*Proceedings of 15th International Marine Design Conference (IMDC-2024)*
*June 2-6, 2024*
*Amsterdam, the Netherlands*

# New Conventions: Intentional Implementation of Set-Based Design Leveraging Point-Based Approaches

**Jonathan E Page[1,*], Warren P Seering[2], Christopher J Higgins[3], and Drake M Platenberg[3]**

***ABSTRACT***

*This article communicates a practical application of Set-Based Design (SBD) during its adoption within the U.S. Navy's latest surface ship design processes. It highlights how the team harnessed SBD to explore the vast design space and make critical decisions. Intriguingly, the article explores the design team's use of point-based designs (PBD), a counterintuitive approach within the context of SBD. It delves into the rationale behind their implementation and the diverse purposes they serve in bridging knowledge gaps such as prototyping, integration, and evaluation of set interfaces. Furthermore, the article presents a preliminary ontology outlining various use cases for these PBDs. A key focus of this article is understanding how these PBDs coexist within the larger set-based construct while recognizing their inherent differences. The article examines the utility and boundaries of this approach, shedding light on the pragmatic interplay.*

## KEY WORDS

Set-based design; point design; risk management; ship design

## INTRODUCTION

This article shares some observations made through action research while designing the U.S. Navy's next class of destroyers, currently designated DDG(X). The article first reviews some principles of point- and set-based design methods. These reviews intend to ground the reader in the methods and lead to a comparison of the two. Next, the article introduces the design effort for DDG(X), including excerpts of the fundamental philosophical, organizational, and methodological principles. Finally, the article reveals the authors' use of point-based designs and methods within their set-based design construct. It instigates an important observation that the two methods are not necessarily mutually exclusive and can be complimentary or necessary.

## POINT-BASED DESIGN AND THE TRADITIONAL DESIGN SPIRAL

Gale (2003) documented the known point-based design (PBD) methods for ships that had been developed and refined over decades. Figure 1 presents a typical formulation of that process. Despite the appearance of a spiral, the process is linear. Each discipline performs its tasks in succession and passes its products to the subsequent discipline. Each loop through the spiral includes changes invoked by previous stages and adds fidelity to the design in their domain. This process iterates through these steps and spirals until the design converges to the final solution.

Ship design methods evolved to this over time because of the complexity of the undertaking; there is no simple or finite set of equations that define a ship. Instead, each discipline involved in ship design has unique equations and relationships dependent on the other disciplines. This method may suffice with a competent and practiced workforce because their initial educated guesses for the ship's parameters are well-informed. That situation dictates minimal rework and refinement as the design converges. Therefore, as the name suggests, this method creates and analyses one point in a basically infinite plane of possible ship designs. If any parameter changes substantially (an ill-defined modifier), a new point in that design space must be explored because it drives a different balance through the various parameters of the various disciplines.

[1] University of Michigan (Naval Architecture and Marine Engineering, Ann Arbor, United States); ORCID: 0000-0001-7779-4169

[2] Massachusetts Institute of Technology (Mechanical Engineering, Cambridge, United States); ORCID: 0000-0001-8846-4851

[3] United States Navy (Naval Sea Systems Command, Washington, DC, United States)
* Corresponding Author: jonpage@umich.edu

Submitted: 3 March 2024, Revised: 1 May 2024, Accepted: 3 May 2024, Published: 23 May 2024

Conference paper, DOI: https://doi.org/10.59490/imdc.2024.879 e-ISSN: 3050-486

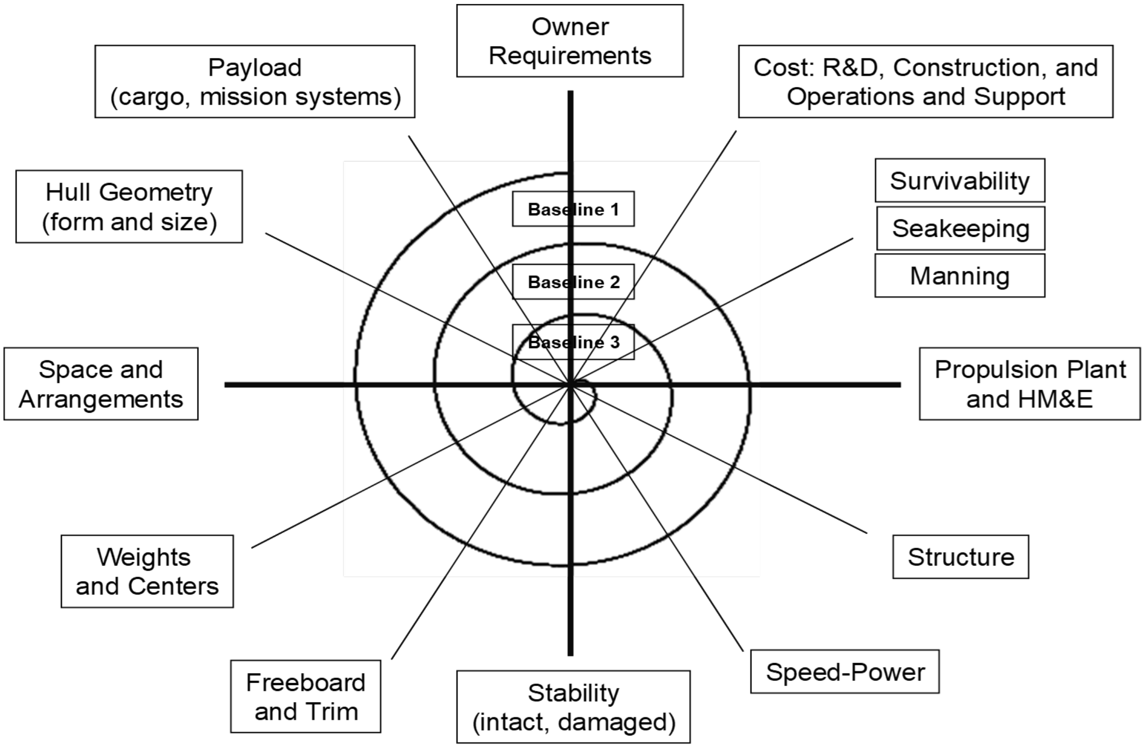


**Figure 1: The Design Spiral in Point-Based Design** *(Gale, 2003)*

A point-based method may be appropriate when the requirements present a well-understood design problem with known technologies. In this case, the characteristics and values within the disciplines can be easily determined or derived from a parent design with similar requirements. Even if one or two parameters change, for instance, a new propulsion engine to meet emissions standards and a new mission system topside, the boundaries of the new design space may only require minimal expansion from existing knowledge. Once again, point-based methods can be used effectively in these conditions. Similarly, suppose the requirements call for optimization within the design along one or two parameters, such as speed and survivability. In that case, point-based methods may be appropriate since other disciplines will acquiesce to optimizing those parameters.

