

Defining an Enterprise Lead Systems Integration (LSI) Framework

Warren K. Vaneman
Systems Engineering Department
Naval Postgraduate School
Monterey, CA, USA

Ronald Carlson
Systems Engineering Department
Naval Postgraduate School
Monterey, CA, USA

Abstract— In the modern operational environment, multiple systems forming a System of Systems (SoS) are required to satisfy the spectrum of capabilities needed to satisfy the mission. Accomplishing the mission has always been a SoS endeavor, where integrating multiple systems into a SoS has been left to small communities of “hero engineers,” or to the operators responsible for the mission. The acquisition and management of these mission capabilities across the SoS lifecycle requires the complex integration of interdependent new and legacy systems from the lowest component level to the highest enterprise level. In 2008, Congress directed government organizations to adopt a Lead System Integration (LSI) process to address the issues with the acquisition, development, and integration of a SoS. This paper introduces an LSI Enterprise Framework that identifies the various levels of interaction, organizational functions, and the universal resources to ensure the enterprise provides a SoS that delivers the required capabilities.

Keywords— *System of Systems; System of Systems Engineering and Integration; Lead Systems Integration*

I. INTRODUCTION

“Our current system is like a machine to which we just keep adding important and wanted items but without a cohesive strategy for an elegant, interwoven system. Considered on their own, the addition and growth of individual elements may be useful. But when ownership organizations do not see how their contribution fits into the whole and think their element is an end-state in itself, effective communication and executions are inhibited.” [1]

Modern complex mission capabilities are fundamentally achieved with multi-mission, highly interoperable system of systems (SoS). A SoS is a set or arrangement of systems that results when independent, and task-oriented systems are integrated into a larger systems construct, that delivers unique capabilities and functions in support of missions that cannot be achieved by individual systems alone [2]. The challenge of integrating these disparate constituent systems into a SoS, is that they are developed and procured asynchronously, usually by different program offices, and often across different enterprises. As such, these high criticality SoS carry a high consequence of failure if not achieved successfully.

System of Systems Engineering and Integration (SoSE&I) was conceived for the planning, analyzing, and integrating of constituent systems into a SoS capability greater than the sum of those systems [3]. The SoSE&I methodology is built on a

Systems Engineering “Vee” process model (Fig. 1), that includes the engineering activities that take place during the lifecycle of a SoS [2]. The goals of SoSE&I are:

- Define engineering and management processes that must occur across the SoS lifecycle;
- Increase speed of delivering the SoS capabilities to the operational environment;
- Approve affordability across the SoS lifecycle;
- Improve SoS agility by predicting and managing emergent behavior;
- Maximize the “value” for the SoS by delivering the capabilities needed for mission success.

While SoSE&I provides the methodology throughout the SoS lifecycle, it does not specify who is responsible for accomplishing these activities. In fact, SoS integration is frequently left to small communities of system developers or by the operators themselves.

While the SoSE&I “Vee” was developed for the design and management of a SoS, it can also be applied to complex systems, where the system structure and interfaces is so significant that the major sub-systems often are considered systems in the own rite. The Lead System Integration (LSI) activities can be applied equally to a SoS or complex system, but will be referred to a SoS throughout this paper.

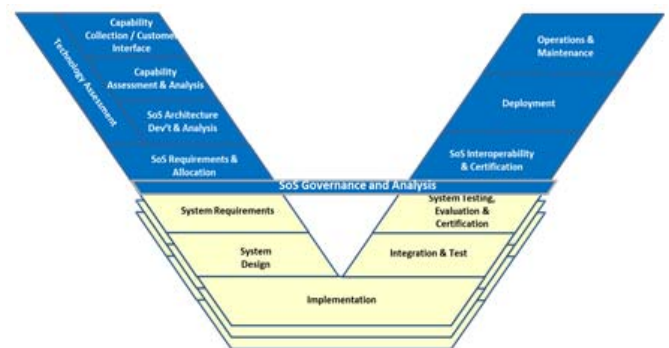


Fig. 1. SoSE&I “Vee” Model [2].

