Addressing the System of Systems Challenge

Dr. John Boardman
Elipsis, Inc.
www.elipsis.com

Mark A. Wilson
Strategy Bridge International, Inc.
www.strategybridgeintl.com

Allen Fairbairn
John Boardman Associates, Ltd.
www.jba.net

Abstract. The top-down systems engineering strategies that worked well in the 20th century do not adequately address the complexity and interdependencies of today's higher-order systems that often must operate seamlessly with huge and disparate legacy systems. Soft management issues, such as organizational culture, also play a vital role in the organization's system of systems, often called the extended enterprise. Strategy must be formulated—and appropriate deployment mechanisms designed—for the system of systems level.

System of Systems (SoS) has become a popular buzz word among systems engineering practitioners, but what does it really mean? Some professional literature traces the origins of SoS to problems with systems that "nobody owns,"[1] while the U.S. Department of Defense has recognized that Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) systems must be seamlessly integrated with weapons systems in order to achieve optimal SoS effects. This paper offers an explanation of key SoS concepts and proposes some useful practices that the authors have found useful to address the complexities of large-scale integration of SoS assets across the extended enterprise (EE).

Warning Notice

This paper is a statement of the authors' beliefs and has not proceeded out of a rigorous series of hypotheses generation, testing and rework. We are not providing case study material, and we are not rehearsing the labors of years of research. We are, however, attempting to provide a thinking framework to address engineering challenges for which the established SE protocols may, at their very heart, be inadequate. We do not agree with those who contend that extending basic SE

---

[1] R. Schumann. *Developing an Architecture that No One Owns: The US Approach to System Architecture,* Proceedings First World Congress Appl. Transport Telematics Intelligent Vehicle-Highway Systems, Paris, France, 1994.

methods and techniques to SoS engineering is a largely trivial exercise. We believe that there is more to SoS engineering than meets the eye—given that what the eye meets is largely subjective…and therein lies the problem.

What is a System of Systems?

As a group, systems engineers are not especially known for their ability to converge on a single definition of a given concept, but it is useful for our purposes to adopt a working definition of a SoS. The Department of Defense has adopted the following definition:

> *System of systems engineering deals with planning, analyzing, organizing, and integrating the capabilities of a mix of existing and new systems into a system of systems capability greater than the sum of the capabilities of the constituent parts.*[2]

In an attempt to clarify the difference between a generic "system of interest" and a "system of systems," we propose the following definition:

(1) A SoS consists of an assemblage of systems, each system capable of – and usually practicing – a separate, independent existence;
(2) The individual component systems are self-sustaining and purposeful *without* the SoS;
(3) The systems comprising the SoS can be connected and reconnected to produce different, and often unanticipated, effects; and
(4) The SoS exhibits the property of "emergence," which means that there are properties characteristic of the SoS that are irreducible and not evident in the component systems.

For example, we would classify the Internet as a SoS since it meets all four elements of our proposed definition. Specifically, the individual computers and routers that comprise the World Wide Web can operate quite successfully in a "stand alone" fashion, yet when connected together create various effects not evident in the component systems. Another example of a SoS is the North American air traffic control system, whose radar systems, airport systems, and airplane systems can operate effectively either as members of the SoS or independent of one another (not that we are suggesting that mode of operation!). On the other hand, the gas turbine engine on a B-747 is *not* a SoS, since it does not function independent of the airplane system of which it is an essential part. Similarly, the human heart, while a remarkable system in its own right, cannot effectively function independent of the human body system.

Whereas classic systems engineering focuses primarily on the realization of a product system, SoS ventures further in addressing overall SoS context from a <u>holistic</u> perspective, including all relevant "people" systems (and not just users in the loops of product systems). When we use the term "Extended Enterprise" (EE), we mean the context for the complete set of activities undertaken by companies operating together in a given market environment to deliver products

[2] DoD Acquisition Deskbook on line, November 2004

and/or services to customers and users.[3] We believe that systems engineers must evolve "state of the practice" thinking and analytic techniques in order to address these SoS challenges from this holistic perspective.

