INTERNATIONAL DESIGN CONFERENCE – DESIGN 2022
https://doi.org/10.1017/pds.2022.253

Starting and Scaling a Set-Based Design Method for a Maritime System of Systems: Designing a Modern Warship

J. E. Page [1,✉], C. J. Higgins [1] and W. P. Seering [2]

[1] United States Navy, United States of America, [2] Massachusetts Institute of Technology, United States of America
✉ pagefactor@outlook.com

Abstract

There has been a growing body of literature and use cases for set-based design since its introduction in the 80s. Few studies or use cases involve highly complex systems, though, except for the hallmark work regarding Toyota in the late 90s. Over the last three years, the US Navy used set-based methods to design a complex system of systems: a warship. Their experience provides insight into the scalability of the method and design management considerations relevant to the start of similar projects.

Keywords: *collaborative design, conceptual design, design methods, design management, complex systems*

1. Introduction

1.1. Set-Based Design (SBD)

Set-based design is a process for designing that rests on the use of methods for reasoning about sets of design alternatives. Set-based design was introduced to the engineering design community in 1989 (Ward and Seering), contributing to addressing the need (Simon, 1996) for a body of partly formalizable, partly empirical, teachable doctrine about the design process. The key ideas behind set-based design are that all viable options should be considered and that no options should be eliminated from consideration unless and until there is a logical reason for doing so (Sobek, Ward, and Liker, 1999; Singer, Doerry, and Buckley, 2009).

During the 1990s, when Toyota's product development methods had been identified as innovative and effective, empirical studies of Toyota's processes (Ward, Liker, Cristiano, and Sobek, 1995) showed an alignment between those processes and the process of set-based design. Consequently, SBD came to be seen as a vital element of the Toyota Product Development System (Morgan and Liker, 2006). Subsequently, SBD methods have been employed in a wide array of fields (Raudberget, 2010; Doerry and Koenig, 2019; Parrish, Wong, Tommelein, and Stojadinovic, 2007). The recent paper by Toche (Toche, Pellerin, and Fortin, 2020) provides an extensive review of this work. It concludes that few complete examples of SBD employed to design large complex systems have been documented. The objective of this paper is to document one such case.

As executed in this study, the set-based design process begins with a curated selection from the complete set of design alternatives that address the perceived need. The project is complex enough to invoke a system-of-systems approach, with the curated selection of sets representing systems or subsystems within. Subsets representing regions of alternatives that prove to be infeasible are eliminated from the original set(s). Any team engaged in the design process can, with proper justification, declare a subset of designs infeasible. In our work, most design teams, called set teams or domain teams, are organized around a specific engineering discipline (referred to as a domain) germane

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

to ship design. Examples are the marine engineering team and the warfare systems team. Each is empowered to propose the elimination of subsets of their design space based on the assertion that they are dominated by other subsets concerning satisfying the need as then understood. While a single set team can show that a subset of alternatives is infeasible, proving that a subset is dominated often requires consensus across set teams. What's good for one set team might not be suitable for another. Reasoning on sets of alternatives requires the specification of abstractions that define the sets in question. For example, when considering the set of power systems for a ship, that domain team might decide that the subset of diesel power systems, an abstraction representing many instances, dominates the subset of nuclear power systems. Alternately, the team may decide that the complete set of diesel power systems dominates a subset of nuclear power systems but not all such systems. It is important to note that such determinations can be and often are made before detailed analyses have been conducted of all such instances, thus incurring some risk that the best solution, or set of solutions, has been removed from consideration. That risk is balanced by eliminating detailed analysis of infeasible alternatives in later stages of the process. The effectiveness of domain teams in employing set-based design depends strongly on those teams identifying viable levels of abstraction for reasoning about the elimination of subsets of alternatives and on the teams' understanding of the consequent risks.

The process of set-based design proceeds as set teams responsible for the various design subsystems advance their understanding of the viability of subsets of the set of design solutions still in consideration and, based on their increased understanding, propose justifications for the elimination of additional subsets of options. Doing so requires that the teams identify and then resolve knowledge gaps, thus informing the proposals to eliminate additional sets. The process continues until a satisfactory design emerges.

The devil, of course, is in the details. This paper describes how set-based design methods have been employed to concurrently design a surface combatant ship and its requirements for the United States Navy. We present this example to show the set of processes that have been developed for designing a large complex system in this way and to describe some of the challenges associated with employing these set-based processes. Throughout the rest of the paper, we use set notation with braces when referring to a distinct {set}.

