people feared that SBD would not be able to integrate by intersection properly, and the Navy would have the null set to choose from at the end of the process. The {ships} addressed that risk by consistently demonstrating we had feasible, balanced designs we could "fall back on" if the risk became real. However, a third purpose the prototypes served ensured that would not happen. By understanding the design through our SBD efforts and having benchmark, excursion, and other ships, we knew we could not rule out a portion of a design space that left no feasible [benchmark] or [ship]. Hence, the {ships} set served as a check to alleviate that fear and always convince ourselves that we had feasible design points. In other words, we could use them to determine the conditions such that we were approaching a condition of being over-constrained and steer the design efforts accordingly. Lastly, these {ships} helped us understand ship-wide impacts of decisions vice those strictly within a domain. For example, calculating the component costs of the various engineering plant architectures is straightforward. Therefore, calculating the cost impacts to the engineering plant resulting from a requirement change to speed is also straightforward. However, changing equipment to accommodate speed potentially changes structures to support different equipment, intakes and uptakes to allow for different airflow, auxiliary equipment to provide different cooling and fuel, and myriad other things. Therefore, understanding the ship-wide impacts, including cost, is more insightful and is precisely the kind of information that the {ships} provide. Lastly, the {ships} provided a testing ground for prototypes that investigated emergent properties like survivability, flexibility, affordability, and others.

7. Emergent Properties in Set-Based Design (SBD)

Research is what I'm doing when I don't know what I'm doing.

- *Wernher von Braun, New York Times, 16 December 1957*

The dissertation now returns to the original question: how should we treat emergent properties, especially flexibility, in a set-based method? The hypothesis predicted that emergent properties could become their own sets on parity with functional requirements and behave equally to them with communication, intersections, explicit trade-offs, reductions, expansions, and other characteristics of an SBD method. In the end, this research neither proved nor disproved the hypothesis. However, the journey towards this stalemate gives insight into the conditions which could prove or disprove the hypothesis. That insight suggests that the answer one day will validate the hypothesis.

The question and hypothesis originated from the desire to ensure naval vessels remain affordably relevant throughout their service life. Keeping the *first* vessel relevant and affordable invoked SBD to understand cost capability trades, delay decisions until the last responsible moment, and reduce the rework load compared to the traditionally iterative process. Keeping the *last* vessel relevant and affordable invoked flexibility to ensure that we consciously made architectural decisions that provided options to future Navy leaders. Therefore, the interest lay in the intersection of the two ideas. However, the intersection extends beyond flexibility and SBD because we were interested in other emergent properties. Thus, flexibility, survivability, producibility, and affordability all contribute to reasoning about the question.

7.1 Methods to account for emergent properties in SBD

Three treatments exist for emergent properties: as an artifact's characteristic, as a characteristic in each {subset} of the artifact, or as a {set}.

First, one could treat the property as a characteristic of the artifact: in our case, a [ship]. This treatment suggests the property is genuinely emergent, and the net behavior is unknown until the system operates, its parts interact, and the behaviors emerge. Alternately, it suggests an inability to attribute the behavior to system elements or characterize relevant interactions. For example, we might not characterize how armor contributes to survivability separate from defensive weapons or separate from longitudinal structures unless we tested them together.

Therefore, we would not understand how survivable a ship is until we deliver it and fire a missile at it.

Alternately, we ought to be able to attribute emergent properties at lower levels of abstraction and characterize their interactions, whether linear or non-linear. Therefore, the second option treats the property as a characteristic of individual {sets}. This approach allows the designer to trade the properties indirectly. By making an emergent property a characteristic of a {set}, the intersection of two (or more) {sets} reveals the effect on the emergent property. For instance, imagine tracking *producibility* as a characteristic of {hull} and {power generation}. The intersection of these two sets could reveal infeasible regions of the design space: the two do not fully intersect. This infeasible region in {hull} or {power generation} may have dominant *producibility* characteristics and thereby influence the outcome of the intersection. The SSDM may decide to expand the {sets} somehow to explore creating an intersection instead of accepting the fate of dominated *producibility*. Whether or not this is an outcome, tracking the emergent property as a characteristic of {sets} informs decisions in their regard. One potential pitfall of this approach is that each domain requires expertise on the emergent properties of interest. If a domain team must track the characteristics, they must know how to. This approach could involve extra personnel on each domain team to provide the required subject matter expertise. Alternatively, it could require extra training and guidance materials to ensure each domain knows how to evaluate each emergent property consistently and repeatably. We considered this approach for some of the emergent properties, and ultimately some naturally fell into this construct. For other emergent properties, though, we chose a different approach.

Lastly, one could treat the emergent property as its own set. We attempted this with {survivability}, {flexibility}, {producibility}, and {affordability}, which guided the organization of the team represented in Figure 3, Figure 5, and Chapter 4.2. This method explicitly trades the emergent properties with other {sets} in the design space. This desirable outcome presents its own challenges. The largest challenge was defining the properties sufficiently to interact with the other sets. As shared in Chapter Five, sets intersect through common dimensionality: length with length, pressure with pressure. Therefore, for emergent properties like flexibility and survivability, one must define the {set} in a way that allows intersection. For some sets like {survivability}, decades of empirical data and development of standards and manuals defined these characteristics for the domain, or at least gave them a start. That domain could take principles like segregation and separation and translate them into ranges of lengths representing ranges of survivability, for instance. *Reliability, availability,* and *maintainability* are other examples of emergent properties with academic rigor and definition. However, we did not explicitly make them into sets: they followed the second approach when considered.

Knowledge of these three treatment options allows reasoning regarding their use by observing the four emergent properties we treated as {sets} in our process. We attempted to elevate four emergent properties by using the third method. Some of those properties are well-defined domains that could create appropriate attributes that suitably intersect with other sets. However, each of those domains presented unique situations which complicated a clear affirmation or refutation of the hypothesis.

First, {producibility} did not receive resourcing during the observation period, so it cannot contribute to any conclusions. However, there is tremendous opportunity to develop {producibility} with the characteristics that define the domain. It ought to be straightforward to translate the practices and design elements that make a naval vessel easier to produce into attributes that intersect with other {sets} like {hull} or {structures}. Further, this provides an opportunity to engage with shipyards early in the design process and allows them to witness the early-stage trades that eventually transform into the ship specifications they receive. The shipyards know what constitutes {producibility} best. Therefore, allowing their participation is critical to properly defining and integrating this {set} and exposing them to the methodology that leads to exclusions and inclusions of the design characteristics.

Second, {affordability} employed a hybrid approach that used all three treatments. The prominent characteristic of affordability is *cost*. This characteristic deconstructs in several different ways, such as first ship cost, average follow-on cost, component cost, acquisition cost, life cycle cost, change costs, inflation costs, and several others that are well-developed. NAVSEA maintains a Cost Estimating Handbook to aid cost estimators in partitioning the costs.

Aside from costs, {affordability} created other elements and characteristics related to design that could intersect with other {sets}, in line with the third treatment. For instance, the domain team created a list of cost reduction initiatives from other ship classes. From this list, they ruled out those infeasible in our design, such as those involving vessel-specific systems for amphibious ships. They also classified the remaining ideas by their intersections to other {sets}, enabling them to work with the other domain teams to characterize the interactions.

The affordability team also used the second treatment option. They assigned costs to individual systems and solutions options when appropriate and possible at the subsystem level. The marine engineering domain used this widely, tracking costs of individual components as a characteristic in the sets of solutions for {propulsion}, {power generation}, {distribution}, {boat handling}, and others. They typically had cost available as a metric to consider with decisions, explicitly or ‘in the backup.’ The same is true for many other domain teams.

{Affordability} also used the first treatment extensively. Many of the findings from affordability studies resulted from the use of {benchmarks} and {ships}. Out of fairness, the nature of ship design necessitates this. Many elements of a ship couple tightly to other elements such that when a designer changes one element, the change inevitably propagates to other domains. Thus, we often find that the cost changes at the {subset} level create non-linear cost behaviors at the [ship] level. The example at the end of Chapter 6 reveals one sample instance. Changing the speed could cause a change in engines which could cause second-order effects like intakes and uptakes for airflow, which create tertiary effects on structures and arrangements. We only capture these higher-order effects at the [ship] level or across {ships}. Therefore, we must track affordability as a characteristic of that superset because there is still some level of ‘emergence’ at the [ship] level not fully characterized at a {subset} level. We could decouple affordability from the {ships} only at low levels of fidelity. For instance, we could generate a finding that ‘cost goes up’ or ‘cost goes down’ based on {subset} characteristics and analysis. However, since the magnitude of the cost movement typically matters for affordability (±$5M versus ±$50M), the effort requires [ship] level analysis, at least for now.

Next, {survivability} primarily used the first treatment method, where it was a characteristic of a given [ship], like in Figure 12. Specifically, the survivability domain investigated {vulnerability} first. A survey of each {vulnerability} Set Review reveals tight coupling with {ships}. The fourth Set Review starts to show evidence of decoupling with the first finding derived from discussions with TWHs instead of analysis of a [ship]. The team also varied characteristics of their set within an existing tool developed in-house. This effort represents a transition beginning from method one to method three, which shows more promise for the future. After all, many standards, manuals, and empirical data help define {vulnerability} already. Nevertheless, their tools function at the [ship] level since they were built for validation vice creation. Therefore, at the beginning of the process, this domain team lacked tools and information that allowed them to develop {vulnerability} as a unique {set} instead of a characteristic of a {set} or a given [ship] configuration (with an [arrangement], [propulsion], [hull], [structure], [topside], and other defined selections). With their studies, they were using those tools to develop the transferrable knowledge one could use independent of a [ship]. Thus, {vulnerability} shows potential to define itself as a unique {set} to communicate and negotiate with the rest of the team, but was tool-limited and therefore still initially analyzed vulnerability as if it was 'emergent.'