What Makes a System of Systems Different?

Problems cannot be solved at the same level of awareness that created them. Albert Einstein

One SoS attribute that demands we address SoS problems differently is the property of *emergence*. Since emergent properties are, by definition, evident only at the SoS level and not in the component systems, the classic systems engineering techniques of functional decomposition and requirements allocation have been proven inadequate to address challenges related to changing or adding component system elements to a SoS. The implications of this problem are readily apparent to the U.S. Department of Defense, which is grappling with leveraging the huge investments that have been made in legacy component systems while seeking to improve the overall C4ISR SoS.

A second challenge in addressing a SoS is its inherently *abstract* nature. Often, the SoS only exists in concept, and not in the physical world. Stated another way, the SoS does not exist in "reality," but only in "ideality." This presents a particular challenge for engineers grounded in the hardware disciplines. Since we often struggle to visualize the SoS, it can be difficult to establish a common mechanism for SoS stakeholder communication. Moreover, it can sometimes be difficult to even establish a quorum of stakeholders willing to transcend the physical domain in order to establish that dialog in the conceptual or abstract domain, much less identify stakeholders with an overarching knowledge of the SoS in question.

A third challenge in addressing systems of systems is the lack of *ownership* at the SoS level. While "owners" of the component systems can be readily identified, who has ownership of the overall SoS, particularly when the SoS transcends organizational boundaries? Recall our definition of the concept of "extended enterprise" (EE), which requires that we address the organization's entire value chain, some components of which transcend the boundaries of what we normally think are internal to that organization. In other words, we must consider all constituent systems, workers, managers, executives, business partners, suppliers, and customers. The extended enterprise can be optimized *if* all of the constituent systems, groups and individuals are dedicated to optimization of the overall EE system of systems. Of course, self-interest is usually intrinsically opposed to individual element sub-optimization, making overall EE/SoS optimization very difficult in practice.

We assert that uneven incentives for the component elements to fully engage and participate in optimizing the SoS, at the expense of sub-optimizing at the component level, often results in limited "buy-in" to the overarching SoS by the component elements. The *National Commission on Terrorist Attacks Upon the United States*, popularly known as the *9/11 Commission*, addressed this very problem in the Intelligence Community. One of their recommendations was to establish the position of Director of National Intelligence as an overall "Intelligence Czar"

[3] *Enterprise Systemics: Systems Thinking for Plotting Strategy at the Extended Enterprise Level,* by Allen Fairbairn and Andrew Farncombe, INCOSE 2001 International Conference, Melbourne, 2001.

with ownership of the entire U.S. Intelligence Community SoS, the implementation of which is still evolving with the current Administration.

Will the Director of National Intelligence (DNI) truly "own" the entirety of the U.S. intelligence system of systems? Is that even possible, given the myriad of competing agendas among the intelligence EE? For example, will the DNI "own" the Special Operations Forces (SOF) in the Department of Defense who most certainly are part of the intelligence EE? Can the DNI direct these SOF to suboptimize in the interest of optimizing the larger intelligence SoS, perhaps at the expense of the DoD SoS of which the SOF are *also* a part? Can the DNI direct Defense contractor R&D investments and/or processes that are essential to the operation of some C4ISR technologies? Some cynical pundits have suggested that the only thing the Administration and Congress have achieved with the DNI position is to establish yet another layer of bureaucracy in the executive branch—along with more a lot more opportunity for federal contractors to sell services to the government.

If one accepts our premise that today's truly complex problems cannot be dealt with in the traditional "reductionist" methods for which systems engineers are noted, how then can one grapple with SoS issues when emergent properties are evident at various levels of complexity? Building on the work of Peter Checkland and other pioneers in the field of systems thinking and cybernetics, we propose six practices that we have found useful in analyzing and addressing SoS challenges.

Six Useful Practices in Addressing System of Systems Challenges

Practice #1: The Boardman Soft Systems Methodology.