1.2. Ship Design, Arleigh Burke, and DDG(X)

The design of ships, especially warships, has always been complicated; challenging but manageable by using some basic first principles like Archimedes' principle of buoyancy and the balance of center of gravity and center of rotation in a plane. However, with time, the designs have gained complexity as the needs grew to require floating cities that generate enough power for several neighborhoods, carry enough food to last weeks, account for a built-in fire department, and all while floating upright on the dynamic ocean surface. Additional considerations such as regulations for clean air emissions and clean ballast discharges further increase the complexity. For warships, add the complexities of carrying extensive communications equipment, combat systems, sensors, protective measures, and several other considerations beyond those of commercial ships. To top it off, vessels tend to be significant capital investments that last decades, adding uncertainty regarding their mission needs over time.

Under similar conditions and with the added complexity of the Cold War, the United States designed the AEGIS combat system and its host ships: the Ticonderoga class cruiser and the Arleigh Burke class destroyer. These ships and their combat systems have served as the workhorses of the surface fleet for almost 40 years. As expected, though, the requirements for the surface navy evolved over those decades, and the Arleigh Burke class evolved through flight upgrades to meet those changing needs, up through the most recent Flight III baseline. However, the Navy recognized that required future warfighting capabilities challenge the limits of the hull form over its service life. Therefore, the Navy sought a new ship that could deliver the latest capabilities available to the surface fleet and accommodate the uncertain needs of a large surface combatant for its predicted 35-year service life.

To start the process, from July to December of 2018, the Navy formed a Requirements Evaluation Team (RET) to analyze suitable Initial Parameters for a new class of ship, then known only as Large Surface Combatant, now known as DDG(X). The intent is that the ship class will initially match the remarkable capability of the Arleigh Burke class destroyer but change, where necessary, to accommodate future uncertainty, no matter the source (geo-political, threats, socio-technical, mission, etc.). Upon completion of the RET and with approval of the Initial Parameters by the Chief of Naval

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

Operations (CNO), the Navy formed a design team to explore the cost-capability trades and concept designs that informed the requirements of the Capabilities Development Document and will lead to an affordable material solution. The Navy decided that a set-based method was appropriate to manage the complexity and complicatedness of this undertaking.

1.3. Scaling Set-Based Design

The team faced many challenges, including starting the process with few established practices, standards, instructions, or team members experienced with set-based methods; scaling the process from initial explorations to those with increasing levels of fidelity; growing and mentoring the engineering team; performing within the schedule constraints of the larger program; and executing with the resource constraints placed on the design team.

The team did have some additional guidance for their efforts based on articles regarding SBD application in the US Navy. Mebane et al. (2011) describe its use for the Ship to Shore Connector pre-preliminary design stage, a similar application but a less complex vessel. Burrow et al. (2014) describe their use of set-based methods in concept exploration of the Amphibious Combat Vehicle, another use case that is also similar and much less complex. Garner et al. (2015) discuss using the method for their Small Surface Combatant Task Force, which started to approach the complexity and complicatedness of DDG(X) but used the method for requirements generation and cost/capability trade-offs rather than ship design work. Finally, Singer et al. (2009) published a bulletin that serves as a general guide for conducting set-based methods. While all these documents were helpful to guide the leadership and the team, none of them provided directly translatable practices, standards, or tools for the DDG(X) team to implement SBD at the required scale.

Knowing that SBD was a viable method, the team hypothesized that it could scale from a Ship-to-Shore Connector or Small Surface Combatant to manage the decisions and complexity of a DDG(X). However, the team did not strictly test this hypothesis; instead, they assumed it to be accurate and consciously created a process they believed would allow the method to scale as needed. This phase of the effort started in January 2019 and continues today. The process they created addresses risks and allows for learning and modification. Importantly, it is organized so that both process and ship design knowledge are documented to make them transferrable for future ship design efforts.

2. A Scalable Method for Complex Ship Design

The first and second authors initially led the design effort. The Navy selected the second author to be the Senior Ship Design Manager (SSDM), functionally the project's chief engineer. He brings experience leading engineering teams through comparable design and acquisition projects, including the DDG 1000 program. His breadth and depth of knowledge, in addition to the lessons learned and carried forward from previous projects, proved critical to defining the DDG(X) design process. Next, the Navy selected the first author, a Commander who was serving on the faculty at MIT at the time and had wide-ranging experience from design and construction to repair and sustainment across several ship classes, as the Deputy Ship Design Manager (DSDM) to help establish, run, and grow the team. Together they created the design process, formed the design team, led process execution, and ensured process and design documentation along the way.