Recognizing the challenges of multi-billion-dollar SoS acquisitions, in the mid-1990s, the U.S. government embraced the practice of partnering with industry to effectively develop and integrate these SoS [4]. The term “Lead System Integrator” became commonplace in large-scale SoS developments such as the Army’s Future Combat System and the Coast Guard’s Deepwater program [5, 6]. However, after several high-profile SoS acquisition failures, it was evident that a full-scale rebalancing of risks and rewards was needed.

The remainder of this paper is organized as follows: Section II defines Lead System Integration; Section III describes an LSI Enterprise Framework from which SoSE&I process can be executed; Section IV defines the LSI touchpoints, or functions, required for the realization of a SoS; Section V briefly introduces the Universal Enabling Resources that the LSI must be familiar with for to establish a successful LSI; and, Section VI provides the conclusions.

II. LEAD SYSTEM INTEGRATION DEFINED

In 2008 Congress directed the Secretary of Defense to ensure that the acquisition workforce is of the appropriate size and skills necessary to accomplish inherently governmental functions related to acquisition of major defense systems and to minimize and eventually eliminate the use of contractors to perform LSI functions [7].

Lead Systems Integration is an acquisition strategy that employs a series of methods, practices, and principles to increase the span of both management and engineering acquisition authority and control to acquire a SoS or highly complex systems. LSI is effectively a “marriage” of program management and multiple functional disciplines which must work together cooperatively to assert and execute trade space in the SoS given multiple constituent system acquisitions. The LSI function is to assert and execute SoS and stakeholder trade space to affordably optimize integrated mission capabilities across the SoS lifecycle [8]. The roles of the LSI are similar to the roles of any systems engineer or system integrator within a program office. The primary difference is the span of LSI design and integration authority that persists throughout the SoS lifecycle.

III. LSI ENTERPRISE FRAMEWORK

To successfully plan, develop, and manage a SoS, a comprehensive development, acquisition, and implementation strategy is required. The SoSE&I “Vee” defines the activities required to engineer and manage a SoS throughout the lifecycle [2]. The LSI Enterprise Framework defines a means to engineer and manage the capabilities and interdependencies of a SoS, that can be executed by the government LSI, across multiple systems, programs, and stakeholder levels. The LSI Enterprise Framework captures the complex, interdependent, and mission capability areas through four enterprise levels to characterize the systems from the enterprise to the component level [9].

The Enterprise Level is the top layer of the LSI framework that consists of a variety of stakeholders, from one or many organizations that represent the complex, socio-technical

systems that comprises interdependent resources of people, information, and systems that must interact with each other and their environment to achieve mission success [10]. It is at this level where the capabilities required to achieve enterprise mission success are defined, decomposed into mission capabilities, and allocated to the SoS level to be satisfied as mission capabilities.

While the majority of the LSI engineering and management activities occur below the enterprise level, this level is important because this is where organizational, policy, and resource decisions are made for the LSI.

The System of Systems Level is where a collection of supporting constituent systems and programs are brought together to support end-to-end capability effectiveness for the designated mission areas. Accomplishing a mission that cannot be satisfied by a single system alone has always been a SoS endeavor, but integrating the multiple systems together has frequently been left to small communities consisting of a few systems or the operators themselves [11]. Many LSI governing efforts, at the System of Systems Level, involve a collaborative partnership of multiple program offices, versus a more directive effort that may occur at lower program levels [9]. Individual capabilities and functions are allocated to constituent systems for implementation.