Outside those example conditions, point-based design methods can present risk. The highest risk of PBD is forfeiting the globally optimal design. Because the method is generally a refinement of initial rough parameters, many more designs and design parameters are -*not*- considered compared to the single point that -*is*- considered, refined, and analyzed. Another risk is that of rework. Suppose the initial estimates of parameters are not accurate enough. In that case, subsequent efforts through the spirals rework those parameters instead of refining them because there may be initial divergence instead of convergence of the parameters into a balanced design. The risk of significant rework is exceptionally high if the requirements change, as is often the case for naval vessels.

## SET-BASED DESIGN AND ITS TENETS

On the contrary, set-based design (SBD) is a method that considers sets of variables in the design space instead of a single point. The method considers more than one viable option for variables of interest and, depending on the design requirements, potentially for most variables. Ideally, the options evaluated are also diverse (Doerry, 2015), allowing the opportunity for refinement and avoiding analysis that is similar across variables and potentially duplicative and inefficient. Ward and Seering (1993) introduced the engineering design community to SBD and started formalizing this teachable doctrine and methodology in addressing design needs (Simon, 2019). Since that time, several publications have extended its tenets from those origins of detailed design to concept development, preliminary design, and contract design activities across a variety of platforms and industries, most notably with its alignment to Toyota's design processes (Burrow et al., 2014; Garner et al., 2015; Raudberget, 2010; Sobek II et al., 1999). During those studies, SBD was also called set-based concurrent engineering (Sobek, 1996). The recent publication by Toche et al. (2020) provides an extensive review of research on SBD. It concludes that few complete examples of SBD exist regarding large complex systems like naval vessels. Dullen et al. (2021) also provide a review of relevant research, especially relevant to quantitative methods within an SBD construct. Shallcross et al. (2020) assess the state-of-the-practice of SBD. Each article presents that the SBD method incorporates distinguishable tenets, principles, and practices (D. Singer et al., 2017; D. J. Singer et al., 2009).

First, SBD has philosophical tenets worth understanding. The name derives from the primary tenet: consider a range or group of alternatives – a set. A set for a variable or solution should consist of all viable and diverse options. Further, none of these options should be eliminated from further consideration unless there is a logical reason for doing so. That logic ought to be robust such that no new information will likely reverse the elimination of the option(s). That tenet discloses a tangential philosophical point; the method operates by removing infeasible and dominated solutions from the set instead of selecting one that seems dominant or preferred. Therefore, one arrives at decisions by systematically removing the "worst" answer(s) instead of trying to select the "best" or "right" one. The sets exist at every perceivable level of abstraction in the design space, from fully synthesized and converged artifacts to subcomponents several levels down in the design structure. This abstraction range means a design team can create non-traditional sets for trade-space exploration and decisions, such as sets encompassing entire power and propulsion architectures. Such approaches allow comparison and analysis at those abstraction levels instead of component or sub-component levels, a necessary condition for enabling decisions from the limited data and knowledge that is available in the earlier stages of a complex design process. An important implication of this method is that the design manager decides to keep or remove portions of the design space based on data instead of intuition. Therefore, this technique forces important and recurring conversations regarding what type and amount of data is sufficient to support any given decision. Design team leadership, in conjunction with the team, continues to refine the measures of effectiveness and measures of performance that constitute "adequacy." Another tenet of the method is that these decisions are made at the last responsible moment, often leading to delayed decisions and a process in which a design manager keeps more of the trade space open longer. This is difficult for engineers, leadership, and program managers alike since, under the historically used measures of team performance, it may appear the team is not making progress on the design. One last tenet is improved communication because concurrent engineering approaches necessitate this.