Lastly, {flexibility} tried implementing the third treatment exclusively. With this emergent property, though, we discovered it required further development. There are toolsets and analyses often used to analyze flexibility, such as real options analysis and epoch-era analysis. Those analyses use Monte Carlo simulations to randomly create multiple possible futures so that one might understand the decision space most robust to future uncertainty. These tools also cause the designer to focus on the rulesets driving future decisions (e.g., when a condition 'x' is met, *switch* to this system or that concept of operations, or when a condition 'y' is met, *expand* to this capability). An impediment with {flexibility} in SBD, though, was understanding the intersections it could have with other {sets}. Therefore, we created a framework to help define {flexibility} to a level that enabled intersections with other {sets} and honored the unique aspects of this property.

7.2 The flexibility framework

We created this framework in a collaboration between the design team and external subject matter experts (SMEs). We adapted their existing research into a practical approach that captured the necessary characteristics for flexibility analysis and set up the information for multi-attribute and multi-domain negotiation (Fitzgerald, Ross, and Rhodes 2012; Fitzgerald and Ross 2014; 2015; 2019; Rehn et al. 2019; Ricci et al. 2014; Schaffner et al. 2013). In past efforts, the Navy defined and analyzed solutions to known (or high probability of occurrence) future events such as upgrading computing infrastructure or changing a particular combat system element. These led to solutions like flexible infrastructure, modularity, and policies related to service life allowances for space, weight, power, and cooling. However, the team understood the need to move beyond the known solutions and further define the appropriateness and quantity of those solutions using both qualitative and quantitative measures. In a summary compendium, Ross and Fitzgerald (2022) provide an extensive account of the framework and its underlying principles.

First, we created a mission statement around the framework:

> *To sustain value in the face of perturbations, we must create coverage by preparing to respond on a matching timescale.*

This mission statement originated from discussions around the operational framework depicted in Figure 16. This framework communicates the life cycle of a system from a 'flexibility' mindset. Uncertainty and perturbations exist throughout the lifecycle. Fundamentally, one does not know if or when they occur. However, one wants the ability to respond on a practical timescale to that perturbation. These responses could invoke change through flexible architectures (like switching a combat system module) or require resistance through robust architectures (like armor). Therefore, one generates option ideas and assesses their coverage of the perturbations. Knowing valuable options allows one to generate and assess preparations to include in the ship's architecture to enable the option or make exercising the option more efficient, more affordable, or both.

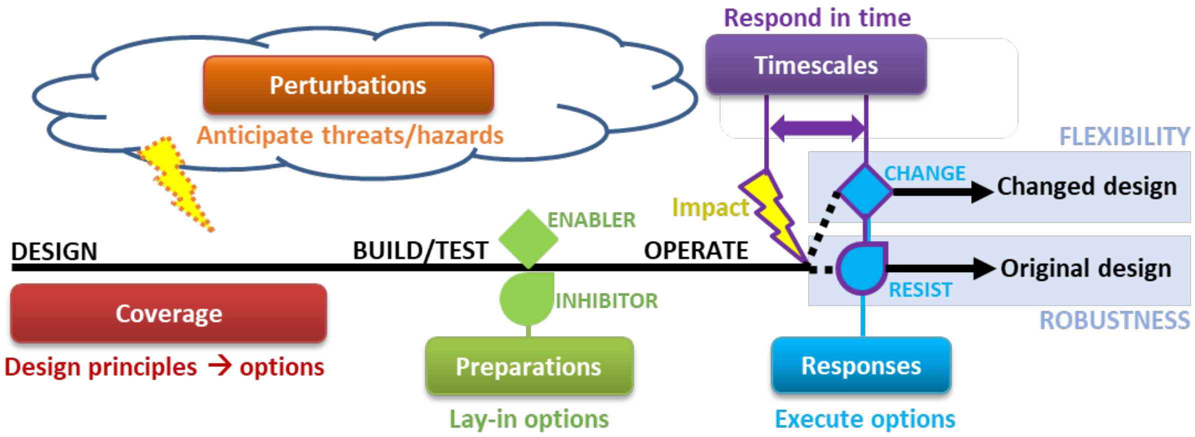


Figure 16: Operational framework for flexibility

To accomplish our mission statement, the flexibility leadership created the analysis framework in Figure 17. It shows the basic procedural steps and the inter-relations between the various parts of the analysis framework. It also shows a clear relationship to the operational framework. It shares the central role uncertainty and perturbations take in analysis; after all, flexibility has no value without uncertainty. One also notices the beginning of {subsets} within {flexibility} that will enable communications with other domain teams: the process steps in the analysis framework result in several lists which can translate into {subsets}. First, the team creates a list of perturbations, which can become {perturbations} once characterized appropriately. Second, the team creates a list of {options} that cover those perturbations. Next, the team creates lists for {preparations} and {responses}. Those lists and {subsets} create good starting points to create intersections. Timescale represents a characteristic (measured in seconds, minutes, hours, days, months, or other appropriate measures) within the aforementioned {sets} that allow them to intersect within the domain. In other words, if the timescale for the response exceeds the timescale for the perturbation, we should not keep that solution chain with the given options and preparations since we could not execute it on time. Synthesis and trade-offs encircle the framework to represent the continuous analysis occurring, both internal and potentially external to the domain.

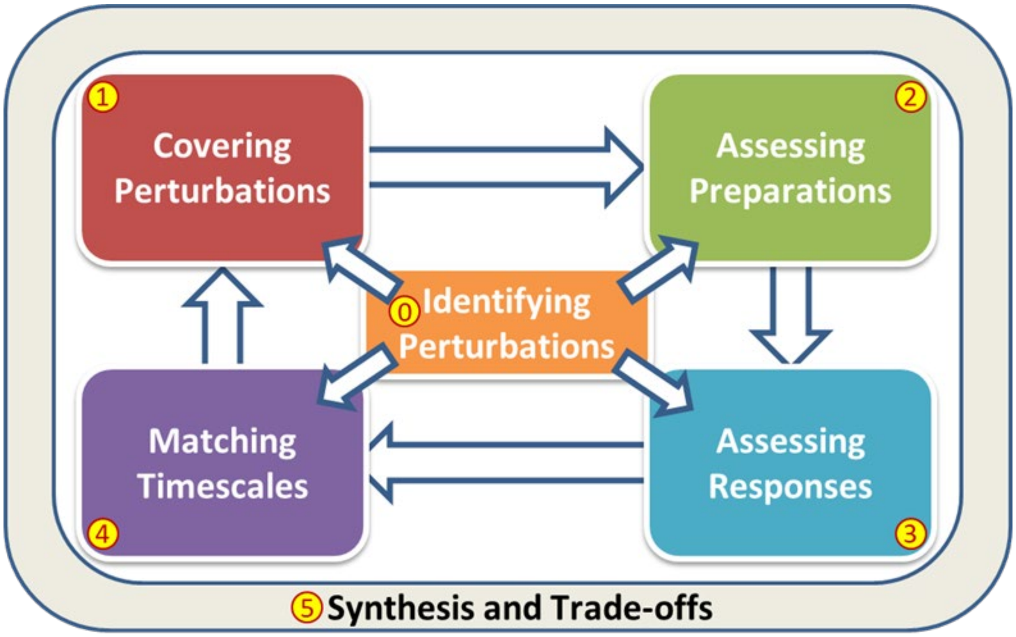


Figure 17: Development and analysis framework for {flexibility}

The synthesis and trade-off step within the framework presents a crucial element. It measures the value of the flexibility, both in quantity and appropriateness. One should not simply take the lists of perturbations that create the list of options requiring the list of preparations and decide to include all preparations in the architecture. Within this analysis step, one determines such things as: if the cost of the option is worth its benefit; if enough resources (space, kg, power, funding, ...) exist to execute the option; if the {set} contains mutually exclusive options (e.g., would occupy the same space when executed); or if there are redundant options that decrease value. These value analyses drive an essential intersection with {affordability} to understand the typical trade-off in the flexibility domain. This trade-off occurs between acquisition costs and life-cycle costs, since the Navy installs the preparations enabling the options up front, but leadership typically exercises the option during the operations phase of the life cycle. Uncertainty complicates this trade-off further because leadership may never execute the option if they never encounter the perturbation. This complication leads many to falsely believe the acquisition cost of the option is inappropriate or difficult to justify. The analyses (such as real options and epoch-era) help quantify these critical aspects so present-day leadership can balance value, cost, and risk when determining whether to retain coverage of a perturbation or not.

7.3 Execution of the framework

Flexibility leadership intended a straightforward execution of the framework. We started at the beginning with perturbations and systematically populated {flexibility} with necessary characteristics and data. The eventual analysis would help determine which options and preparations offered value and inform decisions to retain or remove those elements of the design space. However, the team did not fully follow the framework as intended.

