

# Systems of Systems and Net-Centric Enterprise Systems

**Dr. Judith Dahmann**

The MITRE Corporation  
7525 Colshire Drive  
McLean, VA 22102-7539  
[jdahmann@mitre.org](mailto:jdahmann@mitre.org)

**Kristen J. Baldwin**

U.S Department of  
Defense  
Washington, DC  
[kristen.baldwin@osd.mil](mailto:kristen.baldwin@osd.mil)

**George Rebovich Jr.**

The MITRE Corporation  
202 Burlington Road  
Bedford, MA 01730-1420  
[grebovic@mitre.org](mailto:grebovic@mitre.org)

## Abstract

This paper describes the characteristics of systems of systems (SoS) and net-centric enterprise (NCE) systems, examines their similarities and differences and highlights the implications for systems engineering and acquisition. The paper begins with a review of our current understanding of SoS and the implications for SE. It then looks at the characteristics of NCE systems and compares them with the characteristics of SoS. The paper closes with a discussion of the implications for systems engineering and acquisition.

## Introduction

With the evolution of the understanding of operational capabilities in the US Department of Defense, there is increasing attention focused on the challenges of engineering independently useful systems to work together to meet user needs. Activities in support of the development of a DoD guide for systems engineering (SE) of systems of systems (SoS) [1] examined the shape of today's SoS. An extended taxonomy of SoS identifies the different forms of SoS and their implications for systems engineering. [2] A model for SoS SE provides a framework for applying SE processes to SoS [3]. This work highlights the need to push systems thinking beyond the traditional arena of new system development and acquisition to address the reality of today's system challenges of integrating and evolving existing systems to meet changing needs.

One of the emerging principles for SoS is that SoS architectures which are both open and loosely coupled tend to be more successful since they are tolerant of change in parts of the SoS and open to extensions to the SoS, with minimal impact on other parts. In networked information-technology (IT) systems, this principle has taken the form of network "services" which are discoverable by authorized users and which can be invoked as needed. The services may be created and maintained by different organizations, and may be added or changed for a variety of reasons. The systems engineer in these 'network centric enterprise' (NCE) systems is responsible for defining the protocols and standards which are needed for users to access and use the services. In an NCE system, services may be applied by different users in different ways (different sequences) to yield potentially different results. These NCE systems offer a flexible,

extensible approach to rapidly create and evolve system functionality to meet user needs which change on a regular basis.

## Systems of Systems and Systems Engineering

An SoS is defined as a set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities [DoD, 2008]. While DoD acquisition largely continues to emphasize development of individual systems, it has become increasingly recognized that for priority capabilities it is important to provide management and systems engineering to the ensembles of systems which work together to support user capability needs.

In DoD today there are several types of SoS [4, 2] as shown in Table 1. The Future Combat System is a well known example of a “directed SoS”. Communities of interest (COIs) are examples of DoD “collaborative SoS” and the Global Information Grid is the predominant DoD “virtual SoS”. Increasingly DoD is facing the challenges of “acknowledged SoS” which have recognized capability needs, management, and SE at the SoS level as well as autonomous objectives, management and technical development approaches of the systems which contribute to the SoS capability objectives. Examples of this type of SoS are the Missile Defense Agency’s Ballistic Missile Defense System, the Air Force’s Air Operations Center (AOC) and the Navy’s Naval Integrated Fires Counter Air capability.

**Table 1: Types of Systems of Systems [5]**

| Type                 | Definition                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Virtual</b>       | Virtual SoS lack a central management authority and a centrally agreed upon purpose for the system-of-systems. Large-scale behavior emerges—and may be desirable—but this type of SoS must rely upon relatively invisible mechanisms to maintain it.                                                                                                                                                                                |
| <b>Collaborative</b> | In collaborative SoS the component systems interact more or less voluntarily to fulfill agreed upon central purposes. The Internet is a collaborative system. The Internet Engineering Task Force works out standards but has no power to enforce them. The central players collectively decide how to provide or deny service, thereby providing some means of enforcing and maintaining standards.                                |
| <b>Acknowledged</b>  | Acknowledged SoS have recognized objectives, a designated manager, and resources for the SoS; however, the constituent systems retain their independent ownership, objectives, funding, and development and sustainment approaches. Changes in the systems are based on collaboration between the SoS and the system.                                                                                                               |
| <b>Directed</b>      | Directed SoS are those in which the integrated system-of-systems is built and managed to fulfill specific purposes. It is centrally managed during long-term operation to continue to fulfill those purposes as well as any new ones the system owners might wish to address. The component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the central managed purpose. |