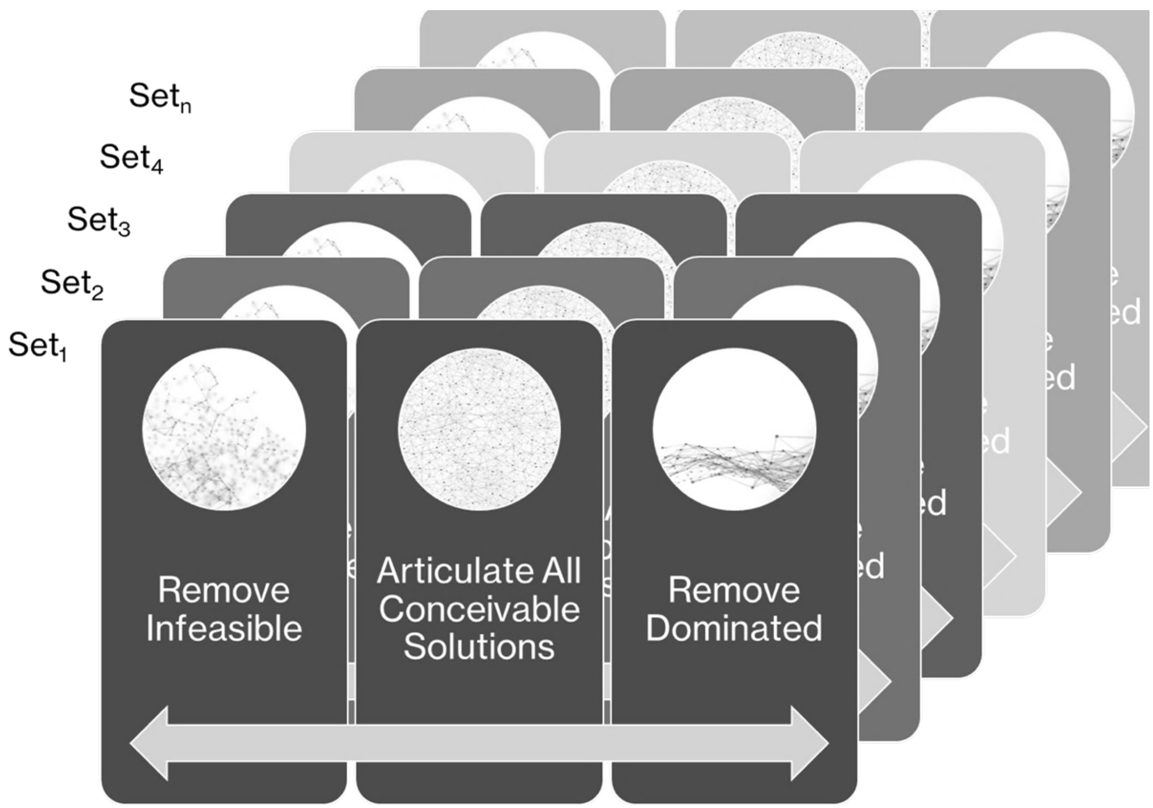


**Figure 2: Conceptual Structure of SBD Method** *(Page & Seering, 2023)*

Figure 2 shares a new conceptual structure of SBD that utilizes and moves away from the usual cited construct from Bernstein (1998). The figure first relates that several sets are in consideration at once, labeled $\text{Set}_1$ through $\text{Set}_n$. The "points" within each set represent the different options within that Set space, thus the Figure labels the middle portion in each Set with "Articulate All Conceivable Solutions." The other portions of a Set space represent analyses that remove portions of the fully articulated Set due to either infeasibility or dominance. The different approaches to removing portions of the fully articulated trade space are neither linear nor sequential. The removals result from analyses that close knowledge gaps in the space.

Additionally, each Set represented may have intersections to the other Sets along shared dimensions. For instance, each point in $Set_1$ could have a connection with one or several points in $Set_2$ and one or several points in $Set_n$. This creates an intricate tapestry of connections between solutions in each space with each other, invoking concepts related to Hilbert spaces to bring multi-dimensional relationships into lower-dimension arithmetic.

Second, Sobek et al. (1999) suggest several fundamental principles of SBD, which have withstood significant modification (Ghosh & Seering, 2014). First, they discuss mapping the design space, which includes defining feasible regions for the parameters under evaluation, exploring trade-offs within that region using multiple alternatives, and communicating those sets of possibilities. Their next principle suggests that the design combines and integrates the feasible and non-dominated regions of the various parameters by intersection – a difficult principle with the tightly coupled and complex design of a ship where thousands of parameters exist to intersect with each other in numerous primary and secondary ways. This principle expects imposing minimum constraints for these intersections to avoid unnecessarily deciding on parameters before the time is right and potentially invoking rework. This principle also seeks conceptual robustness, meaning designs will work regardless of the various parameters' outcomes. Invoking conceptual robustness in design decisions requires a process that supports the knowledge structures of the decisions in a way that enables this. The last principle is establishing feasibility before commitment. This principle starts with gradually narrowing sets while increasing detail. Doing so saves the computationally expensive analyses for the smallest number of solutions. It eliminates more significant swaths of the design space using first-principle and computationally inexpensive analyses at the beginning of the process. This principle also means that once the design manager decides to remove a portion of the design space, the team must stay within the boundaries of the new design space once committed. The team addresses uncertainty around these commitments at gate reviews with appropriate criteria, data, and experience communicated.

Third, following these tenets and principles reveals practices common to SBD (Ghosh & Seering, 2014). One practice is that SBD emphasizes frequent, low-fidelity prototyping to gain new knowledge on the design space and its integration. While Toyota executed this practice physically with clay models and complete prototypes, the practice does not preclude virtual prototyping and invites an intersection between SBD and model-based systems engineering. Another practice with SBD is under-defined system specifications; the method is tolerant to this condition. If the design manager deems one characteristic infeasible, then every design that uses that characteristic is also infeasible. The SBD method instills better communication practices among subsystem teams when executed properly. Another critical SBD practice emphasizes documenting lessons learned and new knowledge. This documentation allows the design teams to reapply the knowledge in other projects and understand decisions for the life cycle of the artifact. SBD enables decentralized leadership and distributed teams through concurrent engineering and improved communication practices. SBD also allows suppliers and subsystems to explore their versions of optimality within the communicated boundaries instead of attempting optimality at the system-of-systems level for a complex product like a naval vessel. Lastly, the SBD method promotes flow-up knowledge creation from the various disciplines' engineers and creators involved instead of flow-down decisions from design management.

Altogether, the tenets, principles, and practices provide a view of SBD and how the method manages complexity, uncertainty, communication, decision-making, and design space exploration. SBD presents a natural step forward in managing these aspects of a complex system design, but it may only sometimes be appropriate. Therefore, the two methods deserve a comparison and contrast.

## COMPARISON

The preceding sections revealed aspects of PBD and SBD that allow one to compare and contrast the methods. Both are methods developed to manage the complexity of a tightly coupled design, in this case, that of a naval vessel. Both start with relatively little information and low-fidelity analyses for the first decisions. The similarities quickly end, however. Singer et al. (2009) provide a good accounting of the differences between the two methods, including the points made here and derived from Bernstein (1998).

In the face of little information and low fidelity, a PBD method selects one point in the design space to iterate around and refine with increasing levels of fidelity. In contrast, SBD uses the information to decide what areas of the design space to stop exploring but keeps many options open for longer. This difference generates a key divergence in the performance of the methods. PBD commits to cost with selection decisions early in the process, with less available information and fewer completed analyses, thus making rework more likely and expensive. Put another way, SBD encourages management to make the right decision the first time by delaying until the appropriate information is available and shifting their influence and cost obligations later in the process.

Another point of comparison between the two methods considers the number of variables involved. With PBD, only a few design space parameters should be independent variables. The method still analyzes, refines, and adjusts all necessary parameters. However, most are dependent variables adjusted due to decisions made and optimizations elsewhere in the design space. This condition applies at both system and subsystem levels. The dependent parameters are known, specific technologies that are well understood and only require refinement instead of exploration. Alternatively, SBD encourages variations and independent variables to explore a broader solution set across more parameters. Both methods recognize tightly coupled parameters, but PBD controls variation through early decisions, while SBD encourages variation and global optima exploration across several parameters. Therefore, SBD is better suited to design challenges where requirements conflict or the technology or design challenge still needs to be better understood.

Additionally, the two methods differ based on idea generation. PBD methods generate ideas from the tacit knowledge of the individuals involved, typically derived from previous work experience. To reach a solution, the design team iterates around this point solution. Suppose the solution does not and cannot converge. In that case, the team may invoke brainstorming techniques to determine the rework necessary in the design, whether a new technology solution or changing a previously dependent variable to an independent one. Alternatively, SBD defines a feasible space with boundary conditions and systematically reduces it by removing infeasible or dominated regions. Thus, the SBD method promotes ideation up front so that the team considers more solutions and possibilities from the outset.

