Michael E. Buckley and Dr. David J. Singer, University of Michigan, Dept of Naval Architecture & Marine Engineering

Capturing Decision Making Needs for Design Affordability and Flexibility: Bridging the SE-HSI Gap

ABSTRACT

In the design of complex systems under conventional practice, an effective embrace of system and system users that drives requirements from the top down is difficult and rare. In fact, perhaps the single greatest technical challenge in engineering complex systems is that of building systems around the people who will be using them. For naval vessels, the challenge is to build the ships around the sailors who will be sailing them 30-50 years after initial commissioning. Understanding the needs of the sailors over the ship's entire service life is one of most important concerns in budgeting for total life cycle costs. In this effort, Systems Engineering and Human Systems Integration (HSI) methods and tools are often applied separately, with many HSI concerns addressed after the fact instead of when it would most benefit design, contributing to operability issues and complicating initiatives for more affordable ships.

Like many others, this paper affirms that an integrated Systems Engineering-HSI approach is fundamental to addressing these challenges, but it also recommends a structured, top down, decision oriented method for achieving this end. The method presented provides a framework for realizing improved traceability from desired capabilities to functional requirements and supports a robust negotiation for flexible and affordable solutions. This paper describes an approach based on understanding the decision landscape of system users/operators as decision makers and founded on the premise that every system can be described as an evolution of knowledge punctuated by decisions supported information and expertise. The paper explains how the method, known as Decision Oriented Systems Engineering (DOSE), bridges the gap from desired capabilities to functional requirements, how the approach closes the Systems Engineering-HSI gap, and in so doing provides several design benefits.

INTRODUCTION

The positive performance impacts of a sound embrace of Human Systems Integration concerns have become generally accepted within the DoD acquisition community. While anecdotal performance benefits have helped to build acceptance, organizational changes to better accommodate HSI needs within the NAVSEA SE organization and the acquisition process signify another level of acceptance. In 2009, the Navy published Secretary of the Navy Instruction (SECNAVINST) 5000.2D. An addendum to Office of the Secretary of Defense's (OSD) HSI Plan, it requires "resource sponsors and program managers (PMs) to initiate an (HSI) effort as early in the acquisition process as possible and address HSI throughout all phases of the acquisition process to optimize total system performance, minimize total ownership costs, and ensure that the system is built to accommodate the characteristics of the user population that will operate, maintain, and support the system."

The perceived cost savings utility of HSI analyses is also gaining impetus. Wallace et al. (2007) advocate an address of HSI concerns very early in the science and technology (S&T) process, citing the "harsh reality" that manpower costs account for over 67% of the Navy's total budget. As one example, Wallace et al. (2007) cite the typical focus on the development of engineering displays during the S&T phase in order to get hard science issues resolved, while ignoring the needed development of operational displays and associated operational architecture. This delay results in increased development and training costs.

Sizeable cost savings result from an early and robust analysis of HSI concerns (Landsburg et al. 2008). These concerns include manpower, personnel, training, safety, occupational health, habitability and quality of life, personnel survivability, and human factors engineering (Malone and Carson 2001, Pew 2008). If sizeable cost savings are possible in this manner it is hardly arguable that more significant positive impacts to affordability are possible with a truly integrated SE-HSI design approach. Along these lines, Malone and Carson (2001) recommend a top-down, systematic approach, known as Top-Down Requirements Analysis (TDRA), for the conduct of such analyses.

TDRA begins with a function analysis step. In this step, functions to be performed are identified, organized and decomposed into functional flow block diagrams that are iteratively developed to build a system model. Additional steps identify high driver functions, define mission scenarios, conduct mission analysis, allocate functions and define roles for humans, etc., and culminate in a manning model that is defined, evaluated under simulation and validated to confirm workload and training requirements (Malone and Carson 2001). This paper proposes a decision modeling approach for the function analysis step, using a method known as Decision Oriented Systems Engineering (DOSE). The DOSE method facilitates construction of a simpler system model to support requirements analysis and allocation; why and how it is simpler is the subject of this paper.

The DOSE method has been used successfully in a variety of venues for more than a decade, from requirements analysis during the early phases of the DDG 1000 Program to very recently, a process design task for the Navy's first implementation of Set Based Design (see Buckley and Peretin 2008, Singer, Doerry, and Buckley 2009, Mebane, et al. 2011). In the early phases of DDG 1000, the method was used to capture and document the decision-making requirements for a large fraction of the ship's planned operational capability (Buckley 2003). In the more recent process design task for the Ship-to-Shore Connector, the method was used to develop a Set Based Design implementation tailored to the program's needs.

It is the goal of this paper to describe DOSE in the context of challenges in current practice in integrating Systems Engineering and HSI methods and describing how a structured, systematic, decision oriented approach to the problem of abstracting system requirements can address some of these difficulties, thereby increasing design affordability and flexibility in the process. Against this backdrop, the DOSE method is summarized and some of the benefits to design affordability and flexibility explained.