2.1. Set Up

Starting the process fresh with a nucleus of only two people introduced myriad administrative and operational considerations. Administratively, team make-up, team structure, meeting structure, timing and periodicity, policy, briefing formats, budgets, internal and external reporting, documentation, and training all required attention and planning. Operationally, it was necessary to identify knowledge gaps, critical decisions, appropriate abstractions for beginning each {set}, plans for {set} propagation, resource needs for each of the set teams, and, critically, which knowledge gaps, decisions, and analyses came ***first*** out of the thousands that must eventually be made on a ship design. Coupled in both aspects of this planning was the requirement for it all - the team, the process, the analyses - to scale and accommodate lower levels of abstraction, higher levels of fidelity, and more documentation, and not collapse under its weight.

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

Operationally, we used the Top-Level Requirements and factors raised by stakeholders combined with our knowledge to generate risk assessments, critical path activities, and appropriate high-level abstractions to form the basis of the initial knowledge gaps and analyses. From those, we established 12 initial {sets}, representing the curated set of systems and subsystems that we believed required initial exploration and reduction. These included the basic {hull form}, fundamental {propulsion architecture} (mechanical, hybrid, integrated power system), {electrical power generation} and {distribution}, {boat handling system(s)}, {topside arrangements/geometry}, {warfare systems}, and the emergent properties of {flexibility}, {affordability}, and {survivability}. Including emergent properties as their own sets was our approach for explicitly considering them in the larger design space as negotiable with other functional requirements. We had confidence that future {sets} would emerge from these and that no others were initially required. Based on this planning, we assigned the required first studies to the leads.

We formed the team with the structure depicted in Figure 1. Each part of the structure initially had only two to four personnel assigned, except for producibility, which we could not staff until later in the process. Each area had a lead given administrative and operational control of their domain and the sets within it (examples follow in Figure 2).

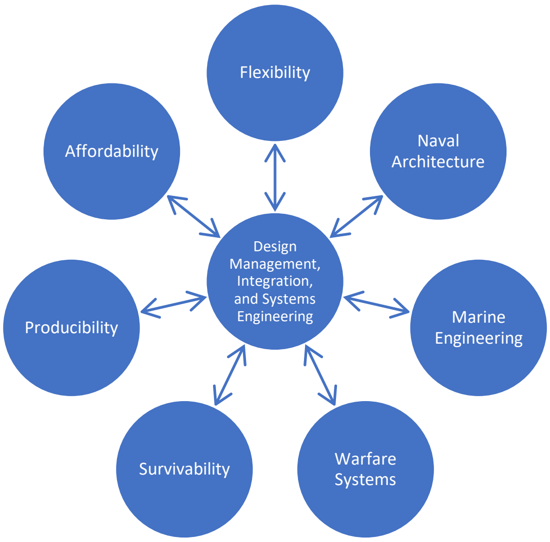


Figure 1. Initial design team organization

Figure 1 intentionally depicts that the various set teams were disconnected at the beginning. All communications came through the two of us at the hub to manage information flow in the critical and fragile early days. We wanted to ensure {sets} were individually developed as fully as possible using relevant domain knowledge and analysis without over-constraining their exploration due to opinions from other domains or *a priori* evaluation of a potential solution. We intentionally did not fill the position of the Design Integration Manager (DIM) for a year (a position which might typically be one of the first filled) to highlight this philosophy and ensure intersection/integration discussions occurred with us. We were not focused on developing ship concepts, despite being in the concept design phase, but rather on developing the {sets} that defined the concepts. For instance, the {hull} set tracks a characteristic for length overall, the {arrangements} set tracks subdivision length, and the {power generation} set tracks a characteristic for engine length. Depending on which characteristic is allowed to dominate, the engine length characteristic could drive the overall length of the ship through the subdivision length, or the subdivision length characteristic could eliminate certain engines from consideration because of their size. It is our view that at the beginning of a process like this, no viable {subsets} within a {set} should be defined as infeasible by another set team. The design of a ship is so tightly coupled and interdependent that allowing rules and opinions from one domain to influence other domains may over-constrain the problem such that the solution space becomes unnecessarily small or even the {null set}. However, an alternate view is that they also reveal regions where {sets} may not currently intersect, but where an expansion (through research and development, market research, critical thinking, brainstorming, constraint relaxation, and other methods) of one {set} could

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

create a new, viable, even dominant region of intersection. We were careful to use this philosophy and not rule out any {subset} too soon unless the internal domain knowledge for that {set} revealed that no assumption or exploration could ever make a {subset} of that {set} space feasible.