The Boardman Soft Systems Methodology (BSSM) technique, a derivative of the seminal work by Peter Checkland, is a meta-process for examining the essential actors, actions and relationships in an extended enterprise. The technique uses concept mapping to construct conceptual SoS models that can be especially useful in assisting systems engineers visualize and grapple with SoS complexity, including legacy component elements.

The technique uses seven steps:

Step 1: Develop an understanding of the problem situation.
Step 2: Express the problem statement.
Step 3: Describe the overall system components using structured text.
Step 4: Translate the structured text into a SystemiGram™ design.
Step 5: Establish a dialog with the SoS stakeholders.
Step 6: Explore options for feasible changes to the SoS within the context of the current SoS.
Step 7: Take action appropriate to the problem statement expressed in step 2.

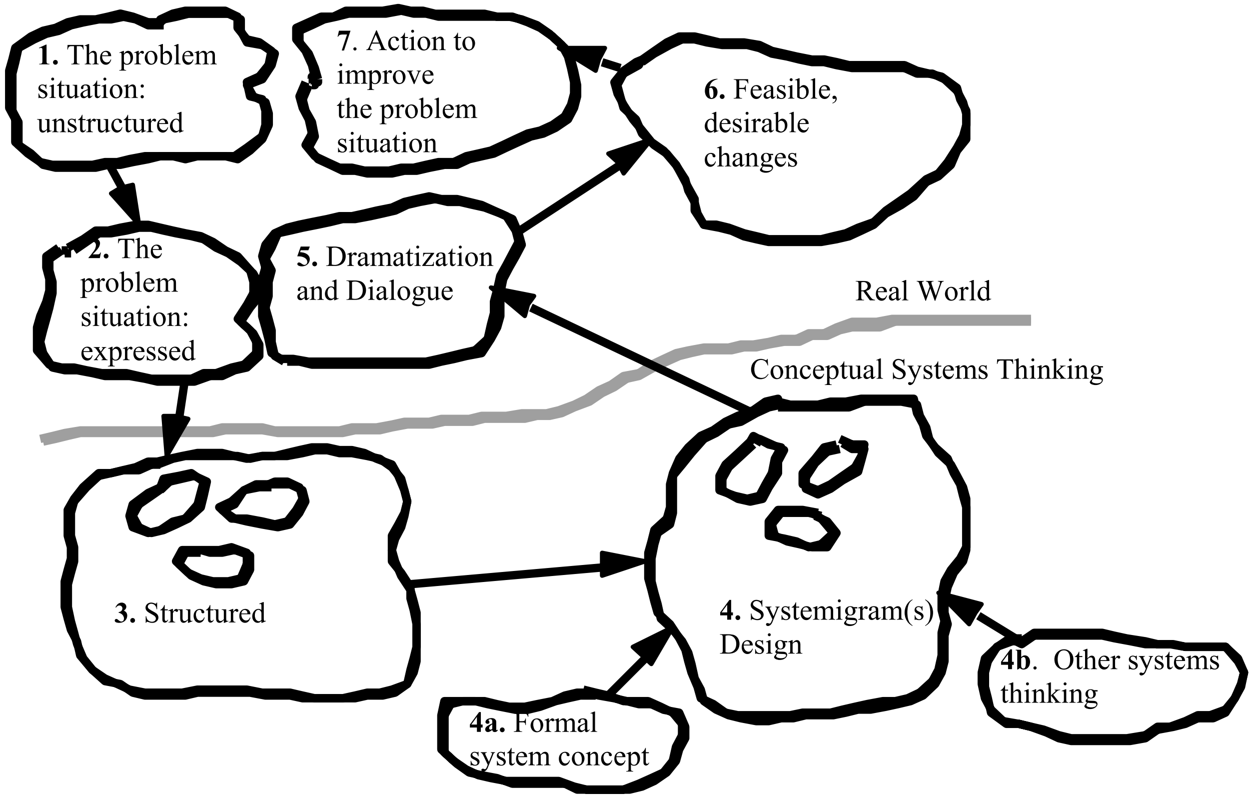

Figure 1. The Boardman Soft Systems Methodology.