The DOSE method involves the construction of a system model built around the decisions the system users need to manage. This decision model of the system is built on decisions represented as decisions made (not the choices), and are captured in a knowledge map punctuated by these decisions and supported by the necessary information and expertise to make them. A detailed description of the method can be found in Buckley 2002, Buckley 2003, Buckley and Stammnitz 2004. This paper describes how the system abstraction techniques of the DOSE method improves the engineer's ability to conduct top-down design, improve design flexibility and affordability while providing the cognitive engineer with a convenient framework with which to deal with various HSI concerns.

THE SE-HSI GAP

Much has been written about an apparent gap that exists between Systems Engineering and HSI practice (Hoffman and Deal 2008, Roth and Pew 2008). Some of the discussion centers on cultural differences that exist between the two disciplines. System engineers are trained to examine systems abstractly; HSI engineers attempt to deal with the full complexity of human – system – environment interactions. System engineers use functional analysis methods to help deal with complexity and derive a design; HSI engineers use functional analysis to inform the design with the necessary changes to enable or enhance operability, habitability, etc., (Malone and Carson, 2001). However, the difference is more than cultural, it is also philosophical. System engineers opt for structured, simple abstractions to span required functionality. HSI engineers focus on "influencing the design to optimize system manning, training, safety, [personnel] survivability, and quality of life" (Malone and Carson, 2001), among other things. A method to achieve the human-centered design goals of HSI within the structured framework of a top-down system engineering effort is highly desired. This paper describes such a method.

SE CHALLENGE OF HUMAN-CENTERED DESIGN

One of the contributors to the SE-HSI gap is the human-machine allocation difficulty. In the normal course of a functional analysis effort, a system is decomposed into functions and sub-functions, etc., with the ultimate intent of packaging functions into components that meet requirements, while minimizing coupling problems and maximizing producibility and affordability.

Consider the notional functional decomposition of Figure 1. As an example, assume that the blocks outlined in black signify functions requiring a human computer interface (HCI). Usually such decisions are made after the functional decomposition has been developed. It is a complex task to attempt to arrange or group functions after the fact in order to design an operable, producible HCI. This paper describes a method that facilitates discovering the needed human-machine interactions up front as a structured part of functional analysis.

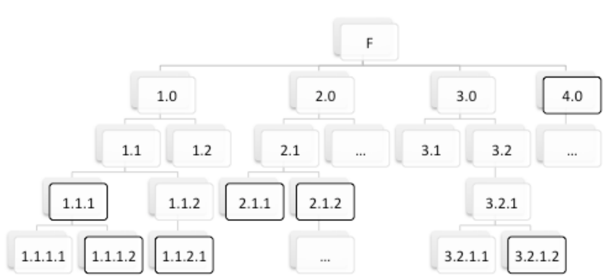

Figure 1 Notional functional decomposition highlighting instances requiring human computer interface (HCI) requirements

In addition to the arrangement of functions for an effective and efficient (HCI), there is the broader challenge of efficient packaging of functions to minimize coupling and dependencies.

The theoretical number of packaging options for grouping *n* functions is:

$$P_n = (n!/2) + 1, \text{ where } n > 3, \text{ and}$$

P_n = number of possible ways that the *n* functions may be grouped.

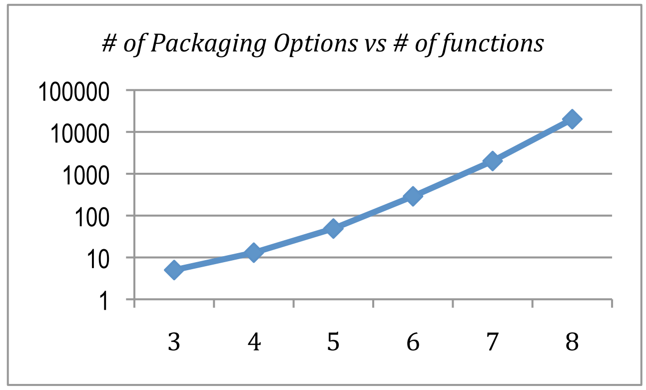

Figure 2 An explosion of packaging options

Other constraints (i.e, dependency and required sequencing) aside, Figure 2 shows the explosion of theoretically possible packaging options, assuming a given function is used only once in a packaging option. Constraints like dependency and required sequencing reduce the number of feasible options. On the other hand, in an actual system, a specific function can be invoked many times and in many components, greatly increasing the number of possibilities. Figure 2 is only included to provide a sense of the exponential growth extant, with increasing number of functions, in the problem of packaging functions for efficient organization.