The process began in earnest with the workshops mentioned in Section 4.5 Training and mentoring the team, collective learning. We intended the first workshop to introduce a diverse grouping of the design team and stakeholders to the flexibility framework and then help the flexibility domain team with the initial data to populate {perturbations} and {options}. The workshop lasted two days in April 2019. On the first morning, the participants received training on flexibility and set-based design principles. Then, we divided participants into five teams to perform guided exercises. Before each exercise, the team received instructions on the format and guidelines to conduct the exercise. The exercises entailed various activities to help brainstorm perturbations, map them to options, and appropriately characterize the options and perturbations. The workshop succeeded in producing a few hundred {perturbations} entries and almost two hundred {options} entries across the five teams. Even after removing duplicate entries across the teams, the {sets} contained over one hundred entries each. This outcome provided the confidence to conduct another workshop within a specific domain to extend further into the framework.

In June 2019, the flexibility team conducted the second workshop. This workshop happened at Naval Surface Warfare Center Philadelphia with the marine engineering team and some of their key stakeholders. Some of those present did not attend the first workshop, which provided an opportunity to hone the training and incorporate lessons learned from the first workshop. Also, this workshop took advantage of using information from the first workshop as a starting point, so the training examples were curated to maximize relevance to those in the marine engineering domain. This workshop ran for two-and-a-half days. The teams in this workshop used the elements of {perturbations} and {options} relevant to {power generation}. They repeated some brainstorming activities from the first workshop with that intersection as a starting point example. They also extended the framework by assessing coverage between options and perturbations, further characterizing the options, assessing preparation costs, assessing three metrics of the response, and assessing timescales. The workshop succeeded in extending the framework, including to individuals introduced to it for the first time. The team created another large data set in a relatively small timeframe. They also further characterized the sets and enabled analysis. Through their participation, they agreed with the value of the approach and looked forward to continuing through this framework to quantify the right intersection of {flexibility} and {power generation}.

An interaction from the second workshop created the third. During the workshop, the participants and flexibility leadership discussed the matter of perturbation sources and their flow into the design space. The discussion originated from internal versus external sources of perturbations. The participants pointed out that the perturbations they address are rarely external but frequently internal perturbations that follow from external perturbations. For instance, new enemy capabilities may drive improvements in radar, which require more (and cleaner) power to operate, which creates a perturbation to {power generation} to provide the necessary power. This conversation revealed the value of a Cause-Effect Map to our design team since we realized the marine engineering team is not the only team subject to this phenomenon. Thus, members of the flexibility team met in Cambridge, Massachusetts, in September 2019 to 1) introduce cause-effect mapping, 2) attempt the first map, and 3) identify any pitfalls that might hinder progress.

The second year's resource reductions and leadership changes caused significantly slower momentum from the workshop, especially as the focus shifted to material preparations. This led to one other effort to revitalize the cause-effect map in the summer of the second year. The outputs of the third workshop remained stagnant for several months, but an opportunity arose with Navy officers at MIT looking to conduct summer research. The flexibility team had products ready for validation and consumption, and we quickly teamed with the students for this win-win scenario: they received relevant research work that they could continue, and the flexibility team received three bright students to continue the vital work of extending the domain knowledge and creating products for the design team to use. They produced a network diagram, corresponding lists of perturbations and options, and a lessons learned document that captured the pitfalls they encountered from working on the diagram for several weeks (compared to several hours at the workshop).

Around the workshops, the flexibility team continued refining the sets and working with other domain teams to populate them. The team realized that the magnitude of an element affects the valuation of the {options} and {preparations}. As a result, {perturbations} expanded into the thousands. For instance, a new power load of 50kW may invoke a 'resilient' approach where the existing services can absorb the additional load within service life allowances, but an additional load of 1MW or more may invoke a different strategy. However, to keep the cause-effect map usable and straightforward, neither the team nor the students accounted for these magnitude differences.

Near the end of the first year, pressure also increased substantially to provide examples of flexibility in the ship's architecture. In the language of the framework, this meant selecting the preparations. Team leadership also changed for the first time. These events changed the team's activity to analyze preparations using traditional methods. The focus shifted to five {subsets} and focused on the technical feasibility of material selections and characteristics that could define flexibility in the ship instead of the value they provided. Therefore, the team cast aside the framework and has yet to return to it and continue the initial progress they realized. This situation created the unfortunate result in some instances of the team suggesting potentially redundant solutions. Leadership intervention prevented the selection, but the event served as a reminder of the framework's value. The domain experienced another leadership change soon into the second year, contributing to the continuation of legacy analysis methods. Finally, dwindling resources in the third year caused design team leaders to focus on other domains, disbanding the flexibility team altogether.

7.4 Lessons from executing the framework

The first lesson is that the flexibility domain almost successfully implemented the third treatment of emergent properties in SBD, but not entirely. The domain mapped their design space during the first workshop and throughout much of the work the first year. They defined feasible regions. They explored trade-offs and multiple alternatives, especially when expanding to thousands of instances in {perturbations}. They communicated using sets of possibilities throughout, even once they dropped the framework. They looked for intersections of feasible sets during the Philadelphia workshop. They naturally imposed minimum constraint in the {sets}: {perturbations} captured future uncertainty and informed {options} which intended to remove constraints as much as minimize them and led to ranges of {preparations} for consideration. They sought conceptual robustness in some of the {subsets} they created when the focus shifted to the material {preparations}. For example, the {capability expansion option} intends to accommodate expanded capabilities within a new subdivision of the ship, requiring robust solutions due to the uncertainty of the desired capability in the future and the rest of the ship design (LaGrone 2022). They were poised to integrate by intersection with the work they completed within the analysis framework but could not take that step before efforts shifted focus or started winding down due to resource decisions at higher levels. Therefore, they also could not demonstrate establishing feasibility before commitment. The data and framework stand ready for a new team with new leadership to pick up these efforts again and see it through the rest of the process in both SBD and the flexibility framework.

A valuable outcome of the flexibility team's interactions with the rest of the design team was the proliferation of the flexibility language to the design team. Other teams used the language from the workshops and team interactions. Hearing other domains speaking of flexibility, perturbations, and options within their trade space rewarded the flexibility domain leadership for their efforts to create the framework and training materials. In addition, it demonstrated that the larger team understood the vision and wanted to follow through with it.

Discussions from the various workshops and interactions revealed the need to separate flexible engineering from necessary engineering. Several participants and team members made a comment along the lines of, “we know we need to account for ___ scenario.” Inevitably, the scenario mentioned had not happened yet, and this condition was confounded with uncertainty, vice a future known change. This very frequently occurred in the warfare systems domain, where we ‘knew’ we would upgrade or replace a system when a higher value system became available in the future. Interestingly, the crux of that statement relates that something is known: it has no uncertainty. Therefore, it should have no place in the flexibility framework. However, the framework still accounts for these scenarios. After all, we could not be certain of the impact, magnitude, and timing. Therefore, the flexibility leadership had to redirect the team’s thoughts to focus their efforts on quantifying those aspects. Alternately, if it was impossible to redirect the thoughts this way, we redirected the requirement to the alternate domain.

Treating the property as its own functional area was valuable for this developmental domain. The team developed a framework, populated the framework, conducted workshops, educated the team, and created {sets}, maps, and other useful products. In other words, they worked like the other functional areas and, in many ways, had more work since they were building the framework and executing it concurrently. Under these circumstances, making them a functional area instead of a specialty team at the “hub” or requiring a ‘flexibility SME’ in each other domain proved appropriate. The domain still requires subject matter expertise, so expecting each other domain to understand and execute the flexibility framework will take more effort and training for the broader naval engineering community.

All team members struggled with the vastness of uncertainty. First, the ship is subject to uncertainty regarding budget, politics, missions, friendly technologies, adversary technologies, adversary actions, and environmental conditions at the system of systems level. Further, systems and subsystems have uncertainty, especially in concept design which lacks a baseline design. SBD seeks to rule out areas of the design space vice selecting them. Therefore, many domains struggle to characterize the uncertainty prevalent with the lack of design decisions that provide selections and constraints. The interconnectedness of a complex design like a naval vessel multiplies this effect farther into perturbation chains, like the bullwhip effect. Many of these discussions and comments happened before the team completed our first benchmarks or successfully reduced {sets}. Therefore, these conversations faded with time as the team became more familiar with the operation of SBD and flexibility.

7.5 Summary of emergent properties and SBD

To summarize, the research explored the possibility of treating emergent properties like functional requirements in an SBD construct. To determine the answer, it proposes criteria: we ought to treat the property like any other {set}. It should define {subsets} and characteristics, communicate and negotiate with other teams, and follow the principles of SBD outlined in Chapter Two. The research attempted to test this on four separate properties: producibility, affordability, survivability, and flexibility. The results concluded that the hypothesis could not be supported but showed promise that it could be in the future.

Producibility does not contribute to the answer because it lacked resources during the observation period.

Affordability does not support the hypothesis because the team took an appropriately hybrid approach, which treated it as a {set}, as a characteristic of other {sets}, and in many cases, as an emergent property measured after the design of a full virtual prototype. However, when treated as its own set, {affordability} followed the principles of SBD.

Survivability does not support the hypothesis because the available tools limited them from integrating by intersection. However, they did well in using the existing pre-integrated tools to develop knowledge for future use that helps transition the domain to better SBD practices.

Flexibility does not support the hypothesis because its efforts shifted and eventually terminated before the team achieved complete success. However, we have a documented framework and data set ready for execution if resources align to the effort once more.