Practice #2: Framing.

How one defines the problem will define the available alternatives from which a system can be architected. The "frame" is the overall context for the engineering challenge under consideration. What is the ultimate objective of the solution? What is the root-cause of the issue/problem? Systems engineers sometimes lock onto a particular solution frame without critically examining the overall context and objectives of the solution itself.

The proper solution frame opens up the spectrum of possible solutions, allowing creative possibilities to emerge. Many systems engineers focus on solutions instead of taking the time to step back and really examine the underlying goal(s) of the meta-problem. If the frame is constrained to answering the question "what's wrong," the range of possible solutions is typically constrained by the alterative. The goal should be defined as concept-independent, so that solution alternatives don't begin taking you down a path that may force you into a predetermined architecture.

Our recommended framing technique is to engage in a context-setting dialog with stakeholders.

Open-ended questions that invite alternatives, using verbs such as "how" or "what" helps keep architectural options open and can help systems engineers avoid prematurely driving to an overly narrow set of alternatives. One simple technique is to keep asking the question "why?" until it doesn't make sense anymore. This technique is sometimes referred to as the "Five Whys" tool because often one may ask the question "why?" up to five times before getting to the root context of a problem or issue. Other context-setting questions include:

- What is the ultimate SoS objective?
- What is the crux of the issue or problem?
- How does the system affect other systems?
- What information do we have about similar problems or systems?
- How will we implement the solution?

Practice #3: Abstract Thinking.

The central premise of analytic reductionism (system decomposition) is that the tangible system components are unchanged/not distorted by the fact of decomposition. While this is a generally reasonable assumption for simple systems, the assumption does not hold true for many complex systems and certainly not for systems of systems. Often, the SoS only "exists" in a conceptual frame, rather than in the physical world. In other words, one can generally tactilely "sense" the system components, but the SoS itself is often an abstract concept.

"Abstract Thinking" can be described as the ability to integrate specific information together with our own knowledge and experience in order to draw conclusions at the conceptual level. For example, the United States Navy has declared that FORCEnet is

> *…the operational construct and architectural framework for naval warfare in the information age that integrates warriors, sensors, networks, command and control, platforms, and weapons into a networked, distributed combat force that is scalable across all levels of conflict from seabed to space and sea to land.*[4]

When FORCEnet is briefed by the U.S. Navy, it is generally described as the architecture that enables "Sea Power 21" and network-centric warfare. Though the word description of the FORCEnet concept can be difficult to envision, the picture that accompanies many presentations on FORCEnet may assist some to understand the basic concept, but it doesn't much help the systems engineers move the dialog from the conceptual domain to the systems domain.

[4] Vice Admiral Richard W. Mayo and Vice Admiral John Nathman, *U.S. Navy ForceNet: Turning Information into Power*, *Proceedings*, February 2003.