Although many of the packaging options would be screened as non-viable, it is evident that if the packaging problem could be simplified prior to the development of a functional model, much could be gained. And although techniques such as Design Structure Matrix do exist to speed up the screening and simplification process, the number of theoretical possibilities balloons with any reasonably sized system of functions. The method described in this paper allows this challenge to be tackled (and much of the simplification performed) prior to the development of a functional model, making the challenge of minimizing dependencies more doable.

Although it is obvious that many of the possible packaging options could be easily dismissed due to an apparent non-viability, it is no trivial exercise to focus in on the viable options. To do that, functions need to be well=defined, overlaps eliminated, durations estimated, start and stop criteria established, and feedback loops identified, etc. The many degrees of freedom inherent in the functional model need to be managed for an optimal arrangement to be sought. If dependencies could be minimized in a descriptive framework with fewer degrees of freedom, the job of describing the essence of system operation could be made vastly simpler. The DOSE method characterizes system operation using fewer degrees of freedom than a functional model, or even a component model, though the demonstration of this latter option is beyond the scope of this paper.

THE ESSENCE OF A SYSTEM

It is normal practice to think of a system in terms of constituent functions, perhaps because it is the functional requirements that determine what gets built in today's world. We live in a world of functions, processes, activities, and procedures. However, as soon as we think in terms of functions or activities we are pulled into talking about *how* things could be done. As soon as we are thinking about how things could be done, we are pulled into talking about which systems might be used and in what ways. This is a perfect recipe for premature design decisions leading to early cost commitment, making it very difficult to proactively pursue affordability goals.

What is the essence of a system? How can the essence of system operation be simply described for requirements capture, allocation and analysis for simplification and optimization? In the DOSE framework system operation is modeled as an evolution of knowledge from raw data to mature decisions, punctuated by the key decisions required to get there. DOSE is founded on the premise that every problem, situation, or system has two faces, the real situation in the physical world and people's perception of it. This is an idea which has evolved in different flavors throughout the history of philosophical thought but has also found a home of sorts in system design theory: "Humans do not respond to the physical qualities of things but act on what they mean to them" (see Krippendorff 2006). DOSE provides a framework for capturing and representing that perception as an evolution of knowledge punctuated by key decisions supported by information and expertise.

Before dismissing the importance of the "perception" issue as subjective nonsense, consider this fact: every ship built, every successful system acquisition effort is first and foremost, a socio-technical exercise. Success of the effort is dependent as much on the politics of the acquisition environment, efficient program execution, and effective cost negotiation in the face of uncertain risk as it is on the robustness of the physical solution. All of these challenges are heavily influence by an environment of changing perceptions.

Decision Oriented System Engineering (DOSE) is a method (U.S. Patent 7493298) that provides a decision model representation of problems, situations, and systems in the form of knowledge maps. In a DOSE analysis, decisions are identified, ordered and transformed into decision chains that, in turn, become the backbone of knowledge maps. Whether the intent is to develop a strategic plan, characterize a problem, or describe a system, key cognitive decisions needed to represent the perception are derived (with the help of subject matter experts) along with required information, until a complete picture (knowledge map) is developed.

DEFINITIONS

The decision-making and knowledge evolution perspectives employed in the DOSE method warrant the use of a few new terms and some tuned definitions. For example, in many circles a *decision* is synonymous with *choice*. However, in the DOSE method and in this paper a *decision* refers to a named set of choices. A few additional definitions useful in the description of the method follow.

The broadest term in the DOSE framework is *Responsibility*. In the DOSE method desired operational capabilities (perhaps from an evolving Concept of Operations) are reframed as responsibilities and each responsibility is mapped as a progression of key decisions. A *responsibility* is assignable to an individual or team and one can expect to be held accountable for success or failure in managing a *responsibility*. A *responsibility* is usually defined with implicit or explicit standards for success. A responsibility that is fully characterized and accepted as part of the design becomes a decision-making requirement.

In DOSE *Decisions* are not choices, they are names given to cognitive determinations made with regard to sets of choices. An example of a key *decision* in the DOSE framework might be "Hostile Intent Assessed" or "Situation Assessed". Note that the choice made is not what is important in this abstraction of system operation. By avoiding the enumeration of possible choices and outcomes the characterization is unnecessary and the resultant decision model of the system is simpler.

Decision Chains are ordered sequences of key decisions and decision products.

A *DOSE Knowledge Map* is a knowledge map that follows the conventions of the DOSE method. It is a directed graph network of nodes representing information products, specific skills and experience and key decisions (represented as decision products) organized according to precedence.

DOSE OVERVIEW

As stated above, the DOSE method involves construction of a system model in the decision domain. The result is a DOSE Knowledge Map depicting what is necessary to support system operation, but without detailing specific processes. Figure 3 below details the conventions used in a DOSE Knowledge Map and notionally depicts what a portion of a completed Knowledge Map looks like.