In addition to the team structure and the leadership selection, we put effort into developing policy and conducting training. We created the initial Design Guidance Memorandums (DGMs). These DGMs promulgated policy and guidance for the team to use in conducting business. They included such topics as defining critical systems and non-developmental items and direction regarding when development is acceptable; the digital engineering strategy, initial ship specifications to use which helped guide assumptions in analyses; design margin policy; and, of course, how to document meetings, minutes and decisions, and route and store the information for posterity. The DGM for Set-Based Design Method Guidance is 13 pages in length with its appendix. It communicates the philosophy of SBD and why we chose to use this method vice point-based or other concurrent design methods. It laid out the basic actions of establishing a design space, creating metrics and criteria to address a knowledge gap, documentation of assumptions, analysis methods and ways of documenting the findings, conducting a Set Review, and routing a consequent Design Decision Memorandum (if appropriate, to memorialize that outcome). It also addressed how and when to integrate by intersection and provided an attachment to guide the flow of information during a Set Review and reach a decision. It included such policy as any team member can choose to keep a portion of the design space alive if they find it valuable for any reason, but only the SSDM or DSDM can decide to remove any portion from further consideration.

Lack of a design site helped management effectively keep the initial inter-team communications to a minimum; very few team members were co-located at this stage. While we had a vision to eventually populate a design site and allow high-frequency communications as the team grew and complexity multiplied, COVID prevented this from happening.

One notable exception to integration and communication limits is our use of a {ship} set and an {affordability} set. The elements of the {ship} set represent balanced ship concepts that meet all essential naval engineering criteria such as floating, floating upright, making speed, generating enough power, and having enough space and volume for notional systems. The {ship} set is made up of components from other {sets} and acts as a feasibility check to show we can create conceptual ships that can meet all the requirements with a particular set of selections from each of the explored domains. It provided insight into how the design met the requirements and how well. The {ship} set offered a new alternative to the typical ship design process that develops and iterates on several ship concepts to analyze one problem at a time.

Within the {ship} set, we developed benchmark and excursion ship concepts to produce a metric for use by the other domain's {sets}. Benchmark ship concepts represent the full range of the open requirements and {set} space; for example, one benchmark may be developed for a ship that carries the least amount of payload and another for the most amount of payload. Whenever a set lead wanted to understand their {set} effects on the system-of-systems (ship), they requested excursions from the benchmarks to analyze that impact. For example, the {propulsion} set used excursion concepts to assess the total system effects of mechanical, hybrid, and electric propulsion systems. We determined metrics such as 'ship acquisition cost delta' to measure the {set}. If the analysis of a particular {set} demonstrated a similar impact across all benchmark ships, we determined it was a stable relationship against the whole {set} space and documented that finding (e.g., cost of increasing the sustained speed requirement across all {propulsion architecture} sets).

Thus, because the {affordability} set and {ship} set encompassed several other {sets} and sought to differentiate between different elements within those {sets}, we naturally allowed communication and integration to occur in a controlled way. One other outcome of this process is the observation that we have not yet combined elements of {sets} into a {ship set} that is has been found infeasible. This addresses a risk that some people communicated to us regarding SBD with a ship, thinking that if one only rules out {subsets} of {sets} without integrating them, then the process could play out and result in selections that do not integrate into a viable, balanced ship in the end. The use of benchmarks and excursion concepts to produce full-ship metrics attributed to individual sets in individual domains addresses the understanding of the secondary impacts of integrating multi-domain {sets} on the total ship and acts as a virtual prototype in advance of potential physical prototypes.

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

2.2. Un-Training and Training

We knew that the team had limited familiarity with SBD. Therefore training was paramount. We didn't anticipate how much ***un***-training would need to take place to start efficiently achieving results. Some team members continued conducting the same activities as before and labeled them (incorrectly) analyses on {sets}, then tried to present their information in our given format, which evokes the analogy of a square peg and a round hole. Some team members had experience with other versions of SBD from other projects and tried to use that same process within ours, which also did not work, since that previous work was for requirements, not design, and was at a different level of fidelity and abstraction. Some members were able to mentally transition from a point design to design using multiple points but could not make the complete mental transition to {sets}. Others outright claimed that SBD could not work for their domain or {set} for some underlying architectural reason, notably with various {arrangements} sets where the typical practice is to take others' design decisions and place them logically in relation to each other within the larger system. These were mental roadblocks to fully understanding SBD or lack of appropriate tools to execute it.

One practical training point is about what constitutes a {set}. Indeed, much of the literature on SBD directs one to identify and examine the design space, as if this part of the process is self-evident. Some considered a {set} to be an enumeration of all possible material solutions in their given area. This definition, though, ignores behavioral, temporal, functional, and other aspects of the {set} that could be of value. Also, complete enumeration of material solutions may unnecessarily complicate the analysis since many of those enumerations may not provide diverse alternatives and only waste computing time. A {set}, rather, ought to be a grouping of appropriate characteristics and attributes of a design element. These characteristics ought to be informed by known material solutions in the particular design space or R&D pipelines. The {set} serves to close knowledge gaps and leads to decisions relative to baselines of the design through methods like controlled convergence.