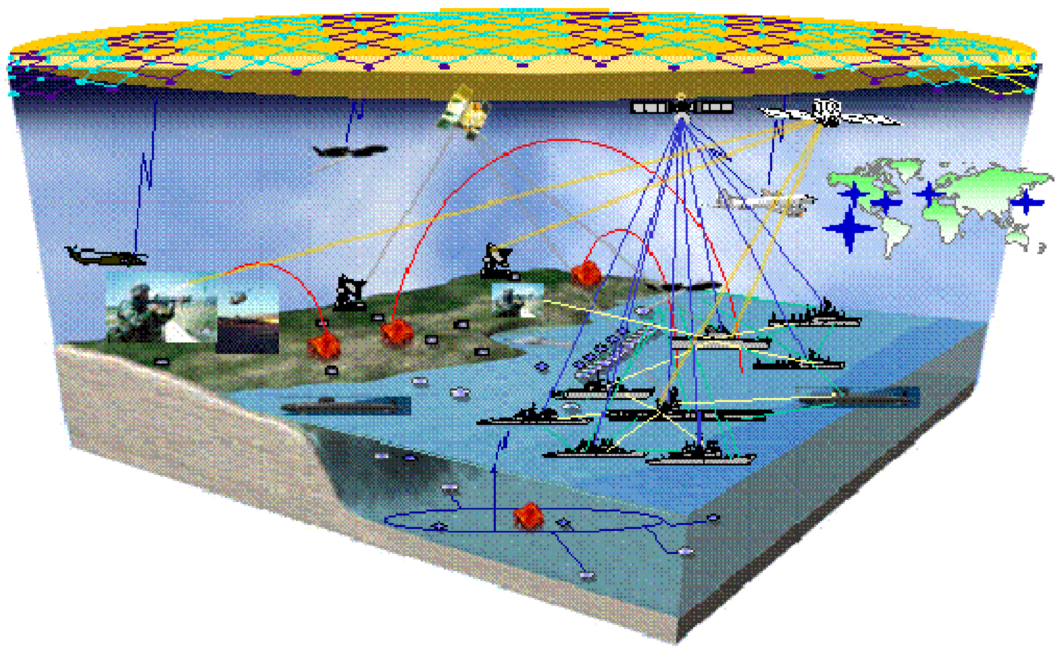

Figure 2. The U. S. Navy's FORCEnet Concept.

Practice #4: The Systemigram™ Technique for Visualizing a System of Systems

We suggest that the SystemiGram™ technique is a useful mechanism to display relationships among the component systems in the context of overall SoS objectives in an easy to digest manner. It allows the systems engineer to visualize the complexity of the SoS by creating a manageable and "manipulable" model of the SoS. At that point, the systems engineer can engage stakeholders in a useful dialog to increase understanding and establish a solid foundation upon which the SoS can be realized. In effect, it can illustrate the concept of operations (CONOPS) at the organizational level, creating the OV-1 artifact using DoD Architecture Framework (DoDAF) terminology.

The SystemiGram™ addresses three aspects of the overall SoS under consideration:

1. Structural design
    a. Condensing factor: making the problem as simple as possible, but no simpler.
2. Dynamic design
    a. Displaying "scenes" that describe the SoS relationships among component elements.
3. Dialog
    a. Telling the story in a way that enables stakeholders to engage in the solution development. In other words, the interactive dynamics that can lead to restructure of the model.

Some useful guidelines for creating a Systemigram™ include:

- Remember that the model is not reality. Rather it's an insightful commentary on reality that serves the purpose of shaping future reality with greater effect.
- Ensure key sentences are recovered in specific scenes (*fidelity*).
- Use background to link previous sentence structures where this helps current scene (*emphasis*).
- Attempt a storyboard effect so that each scene leads naturally to successors thereby conveying added meaning to the 'drama' (*insight*).
- Look for new aspects which the whole might suggest that would not be obvious from prose (*value-added*).

Let's examine how the Systemigram™ technique might be used to display the complexity associated with the U.S. Air Force SoS strategy for "NetCentric" warfare. Drawing from some recent open source documents describing the USAF C2 Constellation Program, we can translate the following condensed version into a Systemigram™:

…the USAF Combat Strategy is to develop Constellation Net in support of Air Force ConOps and Joint Vision 2020 that will lead to Joint Force Victory. The strategy must address information paralysis caused by stove piped networks that cause tribal warfare resulting in data-overloaded warriors, who operate in a camouflaged information environment and lengthy kill-chains which debilitates combat capability needed to secure Joint Force Victory. Strategy seeks to achieve predictable battle space awareness (PBA) and effects-based operations (EBO) that are required by USAF ConOps and Joint Vision 2020, by developing horizontal integration to overcome stove piped networks. Strategy will adopt a staged process by means of (#1) defining network architectures for stove piped networks and by developing technology and standards for information sharing across stove piped networks and hence defining a network-of-network architecture; (#2) achieving current system compliance with the network-of-network architecture; and, (#3) acquiring next generation systems that will leverage off the network-of-network architecture by possessing in-built battle management peer-to-peer capability which achieves horizontal integration.