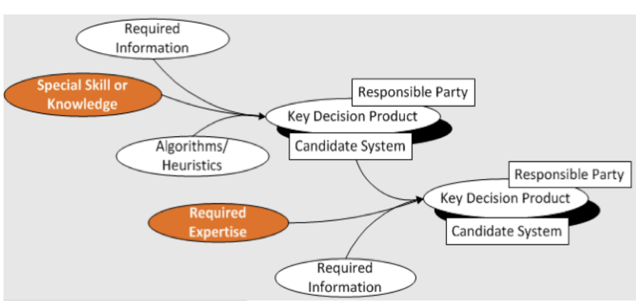

Figure 3 DOSE Knowledge Map conventions

Only the evolution of data to information to decisions and what is required to support that evolution is captured. However, it is captured in a series of steps that involves the iterative development of three artifacts needed to complete the decision model: Decision Chains, Knowledge Evolution Summaries, and the DOSE Knowledge Maps. As stated earlier the details of constructing this decision model have been described elsewhere, however a summary of the method is included here to provide the backdrop for the purpose of this paper: an explanation of why the method can support development of more flexible and affordable designs for ships and other complex systems.

In a completed DOSE Knowledge Map, all nodes represent information products, some of which may be key decision products, supporting information, or special skills, expertise, etc. Shadowed nodes identify decision products key to the knowledge evolution. Arrows indicate precedence relationships and identify a transition of knowledge from a lower to a higher state. The arrows do not indicate an actual *flow* of information, as in a data input, although the transition may imply need for a process with inputs and outputs that can be defined later. Resulting knowledge maps can be translated into simpler process models.

As it is important to capture human cognitive elements along the way, wherever specific human knowledge, skills, or abilities are identified as required context for a decision or information, red nodes are used to signify these contributions. Figure 4 shows the steps in the process and supports the summary description below.

Step 1: *Partition decision responsibilities and identify knowledge cells; identify decision objectives and high-level quality targets for the responsibilities.*

The process begins with a partitioning of decision responsibilities for system operation into mission areas and problem domains (knowledge cells). Requirements guidance documents and operations SMEs are consulted in this effort. The partitioning effort should result in responsibilities that are reasonably assignable to individuals or teams.

Step 2: *Capture key decisions; order decisions to build decision chains; establish quality targets.*

With responsibility sets partitioned and defined, one proceeds with the capture of key decisions to build the decision chains that will ultimately become the backbone of the knowledge map. This is done for each responsibility. In naming decisions we look for labels that would be appropriate for a decision made, making every effort to avoid worrying about specific choices.
The success of any system is measured by how well people using the system manage their responsibilities. A high-level statement of how well the knowledge cell should be managed is developed. Following this, "Quality Targets" are identified to set a target for how good each decision must be.

Step 3: *Capture/ resolve required decision context; develop initial DOSE knowledge maps and knowledge evolution summaries*.
A table is developed containing the decisions and decision products identified as essential to completing the decision chain/s. This is the Knowledge Evolution Summary and is used to help flesh out the full context of information required to support key decision chains. In this step the focus is on capturing the requirements and the job is done when the required decision context is considered complete.