The System Level is where a combination of functionally related physical elements are integrated into a usable, system to achieve the system capability. In this level, the emphasis is on traditional systems engineering and development activities. However, two significant roles are important to the LSI. First, the LSI must ensure that the SoS level organization has sufficient insight into the individual programs within the SoS to understand the functionality and interoperability that will result from the engineering and design effort. Second, the LSI must ensure a strong governance model is in place that provides the technical authority to govern system baselines so that the system delivered for integration into a SoS meets the requirements that were allocated to it [2]. In addition to the LSI’s role in ensuring system integration to a SoS, a LSI may be used for the engineering and development of a complex system, where the system is composed of major sub-systems, and a large number of interacting components.

The lowest level of the LSI Enterprise Framework is the subsystem/component level. This level consists of the allocated sub-systems and components that by themselves may, or may not, provide a usable standalone end product. These are the lowest level building blocks required for any LSI effort and may be managed by a team in a larger program office, or may be managed separately by sub-system program offices [9].

LSI functions in various levels occur at different places within an organization. For complex systems, these activities often occur within the program office. For SoS, the LSI function may occur as an executive level above the program office where they can exercise technical governance. Within each level, various types of LSI interactions and complexities are captured, and have multiple interfaces between program offices at the next lower and next higher levels to ensure proper

communication and coordination is occurring throughout the SoS. Fig. 2 depicts the System and LSI Levels [9].

The LSI has three especially important roles across the framework levels. First, the LSI must have sufficient insight of lower level of system decomposition. Second, the LSI must understand the role of the constituent systems in the SoS capabilities and requirements. And, third, the LSI must ensure there is a strong governance model that provides technical authority with a strong voice within the development and acquisition of constituent systems.

IV. LSI TOUCHPOINTS

Given the breadth of a SoS acquisition effort and recognizing that a government LSI's resources to manage an effort are limited, an LSI must be able to efficiently focus on the highest payoff "touchpoints" of control or influence to assert and execute trade space - aligned across the enterprise – to enable organizational agility. Although previous research has discussed inherently governmental functions for an LSI at a high level [12], there has been unclear specific applicability to current program processes and organizations – and some definitions also did not fully account for multidisciplinary functions that extend beyond systems engineering [9].

The LSI Enterprise Framework defines 12 key touchpoints that apply across all domains as the essential "high payoff" functions and activities. These LSI touchpoints are the functions that assert and execute SoS, complex system, and stakeholder trade space to affordably optimize integrated war fighting capabilities across the system of systems lifecycle. These touchpoints do not necessarily define new processes, but do identify how existing processes can be enhanced and used more efficiently [9]. Fig. 3 depicts the traditional program office functions versus touchpoints required for a LSI.

1) *LSI Process Management.* Responsible for mission wholeness, the LSI defines how their processes interface and interact with legacy processes across multiple stakeholders to meet unique SoS mission capabilities and trade space objectives. These standard work processes document the most efficient known method to produce a system or service, eliminating procedural waste and establishing a baseline for future process improvement initiatives. Standard work packages define process trigger conditions, objectives, enabling factors, inputs, functions, outputs, interfaces, and

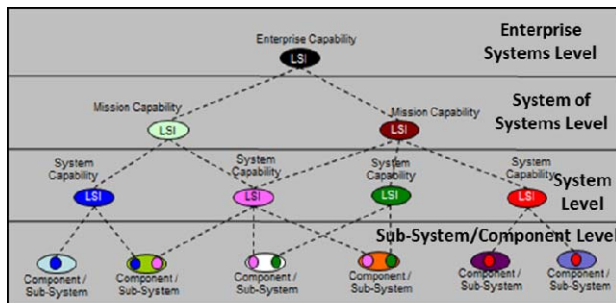


Fig. 2. System and LSI Levels.