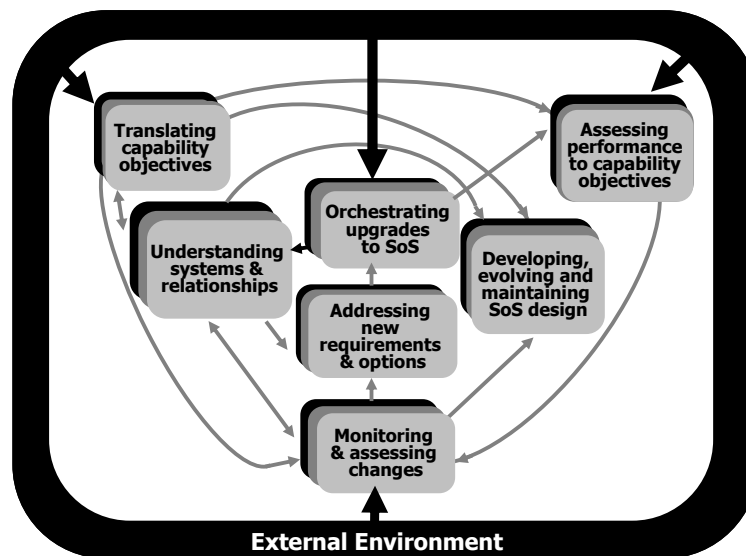
In the DoD a typical strategy for providing end-to-end support for new capability needs is to add functionality to systems already in the inventory. In most cases these systems need to continue to meet their original requirements. Consequently, the ownership or management of these systems remains unchanged and they continue to evolve based on their own development and requirements processes and independent funding. The resulting dual levels (shown in Table 2) of management, objectives, and funding create management challenges for both the SoS and the systems, especially when their objectives are not well aligned. These management

challenges in turn pose technical challenges for the systems engineers, especially at the SoS level.

**Table 2: Comparison of Systems and Systems of Systems [1]**

| Aspect of Environment                          | System                                                                                        | Acknowledged System of Systems                                                                                                                                                                                                                                                             |
|------------------------------------------------|-----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Management &amp; Oversight</b>              |                                                                                               |                                                                                                                                                                                                                                                                                            |
| <b>Stakeholder Involvement</b>                 | Clearer set of stakeholders                                                                   | Stakeholders at both system level and SoS levels (including the system owners, with competing interests and priorities; in some cases, the system stakeholder has no vested interest in the SoS; all stakeholders may not be recognized.                                                   |
| <b>Governance</b>                              | Aligned PM and funding                                                                        | Added levels of complexity due to management and funding for both the SoS and individual systems; SoS does not have authority over all the systems.                                                                                                                                        |
| <b>Operational Environment</b>                 |                                                                                               |                                                                                                                                                                                                                                                                                            |
| <b>Operational Focus</b>                       | Designed and developed to meet operational objectives                                         | Called upon to meet a set of operational objectives using systems whose objectives may or may not align with the SoS objectives.                                                                                                                                                           |
| <b>Implementation</b>                          |                                                                                               |                                                                                                                                                                                                                                                                                            |
| <b>Acquisition</b>                             | Aligned to ACAT Milestones, documented requirements, SE with a Systems Engineering Plan (SEP) | Added complexity due to multiple system lifecycles across acquisition programs, involving legacy systems, developmental systems, new developments, and technology insertion; Typically have stated capability objectives upfront which may need to be translated into formal requirements. |
| <b>Test &amp; Evaluation</b>                   | Test and evaluation of the system is generally possible                                       | Testing is more challenging due to the difficulty of synchronizing across multiple systems' life cycles; given the complexity of all the moving parts and potential for unintended consequences.                                                                                           |
| <b>Engineering &amp; Design Considerations</b> |                                                                                               |                                                                                                                                                                                                                                                                                            |
| <b>Boundaries and Interfaces</b>               | Focuses on boundaries and interfaces for the single system                                    | Focus on identifying the systems that contribute to the SoS objectives and enabling the flow of data, control and functionality across the SoS while balancing needs of the systems.                                                                                                       |
| <b>Performance &amp; Behavior</b>              | Performance of the system to meet specified objectives                                        | Performance across the SoS that satisfies SoS user capability needs while balancing needs of the systems.                                                                                                                                                                                  |