Each vignette reveals that the hypothesis is challenging but achievable. The hypothesis essentially asks if one can make emergent properties less emergent. With appropriate rigor, the method can support an affirmative answer. “Design for X” studies and papers attempt this outcome, also. SBD provides a method where each important emergent property or “design for x” gets an equal negotiating position in the design process. The various domains need to begin the work of translating their properties into the characteristics that define their {sets}. This translation will allow their seamless operation with the principles of SBD. The functional {sets} struggled with this translation, and these properties have the added difficulty of being considered emergent. We must make them ubiquitous instead of emergent for the principles of SBD to work correctly.

For emergent properties of interest, the organizational construct making them their own functional area worked for our team. Each area received the appropriate attention and achieved results in its domain. They discovered the needs for communications and negotiations and incorporated them. This organizational construct also kept the team's hub simple instead of adding teams of SMEs to the hub. Until the property is truly ubiquitous, expecting each domain team to analyze it correctly and consistently involves risk. These are organizational decisions for design team leadership to consider unique to their situation. For instance, {affordability} is ubiquitous enough that each of our domain teams could have characterized it themselves with appropriate interactions to cost SMEs. Nevertheless, we still wanted the effort to have its own functional area and studies because it deserved that place of prominence for our effort.

8. Other insights and future work suggested

Some of the observations and lessons do not fit naturally in the flow of the preceding chapters but deserve mention. Therefore, this chapter shares them in no particular order.

The first considers leadership acceptance. They learned the SBD method and outcomes along with us. They rarely joined the day-to-day execution but still received updates periodically. These updates, naturally, differed from the usual briefs for a program at this stage. We had {sets}. We decided what the ship was *not* instead of starting with what it could be and iterating. Therefore, we had to teach leadership our method of SBD and frequently defend the seeming lack of results. This scenario changed at the end of the first year. By then, we had fully processed nine of the DDMs (although we had approved 26). Those DDMs represented decisions to remove portions of the design space based on knowledge gaps and informed the requirements. We set up weekly meetings with stakeholders to brief this information and how the set-based process led to the knowledge. This energized the process and leadership as they saw the value we derived from the outputs of the process. The stakeholders kept engaging and even proposed knowledge gaps for the team to close related to requirements. We had active, fruitful dialogue from that point forward, which also contributed to the gains from Year 1 to Year 2. This vignette shares the insight that leaders and stakeholders need patience at the outset of a process standup like this, and the team should engage them quickly when results start to arrive.

The second considers tools. Bluntly, the tools available to the team did not accommodate SBD very well. Over the years, engineers and coders developed many, if not all, of the tools for point designs or validation of point designs. The design team adapted these to suit our needs and produced results to the best of their ability. However, the team also spent time and effort developing bespoke tools because some could not adapt well enough. Chapter Seven communicated how this affected the treatment of some emergent properties, but the issue was widespread. {Machinery arrangements} provides another case to this point, but one within marine engineering. In this case, the engineer understood exactly what we were asking for in terms of her {set}. She wanted to work correctly in a set framework, but a lack of funding prevented her from developing the appropriate toolset in time. Unfortunately, the development time required did not support the development time for DDG(X), and she had to adapt legacy tools inefficiently to support our team's effort. Investment in design tools to support set-based methods instead of adapting tools intended to create or validate points or to create or validate *many* points is critical for SBD efforts.

Along the lines of tools, SBD matches naturally with MBSE. We were establishing our digital engineering practices and our design practices concurrently, so this observation is not conclusive, but the approach hinted at success. MBSE as an approach with traceability, interactions, intersections, coupled analysis methods, and documentation seems to present a worthwhile companion to the SBD method. This interaction deserves further study and an operating model.

On a related thought, research on SBD interactions with systems engineering processes could bear fruit. Specifically, the activities at the bottom of the systems engineering "V," which typically represent turnover from designers to producers, deserve more attention. For example, SBD could 'smooth' the discontinuity of the 'V' and create a 'U' by including the producers, integrators, and vendors on the 'requirements down' part of the model. Further, studying if the SBD method can reduce rework on the 'realize up' half of the model could be insightful. Finally, another interesting research topic considers the role of prototyping in smoothing the discontinuity and how much virtual prototyping can take on that role over traditional land-based testing, engineering development models, and other physical models.

Another note related to tools associates with the cause-effect map introduced with the flexibility framework. The design team should continue the work of maintaining that as a living document. Further, they should incorporate parallel chains that account for interesting magnitude differences in the perturbations.

In general, the team should continue documenting and sharing how our SBD method transitions with time. Interesting observations would include: 1) the role of SBD on the back half of the 'V, if any, 2) how and when SBD transitions away or stays in some form through detail design, 3) if it remains, if the nature of SBD is different in 'requirements' and 'realization' phases, 4) any other issues of emergence that we could have handled differently. Further, as mentioned in Section 4.1, the team could observe metrics such as change avoided, rework minimized, and effects on efficiency of downstream design and engineering processes. The team can collect this information as the process continues to play out and contribute to the existing research needs to extend the application of SBD beyond the conceptual design phase (Toche, Pellerin, and Fortin 2020).

Within the conceptual design phase, research on the intersection of SBD with two other principles could provide value. First, our process to determine the first decisions and {sets} for development resulted from the risks and mental rigor we inserted and relayed in Chapter Four. However, a proper Design Structure Matrix (DSM) and accompanying Task Structure Matrix (TSM) could provide more rigor to our team or any team. These products could also lead to an informative decision tree that pulled in real options thinking and analysis. Real options analysis helps in making decisions under uncertainty. Applying its principles towards the uncertainty of design decisions yet to be made could aid in risk reduction by helping develop criteria and decision trees stemming from the DSM that can reveal the right timing to make decisions or keep options open. One could treat each decision as an option to keep or remove portions of the design space for further consideration, which involves uncertainty prevalent in the design space. Because many design decisions are contingent on previous decisions and there are multiple sources of uncertainty, there could be an intersection with compound rainbow option thinking and toolsets from real options that are useful in exploring the design and decision-making processes.

The next observation offers a tweak to the SBD principles the design community uses. In our case, we did not map the entire design space simultaneously. Instead, we mapped the portions we needed to close knowledge gaps. The tweak necessary for consideration is to add ‘at the right level of abstraction.’ For complex projects like a naval vessel that is a system of systems, a complete map of the design space will take years. Therefore, expectations should shy away from producing volumes of data initially and should focus on creating the most valuable data first. Thus, one should map the design space -at the right level of abstraction- to move the design effort forward. One should base the mapping on driving factors, not simply the information available. In one instance, for our team, while the length of gas turbine engines was readily available information, it did not contribute significantly to the decision regarding power plant architecture. Higher abstraction metrics and characteristics informed higher abstraction decisions. Alternatively, the boat handling team used basic analysis like “pro/con” and fleet experience metrics to drive initial decisions in their space on the location of boats. Subsequently, they added the data for size and weight once that was more relevant to the decisions at hand.

Another observation confirms that the original criteria from Sobek et al. (Sobek II, Ward, and Liker 1999) appear necessary and sufficient for SBD. This effort succeeded in following the principles to the best of our ability. As a quick review from Chapter Two, the principles are: mapping the design space, integrating by intersection, and establishing feasibility before commitment. Of course, the devil is in the details of each of those, and this dissertation provided as much of that detail as possible. The research and reasoning about our process do not reveal any critical missing principles and hopes to reveal considerations for organizations beginning an SBD journey. According to some proposed metrics, we met the criteria for SBD rigor (McKenney, Buckley, and Singer 2012; Singer et al. 2017). Those authors present five SBD elements, each with three levels of potential rigor, similar to a capability maturity model. The highest scores in the suggested rubric correspond to the ability of a third party to review products and decisions. This research accomplishes that, with the caveat that the third party reviewer needs appropriate clearance and non-disclosure agreements in our case, of course.

Related to the original research of SBD and the observations of its use at Toyota, we observed differences worth noting. First, when researchers observed Toyota, they observed a well-developed and stable process. Toyota had developed it over the years and refined it to their needs. In contrast, our effort established a process. Our process requires refinement and learning to reach the same stability observed at Toyota. We must tailor it to match the culture and engineering workforce of the US Navy engineering enterprise. Therefore, it will operate differently, but hopefully with similar results: getting better products to market faster than our competitors.

Further, Toyota does not start the process and then finish it; SBD is enduring for them. The US Navy has the opportunity for a similar construct. It should take advantage of every opportunity to establish an enduring effort to create transferrable design knowledge for future ship design efforts. SBD will be less effective and provide less learning if managed and resourced with each new program instead of as a foundational capability.

Lastly, this effort applied rigor towards defining a {set}, but more is warranted to generalize methods for establishing {sets} and {subsets} and characteristics in a consistent way.

9. Summary and Conclusion

The Navy faces significant challenges. They are in a strategic competition with capable peer adversaries with industrial capacity and technology capabilities. The Navy's investments, especially for combatants, are large capital outlays. Those investments have long service lives rife with uncertainty and complex designs matched by a complex acquisition process.

