

# Key Challenges and Opportunities in 'System of Systems' Engineering

**Jack Ring**

Innovation Management  
32712 N. 70th St.  
Gilbert, AZ 85234  
jring@amug.org

**Azad M. Madni**

Intelligent Systems Technology, Inc.  
3250 Ocean Park Blvd., Suite 100  
Santa Monica, CA 90405  
amadni@intelsystech.com

**Abstract** – *System of Systems Engineering (SoSE) extracts value from existing assets and designs new assets to be more easily re-purposed, than has been the case. One way SoSE arrives at a System of Systems (SoS) is by interfacing or incorporating existing systems. Another way is by “harmonizing” a set of holons. Either way managing the on-going evolution of the SoS is more challenging than initializing the SoS. This paper presents the challenges and opportunities for the next generation of concepts, principles, methods and tools that are needed for creating and sustaining SoS's.*

**Keywords:** system-of-systems, systems engineering, system-of-systems engineering, architecture, holons, systems complexity, unintended consequences

## 1 Introduction

One of the more successful, large-scale SoS is commercial air transportation. The commercial air transportation system is a system of airports, airlines, airplanes, airplane manufacturers, air traffic control and a host of other aviation-related systems. Each has distinctly different but highly coherent purposes. However, even this system is signaling 'stall' conditions. The less successful SoS is the K-12 education system in the U.S., with the healthcare SoS not faring much better. These instances serve to highlight both the challenges and the opportunities for SoS success; notably, the value of the respective purposes of the constituent systems as evidenced by their behaviors, and the degree to which these behaviors can be harmonized and augmented to achieve the desired behavior of the composite SoS.

In light of the foregoing, we need to develop a strategy to create and evaluate the SoS's of the future. To this end, we could: a) use COTS; b) reuse major chunks of existing systems; c) repurpose existing systems; d) transform existing systems into 'holons;' or e) create new 'holons.' Holons are subsystems of a new system yet each 'subsystem' can still fulfill its original mission while simultaneously participating in fulfilling the new mission. [1]. This paper examines these different approaches and identifies promising directions for the future. Regarding the latter, future SoSE must have the requisite variety to

cope with the following challenges relative to current SE practices.

## 2 Key Challenges

Perhaps the most challenging problem results from incongruous effects that result from “unnatural juxtapositions” of systems which inevitably lead to increased scope and complexity and thereby:

- a) increase the “unknownness” of the reused systems and the “unknowability” of the SoS;
- b) increase the chance of latent error, bugs, or mismatches;
- c) increase the number of ways the SoS can fail;
- d) decrease the user's ability to discern failures; and
- e) increase the negative ramifications of failures, increase the need for complex, adaptive and self-adaptive kinds of systems.

But there are issues beyond complexity that need to be addressed. These include: ambiguity; human social dynamics; sustainability; and methodology (Table 1).

Table 1. Challenges Beyond Complexity

- Ambiguity
  - precluding a large percentage of design errors, project cycle time, and cost overruns by ensuring that all responsible and affected participants in a project are working in a common semantic space (e.g., unified language and mental models even if spatially and temporally distributed)
  - ensuring adequate, accurate and timely data to all parties, at all times and locations
  - increase in the ambiguity in configuration management thus reducing both the producibility and sustainability of the SoS
- Human Social Dynamics
  - increase in the person to person (actually stranger to stranger) interactions throughout the SoS thus the likelihood of operational error
- Sustainability
  - managing the evolution of system value as the thousands of design decisions unfold and myriad change proposals are judged
- Methodology
  - assessing in real time the SoSE situation and adapting the SoSE capability via personnel assignments based on teleonomics [2] principles

(Table 1. continued)

- maximizing practitioner productivity and innovation
- creating not just design documents but 'learning modules' that are most useful to those who must design, develop, assemble and test the system
- ensuring that SoSE practitioners learn at the rate of opportunity
- offsetting the forthcoming reduction in SoSE practitioner population with advancements in productivity and innovation.

It is prudent that systems practitioners exercise caution and care when creating a SoS. While current practices suffice for simple system configurations, a SoS typically demands better theory, methods, and tools to prevent unintended consequences. And finally, we are forced to acknowledge that a static model of the system is not only insufficient but also leads practitioners to serious misunderstandings and under-conceptualization of the solution. Table 2 presents current practices (left column), the necessary SoSE practices (center column), and key opportunities for improving current practice (right column).

Table 2. Stretch Goals for SoSE

SE Current Focus	SoSE Focus	Carpe Diems
<b>Identification</b>		
Acquisition Sponsor	Variegated stakeholders	Facilitation of group solutions to complex situations. [3] Multi-stage SoSE.
What's the Problem?	What's the opportunity?	Understanding the way we think. Applying Contrarian thinking. Avoiding Groupthink, Clanthink and Spreadthink.
"System Shall" Requirements	Value proposition	Measures of effectiveness
<b>Design/Architecture</b>		
Focus on mission system	Focus on Whole system	Mission system, Operational availability system, Test system(s), Operator preparation system, Production system
Conformance to Requirements	Do Until: Sufficient Requisite Variety, Parsimony and Harmony	Axiomatic design multi-objective maximization Analytic/Intuitive Rigor/Heuristics
Top-down	Allow for heterarchical structures as well as hierarchical	e.g., Process-oriented object-based chaotic forms
Single system (of systems)	Federated or holonic web of systems	Theory, principles and rules for partitioning large, complex systems