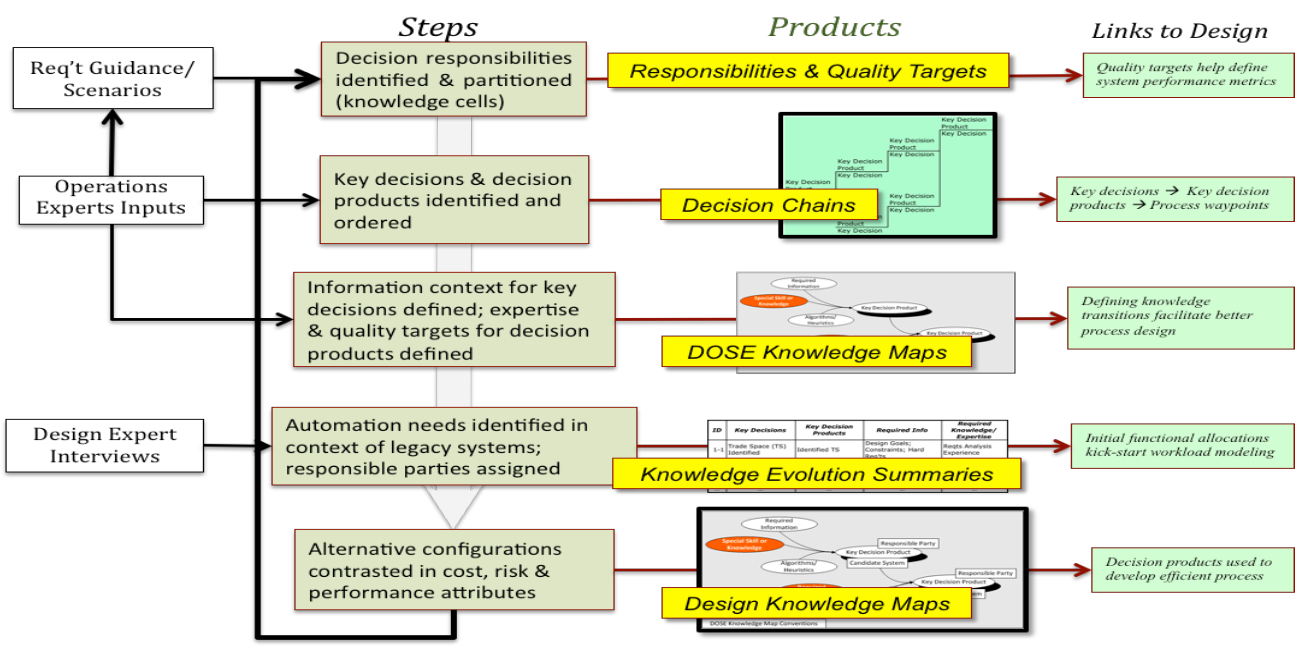

Figure 4 DOSE method overview

As the decision chains are developed, it is important to focus on what is needed to support the humans working with the new system and what would constitute successful management of the decision responsibility. Human cognitive requirements are captured along the way, even the "little stuff". This includes anything that might be required in a training program, educational curriculum or experience profile to handle the responsibility. Quality targets for individual decision products are defined and captured in the Knowledge Evolution Summaries.

Key decisions and required context are used to sketch out a first cut Knowledge Map. The Knowledge Map is completely defined from a requirements perspective when all critical contexts necessary to support decision-making requirements have been captured. Context includes relevant information from whatever source, and the necessary human knowledge, skills, and abilities considered important to the decisions.

4: *Function allocation step: identify candidate components, automation opportunities, and essential personnel in the knowledge evolution summaries*.
Completing the Knowledge Evolution Summary facilitates designing for automation and a sound HCI, among other things. Possible legacy systems and

available capabilities are identified and captured for subsequent analysis. This allows a picture of what's needed to support the decision-making needs for each of the key decisions to begin to emerge. Where there is no current capability provided by legacy systems, a new system capability development (or an extension to a current system) may be warranted or suggested.

In rounding out the decision making requirements for the system and checking for completeness, the following questions apply: "How difficult is it going to be to synthesize the decision product based on the available context?" The answer helps suggest needed automation. The answers to other questions round out the discussion: Is the technology available or, better yet, is it practical? Are there alternatives? If all or part of the responsibility is to remain with an individual who should that person be? What skills or experience should he/she possess?

Step 5: *Complete knowledge evolution summaries and knowledge maps.*
Decision products whose generation may be supported wholly or in part by automation are identified for further analysis. This yields specific candidate automation opportunities. Design decisions regarding which information products will be automated and to what level are captured. Consistent with the design decisions, sources of information products are identified, whether they are to be provided by legacy systems, new systems, people, or any combination thereof.

With respect to completeness, the evolution of data and information represented on a DOSE knowledge map represents an evolution of decision contexts to support some set of decisions that should be made for successful management of the responsibility. The knowledge map is meant to capture the "superset" of all information, supporting decision products, human knowledge, skills, and abilities that may be brought into play to make the key decisions. It is recognized that quite often decision makers must make decisions in the face of less information than they want or need. In such circumstances knowledge of the "superset" of information may be useful in helping the decision maker make a more informed decision.

The Knowledge Evolution Summaries record key decisions, required information, and expertise and other associated attributes. Attributes captured include: Required Context, Contributing Elements/ Components, Automation Level or Category, Required Experience or Expertise, Special Skills and Skill Level. These are just examples as the attributes captured can vary a bit from effort to effort; they should be tailored to the needs of the specific domain.

All decision context elements: decision products, information products, expertise, etc., are analyzed for required quality, resulting in specific inputs to performance specifications. This analysis may also impact functional requirements for the candidate system components. For example, when a quality standard is applied to a decision there is usually an associated impact on the quality of information that must be available. This may result in more stringent requirements on source data. Conversely, if many sources of information will be available that may be correlated, with some sources outside the scope of the system under design, perhaps source data specs for sources within the intended system may be relaxed somewhat.

Step 6: *Iterate and simplify; revise / simplify organization concept; revise partitioned responsibilities*
With the completion of step 5, opportunities for simplification or commonality within the responsibility have usually been discovered. In step 6 the focus is across other responsibilities. With the help of Cognitive Task Analysis or a similar detailed analysis of the task and procedures necessary to accomplish a specific job the candidate configurations are examined **across** mission areas and problem domains for redundant responsibilities and opportunities for simplification across responsibilities.