Therefore, we spent quite some time mentoring and training the team, both inside and outside the formal Set Reviews. We held specific meetings and workshops strictly to introduce the process and expectations for performance. During mentoring (especially after unproductive Set Reviews), we would reiterate some of the basics. Frequent topics were: the definition of {set} boundaries (often too tight), ruling out parts of the space rather than choosing (target removing 1/3 to 1/2 of the space instead of 99% of it), choosing the right level of abstraction in the {set} to answer the question (the analysis evaluated something in too much detail or using the wrong characteristics of the {set} space), how to determine feasibility or dominance for the question at hand, the difference between approaches and assumptions, specifying and developing the viable solution space without initial regard to what may be the best solution space (that is what the analysis is for!), and how to rule out using only domain knowledge without intersections and integration. Mentoring continues to this day.

2.3. Executing and Making Decisions

We had our first Set Reviews in March 2019. The first Set Reviews were not actual Set Reviews since they intentionally did not fully follow the prescribed format and only reviewed the assumptions needed for the analysis. This is still true today with each new {set} created. We do this to address the same risk as early integration: assumptions can be narrow and artificially limit potentially valuable regions of the design space prematurely. Therefore, we review and approve all assumptions before analysis to protect against this to the greatest extent possible.

Subsequent Set Reviews are the culmination of the work of the set team. The desired outcome of the Set Review is an approval of an **assumption, recommendation, finding, or decision**. One kind of **decision** is a reduction, which involves removing an infeasible or dominated portion of a {set} space that is not likely to be made feasible or dominant by any future study or information. The {set} reduction is ***implemented across all domains***. The last reduction of a {set} naturally becomes a decision for the {set} and defines that element of the baseline. **Findings** are outcomes of analysis that have ***created knowledge*** for the team to use in subsequent work. **Recommendations** and **decisions** are closely coupled outcomes; the difference between the two is that a **recommendation** is made by one domain with the recognition that there may be effects in other domains that ought to be explored, whereas a **decision** considers the cross-domain implications. Therefore, **recommendations**, like

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

findings, represent knowledge for different domains to potentially use in subsequent analyses, and *the trade space remains open for further consideration, intersection, and analysis*. A **decision** locks in an element of the baseline and ***closes the trade space from any further consideration in any domain***.

In most early Set Reviews, both before and after we allowed intersections, we ruled out portions of the design space on the basis of infeasibility. This outcome is natural and expected. Ruling out based on infeasibility involves only one criterion and usually one domain. For instance, power generation quickly ruled out solar, wind, and nuclear power sources because they could not generate enough power, generate it consistently enough, or were too expensive. Suppose the solution can't generate enough power. In that case, we need not worry about whether it has an ideal weight, cost, or stability characteristics and do not spend the energy to explore those characteristics.

Using dominance criteria to rule out a portion of the design space didn't begin until the second year. This is also natural and expected. Because dominance is a system issue that involves balancing multiple requirements and criteria, using it as the basis for removal requires consensus from a more significant portion of the design team outside one domain and perhaps the entire design team. During the second year, we considered the {sets} developed enough individually to start intersecting across domains. Also, we noticed that a lot of our mentoring covered the same topics repetitively across domains.

Thus, we changed the structure during the second year to require participation from each domain during each Set Review. This accomplished two outcomes. First, when a case example that required just-in-time training happened during one Set Review, the constructive feedback was provided to the entire team at once, so each domain and set team could learn from each other and apply the lessons within their {sets}. Second, the Reviews became a communication and grounding tool for the team. They enabled conversations on the impacts of recommendations and decisions with the entire team. These conversations could take place audibly on the line or via the chat function; either way, they were part of the record for how we substantiated an outcome. They provided a shared knowledge of the design and its activities outside of a given domain to the entire team at once. We find that having that larger perspective on complex designs is beneficial because it helps team members to understand they are not simply a cog helping the overall machine to run; the transparency reveals their significance.

Thus, at this point in the second year, the team started to grow, the process began to scale, and the team's structure began to morph into what it is today, depicted in Figure 2. One can see the underlying structure was primarily retained, but there are a few key differences at the hub of the system and its spokes, which now resemble a rim.