These differences mean that the nature and role of the systems engineer for an SoS takes on different characteristics to address the broader issues across the SoS. There are seven core elements (shown in Figure 1) that characterize SE for SoS. In SoS SE, systems engineers are key players in: (1) translating SoS capability objectives into SoS requirements, (2) assessing the extent to which these capability objectives are being addressed, and (3) monitoring and assessing the impact of external changes on the SoS. Central to SoS SE is: (4) understanding the systems that contribute to the SoS and their relationships, and (5) developing an architecture for the SoS that acts as a persistent framework for (6) evaluating SoS requirements and solution options. Finally, the SoS systems engineer (7) orchestrates enhancements to the SoS, monitoring and integrating changes made in the systems to improve the performance of the SoS. These core elements and their relationships characterize the top level coordinating and integrating role for systems engineering in SoS which is implemented cyclically following a ‘battle rhythm’ driven by the nature of the SoS.



**Figure 1: Core Elements of Systems Engineering for SoS [1]**

Finally, there are a number of crosscutting approaches that appear to be well suited to SE in this environment. [1]

- (1) It is important for SoS SE to address organizational as well as technical issues in making SE trades and decisions.
- (2) SoS systems engineers need to acknowledge the different roles of and relationships between systems engineering done at the systems and SoS levels. It is important for systems engineers of SoS to focus on those areas critical to SoS success and to leave issues related to the constituent systems to their systems engineers.
- (3) Technical management of the SoS needs to carefully consider the level of participation required of the system-level SEs. Enabled by transparency and trust, their active participation should be focused on areas specifically related to the systems and the SoS.
- (4) There is a real advantage to an SoS design based on open systems and loose coupling which impinges on the systems as little as possible, thus providing systems maximum flexibility to address changing needs and technology opportunities for their users.
- (5) Finally, SoS design strategy and trade studies need to begin early and continue throughout the SoS evolution, which is an ongoing process.

### **Net-Centric Enterprise Systems**

Beyond systems and SoS, there is increasing focus NCE systems. Their importance is a consequence of the government leveraging the opportunities provided by commercial network technology to support defense and other information needs and make better use of information across enterprises. An NCE system is an IT system comprised of a set of services available over the network typically using a Service Oriented Architecture (SOA).

In NCE systems, services may be created and maintained by different organizations, added or changed for a variety of reasons, and applied by different users in different ways (different sequence) to yield potentially different results. As is described in the OASIS SOA reference model [6]:

—...SOA-based system is a network of independent services, machines, the people who operate, affect, use, and govern those services as well as the suppliers of equipment and personnel to these people and services. This includes any entity, animate or inanimate, that may affect or be affected by the system. With a system that large, it is clear that nobody is really "in control" or "in charge" of the whole ecosystem; although there are definite stakeholders involved, each of whom has some control and influence over the community.”

Key SOA elements [7] are:

- A software architecture where functionality is grouped around business processes and packaged as interoperable services.
- An IT infrastructure which allows different applications to exchange data with one another as they participate in business processes. The aim is a loose coupling of services with operating systems, programming languages and other technologies which underlie applications.
- A separation of functions into distinct units, or services, which are made accessible over a network in order that they can be combined and reused in the production of business applications. These services communicate with each other by passing data from one service to another, or by coordinating an activity between two or more services.
- Concepts that are often seen as built upon, and evolving from older concepts of distributed computing and modular programming...”

Based on the above technical and management characteristics, NCE systems exhibit differences from traditional systems. Many of these differences parallel the SoS/System differences, including component (Service/Agency) interdependencies, multiple funding and budgeting sources, and evolving user requirements that may not be known or forecasted at the start. Others may go beyond SoS/system differences, including incentives to provide services or efficiently use services developed by others, and funding approaches to meet potentially unpredictable demand for popular services.