The two methods also take alternate approaches to optimization. PBD uses known optimization techniques with fitness functions and preferences and applies them when suitable. They are applied against a single point to refine and optimize given parameters, which will likely sacrifice the performance of the other parameters. On the other hand, SBD creates alternatives and tests multiple parameters in parallel, eliminating infeasible and dominated solutions from further consideration. This approach considers a more global consideration to optimization, striking a different balance in the design space with the interaction of the coupled parameters. To enable this, SBD uses minimum control specifications and constraints to allow the intersections of parameters to create this global optimum and mutual adjustment. PBD, alternatively, imposes maximum constraint early in the process, attempting to ensure functionality and interface compatibility. The constraints in PBD also lead to the behavior that invokes margins and allowances around specific parameters and constraints to account for uncertainty and give design teams "budgets" to stay within. These margins developed empirically over time and are potentially fraught with the same risks as other parameters regarding the educated estimates that create these values early in the process. Alternately, SBD allows the knowledge to develop with time, reducing the uncertainty as much as possible before committing to a design decision.

Within those constraints and uncertainty, a team using PBD methods integrates by using synthesis methods, typically conducted using software codes. If a particular discipline cannot stay within its budgets and constraints, leadership must reallocate or reapportion the risk and budget to another domain. At the end, a prototype may be built for testing and validation purposes, including virtually. However, SBD integrates by intersection. Suppose two elements of two sets do not intersect along a common parameter. In that case, there is clear data to remove those portions of the design space from further consideration, validating the minimal-constraint approach: the design space naturally culls itself as more information becomes available without transferring risk throughout the team. SBD encourages prototyping along the way to generate knowledge that can flow up to higher levels of abstractions and sets in the system of systems. These prototypes tend to invoke lower-cost tests that prove the infeasibility or dominance of a select portion of the design space versus the converged artifact as a whole.

Lastly, the two methods differ regarding risk management. SBD engenders a decentralized risk management approach. Each discipline establishes feasibility before commitment. All disciplines pursue their options in parallel. They have a charter to find solutions robust to physical, logical, market, and design variations. Once the design manager commits to a choice, the team stays within the sets and manages uncertainty at least as often as process gates. PBD requires centrally managed feedback channels with high frequency. The disciplines must respond quickly to changes due to analysis in other disciplines or decisions by management. Table 1 provides a summary of these comparison points. Singer et al. (2009) first developed a similar table.

**Table 1: Summary Comparison of PBD and SBD**

| Characteristic | Point-Based Design | Set-Based Design |
|---|---|---|
| Variables | Very few independent variables, many dependent variables | Many with tight coupling. Most are treated as independent |
| Requirements Stability | Known, stable, thoroughly documented | Uncertain; flexibility and trades are required or desired |
| Initial Knowledge | Well-understood technologies and design challenges; less uncertainty | Many knowledge gaps regarding technology, integration, and the design challenge |
| Finding Solutions | Iterate an existing design based on previous work or educated estimates, modifying it to achieve objectives and improve performance. If necessary, brainstorm new ideas. | Define a feasible design space, then reduce it by removing regions where solutions are proven infeasible or dominated. |
| Communication | Frequent, latest iteration of the "best" idea | Less frequent; sets of possibilities; broadcast knowledge for the team vice the next step |
| Integration | Synthesis and convergence through iteration. Disciplines given budgets and constraints, which can be reallocated to other disciplines | Intersections of sets along matching parameters |
| Selection | Select the "best one" based on a developed decision scheme. Prototype at the end to confirm the working solution. | Concurrent engineering. Eliminate inferior choices. Conduct low-cost tests along the way to generate flow-up knowledge that supports choices. |
| Optimization | Typical methods with fitness functions and preferences are invoked at any level of abstraction in the design. Typically, one or two parameters dominate the function. | Alternatives and disciplines design in parallel. Infeasibility and intersections drive global optimization. |
| Constraints | Maximize constraints in specifications to ensure functionality and interface fit. | Use minimum control specifications to allow optimization and mutual adjustment. Design decisions create constraints and requirements for subsequent analyses. |
| Risk | Centralized and managed with budgets and constraints. | Decentralized, everyone establishes feasibility before commitment, seeks robust solutions, and honors commitments and choices. |

## DDG(X) SBD METHOD AND SCALE

In 2018, the U.S. Navy took action to begin the development of a new class of combatant ships to address uncertainty in future needs and accommodate potential future missions above and beyond those of the current DDG-51 and CG-47 classes (O'Rourke, 2023). A Requirements Evaluation Team analyzed suitable initial parameters. Then, it delivered them to the inaugural design team led by Higgins and Page as the Senior Ship Design Manager (SSDM) and Deputy Ship Design Manager (DSDM), respectively. Their initial charter was to further explore cost-capability trades and concept designs to inform the Capability Development Document. We decided that a set-based method was appropriate to manage the complexity and complicatedness of this undertaking in line with previous Navy efforts (Burrow et al., 2014; Garner et al., 2015). Excerpts from Page (2022) and Page et al. (2022) follow to provide the context for the organization, philosophy, and methodology created.

### Organization

Organizationally, the team used a hub-and-spoke model. This structure balanced the distributed nature of the U.S. Navy's ship design organizations and our initial chosen sets. The SSDM and DSDM designated seven Functional Area Leads for systems engineering, naval architecture, marine engineering, warfare systems, survivability, flexibility, and affordability, as seen in Figure 3. Those leads were responsible for designated sets in their domains. Leadership planned to co-locate the team at a design site for high-frequency communications and collaboration, but resources prevented that initially, and then COVID-19 precluded co-location. So, the set teams operated from their designated work sites in Pennsylvania, Maryland, Virginia, and Washington, DC, with little opportunity for face-to-face interaction. This lack of co-location presented an unanticipated opportunity to demonstrate the concurrent engineering aspects of Set-Based Design.