To address these, we instituted an SBD method for the next large surface combatant, now called DDG(X). This required establishing an SBD method to scale. Reviews by independent research teams showed a growing body of relevant literature and revealed a lack of a model of operation for conducting SBD, especially when transitioning from traditional methods (Toche, Pellerin, and Fortin 2020; Shallcross et al. 2020; Dullen et al. 2021). This dissertation offers one such model for consideration. We also instituted a flexibility framework to explicitly consider the uncertainty associated with a naval vessel's long development cycle and service life.

The documentation of our process attempts to contribute to many of the identified gaps. First, this documentation and research contribute to approaches for organizational transformation. It discusses some of the risks we perceived in this transformation and how we addressed them using Design Guidance Memoranda, procedures, and policies. It also reveals how we organized our team and our considerations, such as closely matching the organizational structure already in place with the parent organization. The research also contributes thoughts regarding reusable design knowledge and its management. It offers different types of reusable knowledge and considerations for how to create the knowledge, including with careful control and selection of assumptions and through our use of {benchmarks}. With {benchmarks} and land-based testing aspects, this research also discusses prototyping and its role in concept design. The research also creates and observes cross-functional interactions among the organization's functional areas we established. It reports on the intersection activities and how Set Reviews grounded and unified the team and the overall effort. This research documents a functional model representing those cross-domain communications, overlapping, narrowing, refinement of sets, institutional learning, and all other tenets of SBD.

Importantly, the research and associated metrics show that the process can scale. For example, we started with eleven {sets}, but the team now has over 200 identified {sets}, and at least 30 are actively exploring the design space at one time. Further, the team started with around

20 personnel and grew to over 100 personnel. The organizational and process structures sustained themselves through this growth. They also sustained themselves once we turned over control of the Set Reviews and the SBD process to others, retaining the decision authority with the SSDM.

Further, we accomplished this research with a representative and diverse sampling of the naval engineering workforce. Also, a point deserving continued emphasis is that we accomplished this remotely. The pandemic forced our team to conduct business remotely and virtually, and this process fared well and even excelled under these conditions.

The focus on knowledge gaps was another key learning point. The team had tools that could have created thousands or millions of data points. However, our approach sought to produce the right data at the right abstraction that closed an existing knowledge gap and led to a decision. It focused on creating valuable data and mapping the design space using driving factors instead of available information. We documented those decisions in a Design Decision Memorandum, a critical artifact of the process. It provided an auditable trail of technical information, assumptions, conversations, and even chat logs regarding the ship's design for posterity. Those Memoranda contain explicit reusable design knowledge.

The research contributes thoughts and procedures on integration by intersection. First, it offers a definition for sets, which is not as prevalent in the literature as one might think. This paper suggests we think of {sets} as useful decompositions of groups or points into {subsets} and characteristics. They are not passive groupings of alternatives collected after creation but intentionally created to close knowledge gaps. It also clarifies that, in many cases, one could think of points as [vectors] or {sets} themselves. It explains sets versus points and many points. It aids the reader in visualizing {sets} as we think about them and the structure we put around them. It discusses the intersections occurring at the {subset} level and that they occur across {subsets} but not within them. In reasoning about intersections, it shares that the Venn diagrams of intersections are not nebulous groupings but characteristics of different domains that match dimensionally. Therefore, a necessary effort is to determine in which dimensions two domains intersect and partition the domains and {sets} down to that level. The paper also shares that it is as much about reasoning about intersections as it is integration by intersecting because it could be any number of other functions that could relate the characteristics of the two domains, including greater than, less than, offsets, and other things. It points out that reasoning about intersections can lead to contractions of the design space, gradually reducing it to a baseline value, but also leads to expansions to open new intersections of interest or value. Lastly, the paper shares that our negotiations and intersections were informal. We had no formal toolset in place beyond our Set Reviews and their format and the DDMs, but MBSE shows promise in this regard.

The research also introduces a special case of intersection unique and perhaps counter-intuitive to SBD: using point designs. First, it distinguishes between SBD, PBD, and many-points-based set reduction. These are important distinctions because to understand the design, one should reason about it before integration and synthesis, not after. However, we created virtual prototypes called {benchmarks} to understand specific aspects of the design, especially non-linear ship-wide impacts of decisions and knowledge of certain emergent properties. {Benchmarks} created starting points for excursions that create reusable design knowledge by taking an all-else-equal approach to the virtual prototype development. They also helped us understand that we still had feasible outcomes, which helped address some people's perceived risk that SBD would rule out all viable solutions. Instead, these point designs helped us understand when we were approaching an overly constrained state and steer design decisions and the process accordingly, including stakeholder notifications and discussions.

Finally, the research addressed the original motivating question and revealed that the effort came close to treating emergent properties as {sets}. However, each experiment had unique circumstances that prevented the complete validation of the hypothesis. The paper shares the three basic treatments for emergent properties in SBD: a characteristic of the artifact, a characteristic within the {sets} and {subsets} of the artifact, or a {set}. In some cases, available design tools prevented treating the property as anything but emergent. In other cases, the team used a hybrid approach that treated the property with each of the three methods. In every case, during the observation period, each observed property could not follow all the tenets of SBD, especially integration by intersection. This outcome couples with reasoning about integration and the need to define the intersecting characteristics of each {set}. The team and the Navy can overcome this last hurdle with the right focused efforts.

The paper also shares the flexibility framework, a special collaboration that explicitly values flexibility in the design and supports its treatment as a unique set. Flexibility had many victories on the team, including the proliferation of the philosophy and language of the framework. The team built a structure for {sets} that help define this emergent property down to constituent characteristics that could intersect with other {sets} and allow for explicit negotiation of this aspect of the design with others. However, successful valuation of options and preparations and integration through intersections with other domains remains elusive until the efforts restart. In all, the framework represents a positive step forward to incorporating the right amount of flexibility in the architecture of a naval vessel. The framework's explicit representation of uncertainty via {perturbations} provides a necessary enhancement to previous Navy efforts.

The experience revealed the need to develop different tools for an SBD method. The tools we had worked well in point-based approaches. We adapted them as best as possible, but they were not well-suited for the SBD method.

Lastly, the experience revealed the value of an action research approach for design studies. We had no choice but to build the design machine as we ran it. Conducting side-by-side studies of a design process requires too many resources and is unlikely to happen. The action research approach used frequent reflection and action to always conduct the process in a manner we thought was the best. It encouraged frequent experimentation in an active process. Luckily, our process had many opportunities for small experiments of the process, usually more than once per week. Lastly, it used a case study-like approach to provide reasoning about the process for generalizable and transferrable knowledge.

The Navy is on its way to meeting the challenges it faces in the best ways possible. Toyota used the method successfully to deliver better products to market faster in its global competition. Hopefully, the Navy treats this as the beginning of their journey to develop a set-based engineering practice that affordably and efficiently delivers necessary capabilities. Naval vessels are complex, but we demonstrated that the methods work and scale to meet that complexity. Significantly, it enhanced the dialogue between stakeholders and especially with those in charge of requirements early in the process, a success that hopefully keeps momentum. The action research methodology helped us learn substantially along the way. It was a privilege to capture a subset of those lessons within this document to share with others considering starting a similar journey. Other organizations may decide which risks and processes fit them appropriately according to their structure and culture.

References

Adams, Kevin MacG. 2015. *Nonfunctional Requirements in Systems Analysis and Design*. Topics in Safety, Risk, Reliability and Quality: Volume 28. Cham : Springer, 2015.

Bernstein, Joshua I. (Joshua Ian). 1998. "Design Methods in the Aerospace Industry : Looking for Evidence of Set-Based Practices." Thesis, Massachusetts Institute of Technology. https://dspace.mit.edu/handle/1721.1/82675.

Broome, Granville W., David W. Nelson, and William D. Tootle. 1982. "THE DESIGN OF VARIABLE PAYLOAD SHIPS." *Naval Engineers Journal* 94 (2): 147–78. https://doi.org/10.1111/j.1559-3584.1982.tb02436.x.

Burrow, John, Norbert Doerry, Mark Earnesty, Joe Was, Jim Myers, Jeff Banko, Jeff McConnell, Joshua Pepper, and Tracy Tafolla. 2014. "Concept Exploration of the Amphibious Combat Vehicle." In *Day 1 Wed, October 22, 2014*, D011S001R007. Houston, Texas, USA: SNAME. https://doi.org/10.5957/SMC-2014-T19.

Cloft, Penny W., Michael N. Kennedy, and Brian M. Kennedy. 2019. *Success Is Assured: Satisfy Your Customers on Time and on Budget by Optimizing Decisions Collaboratively Using Reusable Visual Models*. First Edition. New York, NY: Routledge, Taylor and Francis Group.

Commander Naval Surface Forces. 2021. "Surface Warfare: The Competitive Edge." Commander's Intent. San Diego, CA: Department of the Navy. https://www.surfpac.navy.mil/Portals/54/Documents/CNSP/SNA/SNA-2022/Competitive-Edge-Paper-Approved-Dec-20th.pdf.

Copeland, Thomas E., and Vladimir Antikarov. 2003. *Real Options: A Practitioner's Guide*. New York: Texere.

De Weck, Olivier L., Daniel Roos, and Christopher L. Magee. 2012. *Engineering Systems: Meeting Human Needs in a Complex Technological World*. Cambridge, Mass: MIT Press.

Dixit, Avinash K., and Robert S. Pindyck. 1994. *Investment under Uncertainty*. Princeton, N.J: Princeton University Press.

Dodaro, Gene, ed. 2021. *Weapon System Annual Assessment*. GAO-21-222. Washington, DC: Government Accountability Office.