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. They were still "hub" activities, but also took on some sets of their own, like {requirements}, {topside}, and {arrangements}. 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 and handled all formatting, structure, content, 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 management of the outcomes and associated documentation. The SSDM still retains decision authority to approve or disapprove all proposed outcomes and possesses the sole authority to deconflict dominant solutions that are opposed between the {sets}. The Design Integration Manager and SBD Lead also generated, tracked, and maintained the running list of knowledge gaps. This was also a crucial activity because the knowledge gaps informed the next {sets} that required development or the subsequent analysis needed within an existing {set}.

Second, the spokes/rim of the new structure changed fundamentally. While we retained the same functional areas as initially set up, the set teams started to share {sets} and information among each other as one of our forms of communication. Figure 2 presents some of these {sets}. The outer ring encloses these to represent that these characteristics, variables, and {sets} are communicated or negotiated with other set teams (Singer et al. 2009) in contrast with internal variables for the use of only one set team. Of note, this is a mere sampling. As of the time of this writing, there were close to 150 {sets} that the team had defined and in which some definition or analysis has taken place, with almost 40 of them actively engaged in development, analysis, or Set Review preparations.

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

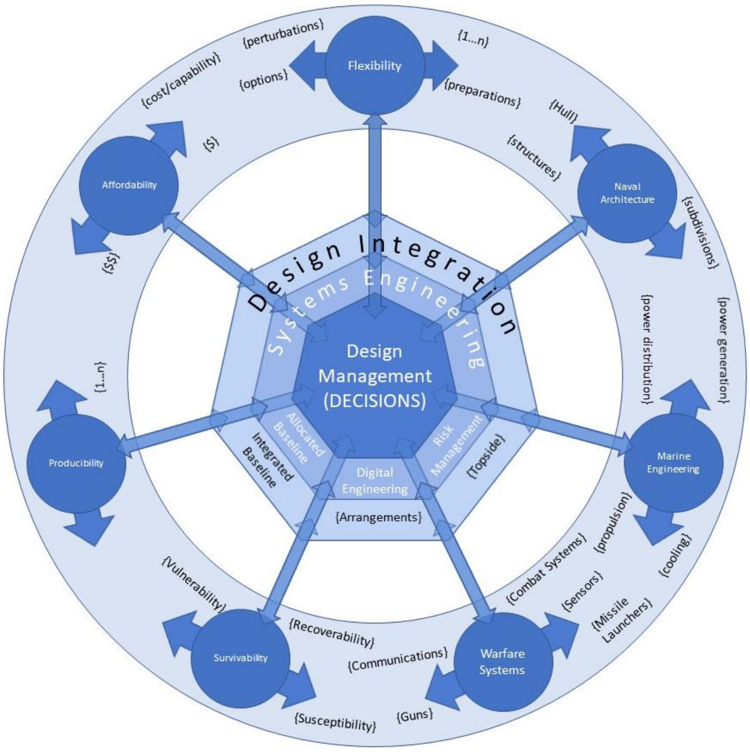


Figure 2. Evolved and scaled design team organization

Of course, the global pandemic was in full swing around this time, with very few people allowed to go to work, let alone in large groups. Our team was now between 80 and 100 full-time equivalents. In this instance, remote work came with a silver lining: the collaboration tools we adopted to conduct design reviews allowed text-based chat and the complete text capture of the chat room. The text-based chat function enabled the high-frequency communications desired without interrupting the presenter and provided some ready-made meeting minutes that captured the conversations and considerations taking place outside of but still relevant to the review.

These communication practices were one of many risk management techniques. We could manage the risks of ruling out a {subset} of alternatives that might be helpful to another set team by having members of that set team or functional domain present and empowered to communicate in any of several ways. Prototyping, and especially virtual prototyping, is another significant risk management undertaking. For the hull form, we will build scale models to test characteristics of its shape in tow tanks for resistance and wave tanks for stability. We will also build a representative power and propulsion plant at a land-based test site to reduce integration risk and test design alternatives. The team is extensively modeling both sets to ensure the prototypes we build provide the maximum knowledge and benefit for the Navy.

The structure of the team and execution of the process remain in place today. The DSDM received new orders but remains involved with the team in his new role. His replacement carried on the task seamlessly and continues to adapt the process as the design progresses. He and the SSDM continue to mentor, untrain and train the team, and document lessons and design artifacts. The Navy also established a program office to manage the acquisition of the platform.