Some current internet-based systems (e.g. Google) are examples of NCE systems. NCE systems are in use by US intelligence agencies for a number of different purposes and have proven to be a flexible, extensible way to rapidly create and evolve system functionality to meet user needs which change on a regular basis. In the DoD, moving to NCE systems is seen as an objective, particularly for command and control (C2) systems. For example, the objective of the Net-Centric Enterprise Services program is to develop and field core services which can be applied in a wide variety of application areas across a wide range of users. Another example, the Net-Enabled Command and Control system (NECC) is intended to replace the current Global Command and Control Systems (GCCS) family of systems with a set of services which can be accessed and employed by GCCS users to implement the activities now done using the GCCS family of systems. The benefit of the change is in the added flexibility, extensibility, and maintainability that the services based approach provides over the current systems.

The systems engineer of an NCE system is responsible for defining the protocols and standards which are needed for users to access and use the services. This is often described as ‘enterprise’ engineering and is defined as “cyclic process of managing the evolution and interconnection of the enterprise”. [8] The systems engineer of an NCE system is responsible for a set of agreed upon specifications which ensure users in their community can access and employ relevant services. As with SoS, they analyze service functionality and performance to address evolving user needs. They may also work with the NCE system manager to develop Service

Level Agreements (SLAs) with service providers to ensure the services meet the needs of the NCE systems users.

## **SoS and NCE Systems – Comparison and Implications**

How do NCE systems compare to SoS?

First, there are some strong similarities particularly in the challenges they face in development and operation. This is because they both operate in a similar management context. In many cases neither the SoS nor the NCE system authority control the components that are core to their success. Many NCE systems share the characteristics of acknowledged SoS in that NCE services may be developed and implemented by different organizations for different purposes and these organizations may evolve these services over time to address both needs and opportunities. In fact SOA has evolved as a technical approach to address just these issues in an IT environment. As described by Altman, SOA is best viewed as “about cross-application development.” [9]

From this perspective, NCE systems can be viewed as an example of an SoS, particularly for IT-based systems or SoS linked by distributed software. SOA can be viewed as an approach to architecting information based SoS which meets the characteristics of “an SoS design based on open systems and loose coupling which impinges on the systems as little as possible, providing systems maximum flexibility to address changing needs and technology opportunities for their users.” [1] As previously noted, this is a characteristic of successful approaches to SoS architectures.

Ironically for NCE systems, the SOA architecture approach is best suited to integration across heterogeneous distributed systems (‘architecture of architectures’); however, the governance that accompanies such situations in the DoD are those which have the biggest problems with cross-systems governance.

Governance and management issues for SoS and NCE systems are a driver for the technical considerations in systems engineering for both SoS and NCE systems. The implications for systems engineering in both cases is that the systems engineer in SoS and NCE systems plays a broader role. They work across components to define the top level capability needs. They coordinate the development and evolution of the components in the context of a common architecture while considering the range of issues beyond technical (e.g. political, resource) which impact the prospects for success and progress toward broader enterprise capability objectives.

The biggest difference between SoS and NCE systems is that the services in a fully implemented NCE system are not independent systems as traditionally defined. Ideally, an enterprise SOA implementation is developed by a portfolio of projects involving both distributed applications and services. These services are often developed or acquired in parallel, and they need to be woven together to form a shared fabric of integrated and consolidated distributed computing rather than individually developed as disparate, stand-alone applications and their components. They are services implemented in software available on the network to be accessed

on an as-needed basis. Although NCE systems may begin as a set of independent systems, with a fully implemented SOA, the originally independent systems become part of a larger set of interdependent components. In effect, they no longer operate on their own, and are no longer an SoS.

## **Acquisition Challenges and Approaches**

The ‘acquisition’ of systems of systems is somewhat a misnomer since in SoS most of the functionality is already available in fielded systems (which have already been ‘acquired’) or in systems which are themselves in acquisition. The SoS manager and systems engineer work with the owners of the constituent systems to evolve these systems to meet capability needs of the SoS. The current DoD acquisition system is designed for the creation or upgrade of individual systems and the major acquisition milestones and processes are not well matched to the cyclic nature of SoS evolution. Often there is an organization outside of the acquisition process which looks across the SoS and levies requirements on systems (which may be in acquisition or in maintenance) to meet the needs of the SoS. In a number of cases, when the investment needed for the SoS is large, an acquisition program has been formed to address these SoS needs, but typically the acquisition program focuses on the new components or major system upgrades needed for the SoS rather than the SoS as a composite enterprise.