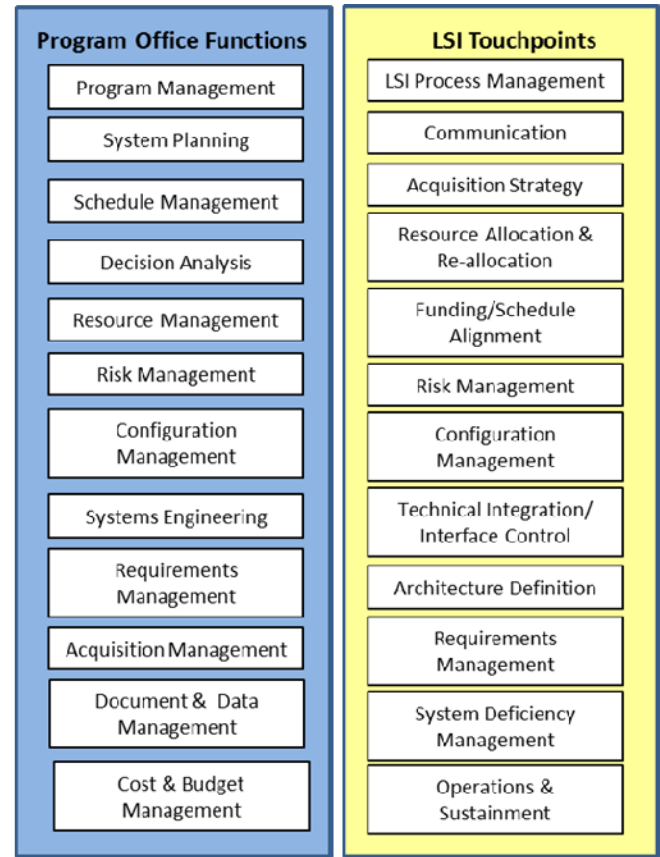


Fig. 3. Program Office Functions vs. LSI Touchpoints.

process time. Furthermore, these standard work processes are the foundation of effects-based staffing, which is critical to defining the skills and resources required to build and maintain an acquisition workforce capable of executing an LSI acquisition strategy [9].

2) *Communication.* The LSI serves as the primary interface and facilitator across a diverse stakeholder constituency. Communications for the LSI is essential to manage multiple stakeholders and enable the organization to react with agility to requirements, design, or emerging stakeholder needs. All members of the LSI team should have a common understanding of assumptions, limitations, and constraints across the SoS [9].

The continuous evolution of SoS capabilities, priorities, mission environments, assumptions, constraints, and threats, mandates unprecedented organizational alignment and enterprise agility. Due to the number of typically "stove-piped" teams and program offices, the need to communicate effectively is a key to success. The desired end state of this communication touchpoint is full programmatic, technical, and organizational alignment between the LSI acquisition objectives, and the objectives of the constituent systems [9].

3) *Acquisition Strategy.* The LSI should serve as the principal SoS acquisition strategist. While the U.S. Government has been assembling SoS for decades, there is often no overarching acquisition strategy. Given their broad

responsibilities, the LSI is often in the best position to develop an overarching acquisition strategy that can be implemented across multiple independent and asynchronous programs and stakeholders to achieve the desired mission capabilities within the resource constraints [9].

4) *Resource Allocation/Re-allocation.* The LSI is the primary arbitrator of enterprise resource allocations and re-allocations between constituent SoS elements and stakeholders. Requirements and risk mitigation plans should be properly funded across the integrated mission architecture in accordance with a LSI value maximization strategy to achieve the desired capability outcomes. Given the inherent volatility, uncertainty, complexity, and ambiguity of SoS mission environments, allocation of requirements and resources is an iterative process that occurs throughout the mission lifecycle [9].

System of Systems asynchronous development schedules add a new degree of complexity to LSI resource allocation and re-allocation functions. Given the broad scope of constituent systems encompassed within many SoS mission architectures, it is unlikely that all elements will be in the same acquisition phase. In order to optimize SoS mission value across the SoS trade space, the LSI should also consider the overall mission readiness throughout the SoS lifecycle, including existing legacy operations & sustainment activities [2, 8].