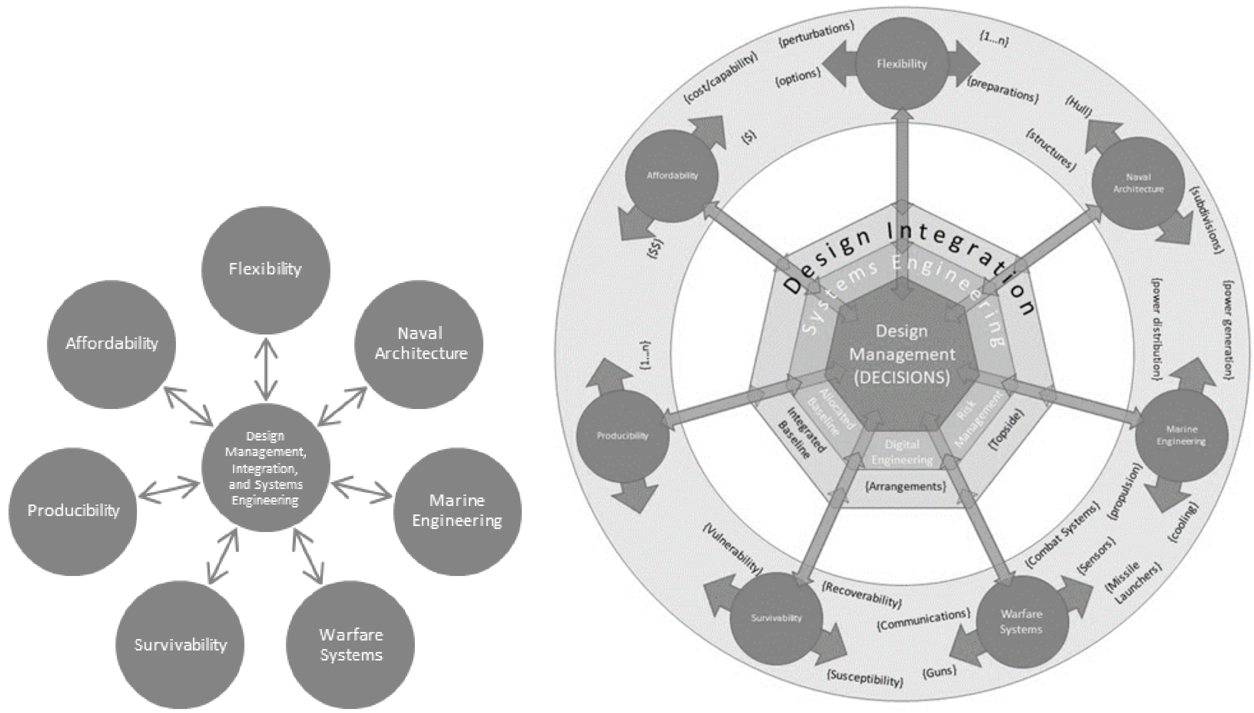


**Figure 3: Initial Design Team Organization and Scaled Design Team Organization**

With time, the team expanded, explored more sets, and the organization and process scaled. Figure 3 also reveals the scaled structure of the organization, which retains much of the foundational framework. The hub expanded and discretized the systems engineering and design integration functions distinct from design management. The spokes of the framework are now connected, forming a wheel representing the communications and negotiations delegated to the domains from the hub.

## Philosophy

The SSDM and DSDM created a design philosophy and promulgated it through a series of Design Guidance Memoranda (DGM). There were initially seven, but that has expanded to include many more. The topics include meeting minutes, routing processes, digital engineering strategy, testing and test objectives, margin management, team training, critical systems, and, of course, how to implement the set-based method for DDG(X). Fundamentally, the philosophy was to manage the risk 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. The SSDM and DSDM perceived risk with knowledge capture, configuration control, an overly constrained design space, design rework, team communication, and a method unfamiliar to many within the Navy organization. These DGMs addressed those risks and others by instituting practices and providing foundational knowledge and desired behaviors.

The second DGM in the series was especially important because it communicated the SBD philosophy and process to the team. It promulgated the policy that every team member was empowered to retain a valuable portion of their trade space, and only the SSDM (or his delegate) could remove a portion of the design space from further consideration. It also included a 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. The assumptions served as placeholders for data not yet known or available to the teams, and they had a potential to affect the boundaries and constraints of the analysis. It established the guiding principle to explore the various domains and sets using appropriate methods. It dictated that sets ought to be no bigger than necessary to analyze an appropriate level of abstraction to close a knowledge gap. The SSDM and DSDM also 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 close a knowledge gap or extended studies that did not reach conclusions on time. The DGM also included a glossary of the new terms representing this new way of thinking. It established the concept of a Set Review with the desired outcome of the Reviews being these decisions and approvals.

## Method

The next step in executing SBD with the DDG(X) design team was to meet with the functional area leads depicted in Figure 3 and their set team leads to discuss how to apply this method in their set. These meetings served several purposes, including training, calibration, brainstorming, and just plain storming (from the ubiquitous form-storm-norm-perform model of team dynamics). The group would discuss the first knowledge gaps to address in the design germane to their domain and move forward with agreement on the proper analysis to resolve that knowledge gap, the assumptions necessary to accomplish that analysis, and the metrics and criteria appropriate for the analysis. Reaching mutual understanding often required several meetings with leadership and set teams.

These meetings served as the basis for what became Set Reviews. The Set Review was a structured meeting that provided consistency to the presentation of information to the SSDM for decisions. The Review served other purposes, too. It reinforced training and developed the team's familiarity with our SBD process. Eventually, it was the forum where the design team discussed integration and the intersection of overlapping parameters across domains. For each Set Review, team leadership expects a set team to review eight fundamental elements. The DGM on SBD directed these elements to the team and provided a case example for them to follow.

Each Set Review started with the <u>purpose</u> of the study (i.e., which knowledge gap the team is addressing) and of the Set Review (e.g., interim review, approve assumptions, decision review, set reduction). Next, the Set Review shared the team's <u>approach</u> to closing the knowledge gap. Next, the Set Team briefed the <u>assumptions</u> necessary to use that approach. Subsequently, they communicated the <u>criteria</u> for developing their findings, conclusions, decisions, or recommendations. Then, with that background in place, the team could share the <u>analysis and results</u> followed by the <u>outcomes</u> (e.g., findings, conclusions, recommendations, and decisions to remove portions of the design space). After that, the team shared any <u>uncertainty</u> about the outcomes that may influence the outcomes or future analysis. Finally, the Set Review ended with the anticipated <u>next steps</u>, including further analysis within the current set, recommendations for the following knowledge gap or set to explore, or others.