Applying our guidelines, we derive the following picture of the SoS:

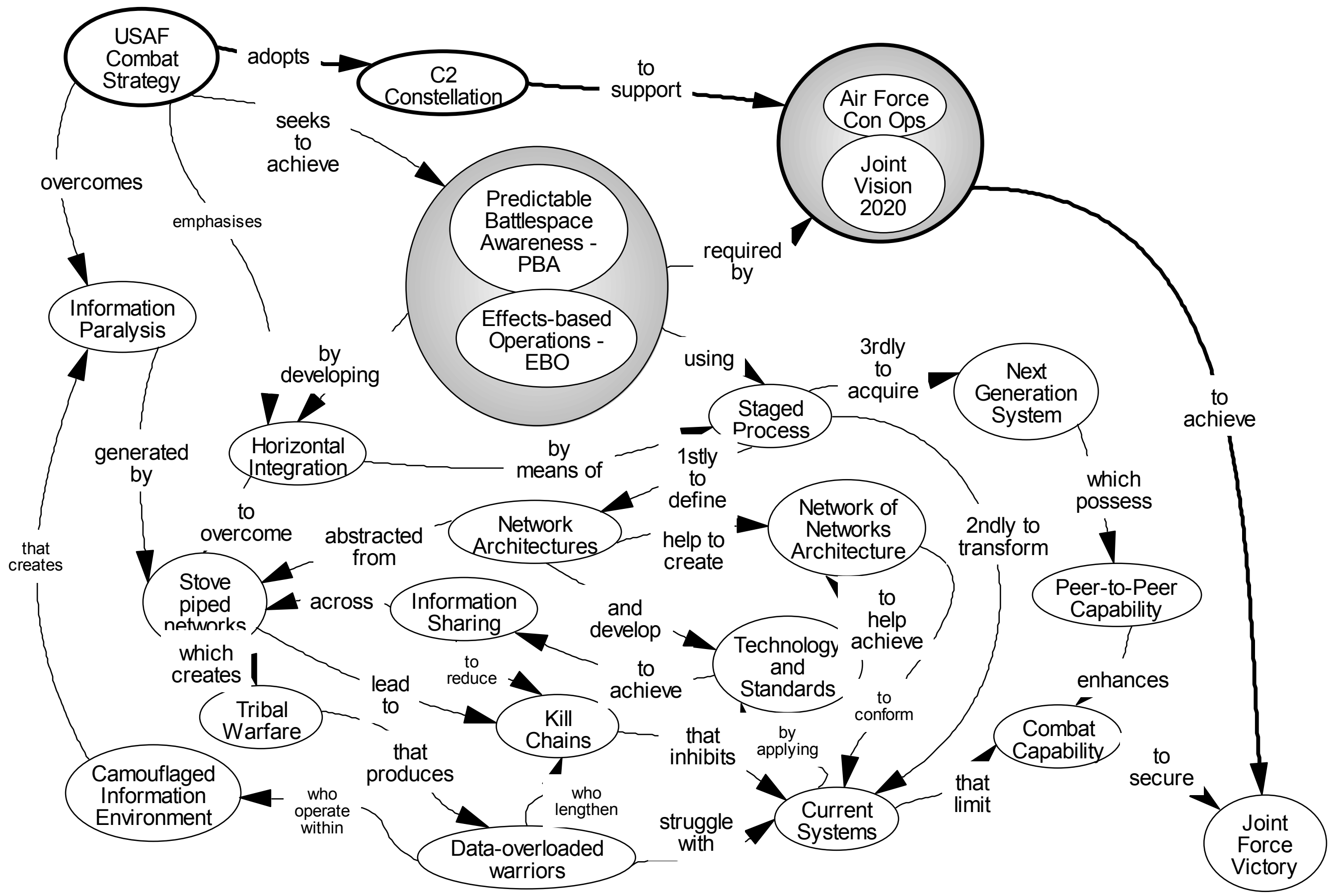

Figure 3. A View of the U.S. Air Force Strategy for NetCentric Warfare.