5) *Enterprise Funding and Schedule Alignment.* The handling of funding is an inherently governmental function. Enterprise funding and schedule alignment is especially challenging for the LSI since resources are usually budgeted by the resource sponsors to specific programs and systems, and not the SoS to satisfy enterprise or mission level capabilities. The LSI should be aware of dynamic funding, and schedule changes across multiple programs, and must align multiple asynchronous schedules of the constituent systems it may control [9].

6) *Risk Management.* Risk management for a SoS is more complex than for traditional systems. Since there are likely many interdependent stakeholders and constituent systems in this effort, the LSI should expand the traditional definition of risk management from the system level and focus on risks at the SoS level. LSI risk management must maintain visibility of risks and opportunities of all systems and critical subsystems, across the SoS trade space. The LSI defines alternative mitigation strategies to combine and normalize these risks across the SoS trade space [9].

7) *Configuration Management.* Configuration management (CM) is the application of appropriate resources, processes, and tools to establish and maintain consistency between the system requirements, the system, and the associated system configuration information. This CM definition must be expanded to address the asynchronous CM across multiple interdependent stakeholders and constituent systems. This asynchronous CM is especially complex for a LSI that must establish and maintain the overall SoS CM baseline throughout the lifecycles for all system baselines. Since multiple system program baselines contribute to mission success, the LSI's CM baseline may change dynamically [9].

Another challenging characteristic for the LSI is that each system may have a different way of managing their CM. Within each system there may be multiple "as designed", "as built" and "as delivered" configurations that may not be evident during the verification and validation phase. The LSI should be able to adjust for dynamic SoS loading to accommodate for these changes and constantly monitor and communicate within the team to ensure the SoS capability is still maintained [9].

8) *Technical Integration and Interface Control.* Technical integration and interface control has a more significant role for the LSI bringing together a SoS, or complex system, than in traditional systems engineering. Since technical trade space management for a SoS occurs at the interfaces between constituent systems, the LSI should focus on enterprise technical integration and interface control. This effort is far more complicated than a traditional acquisition effort, since the technical maturity of the constituent systems within the SoS may be at different levels, and may also be changing at different rates [9].

For systems early in the design lifecycle, the LSI should strive to ensure that the widest degree of interoperability is planned so that the system can participate to satisfy a wide range of mission capabilities. The LSI should work to influence design trade space decisions in systems that are in-development to promote risk reduction for interoperability issues. For the legacy systems, that may have been developed and fielded without consideration for inclusion into a SoS, the LSI should evaluate the system capabilities and technical scope, to determine if the system can be used in a SoS to support mission capabilities, and recommend potential solutions that would enable the legacy system to contribute to the SoS.

9) *Architecture Definition.* An architectural definition for a SoS, preferably developed and hosted in a Model-Based Systems Engineering (MBSE) environment, is essential for engineering, analysis, and management of the SoS. The SoS architecture provides a technical blueprint of the SoS, showing the traceability of functional and derived relationships among all constituent systems. The architectural viewpoints enable stakeholders to visualize, define, and bound the component systems, SoS, and identify integration points both inside and outside the systems. From these views, system interoperability issues can be identified. With proper CM, and use of compatible databases, new systems entering the SoS family may more easily integrate from an LSI standpoint, and where all disciplines can see integration impacts, dependencies, and interoperability concerns [9].

The LSI should place significant effort and value in the overall SoS architecture, since this architecture defines the interfaces for trade space management across the SoS. When defining and maintaining this SoS architecture, the LSI may be required to integrate the architectural data in various architecture tools used by different stakeholders and establish some method for overall architecture configuration management across these multiple databases and stakeholders [9].