To document the useable, and hopefully re-useable, knowledge, a successful Set Review culminated in a Design Decision Memorandum (DDM). These DDMs are critically important because they record the team's progress, capture the team's work, and record the outcomes of the Set Reviews for posterity. They were widely available to the team and the community of stakeholders surrounding this effort, including resource sponsors, requirement generators, subject matter experts outside the design team, and acquisition professionals. Of course, not every Set Review generated a DDM as the team instantiated this SBD process: through the first year, there were 66 Set Reviews and 26 DDMS (39%); through the second year, there were 139 Set Reviews and 75 DDMs (54%), and through the third year there were 156 Set Reviews and 86 DDMs (55%). In addition to the increase in successful outcomes with time, another important metric related to the process was a decrease in the time to achieve the first DDM, lowering from about 49 days in the first year to about eight days by the third year. In other words, when the team first presented a new set or analysis, it used to take almost two months of rework and adjustments to reach a satisfactory level of information for a successful outcome. However, by the third year, the Set Reviews were more productive as the team learned from each other and leadership the standards necessary for a successful Set Review.

## Risk and the First Sets

While the DGMs addressed many risks and set policies in place that answered many questions of the growing team, there was one other philosophical question to answer that did not necessarily belong in a DGM: Where do we start? In other words, what are the first sets? Do we start with every possibility in every set? The answer to this last question is emphatically "no." A naval vessel is sufficiently complex that doing so is impossible. The answer to the other questions, generalized to any naval vessel, is that it depends. Generally, the requirements will drive the answer. An aircraft carrier will have a different start than an amphibious vessel, which will have a different start than a submarine, which will have a different start than a combatant. Also, generally speaking, the answer is outside the Work Breakdown structure of the vessel. Instead, *knowledge gaps* are a vital concept that helps determine the first sets and subsequent sets. The SSDM and DSDM used the requirements, known risks, critical path analysis, external sources, and our judgment and experience with previous projects to derive knowledge gaps and thus the first sets.

One example of requirements dictating the first sets derives from the combat systems domain. The requirement stated that the ship would use the latest variant of the AEGIS combat system but incorporate a service life allowance to change and expand in the future (LaGrone, 2022). One might think there was little work left and few knowledge gaps; however, although that requirement defined some of the material solutions, the team still needed to determine the placement of the sensors in the suite and define what service life allowance for the system in terms of computing power, space and volume, weight, arrangements, electrical power, and cooling. Of course, the initial placement of sensors depends on superstructure geometry, so this drove another of the initial set explorations due to the tight coupling between these parameters, which were equally undefined at the start. Therefore, warfare systems and combat systems were one of the first sets, and teams formed in this domain to explore these knowledge gaps.

An example of the critical path dictating one of the first sets is with the hull. Despite the fidelity of modeling tools available for hydrostatics and hydrodynamics of a vessel, the U.S. Navy still conducts model tests of hull forms at the Naval Surface Warfare Center, Carderock Division, to explore performance ranges, reduce risk, and confirm the expected performance under various conditions. This test requires building a properly scaled model (which takes time) based on the hull form's final dimensions, which requires analysis of many other coupled parameters throughout the rest of the ship design (and, therefore, also takes time). The testing takes time, as does analyzing the data and many other activities in this process. Knowing this, the hull form definition and testing was a critical path item and, therefore, one of the first sets explored.

Judgment and experience with previous projects led the SSDM and DSDM to include boat handling (the launch and recovery of the ship's boats) as one of the first sets. One may think this discipline should not entertain the same regard as the hull form, combat system placements, or power and propulsion architecture. However, in this case, experience from a previous ship class where boat handling systems had proven difficult and caused significant and expensive rework informed this process of the breadth of potential solution space and the varying degrees of impact on hull form and other major systems, revealing the potential risk that warranted early consideration of the boat handling subsystem. After all, the team had not yet ruled out any parameters with the hull form, arrangements, or other domains that may have removed the risky solution. Further, SBD allowed exploration in this domain for solutions that may have precluded the issues that developed in the previous class.

The SSDM and DSDM chose eleven initial sets and their constituent knowledge gaps. They were:

1. **Hull form** began by exploring three basic shaping choices that affected **survivability**. **Hull form** was also on the critical path because of model testing.
2. **Survivability** began with certain **susceptibility** studies and the gathering of 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 risky area with the potential to be on the critical path due to land-based testing.
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 tracked 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 deckhouse size, shape, and location measures.
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. This was the first time anyone had 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**.
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 to validate and integrate other **sets**.

This final set, the **ship** set, is particularly interesting because it is a set of synthesized, converged, and balanced designs. In other words, it is a set full of point designs.

## DDG(X) USE OF POINTS WITHIN THE SBD CONSTRUCT

The use of PBD within SBD may seem counterintuitive. After all, the summary comparison of the two methods presents a case in which the two have few similarities and many differences. That comparison and this article - so far - imply that the two methods are mutually exclusive. This section aims to dispel that thinking. When one carefully crafts the interactions of the two methods, they are complementary. Integrating the two methods requires mindful development of the point designs to fulfill the closure of particular knowledge gaps in a given set in a calculated way, similar to any other analysis a set team may conduct.

The team's discovery of this interaction was both reluctant and serendipitous. It emerged as a discovery due to the final two listed sets of our initial effort: **affordability** and **ships**. Early in the timeline of the design effort, the team's charter was to explore cost-capability trades, which initiated the **affordability** set. In line with the process, discussions ensued to determine the proper analysis, metrics, criteria, assumptions, and approach to resolve the knowledge gaps and inform requirements regarding affordability. The team quickly realized that affordability, like many other emergent properties of a vessel, is a system-level attribute that is difficult to allocate cleanly to subsystems, but the team needed to evaluate cost and affordability at both discrete and integrated levels of abstraction. Many affordability attributes are tightly coupled with other design aspects, often non-linearly, such as speed, engines, and hull form that have cubic behaviors with resistance and discrete increases in cost due to engine sizing. This abundance of attributes certainly invoked SBD but created the conundrum of understanding affordability at the system (vessel) level to include the secondary and tertiary effects to cost that changing certain parameters and analyzing them may have. At this point, when trying to stand up an SBD method, creating point designs to understand cost and affordability seemed paradoxical. Reluctantly, the team moved forward with this approach, understanding that these points would be more tightly controlled and require faster creation than in previous efforts. The emphasis of the points was on realistic representation of the open design space across all domains while isolating a particular set or characteristic of interest.