The Single Integrated Air Picture (SIAP) is an example. Its objective is to provide a capability for all users to have a common and accurate air picture by ensuring the information shared on friendly and enemy air objects is exchanged and managed so a coherent cross platform picture is available. The participating systems include sensor and weapons platforms and systems on the ground and in the air. The SIAP acquisition program focuses on the development of core software needed for all participants in the SIAP and which will be instantiated in the SIAP constituent systems which are acquisition programs themselves. External oversight mechanisms have been established to work across program boundaries to develop and deliver SIAP capability.

As with SoS, the NCE systems have faced challenges in DoD acquisition, largely for the same reasons. As the executive summary of a recent AFEI project [8] states

–The challenges to this [NCE] migration are significant: a DoD acquisition system that is predicated on a high degree of independence between systems; an ‘a priori’ lack of clearly defined system interfaces; an emphasis on development rather than operations; and a lack of incentives for system owners to provide capabilities to any user beyond their predefined base. The DoD must overcome these challenges to achieve the Service Oriented Architecture (SOA)-based environment it desires in the future.”

NCE, like SoS are based on incremental changes to current systems and development of new services to support larger enterprise capability needs. This “overlay” across the current procedures for acquiring systems is different in kind, more continuous than phased, and more evolutionary in terms of requirements and delivery of capability.

The experience of systems like Google has been offered as a model for how to develop and field NCE systems. Understanding their approach makes clear some of the stresses of the current

DoD acquisition processes as applied to NCE systems. These commercial systems do not begin with requirements. Rather, they identify opportunities and create services which are offered to users. Those that have value are used and sustained. In this way the NCE system evolves over time based on interaction with users. This approach differs markedly from the current defense acquisition process. The Net Enabled Command and Control (NECC) System for example has been presented as a replacement for the current Global Command and Control System (GCCS) which is a currently fielded family of systems. This means that there are a set of base NECC requirements which need to be addressed in the delivery of the system that cannot be addressed by a ‘Google-like’ development approach. Further, the basic services needed as the foundation for NECC application-based services have not yet been developed and fielded as common services available across defense environments. Without these basic services, different organizations have developed their own renditions and hence multiple variations of the basic services are being developed that are not commonly available across the enterprise.

A number of the SoS programs studied in the development of the SoS SE Guide have taken an incremental approach to achieving capability needs. This allows for less disruption to existing legacy components, nearer-term payoff or delivery of initial capability, and a more successful ultimate capability implementation. It turns requirements evolution to an advantage by using it as a way of acquiring knowledge over time. The initial step in this incremental approach is to federate the existing capability delivery systems to meet the broader capability needs. Trades consider the impact to the existing systems while attempting to maximize the synergy gained through their federation. At the same time the individual system programs are implementing a long-term approach to migrate to a NCE system. An example of this approach is the AF Air Operation Center (AOC) program. It began by applying a federated architecture leaving original systems ‘intact’ and sharing data to meet common needs with the plan to ‘migrate’ to a SOA NCE architecture. Over time this approach expands the degree to which the systems work cooperatively and their ability to share with add user/systems.

## Summary and Challenges

In summary, we find many common characteristics and common challenges between NCE and SoS. From an organization and governance perspective:

- Neither owns or controls their components
- Capabilities or services are provided by different owners
- Because capability needs are not fully known, funding and budget estimates are not knowable to meet the final scope of requirements

From a SE perspective:

- Engineers must play a much broader role, across multiple systems
- SoS and NCE engineers must focus on cross-cutting capability needs
- Engineers play a key role in negotiating and coordinating these needs and their solutions over time, establishing an architectural approach that allows them to do so.

SoA offers a technical approach to address some of the organizational and governance issues for the IT domain, and respond to the loosely coupled architecting needs of a SoS.