DECISION-ORIENTED DIFFERENCES

At first glance applying the DOSE method may seem like extra work, extra steps in an already complicated multistep process of ship design. Before arriving at this conclusion consider some of the differences afforded by the method and the resultant benefits.

Design processes and techniques generally assume a stable understanding of system needs, and justifiably so. Ultimately, some static assumptions must be made in order to get something built. However, the decision-oriented approach of the DOSE method offers a higher level of abstraction of system operation than a functional representation, it is less sensitive to the need for change due to technology upgrades. This is due to the fact it is focused on what decisions the decision makers need to navigate and these *operational* decisions evolve much more slowly over time than do the technologies of the supporting systems. Any design founded on operational decisions is more stable in the face of technology upgrades and inherently more flexible to these same technology upgrades. Further, since a DOSE decision model founded on operational decisions provides a more stable design foundation over the system's life cycle, it also enhances traceability over the same life cycle. A design founded on operational decisions is also more easily optimized in terms of required processes.

What is true for the operational design is equally true for the design process. Using the same decision modeling approach, the higher level of abstraction provides more flexibility for similar reasons as above.

Every complex design is, first and foremost, a socio-technical challenge. The social and political challenge of program execution in the face of ever-evolving requirements and dynamic political and economic contexts can often outweigh the difficulty of the technical challenge of building a ship to a given set of requirements. The socio-technical challenge of design is as much a problem of perception management as anything else. The DOSE method provides for mapping this perception for more effective management.

In the decision oriented approach of the DOSE method the essence of system operation is summarized by a network of key decision nodes supported by information and expertise. The only relationship among the nodes is one of precedence. Though this decision model representation is more abstract than a functional representation, it is also more removed from the functions required of specific systems.

There are more degrees of freedom to manage in a functional model than a DOSE decision model. It is a simpler matter to minimize information, decision and expertise dependencies in the DOSE decision model than it is to optimize the arrangement of a functional model because the information model reflects only a fraction of the complexity of an equivalent functional model. See Table 1 below for a relative comparison of the complexity of each in terms of degrees of freedom.

The bold text of Table 1 indicates the subset of factors dealt with in a DOSE decision model. The number of factors is less than half of the total that contribute to the complexity of a functional or process model. Consequently, it is simpler and more efficient effort to first build and optimize a DOSE decision model than it is to build and optimize a functional model alone.

Table 1 Complexity of DOSE vs functional modeling

Factors	DOSE Model	Functional Model
Levels of information	DOSE Model	Functional Model
Info dependencies		
Decision milestones		
Decision products		
People assignments		
Feedback loops		
Activities		
Activity definitions		
Start/stop criteria		
Timing requirements		
Activity durations		
Activity dependencies		
Resource allocations		
Procedures/Protocols		

The higher level of abstraction inherent in the DOSE decision model translates to fewer entities needed to capture the essence of system operation. In the DOSE analyses conducted in support of early DDG 1000 phases, it was observed that the decision model usually required between an order of magnitude and an order of magnitude and a half fewer key decision products than functions to span the same operational functionality. In other words, while the decision models were requiring anywhere from a 1-2 dozen key decision products to span a given responsibility, the sister functional model was requiring 10-50 times as many functions to span the same functionality.

Consider the challenge of packaging functions to minimize dependencies (and potential coupling problems) and improve system producibility. Figure 2 clearly shows the explosive growth of possibilities with increasing number of functions, assuming just a brute force approach in looking for better packaging options. Clearly, the likelihood of missing very loosely coupled packaging combinations looms large no matter how smart the search engine or organization scheme. Fewer entities to consider in screening packaging combinations translate to a quicker discovery of simpler arrangements.

Whether the system is primarily hardware or software, tightly coupled systems and components (many dependencies) are hard to produce and harder to upgrade. Loosely coupled (very few dependencies) are easier to produce and easiest to upgrade. Fewer dependencies translate to fewer changes to accommodate technology upgrades. This translates to better design flexibility and improved affordability. This is especially true in systems with long life cycles.

A side benefit of the decision-oriented approach is a benefit to operable and efficient human computer interface (HCI) design. By capturing the decision-making needs and the information and expertise to support those needs, HCI content requirements are captured early, making it easier to organize information access and HCI interaction for superior operability.

There is, however, a more fundamental reason that a decision oriented approach facilitates design flexibility. It is because such an approach is focused on system users as decision makers. How will system operators continue to efficiently and effectively use the system over the life of the system, through the many, many upgrades and technology refresh cycles that are bound to occur? The answer lies in the robustness of the human-machine design and its capacity to support technology evolution while sustaining the decision support requirements of the people using the system.

CLOSING THE HSI GAP

Earlier it was noted that DOSE's decision-oriented approach helps close the HSI gap. What is this HSI gap? At one level, system engineers *derive* the design using functional analysis methods, while HSI Engineers usually *influence* the design to deal with HSI issues and concerns. The gap refers to the difference between systematically deriving the design and adjusting a design to meet specific HSI needs.