Practice #5: Scenario Planning.

Scenario planning is a creativity technique that uses the power of imagination and analogy as a way to stimulate creativity and abstract thought. First popularized by Peter Schwartz in his book, *The Art of the Long View: Planning for the Future in an Uncertain World* (Bantam Doubleday Dell, 1995), the technique helps systems engineers articulate their ideas about alternative future environments. The alternate scenarios are developed as a set of "stories" to organize explicit and tacit knowledge—as well as assumptions—regarding envisioned SoS employment and performance.

Coupled with the Systemigram™ technique for visualizing the SoS, scenario planning can be effective for examining real technology and operational options by considering potential interactions among component elements in various future contexts. The future scenarios can foster a common way of communicating about complex SoS interactions, often leading to better, more disciplined development processes.

In practice, scenario developers should be empowered to question current assumptions and practices in thinking about alternative, plausible futures. The goal of scenario planning is not a more accurate picture of future, but a common mechanism for an ongoing strategic conversation about the future. The very act of thinking about plausible futures enables systems architects and managers to be sensitive to changing conditions, thereby enabling them act more quickly—and hopefully more effectively. Interestingly, the *9/11 Commission* reported that a contributing factor to the breakdown in the Intelligence Community SoS leading to the terrorist attacks on the U.S. homeland was a "failure of imagination."[5]

Practice #6: Process Integration

We contend that an analytical/reductionist approach for complex systems of systems has very serious limitations. Not only is this due to the stubborn refusal of prescribed parts to remain stable in their partitioned state and not to interact in unpredictable ways with other parts, but also to the essentially abstract nature of a prescribed overall starting point. Moreover, partitioning based on an abstraction can hardly lead, in itself, to concrete parts, so any certainty or physical attributes attached to the alleged simplification of dealing with a collection of deduced parts, rather than a complex whole, seems to involve a leap of faith at best.

The vast majority of projects are not green field, of the sort that requires an overall abstraction as a starting point. If we are considering a SoS that is an incremental change from an existing SoS (at least 85% reuse from what has already been built), then starting with concrete knowledge at a level where the proposed SoS is comprehensively understood ought to be more effective than an overall "clean sheet" abstraction. Whether the emphasis is legacy, COTS, or System of Systems, the leverage can be off something already known.

"Process" is a useful means of accumulation – in the sense that with process, one thing leads to another until a desired result is obtained. So, if we seek to build up a system from known parts through an integration of process elements, we may have a way not only of putting an effective system together but also developing a time and resource based plan for its achievement. If we add to this notion of process-based planning, the possibility of representing the process integration process itself, so that knowledge may be shared, improvement ideas tested, and activities coordinated, we may have the means of producing a viable, perhaps even optimal, system solution.

Take, for example, the introduction of a new market of a given product – an improved, more advanced, next stage of development engine, handbag, radio set - whatever. While the requirements for the yet to be realized product will have to be abstracted in some form, in order to be shared throughout participating members of the EE, it would be a mistake then to attempt to share out the time, cost or development risk in a prescriptive manner. First, no one person or organization can effectively drive an EE, so the command and control instinct must always be suppressed. In the second place, on what reasoned basis can the overall task be so divided?

[5] *The 9-11 Commission Report: Final Report of the National Commission on Terrorist Attacks Upon the United States*, Official Government Edition, 2004, p. 339.

Better by far to start with a statement of the overall aim say, of reducing time to market by 25%, and then looking at what is known in detail about a "close enough" solution for improvements, which need not necessarily be essentially marginal. Artifacts modeled as detailed process elements can then be compared, coordinated, modified and integrated into a process based plan of implementation, with corporate boundaries crisscrossed as necessary by the requirements of a fully integrated activity set.