Understanding the technical and management characteristics of complex SoS and NCE systems is key to developing approaches to effectively deliver needed capabilities in today's defense acquisition environment. Working across communities to identify challenges and explore possible approaches is critical to providing SE support to the development of user capabilities and effective ways to apply defense acquisition policy to leverage advantages of commercial technologies in military system acquisition.

## References

- [1] Office of the Under Secretary of Defense for Acquisition, Technology and Logistics (OUSD AT&L), 2008, Systems Engineering Guide for Systems of Systems, , Washington, DC: Pentagon, (2008).
- [2] Dahmann, Judith and Kristen Baldwin, 2008, –Understanding the Current State of US Defense Systems of Systems and the Implications for Systems Engineering”, Montreal, Canada: IEEE Systems Conference, 7-10 April.
- [3] Dahmann, Judith, Kristen Baldwin, JoAnn Lane and George Rebovich –A Model of Systems Engineering in a System of Systems Context”, 2008 Conference on Systems Engineering Research, Redondo Beach, CA, April 2008
- [4] Maier, M. (1998); "Architecting Principles for Systems-of-Systems"; Systems Engineering, Vol. 1, No. 4 (pp 267-284).
- [5] Dahmann, Judith, JoAnn Lane and George Rebovich, –Systems Engineering for System of Systems Capabilities”, November 2008 Crosstalk
- [6] Organization for the Advancement of Structured Information Standards (OASIS) , –Reference Model for Service Oriented Architecture 1.0”, Committee Specification 1, 2 August 2006, found at [www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf](http://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf)
- [7] Organization for the Advancement of Structured Information Standards (OASIS) , –Service Oriented Architecture (SOA)”, Technology Reports, last modified 15 October 2008, found at <http://xml.coverpages.org/soa.html>
- [8] Association for Enterprise Integration –Industry Recommendations for DoD Acquisition of Information Services and SOA Systems”, SOA Acquisition Working Group, July 7, 2008
- [9] Altman, Ross (2009), –SOA - Questioning Some Assumptions”, Sun Microsystems Object Management Group Document found at [www.omg.org/docs/soa-c/08-12-01.pdf](http://www.omg.org/docs/soa-c/08-12-01.pdf)

## Biographies

**Dr. Judith Dahmann** is a Senior Principal Scientist in the MITRE Corporation Center for Acquisition and Systems Analysis supporting the Director of Systems and Software Engineering US DOD Under Secretary of Defense for Acquisition, Technology and Logistics. Prior to this, Dr. Dahmann was the Chief Scientist for the Defense Modeling and Simulation Office for the US Director of Defense Research and Engineering (1995-2000) where she led the development of the High Level Architecture, a general-purpose distributed software architecture for simulations, now an IEEE Standard (IEEE 1516). Dr. Dahmann holds a Bachelor's Degree from Chatham College in Pittsburgh, PA with a year as a special student at Dartmouth College, a Master's Degree from The University of Chicago and a Doctorate from Johns Hopkins University.

**Kristen J. Baldwin** is the Deputy Director for Strategic Initiatives for the Director for Systems Engineering in the Office of the Deputy Under Secretary of Defense for Acquisition and Technology. Her responsibilities span both systems engineering and systems integration. She leads the application of capabilities-based planning in the acquisition process, focusing on the integration of requirements, acquisition, and programming. Ms. Baldwin's responsibilities include Systems of Systems (SoS) Systems Engineering guidance, CMMI oversight, and co-chair of the DoD Software Assurance Initiative. Ms. Baldwin received a BS in Mechanical Engineering from Virginia Tech and an MS in Systems Management from Florida Institute of Technology.

**George Rebovich** is a Senior Principal Engineer at The MITRE Corporation. He holds a B.S. degree (Mathematics) from the United States Military Academy (USMA), an M.S. (Mathematics) from Rensselaer Polytechnic Institute, a certificate of Administration and Management from Harvard University and is a graduate of the U.S. Army Command and General Staff College. He has held various systems engineering and management positions at MITRE. He served in the U.S. Army including tours of duty in the U.S. and Europe, and as an artillery forward observer with the 101<sup>st</sup> Airborne Division in Vietnam. He is a former Assistant Professor of Mathematics at USMA.