10) Requirements Management and Concepts of Employment. Once a preferred SoS concept is established, the LSI allocates requirements, functions, interfaces, and constraints across constituent systems. This task is especially challenging since the LSI must consider enterprise requirements management and concepts of employment (CONEMPS) across multiple systems and stakeholders. The stakeholders may each hold different assumptions, limitations, or constraints about the expected use of systems, and the mission requirements for the SoS. Constituent system decomposition and integration may also change dynamically or emerge during the evolution of the mission capabilities during SoS lifecycle. Requirements management for the LSI is further complicated since the allocation of requirements and resources may be iterative and ongoing across elements that the LSI may not control. The LSI should align requirements, assumptions, limitations, and constraints at the capability level for the overall SoS effort. The CONEMPS may be used as one tool to energize early user and resource sponsor involvement to align stakeholders [9].

11) System Deficiency Management. Systems of systems deficiency management, supported by laboratory and operational verification and validation activities, is challenging for LSIs in complex mission environments involving multiple programs and stakeholders. The LSI should determine the impact of constituent systems deficiencies at the SoS level. The LSI should also determine the best way to mitigate these deficiencies. The use of simulations and prototypes representing each constituent system that comprises the SoS is a cost effective method that can be used for early integration risk reduction, may help to refine requirements, identify additional requirements and constraints at the SoS level that may not be apparent at the system level.

Another approach the LSI may employ to manage system deficiencies is to leverage simulations in live, virtual, and constructive environments. The live, virtual and constructive test and evaluation approach permits “confidence” evaluations of SoS capabilities, and allows for systems, weapons, networks, and sensors to be tested against the most realistic environments and concepts of operationally-based scenarios.

12) Operations and Sustainment. The LSI’s challenge of affordably optimizing integrated system capabilities across the SoS lifecycle is more complex than in a traditional acquisition effort since it may involve multiple independently developed support strategies, or existing legacy system support strategies, across the systems in the SoS which may also be at different levels of maturity. The LSI must understand the support requirements for the entire SoS so that the logistical requirements can be allocated effectively to the constituent systems and supporting stakeholders. The logistics support system should be evaluated across the SoS lifecycle to ensure operational supportability with specific attention to minimizing the logistics footprint. Sustainment costs should also be considered during system development and evaluated during testing to ensure that when the SoS capability is

fielded, the sustainment costs to support the system are within the constituent systems and/or the LSI’s SoS budget [9].

V. UNIVERSAL ENABLING RESOURCES

Universal enabling resources are those resources that support LSI-unique execution at any of the touchpoints to assert and execute the trade space. The four enabling resources and inter-related enablers apply at all levels in the LSI Enterprise Framework, and are outside the responsibilities of the typical program offices. However, the LSI must be aware of these activities, and navigate within them. The four enabling resources are hereby introduced to ensure completeness in the discussion.

1) Staffing and Workforce Development – Due to the unique nature of operating in a complex SoS environment, and executing LSI efforts, requires an additional depth of focus and tailored enhanced knowledge, skills, and experiences beyond that required in traditional acquisition programs. Typical organizational functions supporting the touchpoints should be staffed by personnel trained with the requisite expertise and knowledge of operationally relevant environments at the various LSI Enterprise Framework levels [9].

2) Authoritative Data in Context. The complex nature of the SoS environment makes asserting and executing the trade space essential, and creates the need for sound, authoritative data across systems. In any LSI effort, everyone must have the same data, and have a way to validate the authenticity, and accuracy, of the data to be used for decisions. “Authoritative Data in Context” includes a comprehensive integrated set of programmatic, technical, and stakeholder data that enables a shared common understanding of the trade space [9].

3) Policy. Policy consists of the technical, organizational, and legal guidance and constraints of the LSI organization. This may include public law, civil mandates, legal rulings, competency policies, certification requirements, and other overarching guidance that must be accounted for by a LSI when executing any of the touchpoints at any level. These policies provide common guidance across the organizational levels, though the relative impact and flexibility of these policies may vary [9].

4) Resource Management. Resource management includes a cost, schedule, and performance resource triad that captures the relationship between the financial, timing, and capability aspects of the total system. When considered against a set of requirements, the resource triad is necessarily constrained by limiting the available resources to a bounded set [9].