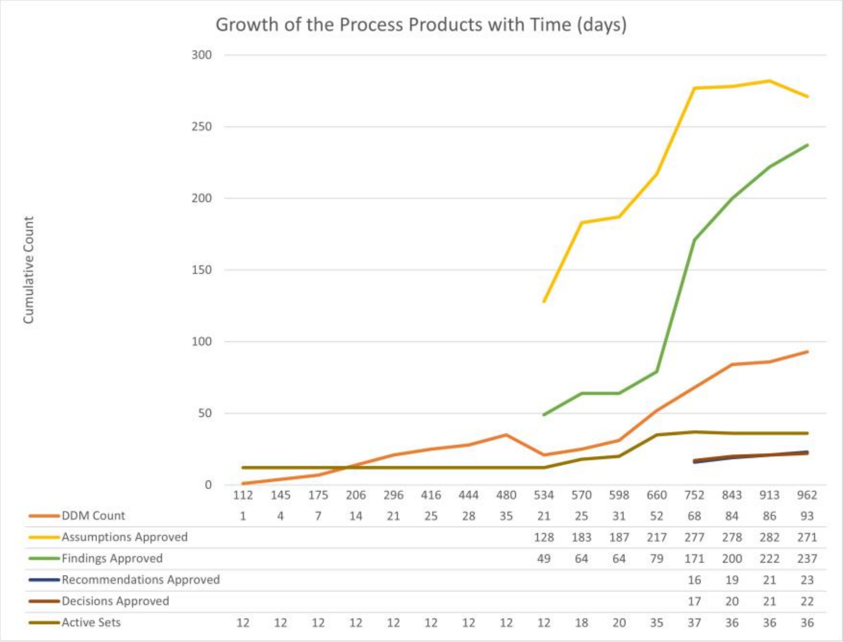
3. Metrics and Observations

Our team and process were able to scale together. We demonstrated the viability of using a SBD method for a system-of-systems level of complexity. The team steadily increased the number of {sets}, Set Reviews, assumptions, outputs, and other decisions as documented in Design Decision Memorandums (Figure 3). We also explored the design space through increasing levels of fidelity and lower levels of abstraction as we closed knowledge gaps and recorded decisions. The team also steadily expanded, involving more people requiring untraining and training, including industry

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

members. The process remained essentially unchanged through this scaling and growth, signaling that it was robust enough to operate at many levels of abstraction, complexity, and volume.

It is also essential to compare these results with previous projects (from Section 1.3). While Toyota performs at a large scale on a complex system of systems and is part of the basis for SBD, very few projects have adopted it at the scale of DDG(X). Ship-to-Shore Connector (SSC) (Mebane et al. 2011) was a 12-month effort that converged ~120 vital design parameters and, thinking combinatorially, ~10^{47} potential design options. The Amphibious Combat Vehicle (ACV) (Burrow et al. 2014) was also a 12-month effort that focused on the trade space of a particular requirement by exploring five major design attributes and included analysis of ~20k different design options. By comparison, DDG(X) concept and preliminary design are approaching 36 months of trade space exploration, and so far, involves over 140 {sets} identified, 22 approved decisions, and on average six Set Reviews per week that increase these numbers. If one enumerates our {sets}, it appears the team is evaluating over 10^{140} possible configurations and actively exploring between 10^{20} and 10^{30} of them in Set Reviews at any given time. Fortunately, SBD ignores those enumerated points and evaluates them in sets instead. Further, SSC developed two baseline models, ACV created 48, and DDG(X) has over 120 and counting. Each of these metrics will continue to grow with time, also, except for these numerous possible configurations, which will decrease over time until one final {set} remains that will be used to write specifications and construct the ship.



	112	145	175	206	296	416	444	480	534	570	598	660	752	843	913	962
DDM Count	1	4	7	14	21	25	28	35	21	25	31	52	68	84	86	93
Assumptions Approved									128	183	187	217	277	278	282	271
Findings Approved									49	64	64	79	171	200	222	237
Recommendations Approved													16	19	21	23
Decisions Approved													17	20	21	22
Active Sets	12	12	12	12	12	12	12	12	12	18	20	35	37	36	36	36

Figure 3. Process Outputs

4. Outcomes

While the efficacy of the scaling may be open to interpretation and opinion, some qualitative observations indicate the process scaled well. First, when we "handed the keys" of the process over to the SBD Lead to run it, the progress through the process did not falter, and he fully understood the philosophical and tactical approach we were attempting. Further, when the program office stood up and assumed their authority in the process, the only requested changes were adding a new signature line to the documentation for the decisions that they wanted to be elevated to the level of the Program Manager instead of the Chief Engineer and adding a meeting with the Program Manager to review such decisions with him. So, the process and its outputs have been structurally stable. The team now often needs more time each week to conduct Set Reviews because of the sheer number.

Additionally, the structure we implemented for SBD has lasting effects on other regimes. For instance, the SBD hierarchy we developed formed the basis of the logical model for our Model-Based Systems Engineering efforts. Further, many assumptions approved during Set Reviews were subsequently validated and evolved into Requirements or Specifications for conversations with Industry partners.