Dullen, Shawn, Dinesh Verma, Mark Blackburn, and Cliff Whitcomb. 2021. "Survey on Set-Based Design (SBD) Quantitative Methods." *Systems Engineering* 24 (5): 269–92. https://doi.org/10.1002/sys.21580.

Fitzgerald, Matthew E., and Adam M. Ross. 2014. "Controlling for Framing Effects in Multi-Stakeholder Tradespace Exploration." *Procedia Computer Science*, 2014 Conference on Systems Engineering Research, 28 (January): 412–21. https://doi.org/10.1016/j.procs.2014.03.051.

———. 2015. "Effects of Enhanced Multi-Party Tradespace Visualization on a Two-Person Negotiation." *Procedia Computer Science*, 2015 Conference on Systems Engineering Research, 44 (January): 466–75. https://doi.org/10.1016/j.procs.2015.03.014.

———. 2019. "Artificial Intelligence Analytics with Multi-Attribute Tradespace Exploration and Set-Based Design." *Procedia Computer Science*, 17th Annual Conference on Systems Engineering Research (CSER), 153 (January): 27–36. https://doi.org/10.1016/j.procs.2019.05.052.

Fitzgerald, Matthew E., Adam M. Ross, and Donna H. Rhodes. 2012. “8.4.1 Assessing Uncertain Benefits: A Valuation Approach for Strategic Changeability (VASC).” *INCOSE International Symposium* 22 (1): 1147–64. https://doi.org/10.1002/j.2334-5837.2012.tb01394.x.

Frye, Matthew C. (Matthew Clinton). 2010. “Applying Set Based Methodology in Submarine Concept Design.” Thesis, Massachusetts Institute of Technology. http://dspace.mit.edu/handle/1721.1/69214.

Garner, Matt, Dr Norbert Doerry, Adrian MacKenna, Frank Pearce, Dr Chris Bassler, Dr Shari Hannapel, and Peter McCauley. 2015. “Concept Exploration Methods for the Small Surface Combatant,” 10.

Ghosh, Sourobh, and Warren Seering. 2014. “Set-Based Thinking in the Engineering Design Community and Beyond.” In *Volume 7: 2nd Biennial International Conference on Dynamics for Design; 26th International Conference on Design Theory and Methodology*, V007T07A040. Buffalo, New York, USA: American Society of Mechanical Engineers. https://doi.org/10.1115/DETC2014-35597.

Gray, Alexander W. 2017. “Applied Set-Based Communications and Negotiation Systems.” *Naval ENgiNEErs JourNal*, 12.

Gray, Alexander Westley. 2011. “Enhancement of Set-Based Design Practices Via Introduction of Uncertainty Through the Use of Interval Type-2 Modeling and General Type-2 Fuzzy Logic Agent Based Methods.” Ph.D., Ann Arbor, MI: University of Michigan. https://www.proquest.com/docview/896357950/abstract/2532E937DE1946D4PQ/19.

Hannapel, Shari E. 2012. “Development of Multidisciplinary Design Optimization Algorithms for Ship Design under Uncertainty.” Ph.D., United States -- Michigan: University of Michigan. https://www.proquest.com/docview/1268742764/abstract/9A3C2528509442F9PQ/3.

Keane, Robert G., Barry F. Tibbitts, and Peter E. Jaquith. 2019. “The Navy’s Ship Design Factory: NAVSEA—The ‘Golden Goose.’” *Naval Engineers Journal* 131 (3): 61–78.

Kennedy, Brian M., Durward K. Sobek, and Michael N. Kennedy. 2014. “Reducing Rework by Applying Set-Based Practices Early in the Systems Engineering Process.” *Systems Engineering* 17 (3): 278–96. https://doi.org/10.1002/sys.21269.

Kitchener, Roy, Brad Cooper, and Paul Schlise. 2021. “What Got Us Here Won’t Get Us There.” U.S. Naval Institute. January 6, 2021. https://www.usni.org/magazines/proceedings/2021/january/what-got-us-here-wont-get-us-there.

Kodukula, Prasad, and Chandra Papudesu. 2006. *Project Valuation Using Real Options: A Practitioner’s Guide*. Ft. Lauderdale, Fla: J. Ross Pub.

Kossiakoff, Alexander, Steven M. Biemer, Samuel J. Seymour, and David A. Flanigan. 2020. *Systems Engineering: Principles and Practice*. Third edition. Wiley Series in Systems Engineering and Management. Hoboken, NJ: John Wiley & Sons, Inc.

LaGrone, Sam. 2022. “Navy Unveils Next-Generation DDG(X) Warship Concept with Hypersonic Missiles, Lasers.” *USNI News* (blog). January 12, 2022. https://news.usni.org/2022/01/12/navy-unveils-next-generation-ddgx-warship-concept-with-hypersonic-missiles-lasers.

McKenney, T, M Buckley, and D Singer. 2012. “Differentiating Set-Based Design from Other Design Methods and the Cultural Challenges of Implementation.” In *Proceedings, 11th International Maritime Design Conference*. Glasgow, UK.

McKenney, Thomas Abbott. 2013. “An Early-Stage Set-Based Design Reduction Decision Support Framework Utilizing Design Space Mapping and a Graph Theoretic Markov Decision Process Formulation.” Ph.D., United States -- Michigan: University of Michigan. https://www.proquest.com/docview/1499244175/abstract/190F38E60D564EE2PQ/1.

Mebane, Walter L., Craig M. Carlson, Chris Dowd, David J. Singer, and Michael E. Buckley. 2011. “Set-Based Design and the Ship to Shore Connector: Set-Based Design.” *Naval Engineers Journal* 123 (3): 79–92. https://doi.org/10.1111/j.1559-3584.2011.00332.x.

Mun, Johnathan. 2006. *Real Options Analysis: Tools and Techniques for Valuing Strategic Investments and Decisions*. 2nd ed. The Wiley Finance. Hoboken, N.J: John Wiley & Sons.

Naval Sea Systems Command. 2008. *Ship Design and Analysis Tool Goals*. Washington, DC: Naval Sea Systems Command.

———. 2014. *NAVSEA Engineering and Technical Authority Policy*. NAVSEAINST 5400.111A, 5400.11A. Washington, DC: Naval Sea Systems Command.

Neufville, Richard de, and Stefan Scholtes. 2011. *Flexibilty in Engineering Design*. First. Engineering Systems. Cambridge, MA: MIT Press.

Oers, Bart van, Erik Takken, Etienne Duchateau, Ruben Zandstra, Siebe Cieraad, Wendy van den Broek-de Bruijn, and Michel Janssen. 2018. “Warship Concept Exploration and Definition at The Netherlands Defence Materiel Organisation.” *Naval Engineers Journal* 130 (2): 63–84.

Office of the Chief of Naval Operations. 2018. *A Design for Maintaining Maritime Superiority*. 2.0. Department of Navy. https://media.defense.gov/2020/May/18/2002301999/-1/-1/1/DESIGN_2.0.PDF.

———. 2020. “Report to Congress on the Annual Long-Range Plan for Construction of Naval Vessels.” 2-295B307. Washington, DC: Department of the Navy.

———. 2021. “Report to Congress on the Annual Long-Range Plan for Construction of Naval Vessels for Fiscal Year 2022.” Report to Congress E-4E1C. Washington, DC: Department of the Navy.

Office of the Inspector General. 2000. *DoD Acquisition Workforce Reduction Trends and Impacts*. D-2000-088. Washington, DC: Department of Defense.

Office of the Secretary of the Navy. 2020. “Advantage at Sea Prevailing with Integrated All-Domain Naval Power.” Department of the Navy. https://apps.dtic.mil/sti/citations/AD1118532.

———. 2021a. *Department of the Navy Research and Development, Acquisition, Associated Life-Cycle Management, and Sustainment Responsibilities and Accountability*. SECNAVINST 5400.15D. Washington, DC: Department of Navy.

———. 2021b. “One Navy-Marine Corps Team: Strategic Guidance From The Secretary of the Navy.” Strategic Guidance 100721. Washington, DC: Department of the Navy. https://www.navy.mil/Press-Office/Press-Releases/display-pressreleases/Article/2805409/secnav-del-toro-outlines-top-priorities-on-maritime-dominance-people-strategic/.

———. 2021c. “Report to Congress on National Surface Warship Engineering and Design Capability.” Report to Congress. Washington, DC: Department of the Navy.

Office of the Under Secretary of Defense for Research and Engineering. 2022. *Systems Engineering Guidebook*. Washington, DC: Office of the Secretary of Defense.

Page, Jonathan (Jonathan Edward). 2011. “Flexibility in Early Stage Design of US Navy Ships : An Analysis of Options.” Thesis, Massachusetts Institute of Technology. http://dspace.mit.edu/handle/1721.1/68572.

Raudberget, D. 2010. “THE DECISION PROCESS IN SET-BASED CONCURRENT ENGINEERING - AN INDUSTRIAL CASE STUDY.” *Proceedings of the International Design Conference*, May, 10.

Rehn, Carl F., Sigurd S. Pettersen, Jose J. Garcia, Per O. Brett, Stein O. Erikstad, Bjørn E. Asbjørnslett, Adam M. Ross, and Donna H. Rhodes. 2019. “Quantification of Changeability Level for Engineering Systems.” *Systems Engineering* 22 (1): 80–94. https://doi.org/10.1002/sys.21472.