VI. CONCLUSIONS

The use of SoS to satisfy mission capabilities has been prevalent in operations for decades. The SoSE&I “Vee” Model took the initial step of defining the engineering activities throughout the SoS lifecycle. However, this process left an unanswered question, “Who is going to perform the SoSE&I activities?”

The 2008 Congressional mandate specifying LSI as an inherently governmental function addressed the “who,” but remained silent about the “how” the LSI was going to function in an enterprise setting.”

This paper defined how the LSI Enterprise Framework could be used to address the engineering management of SoS and complex systems. The complete framework is shown in Fig. 4 [derived from 9]. This framework begins to shape the problem faced by the planning and engineering of SoS, and complex systems, to satisfy a set of mission capabilities, and serves as the basis to identify the “touchpoints” required to achieve these capabilities.

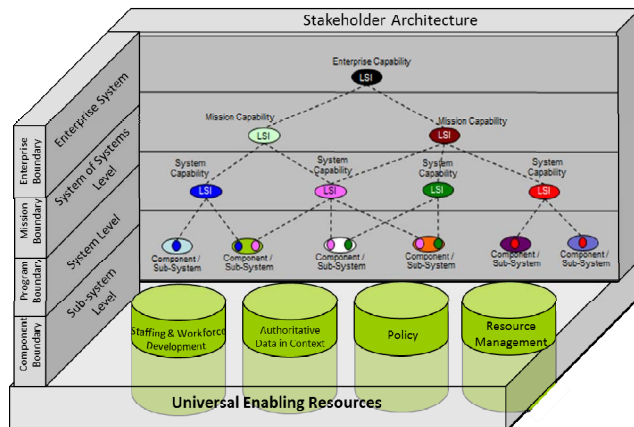


Fig. 4 The LSI Enterprise Framework.

ACKNOWLEDGMENT

The authors wish to thank the Naval Air Systems Command for their sponsorship and support of the Naval Postgraduate School's LSI Certificate Program, and the student research from Cohorts 1 and 2 who made this paper possible.

REFERENCES

- [1] B. Gortney and H. Harris, "Minding the Gaps, " U.S. Naval Institute Proceedings, pp. 36-40, May 2014.
- [2] W.K. Vaneman, "System of Systems Engineering and Integration," Proceedings of the 10th Annual IEEE Systems Conference , April 2016.
- [3] W. K. Vaneman and R. Budka, "Defining a system of systems engineering and integration approach to address the Navy's Information Technology Technical Authority," Proceedings of the INCOSE International Symposium, June 2013.
- [4] K. H. Laudin, "Lead System Integrators: A Post-Acquisition Reform Retrospective," A Publication of the Defense Acquisition University, January 2010.
- [5] C.G. Pernin, *et al.*, "Lessons from the Army's Future Combat Systems Program," RAND Corporation, 2012.
- [6] R. O'Rourke, "Coast Guard Deepwater Acquisitions Programs: Background, Oversight Issues, and Options for Congress," Congressional Research Service, January 2012.
- [7] Public Law 110-181, National Defense Authorization Act for Fiscal Year 2008, Section 802: Lead Systems Integrators.
- [8] NAVAIR Cohort #1, "The Roles of the Government-Led Lead System Integrator (LSI)," Naval Postgraduate School Lead System Integrator (LSI) Certificate Program unpublished report, 25 September 2014.
- [9] NAVAIR Cohort #2, "An Enterprise Lead System Integrator (LSI) Framework," Naval Postgraduate School Lead System Integrator (LSI) Certificate Program unpublished report, 2 October 2015.
- [10] R.E. Giachetti, "Design of Enterprise Systems," New York, NY: CRC Press, 2010.
- [11] .S. Navy Chief Systems Engineer, "Naval System of Systems Engineering Guidebook," Draft 21 Dec 2013.
- [12] Navy Lead Systems Integrator Working Group Final Report, February 2010.