On a more practical level, the HSI gap refers to the problem of transforming desired capabilities into functional requirements while embracing operator needs (including HSI concerns). Refer to Figure 5 to consider the following example: Suppose a desired capability of a new system is an ability to support situation assessment in a specified set of scenarios, situations and constraints. For this example assume the context (required scenarios, situations and constraints) has been fully defined. The system (including the people) must provide a situation assessment capability. As the knowledge map fragment in Figure 5 indicates, the key decision product for this capability could be named "Estimation of Hostile Intent". For the estimate, certain information and expertise are required, and this would be detailed in the information and expertise nodes supporting the key decision product, "estimation of hostile intent".

Figure 5 DOSE Knowledge Map fragment showing 2 key decision products

In contrast, in an equivalent functional model, the notable process node might be labeled *conduct situation assessment* or *assess situation*. Either describes an activity and invites a discussion of "how" the situation assessment will be conducted. It also invites a discussion of how long it will take. This leads to a discussion of tasks and procedures. The difference may not seem like much, but it is much easier to impose a discipline of focusing on "what" needs to be accomplished with the decision model than with its functional model counterpart. In the decision model representation, information dependency is what is captured. "How" the information will be synthesized to produce the decision product is ignored until the information model of the system has been made as simple as possible.

The decision model representation is superior in another aspect: all HSI needs can be shown to obtain from an understanding of the cognitive needs of the user or operator as decision maker. For an operator to perform effectively, certain standards for comfort, safety, endurance, etc., apply, all of which are necessary for him or her to perform satisfactorily.

A SPRINGBOARD FOR HSI

In terms of facilitating a structured and systematic review of HSI concerns and issues, HSI-relevant results of a DOSE decision-making requirements capture include the following:

1. Minimum set of tangible decision products key to the decision maker/s (user/s) for successful handling of the responsibility, defined by the information that needs to be synthesized and expertise required to do so;

2. Knowledge map summarizing the above that is a decision-making requirements roadmap for driving other system requirements, including the analysis of HSI concerns;

3. Description of information content requirements (contained in the knowledge map) for an HCI to support required operational decisions;

4. Summary of skills and expertise needed to support the decisions (contained in the knowledge map) and useful to initiate analysis for training requirements;

5. Network (knowledge map) of operational decisions and required information that provides a visual framework for examining (a) where human judgment is needed, and (b) decision making policies and protocols, required information, and expertise

With the above information a translation to an efficient process or functional model representation is relatively straightforward; only now, process outputs are decision (information) products that directly support the system users, operators, decision makers. The results provide a convenient springboard for examining the various HSI concerns in a top-down manner, as an intrinsic component of the derivation of system requirements. Habitability and quality of life concerns, manpower needs, training requirements, personnel assignments, safety and health issues, personnel survivability concerns, and human factors issues all originate from the envelope of decision making requirements (see Figure 6).

For example, manpower needs are tied to cognitive workload demands of operators as decision makers; Personnel assignments are based on skill and experience levels tied to decision making demands; quality of life is impacted by decision making demands, associated stress levels and the environment for operations. Other pillars of HSI are linked just as tightly to decision-making requirements. The analysis for each category of concern or issue may vector off in different directions, some dependent on others, some relatively separable, but each is traceable to the decision-making requirements for the responsibilities that must be managed.

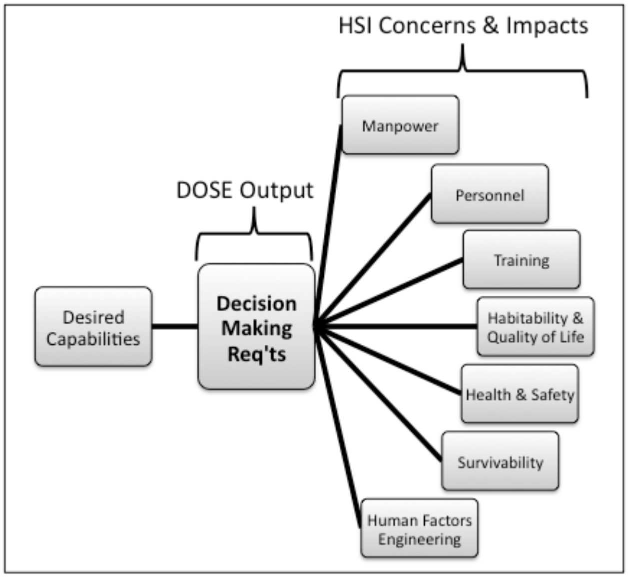

Figure 6 Decision making requirements bridge the gap between desired capabilities and design for HSI concerns

CONCLUSIONS

Decision Oriented Systems Engineering (DOSE) is a new approach to designing systems from the top down. It is a decision punctuated process design technique that forestalls the discussion of specific processes by capturing the essence of system operation as a network of decisions. In this paper an overview of the method has been provided and some of the benefits to design affordability, design flexibility, and the management of design complexity discussed.