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press

The ultimate outcome, though, was that the process scaled up to a complex, system-of-systems level project and controlled outcomes equally well regardless of volume, complexity, or abstraction.

5. In closing, Future Work

There is much work yet to be done, both on the design itself and the use of SBD practices and observations. SBD shall continue to be the basis of domain refinement and decisions through the preliminary design phase. It will be interesting to observe the transition from preliminary design to contract design and the transition out of SBD to more conventional engineering practices. It will be even more interesting to observe if this method produced more robust results against future requirements or design changes that inevitably crop up in complex projects like warship acquisition. The Navy should undoubtedly endeavor to maintain this discipline for future concept and preliminary design work and could benefit from extending SBD practices into other types of analyses.

Some of the elements of our method that we believe translate to other processes and analyses include starting small before growing the team and scaling the process, deriving the first {sets} using criteria such as critical path and risk assessments, grounding further explorations on knowledge gaps, and fully developing {sets} independently without regard to intersections and integration. In conclusion, we wish to note that, though this case study was of the Set-Based Design process applied to a large ship, what we have learned and described here suggests that the method as it has been developed could support the design of other types of complex systems.

References

Burrow, J., Doerry, N., Earnesty, M., Was, J., Myers, J., et al. (2014), Concept Exploration of Amphibious Combat Vehicle, United States Marine Corps, Washington, DC.

Doerry, N., and Koenig, P. (2019), “Implementing Set-Based Design in DOD Acquisitions,” *The 16th Annual Acquisition Research Symposium*, The Naval Postgraduate School, Monterey, CA, 8-9 May, 2019.

Garner, M., Doerry, N., MacKenna, A., Pearce, F., Bassler, C., et al. (2015), Concept Exploration Methods for the Small Surface Combatant, United States Navy, Washington, DC.

McKenney, T. and Singer D. (2014), "Set-Based Design: A concurrent engineering approach with particular application to complex marine products", *Marine Technology*, July 2014, pp. 51-55.

Mebane, W.L., Carlson, C.M., Dowd, C., Singer, D.J., and Buckley, M.E. (2011), "Set-Based Design and the Ship to Shore Connector", *Naval Engineer's Journal*, Vol. 123 No. 3, pp. 79-92. https://doi.org/10.1111/j.1559-3584.2011.00332.x

Morgan, J.M., and Liker, J.K. (2006), “The Toyota Product Development System: Integrating People, Process, and Technology", Productivity Press.

Parrish, Kristen, John-Michael Wong, Iris D. Tommelein, and Bozidar Stojadinovic (2007), “Exploration of Set-Based Design for Reinforced Concrete Structures,” *Proceedings IGLC-15*, Michigan, July 2007.

Raudberget, D.D. (2010), “The Decision Process in Set-Based Concurrent Engineering – an Industrial Case Study,” *Proceedings of DESIGN* 2010, Dubrovnik, Croatia.

Simon, Herbert A. (1996), *The Sciences of the Artificial*, 3rd ed. Cambridge, MA: MIT Press.

Singer, David J., Norbert Doerry, and Michael E. Buckley (2009), “What Is Set-Based Design?” *Naval Engineers Journal*, Vol. 121 No. 4 (October 11): pp. 31–43. https://doi.org/10.1111/j.1559-3584.2009.00226.x

Sobek, D.K., Ward, A.C., and Liker, J.K. (1999), “Toyota’s Principles of Set-Based Engineering,” *Sloan Management Review*, Vol. 40, pp. 67-83.

Society of Naval Architects and Marine Engineers (2017), T&R Bulletin 7-12: Set-Based Design, Society of Naval Architects and Marine Engineers, Alexandria, VA.

Toche, B., Pellerin, R., and Fortin, C. (2020), “Set-based design: a review and new directions”, *Design Science*, vol. 6, pp. 1-41. https://doi.org/10.1017/dsj.2020.16

Ward, A.C. (1989), *A Theory of Quantitative Inference for Artifact Sets, Applied to a Mechanical Design Compiler [Ph.D. Dissertation]*, Massachusetts Institute of Technology.

Ward, A.C., and Seering, W.P. (1989), “Quantitative Inference in a Mechanical Design ‘Compiler,’” *ASME International Design Engineering Technical Conferences*, Detroit MI, pp. 89-97.

Ward, A.C., Liker, J.K., Cristiano, J.J., and Sobek II, D.K. (1995), “The Second Toyota Paradox: How Delaying Decisions Can Make Better Cars Faster,” *Sloan Management Review*, Vol. 36, pp. 43-61.

https://doi.org/10.1017/pds.2022.253 Published online by Cambridge University Press