Before this effort, including in the Requirements Evaluation Team, the Navy created point-based designs with a few exceptions (Burrow et al., 2014; Garner et al., 2015; Mebane et al., 2011). The Navy has a tool that synthesizes the various parameters of a vessel's design into a feasible solution or tells the user why the vessel will not synthesize. They also have a companion tool to change parameters within boundaries and constraints to produce many point designs rapidly and allow comparisons. It can produce 1,000-2,000 point designs per hour if run on high-power computers. This toolset enables a design space exploration based on boundaries at the moment in time coupled with the programmed code, but does not necessarily create new knowledge. The Requirements Evaluation Team generated 25,000 points during its exploration similar to the **ship** set the affordability team created, but none of those points were used in subsequent design phases with the DDG(X) team.

The DDG(X) effort did not require thousands of designs, though; it required carefully curated point designs that could be compared in all-else-equal techniques to generate knowledge that informed affordability measures. Therefore, each analysis required a unique study guide, a document usually a dozen pages or less that defined benchmark vessels, excursions from that benchmark, and what parameters were allowed to vary from one to the next. The combination of the benchmark vessel and the parameters varied as part of the study created the excursion that generated new knowledge of the design within the boundaries and constraints of the excursion. These study guides listed the approved assumptions and boundary conditions for each point design in the analysis. Collectively, all of these point designs constitute the **ship** set.

The **ship** set contains a few variations of point designs. One variation is called a benchmark. A benchmark vessel represents a synthesized vessel that incorporates decisions and outcomes from Set Reviews up to a certain point. Put another way; a benchmark incorporates the knowledge created and decisions made into a representative point design. Due to this approach, only known and approved knowledge structures (or known and approved assumptions) created new product structures. This limitation helped keep rework loops small and ensured that knowledge built upon itself as the process revealed the design. Depending on the knowledge sought in the open trade space, the process may require several related benchmarks. For instance, when exploring the major power and propulsion architecture, the team created one benchmark with a mechanical architecture, another with a hybrid architecture, and another with an integrated electric architecture so that knowledge creation happened across these architectures until the SDM made a down-select decision. That introduces the final variation of the **ship** set, an excursion. This type of point design allowed specific parameters to vary in controlled ways to help analyze the ship-wide impacts of these changing variables.

Another variation of the approach is that a benchmark vessel represents a synthesized vessel that reflects a portion of the ship set trade space, whereas a few benchmarks are developed and maintained to represent the total ship set trade space. Over time, these benchmarks are updated to incorporate decisions and outcomes from Set Reviews. When decisions and outcomes have not been completed (e.g., warfare system selection), different benchmarks will vary the sets they are designed with to ensure representation of the ship trade space. These benchmarks are then utilized to support the determination of the metrics that can only be determined from a synthesized vessel. When a particular set needs to understand total ship affordability or other metrics for comparisons within the set space being examined (e.g., power generation type – gas turbine or diesel), excursion concepts will be developed from the benchmarks that vary one set (e.g., diesel power generation set in an excursion from a gas turbine benchmark). The differences from the excursion concept to the benchmark represent the impact of the set variation, and that difference becomes the finding for that set's impact across the ship trade space. (e.g., diesel generators are much less power dense but more fuel efficient than gas turbines, increasing ship size to accommodate the diesel generators but reducing ship size on fuel demand; benchmark/excursion comparisons will identify the overall impact on affordability).

A practical example helps to illustrate these points and understand the different variations of point designs. Two of the cost-capability trades requiring more knowledge at the beginning were the top speed of the vessel and the architecture of the power and propulsion plant, specifically mechanical, hybrid, or integrated-electric. Naval engineers ought to recognize that 1) changing the propulsion plant architecture is not trivial, 2) changing speed changes costs in non-linear ways, and 3) the change in cost for the change in speed will depend on the architecture. Therefore, the team created a coupled design study that varied each parameter to close this knowledge gap for the team and our stakeholders. At the time, the team had three benchmarks for each propulsion architecture within a common hull form with a common combat system and topside arrangement. Those benchmarks became the standard from which two excursion ships varied in their top speed (six new point designs total). These benchmarks and excursions are still only representatives of the open trade space: the team selected specific power generation and propulsion components that had not been locked in with design decisions, and the synthesized models included inherent assumptions on how to best trade hydrodynamics against installed power since that decision space was also still open. The results allowed the team and our stakeholders to compare three cost curves across variations in propulsion plant and top speed. The team could determine and report which plants were more affordable at which speeds. They could also reveal which plants were more sensitive to speed changes – an essential measure of uncertainty at this design stage. The team could couple this information with others like risk and technology readiness to help in the decisions regarding speed, cost, architecture, and others moving forward from that scenario. The speed and propulsion architecture decisions were made within the next year, resulting in a new benchmark that incorporated those decisions and others into a new synthesized and balanced reference vessel, finishing this practical example.

These benchmark and excursion point designs serve many vital purposes. First, one use case for these point designs within SBD is as a risk reduction measure. There are scenarios when schedules accelerate or time runs out for a given design phase. The benchmarks provide a balanced and synthesized design that *could* proceed to subsequent phases at any time since the benchmark presents a representative point that should meet all the requirements. The number of knowledge gaps unanswered dictates the uncertainty and risk in moving forward. The benchmarks also validate the requirements by showing one possible solution of many.

A second use case is in performing ship-wide comparisons and analyses when only select variables change. This purpose allows the design team to tease out second and third-order effects through the design and their cost implications. This case is essential in designing a naval vessel because of the tight coupling of the design parameters.

A third use case relates to the second: knowledge generation. When carefully curated and analyzed, the point designs generate knowledge on a ship scale to understand requirements and cost impacts. In the vein of knowledge generation, they also provide a means by which a design team can match their knowledge structure with the product structure, at least within the boundaries of the synthesis code, in this case. Further, the point designs allow a design team to test the boundaries of the design space. DDG(X) terms these tests "diversity of thought" exercises. When leadership or the team wants to challenge a requirement, assumption, or technology solution, the use of point designs in the manner relayed here provides a structure to conduct those tests, generate new knowledge, and potentially stretch the boundaries of the design space.