The decision-oriented approach of the DOSE method offers a higher level of abstraction of system operation than a functional representation. This translates to fewer entities to describe and span a given functionality. Fewer dependencies translate to fewer changes needed to accommodate technology upgrades, and this in turn, translates to an easier task of minimizing dependencies for improved producibility, affordability, and design flexibility.

Because of this, the system model so described is less sensitive to the need for change due to technology upgrades. This is due to the fact it is focused on what decisions the decision makers need to navigate and these operational decisions evolve much more slowly over time than the technologies of the supporting systems.

The decision-oriented backbone of the DOSE system model is more stable and less sensitive to changes in technology or environment. For example, a Tactical Action Officer on a naval vessel must be ever vigilant of the operating environment. One of his/her most important responsibilities is a constant and continuous estimation of hostile intent for contacts of interest. This responsibility remains generally constant even as technology advances continue to leapfrog current capabilities.

Finally, the results of a DOSE decision-making requirements capture provide a convenient springboard for examining the various HSI concerns in a systematic manner. By providing a structured framework to capture, analyze and simplify decision-making needs, the full gamut of HSI concerns may be tackled systematically, as part of the normal derivation of the system design. Instead of merely influencing the design, sometimes belatedly, and often at great cost, HSI analyses can drive the design as part of the formal SE design process.

REFERENCES

Buckley, M.E, (2002). “A Method and System for Decision Oriented Systems Engineering”. S/N-10/358,117, Patent Application Publication No. US-2003-0177047-A1. September 18, 2003.

Buckley, M.E., (2003) “DD(X) Program Knowledge Engineering Data and Results”, Northrop Grumman Report, DD(X) CDRL B7, Unpublished Manuscript.

Buckley, Michael E, and Peretin, Jared, (2008) “Ship to Shore Connector Implementation of Set Based Design”, CDI Marine Working Paper 765.002.023-1.

Buckley, Michael E. and Stammnitz, Volker, (2004) “Decision Oriented Systems Engineering (DOSE) – A Structured, Systematic Approach to Function Allocation”, 14th Annual International Symposium for INCOSE, Proceedings, 2004.

Buckley, Michael E., and Womersley, Mark, (2007) “Mapping the Decision Landscape of Port Security Governance Issues”, Maritime Technology (MTEC) Conference Proceedings.

Hoffman, Robert R and Deal, Steven V, (2008) “Influencing versus Informing Design, Part 1: A Gap Analysis”, IEEE Intelligent Systems, Sep/Oct 2008, pp. 78-81.

Krippendorff, K. (2006) *The Semantic Turn: A New Foundation for Design*, Taylor & Francis/CRC Press, Boca Raton, FL/New York, NY.

Landsburg, Alexander C., Avery, Larry, Beaton, Robert, Comperatore, Carlos, Khandpur, Malone, Thomas B., Parker, Christopher, Popkin, Stephen, and Sheridan, Thomas B. (2008) “The Art of Successfully Applying Human Systems Integration”, Naval Engineers Journal, 2008 #1, pp 77-107.

Malone, T.B. and Carson, F. (2001) “HSI Top Down Requirements Analysis”, Naval Engineers Journal, Spring, 2003, pp 37-48.

Mebane, Walter L., Carlson Craig M., Dowd, Chris Dowd, Singer, David J., and Buckley, Michael E. (2011) “Set-Based Design and the Ship to Shore Connector”, ASNE Day 2011 Proceedings.

Militello, Laura G., Dominguez, Cynthia O., Lintern, Gavan and Klein, Gary. (2010) “The Role of Cognitive Systems Engineering in the Systems Engineering Design Process”, International Council of Systems Engineering (INCOSE) Vol. 13 No. 3, pp. 261-273.

“Naval Human Systems Integration Plan (Annex to the OSD HSI Plan)”, (2009) SECNAVINST 5000.2D, Version 2.2, Dept of the Navy.

Pew, Richard W, (2008) “Some New Perspectives for Introducing Human-Systems Integration into the System Development Process”, Journal of Cognitive Engineering and Decision Making, Vol. 2, No. 3, pp 165-180.

Roth, Emilie M, and Pew, Richard W, (2008) “Integrating Cognitive Engineering in the Systems Engineering Process: Opportunities, Challenges, and Emerging Approaches”, Journal of Cognitive Engineering and Decision Making, Vol. 2, Number 3, pp. 161-164.

Singer, David J., Doerry, Norbert, Buckley, Michael E., (2009) “What is Set Based Design?”, ASNE Day 2009-Naval Engineering in Support of 21st Century Maritime Strategy, Proceedings.