Ricci, Nicola, Matthew E. Fitzgerald, Adam M. Ross, and Donna H. Rhodes. 2014. “Architecting Systems of Systems with Ilities: An Overview of the SAI Method.” *Procedia Computer Science*, 2014 Conference on Systems Engineering Research, 28 (January): 322–31. https://doi.org/10.1016/j.procs.2014.03.040.

Ricketts, Myron V. 2011. “NAVSEA Technical and Acquisition Personnel1: NAVSEA Technical and Acquisition Personnel.” *Naval Engineers Journal* 123 (3): 103–18. https://doi.org/10.1111/j.1559-3584.2011.00334.x.

Ross, Adam M., and Matthew E. Fitzgerald. 2022. “Flexibility Framework Compendium.” Cambridge, MA: Diakronos Solutions Incorporated.

Ross, Adam M., and Donna H. Rhodes. 2015. “Towards a Prescriptive Semantic Basis for Change-Type Ilities.” *Procedia Computer Science*, 2015 Conference on Systems Engineering Research, 44 (January): 443–53. https://doi.org/10.1016/j.procs.2015.03.040.

Ross, Adam M., Donna H. Rhodes, and Daniel E. Hastings. 2008. “Defining Changeability: Reconciling Flexibility, Adaptability, Scalability, Modifiability, and Robustness for Maintaining System Lifecycle Value.” *Systems Engineering* 11 (3): 246–62. https://doi.org/10.1002/sys.20098.

Schaffner, Michael A., Marcus W. Shihong, Adam M. Ross, and Donna H. Rhodes. 2013. “Enabling Design for Affordability: An Epoch-Era Analysis Approach:” Fort Belvoir, VA: Defense Technical Information Center. https://doi.org/10.21236/ADA585035.

Schank, John, Scott Savitz, Ken Munson, Brian Perkinson, James McGee, and Jerry Sollinger. 2016. *Designing Adaptable Ships: Modularity and Flexibility in Future Ship Designs*. RAND Corporation. https://doi.org/10.7249/RR696.

Shallcross, Nicholas, Gregory S. Parnell, Ed Pohl, and Eric Specking. 2020. “Set-based Design: The State-of-practice and Research Opportunities.” *Systems Engineering*, July, sys.21549. https://doi.org/10.1002/sys.21549.

Singer, David J. 2003. “A Hybrid Agent Approach for Set-Based Conceptual Ship Design through the Use of a Fuzzy Logic Agent to Facilitate Communications and Negotiation.” ProQuest Dissertations Publishing.

Singer, David J., Norbert Doerry, and Michael E. Buckley. 2009. “What Is Set-Based Design?” *Naval Engineers Journal* 121 (4): 31–43. https://doi.org/10.1111/j.1559-3584.2009.00226.x.

Singer, David, Jason Strickland, Norbert Doerry, Thomas McKenney, and Cliff Whitcomb. 2017. “SNAME T&R Bulletin 7-12 Set-Based Design.” The Society of Naval Architects and Marine Engineers.

Sobek II, Durward K., Allen C. Ward, and Jeffrey K. Liker. 1999. “Toyota’s Principles of Set-Based Concurrent Engineering.” *Sloan Management Review* 40 (2): 67–83.

Specking, Eric, Nicholas Shallcross, Gregory S. Parnell, and Edward Pohl. 2021. “Quantitative Set-Based Design to Inform Design Teams.” *Applied Sciences* 11 (3): 1239. https://doi.org/10.3390/app11031239.

Sturtevant, Glen H, Lance A Flitter, Heath L Marshall, and Jeffrey M Voth. 2014. “Flexible Ships: Affordable Relevance Over the Lifecycle.” *Naval Engineers Journal* 126 (3): 14.

Toche, Boris, Robert Pellerin, and Clement Fortin. 2020. “Set-Based Design: A Review and New Directions.” *Design Science* 6: e18. https://doi.org/10.1017/dsj.2020.16.

Walden, David D., Garry J. Roedler, Kevin Forsberg, R. Douglas Hamelin, Thomas M. Shortell, and International Council on Systems Engineering, eds. 2015. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. 4th edition. Hoboken, New Jersey: Wiley.

Wang, Tao, and Richard de Neufville. 2006. “8.1.3 Identification of Real Options ‘in’ Projects.” *INCOSE International Symposium* 16 (1): 1124–33. https://doi.org/10.1002/j.2334-5837.2006.tb02800.x.

Ward, Allen C. 1989. “A Theory of Quantitative Inference Applied to a Mechanical Design Compiler,” January. https://dspace.mit.edu/handle/1721.1/6977.

———. 1990. “A Recursive Model for Managing the Design Process.” In *2nd International Conference on Design Theory and Methodology*, 47–52. Chicago, Illinois, USA: American Society of Mechanical Engineers. https://doi.org/10.1115/DETC1990-0107.

Ward, Allen C, Jeffrey K Liker, John J Cristiano, and Durward K Sobek II. 1995. “The Second Toyota Paradox: How Delaying Decisions Can Make Better Cars Faster.” *Sloan Management Review* 36 (3). https://doi.org/10.1016/0024-6301(95)94310-U.

Appendix A: <u>Excerpts</u> from Design Guidance Memorandum 002

Subject: DGM-002 Set-Based Design Method Guidance
References:
Enclosures: (2) Glossary of Terms

Purpose

The purpose of this Design Guidance Memorandum (DGM) is to communicate the tenets of Set Based Design and how the Large Surface Combatant (LSC)[9] design team shall implement it in the Pre-Preliminary Design (PPD) and Preliminary Design (PD) phases of our design process.

Background

In the early stages of design, the LSC design team shall employ set-based methods to explore the design space and conduct capability to cost trade-offs to help further refine the requirements and define the feasible design space for PD, Contract Design (CD) and Detail Design (DD) efforts in the future. By using set- based design methods, the LSC Design Team can work within their individual domains concurrently, reducing the overall design time required, base the critical early stage decisions on data rather than conjecture or local optimizations, and consistently and persistently document assumptions, information, analyses, findings, and decisions.

Actions

Establish a Design Space

Within your domain, establish the boundaries and Sets necessary to explore the design space and make decisions. Develop them first with only relevant domain knowledge, without imposing boundary conditions on the exploration of the set from other domains. This will ensure full exploration of the domain's Set. Document the approach used to create the design space and any requirements imposed on your domain.

Document Assumptions

Assumptions shall receive the code "ASM" and follow the format ASM-LLL-####. Be careful to separate assumptions from approaches and requirements and boundary conditions. Assumptions are generally used to complete gaps in an analysis. Approaches communicate how you intend to generate information.

Create Metrics and Criteria

Metrics help track the SBD process, for instance, numbers of viable candidates remaining in a set, numbers of solutions removed from a set, variability of the candidates in the set, etc. Criteria are used to help make decisions to remove infeasible or dominated candidates in a set.

---

[9] Large Surface Combatant was the name we used for the ship before it was declared the DDG(X).

- *Infeasible solutions* do not meet a requirement or do not have viable intersections with any other remaining solutions of other sets.
- *Dominated solutions* do not display desired performance characteristics along several measures, i.e. less efficient AND more costly AND longer repair times AND fails more often… Do not recommend removal of a portion of the design space based on dominance of one measure of performance, because that portion of the design space could perform quite well along another performance parameter that may be equally or more important.

Document Findings

Findings are facts and conclusions that begin to lead towards a decision. A decision could be driven by one finding or several, but they will usually also help to inform the requirements generation process with the sponsor at OPNAV. Findings, of course, are within a given context using certain assumptions that were documented as noted above. This context and the associated assumptions must be captured with the findings so they are not lost and in case the context or assumptions changes with time. This should also drive the behavior that you should not brief information and data unless it represents knowledge gained. Do not show data for the sake of showing data. Pull out what the data is teaching the team and brief it as a finding.

Conduct Set Review

Follow the format in Enclosure (1)[10] as much as possible. The objective of a Set Review is to gain approval of an assumption, gain approval of a finding/conclusion, gain agreement on an approach, or garner a decision from the decision authority, generally the SSDM and any other TWHs deemed appropriate. These approvals and decisions shall be documented in a DDM.

Route a Design Decision Memorandum

Follow the process outlined in DGM-001 to generate minutes and route the decision memorandum.

Configuration Control

All correspondence and documentation related to the Set shall use its designated three-letter code, such as HUL for {hull}, FLX for {flexibility}, and POG for {power generation}. Special categories for codes include {assumptions}, ASM, and {findings}, FND.

Integration by Intersection

Once sets are generated, validated, and reduced individually, the team will progress the design by creating the intersections between one Set and another. These intersections will reveal other areas of infeasible design space (perhaps a POG design area that is too heavy or large to fit within any HUL design areas) and new analyses for dominance of certain combinations over others. The intersections are created by determining what characteristics the two Sets have in common along any measure and approaching that common measure from both Sets.

This Design Guidance is in effect immediately and is the primary method and format for how decisions are to be made regarding LSC design.

(signature)

---

[10] This enclosure has been removed since the example for the design team contains sample technical data.

Glossary

Approach: the process used to explore the design space or conduct the evaluation. Sometimes confused with assumptions. Both approaches and assumptions should be communicated, but an approach does not necessarily need explicit approval. Approaches are generally not subject to change or modification based on new knowledge, although an approach may be repeated with that new knowledge to test whether the outcome of an evaluation is different (a new/different finding).