In essence, these point designs act as virtual prototypes, a purpose SBD encourages. However, while Toyota can afford to (and does) build physical prototypes of cars and subsystems, it makes less sense for the Navy or another ship owner to follow the same practice. Virtual prototyping allows the Navy to "build" a ship and create a digital engineering underpinning for the effort, with digital threads that can carry through into production and sustainment. The value of the prototype is limited in the same way as any model used is: one must understand the assumptions, code, and fidelity of the information involved.

One last case to note is that the point designs help transition SBD from theory to practice. The U.S. Navy had limited exposure to SBD, especially on a large scale for a combatant. Some stakeholders perceived a considerable risk in using SBD on such an important project without more of a track record for the method. These point designs grounded SBD by providing a sense of familiarity. At the risk of speaking for those stakeholders, they appreciated this approach and, for the first time in their careers, saw (literally seeing in the charts and data graphs presented) the effects requirements and assumptions had on cost and the other coupled parameters of the design. This approach was able to visualize a trade space for decision-makers that transparently presented the capabilities, limitations, and assumptions of the data and outcomes. The design team leadership had to carefully remind stakeholders, when appropriate, that these point designs were for knowledge regarding possibilities and causal relationships of parameters and results that lead to decisions but are not necessarily - *the* - solution (notwithstanding the first use case communicated above).

## CONCLUSIONS

While it may seem counterintuitive to use PBD within SBD, the two methods complement each other when constructed in ways that enable their interaction. For complex design challenges like naval vessels, using point design within SBD to generate knowledge and understand causal relationships in the tightly coupled and non-linear trade space benefits the effort. These point designs help reduce risk, act as virtual prototypes, and enable analysis of ship-wide impacts when only a few parameters change. DDG(X) created a preliminary ontology of point designs that suited the team's purpose: benchmarks and excursions. These basic variations facilitated the team's knowledge generation and matched their knowledge structure to the product structure in repeatable and valuable ways. Though the DDG(X) team was reluctant to create point solutions as part of their SBD implementation, it proved a valuable practice worth repeating, with well over two hundred point designs created to explore the space, create knowledge, and reduce risk. The two methods are not mutually exclusive, and this approach can generalize to apply to any complex design undertaking.

## CONTRIBUTION STATEMENT

**Page:** Conceptualization, investigation, methodology, visualization, writing - original draft. **Seering:** Conceptualization, methodology, project administration, supervision, writing – review and editing. **Higgins:** Conceptualization, investigation, methodology, project administration, supervision, writing – review and editing. **Platenberg:** Data curation, formal analysis, investigation, visualization, writing – review and editing.

## REFERENCES


Bernstein, J. I. (Joshua I. (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

Burrow, J., Doerry, N., Earnesty, M., Was, J., Myers, J., Banko, J., McConnell, J., Pepper, J., & Tafolla, T. (2014). Concept Exploration of the Amphibious Combat Vehicle. *Day 1 Wed, October 22, 2014*, D011S001R007. https://doi.org/10.5957/SMC-2014-T19

Doerry, D. N. (2015). Measuring Diversity in Set-Based Design. *Proceedings of ASNE Day*, 14.

Dullen, S., Verma, D., Blackburn, M., & Whitcomb, C. (2021). Survey on set-based design (SBD) quantitative methods. *Systems Engineering*, *24*(5), 269–292. https://doi.org/10.1002/sys.21580

Gale, P. (2003). The Ship Design Process. In T. Lamb & Society of Naval Architects and Marine Engineers (Eds.), *Ship design and construction. Vol. 1* (Vol. 1, pp. 5–1 to 5–38). Society of Naval Architects and Marine Engineers.

Garner, M., Doerry, D. N., MacKenna, A., Pearce, F., Bassler, D. C., Hannapel, D. S., & McCauley, P. (2015). *Concept Exploration Methods for the Small Surface Combatant*. 10.

Ghosh, S., & Seering, W. (2014). Set-Based Thinking in the Engineering Design Community and Beyond. *Volume 7: 2nd Biennial International Conference on Dynamics for Design; 26th International Conference on Design Theory and Methodology*, V007T07A040. https://doi.org/10.1115/DETC2014-35597

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

Mebane, W. L., Carlson, C. M., Dowd, C., Singer, D. J., & Buckley, M. E. (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

O'Rourke, R. (2023). *Navy DDG(X) Next-Generation Destroyer Program: Background and Issues for Congress* (IF-11679; In Focus, p. 3). Congressional Research Service. https://crsreports.congress.gov/product/pdf/IF/IF11679

Page, J. E. (2022). *A Model for Set-Based Design at the System-of-Systems Scale with Approaches for Emergent Properties*. Massachusetts Institute of Technology.

Page, J. E., Higgins, C. J., & Seering, W. P. (2022). Starting and Scaling a Set-Based Design Method for a Maritime System of Systems: Designing a Modern Warship. *Proceedings of the Design Society*, *2*, 2503–2512. https://doi.org/10.1017/pds.2022.253

Page, J., & Seering, W. (2023, August 20). Steps Toward Development of a Comprehensive Set-Based Design Process Model: A Case Study. *IDETC-CIE2023*. https://doi.org/10.1115/DETC2023-115315

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

Shallcross, N., Parnell, G. S., Pohl, E., & Specking, E. (2020). Set-based design: The state-of-practice and research opportunities. *Systems Engineering*, *23*(5), 557–578. https://doi.org/10.1002/sys.21549

Simon, H. A. ; L., John E. (2019). *The sciences of the artificial*. The MIT Press; U-M Articles Search. https://doi.org/10.7551/mitpress/12107.001.0001

Singer, D. J., Doerry, N., & Buckley, M. E. (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, D., Strickland, J., Doerry, N., McKenney, T., & Whitcomb, C. (2017). *SNAME T&R Bulletin 7-12 Set-Based Design*. The Society of Naval Architects and Marine Engineers.

Sobek, D. K. (1996, July). A set-based model of design. *Mechanical Engineering*, *118*(7), 78.

Sobek II, D. K., Ward, A. C., & Liker, J. K. (1999). Toyota's principles of set-based concurrent engineering. *Sloan Management Review*, *40*(2), 67–83. bth.

Toche, B., Pellerin, R., & Fortin, C. (2020). Set-based design: A review and new directions. *Design Science*, *6*, e18. https://doi.org/10.1017/dsj.2020.16

Ward, A., & Seering, W. (1993). Quantitative Inference in a Mechanical Design 'Compiler.' *Journal of Mechanical Design (1990)*, *115*(1). U-M Articles Search. https://doi.org/10.1115/1.2919320