Summary

As systems continue to grow in complexity, we suggest that the top-down systems engineering strategies that worked well in the 20th century will not adequately address the complexity and interdependencies of 21st century systems of systems. Whereas classic systems engineering has been primarily focused on the realization of a product system, we believe that systems engineers must evolve state of the practice thinking and analytic techniques in order to address these SoS challenges from a holistic perspective.

REFERENCES

Boardman, John, *Wholes and Parts - A Soft Systems Approach*, (1995) IEEE Transactions: Systems Man and Cybernetics, Vol. 25, No. 7, July 1995.

Boardman, John (with AJ Cole), *Integrated Process Improvement In Design And Manufacture Using A Systems Approach*, (1996), IEE Proceedings Part D, Vol. 143, No. 2, March 1996, pp 171-185.

Boardman, John (with BT Clegg), *Process Integration And Improvement Using Systemic Diagrams And A Human-Centred Approach*, (1996), Concurrent Engineering: Research and Applications, Vol. 4, Number 2, June 1996. ISSN 1063 293X . pp 119-136

Boardman, John (with DA Ramsay, and AJ Cole), *Reinforcing Learning Using Soft Systemic Frameworks*, (1996), International Journal of Project Management, Volume 14, Number 1, February 1996, pp 31-36.

Boardman, John (with DG Sherman and AJ Cole), *Assisting Cultural Reform In A Project-Based Company Using Systemigrams*, (1996), International Journal of Project Management. Volume 14, Number 1, February 1996, pp 23-30.

Boardman, John (with JS Sagoo), *Towards The Formalisation Of Soft Systems Models Using Petri Net Theory*, (1998), IEE Proc. - Control Theory Appl., Vol. 145, No. 5, September 1998, pp 463 - 471.

Checkland, Peter, *Systems Thinking, Systems Practice,* John Wiley and Sons, 1993.

Fairbairn, Allen and Andrew Farncombe, *Enterprise Systemics: Systems Thinking for Plotting Strategy at the Extended Enterprise Level,* INCOSE 2001 International Conference, Melbourne, 2001.

Mayo, Vice Admiral Richard W. and Vice Admiral John Nathman, *U.S. Navy ForceNet: Turning Information into Power*, *Proceedings*, February 2003.

Schuman, R., *Developing an Architecture that No One Owns: The US Approach to System Architecture.* Proceedings First World Congress Appl. Transport Telematics Intelligent Vehicle-Highway System. Paris, France, 1994.

Schwartz, Peter, *The Art of the Long View: Planning for the Future in an Uncertain World,* Bantam Doubleday Dell, 1995.

The 9-11 Commission Report: Final Report of the National Commission on Terrorist Attacks Upon the United States, Official Government Edition, 2004.

BIOGRAPHY

John Boardman – Professor, Stevens Institute of Technology. His industrial experience includes GEC and MANWEB, a UK regional electrical utility, and his academic appointments have taken him from the University of Liverpool through Brighton Polytechnic, Georgia Institute of Technology, and the University of Portsmouth to Professor of Systems Engineering at De Montfort University, Leicester.

Mark A. Wilson – CEO of Strategy Bridge International, Inc., a firm that delivers consulting, facilitation, and professional development services to enable clients to effectively execute strategy through systems thinking, systems engineering, and project management. He is a Certified Systems Engineering Professional (INCOSE), Certified Quality Manager (American Society for Quality) and holds a B.S. from the U.S. Naval Academy and a M.S. from the U.S. Naval Postgraduate School.

Allen Fairbairn – Managing Director, Elipsis, Ltd, and Director, JBA Management consultants. Elipsis provides professional development seminars, consulting expertise, and innovative software tools focused on enabling clients to effectively manage large-scale and small-scale systems, processes and projects. Mr. Fairbairn worked in various industries before working on the Channel Tunnel Project as Systems Engineering Manager from 1985 to 1998. His work with JBA has included the development of extended enterprise thinking using System of Systems and other representations. He is Past President of the INCOSE UK Chapter and was given the INCOSE Pioneer award in 2004.