Assumption: an educated guess used to fill in knowledge gaps in order to conduct an evaluation. These are subject to change based on future knowledge gained via studies or clarifications. These must be approved by the SDM to create valid evaluations and findings.

Criterion: a rule or principle used for evaluating or testing a set or variable within a set that drives a decision. Criteria isn't simply the characteristic being tested (i.e., weight, specific fuel consumption, area), but the standard upon which it is tested (greater than ___, less than ___, better than ___, in accordance with ___, etc.).

Design factor / design variable: A design factor (or design variable) is a solution parameter, characteristic, or relationship that influences the design at the system level. Examples include but are not limited to: length of ___, weight of___, interface of___, density of___, volume of ___, specific fuel consumption, range, time on station, acquisition cost, carrying cost, sustainment cost, maintenance time, mean time to repair, mean time between failures, expected service life, and many others.

Design space: One or more independent design factors create a design space within a domain. Also, multiple domains combined with their independent design factors create the larger design space. Ultimately, all domains and their design factors roll up into Ship Concepts (SHP) and the Preliminary and Contract Design elements.

Diversity: This is a metric of how different the solutions within a feasible design space are from each other. High diversity indicates that if any one component or technology proves not to work as intended, the subset of the solutions in the design space that do not use the component or technology is viable.

Domain: This is a contributor of technical expertise that can evaluate design factors.

Evaluation: The end results of assessments upon which the data and knowledge are applied against criteria or criterion or standards to inform a lasting decision (a finding).

Feasible: A feasible solution or region is a point within or a region of a design space where domain analysis conducted to date indicates the solution or solutions in the region will work. Future work or additional domain analyses may result in a solution or region currently evaluated as feasible being re-evaluated as infeasible.

Finding: the results of an evaluation. It represents knowledge gained and should be irrefutable within the boundaries of identified assumptions and the communicated approach.

Highly dominated: A highly dominated region of a design space is a region where another feasible region is evaluated as being better in a significant number of variables of interest. There is opportunity for interpretation regarding how many variables must be dominant before assessing a region as “highly” dominant. A sensitivity analysis may reveal a reasonable number of variables to use in this evaluation.

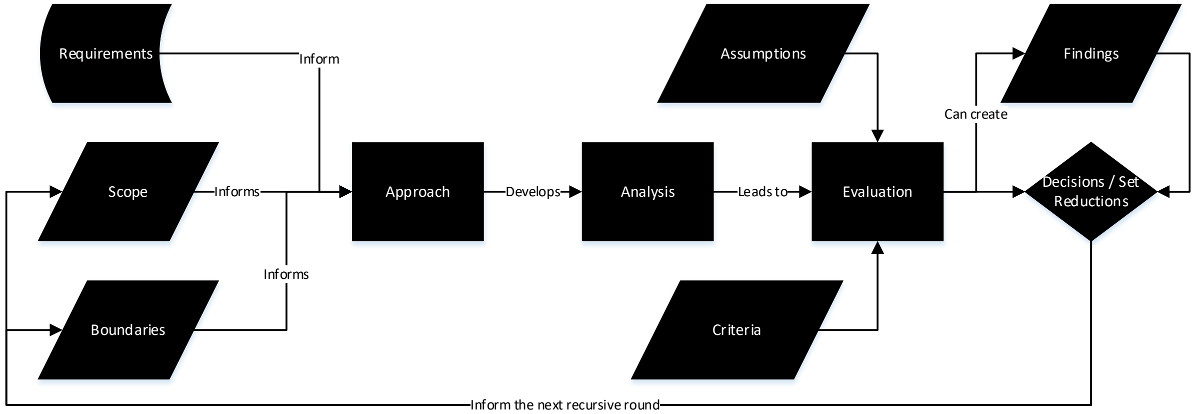
Infeasible: An infeasible region is a region where there is high confidence that a solution does not exist and no new information would alter that assessment.

Stakeholder preference and utility: Stakeholder preferences are those aspects of the system that the customer values. Utility is the relative value the stakeholders place on each preference.

Unexplored: A region of a design space not evaluated by a domain.

Viable: A viable solution is one that, if built, would perform as intended; subsequent analysis and testing will not show the solution to be infeasible. During the early stages of design, one cannot determine with certainty that any solution is viable. Instead, a region of a design space is considered viable if there is high confidence that at least one of the feasible solutions within the design space would prove to be viable if built; it is not necessary to identify which of the solutions are viable, only to show with high confidence that at least one of the many would be viable. An assessment of whether a region of the design space is viable is typically made based on diversity.

Basic Flow Charts of terms through our Set-Based Process



Appendix B: Sample Design Decision Memorandum Form

One can see, the form has ten sections.

The first section captures administrative and configuration control data like the originator and their contact information, the date of the review, the sequential DDM number for that {set}, and if there are any related DDMs.

The second section captures an overview of the meeting and purpose of the DDM. This part summarizes the "Purpose" data from the Set Review and provides rationale for the DDM, such as "successful decision" or "set reduction" or "approved assumptions." This part also records attachments to the DDM, of which there were always a minimum of two: 1) the Set Review presentation (often more than one presentation if the Review was not successful in reaching this outcome the first time around and there was "a comeback") and the meeting minutes (from each relevant review leading up to the DDM). An example of another attachment that was sometimes included would be an e-mail with comments or concurrence from a technical domain manager TWH.

Section Three of the DDM summarizes design impacts. This section was free form and could include impacts internal or external to the domain, including cost or performance attributes.

Section Four captures signatures and comments from team leaders within the domain, then Section Five captures comment, signatures, and concurrence or non-concurrence from any relevant TWHs (technical domain managers).

Section Six shows which functional areas of team should receive distribution of the DDM (or at least awareness that it is available for review and use).

Finally, sections seven through ten capture names, comments, and disposition from the DSDM, SSDM, program office (as appropriate), and resource sponsor (as appropriate).

Large Surface Combatant (LSC) Design Decision Memorandum (DDM) DDM Form Rev3 - 31Oct2019.pdf

Section 1 - Admin / Set Review: Originator

DDM No.: Rev. No.: - Origination Date: Revision Date:

DDM Title: Related DDMs:

Originator: Phone: E-mail:

Section 2 - Overview: Originator

Description:

Rationale:

☐ Technical Requirements ☐ Design Development ☐ Affordability Initiative ☐ Other (specify):

Attachments (list file names):

1.
2.
3.
4.
5.
6.

Section 3 - Design Impacts: Originator

Design Area	Impact

Section 4 - Ship Design Team Review: Subject Matter Expert / Task Lead

Task SME Name: Team: Date:

Comment/Recommendation:

☐ Proceed ☐ Stop DDM ☐ Requires Revision

Task Lead Name: Team: Date:

Comment/Recommendation:

☐ Approve ☐ Disapprove ☐ Disapprove / Rework as New Revision

Task Lead Signature:

Section 5 - Technical Review: Technical Warrant Holder(s)

TWH (if required) Name: Date:

Title/Technical Warrant Area:

Comment/Recommendation:

☐ Concur with DDM ☐ Do Not Concur ☐ Do Not Concur / Rework as New Revision ☐ Concur with Inclusion of Attached Comments

TWH Signature:

TWH (if required) Name: Date:

Title/Technical Warrant Area:

Comment/Recommendation:

☐ Concur with DDM ☐ Do Not Concur ☐ Do Not Concur / Rework as New Revision ☐ Concur with Inclusion of Attached Comments

TWH Signature:

TWH (if required) Name: Date:

Title/Technical Warrant Area:

Comment/Recommendation:

☐ Concur with DDM ☐ Do Not Concur ☐ Do Not Concur / Rework as New Revision ☐ Concur with Inclusion of Attached Comments

TWH Signature:

Section 6 - Distribution: Ship Design Team

Distribution			Area	Name	Date	Endorse?	
☐	☐	☒	Design Management			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Design Integration			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	System Engineering			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Naval Architecture			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Marine Engineering			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Warfare Systems			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Flexibility			☐	☐
Info	Action	N/A	Comments:			Yes	No
☐	☐	☒	Other (specify)			☐	☐
Info	Action	N/A	Comments:			Yes	No

Section 7 - Deputy Ship Design Manager (DSDM) Review

DSDM Name: Date:

Comment/Recommendation:

☐ Approve ☐ Disapprove ☐ Disapprove / Rework as new Revision Requires SSDM Review: ☐ Yes ☐ No

DSDM Signature:

Section 8 - Senior Ship Design Manager (SSDM) Review

SSDM Name: Chris Higgins Date:

Comment/Recommendation:

☐ Approve ☐ Disapprove ☐ Disapprove / Rework as new Revision Requires PEO Ships Review: ☐ Yes ☐ No

Requires OPNAV N96 Rvw: ☐ Yes ☐ No

SSDM Signature:

Section 9 - Program Office Review (if required)

PEO Ships Name: Date:

Comment/Recommendation:

☐ Concur ☐ Do Not Concur Requires OPNAV N96 Rvw: ☐ Yes ☐ No

PEO Ships Signature:

Section 10 - Sponsor Review (if required)

OPNAV Name: Date:

Comment/Recommendation:

☐ Concur ☐ Do Not Concur

OPNAV N96 Signature:

Print Form

Appendix C: Full Set Review Data Extraction Through Year 3