		into manageable modules at all stages of system realization (e.g. diaktotics). Design, build, activate only small systems then systematize into value web. Use Scenario Agents to organize and execute operational episodes across the web.
Framework-driven	Principles-driven	Response Ability Principles [4]
Minimize Complexity	Leverage complexity	Design for agility. [5] System structure zoom and navigation. [6] Make systems self-documenting and self-reporting to minimize need for human discernment of status.
Focus on Informatics, Thermodynamics	Add Teleonomics and Human System Dynamics	Include Cognitive Systems Engineering and extend to Human System Dynamics.
Interfaces	Interrelations, (explicit and implicit)	How to structure context sensitive systems and autocatalytic systems.
<b>Engineering</b>		
Spiral Method	Mesh Method	Genetic Algorithms and Agents [7]
Functional vs. Non-functional	Model-based SE	Wymore's Six [8] Holonomic design
System layers	Distributed Infrastructure	Design for Evolution, DfE
Logical, Physical	Feasible (techno-economic), Buildable (harmonized)	Design decision management. Goal-seeking archetype.
Risks Mitigation	Dynamic Limits Sufficiency	Expectation of System Achieving MOE's.
<b>Activation</b>		
Support	Verify Interoperability Design CCB Whole System. Adapt to evolving situation	Proactive FMEA. Discern POSIWID. Assay system value. Four phase commit sequence for changes.
Disposal	Renewal	Disposal is years later
<b>Learning</b>		
N/A	Actual Models Lessons Learned Facilitated Reflection	See SoSE as knowledge production and utilization. Evolve an SE Education Community. [9]
SE Tools	SoSE Infrastructure	Reuseful Assets Repository

### 3 Law of Unintended Consequences

The Law of Unintended Consequences says that actions always have effects that are unanticipated or "unintended." In the worst case, the outcomes are exactly the opposite of the purpose of the original actions. Often the resultant situation ends up being worse than the situation before the action was taken.

This phenomenon has been around for centuries. In 1692 John Locke, the English philosopher [10] spoke against a proposal to cut the maximum permissible rate of interest. Locke argued that instead of benefiting borrowers as intended the bill would punish the borrowers because people would find ways to circumvent the law and the costs of circumvention would be shifted to the borrowers.

Almost three centuries later, according to Rob Norton [11], a 1968 Vermont law banned roadside billboards and large signs in order to protect the state's pastoral vistas. One unintended consequence was the appearance of large, bizarre "sculptures" adjacent to businesses.

In [12] John Gail gives numerous other examples. Alternatively, one only has to watch large bureaucracies in operation to appreciate how they come to be rich sources of unintended consequences.

In fact, the examples are so prevalent as make the Law of Unintended Consequences seem to be an every day, natural law. However, it is a man-made law and can be repealed. Unintended is simply a code word for not knowing enough about systems or worse, being inept at applying systems engineering principles.

*Unintended indeed.* Recently, a U.S. Congressman complained about corporations taking advantage of loopholes in a recently passed law. Since he was the co-sponsor of the legislation, he felt especially incensed that the law was having an effect that was opposite to what he had intended. One might wonder why the loopholes were put in the law in the first place because loopholes had to pre-exist in order to be found and exploited. Yet there was not a scintilla of realization on the Congressman's part that the source of the loopholes was no other than he and his staff. This is an all-too-common problem when one is dealing with a system-of-systems. Let us dig a little deeper.

Of course the loopholes were not purposefully included when the legislation was written. Loopholes are an emergent characteristic of rule sets that were written for System A but then cleverly applied to a System B.

*Loopholes can be precluded from just happening.* Systems thinkers know they must design a system not just for the situation that exists but for the situations that will

result when the system is put into operation. Humble system thinkers have a label for this, POSIWID, the purpose of a system is what it does (regardless of what the sponsors and developers intended it to do).

*Ignorance of the law is no excuse.* This oft-quoted admonition applies to ignorance of the "laws of systems" as well as to civil/criminal laws. No doubt the lawmaker was trying to do the best job he knew how to do. But therein lies the clue. He simply did not know enough about "systems" to be writing legislation. More than likely he did not have the right educational background, was too shortsighted to worry about potential loopholes, and where they come from.

*Sources of Unintended Consequences.* Rob Norton also tells us that the first and most complete analysis of the concept of unintended consequences was done in 1936 by the American sociologist Robert K. Merton. Merton [13] identified five sources of unanticipated consequences: 1) ignorance; 2) error; 3) imperious immediacy of interest; 4) basic values that are unsustainable; and 5) self-defeating prediction and its dual, the self-fulfilling prophecy.

In 1995 John Warfield introduced a different view by pointing out four ways humans delude themselves [14]. Further, in 'Mentomology, The Identification and Classification of Mindbugs', Warfield provided 25 examples. We believe that these are a valuable checklist for SE practitioners to use in design reviews and approving the release of work products.

### 4 Opportunities

From the foregoing table, it is clear that there are numerous opportunities for more effective SoSE. Therefore, it is important to prioritize and identify a subset of high value opportunities to pursue. Ideally, opportunity prioritization and pursuit should be conducted by a workgroup of informed practitioners and researchers using, for example, Interpretive Structural Modeling [5] or equivalent. Nevertheless, we have identified the following high payoff opportunities for advancing the cause of SoSE.

#### 4.1 SoSE Education Environment

We identify SoSE Education as an early step because of the need for a longer lead time to payoff. The Concept of Operations (ConOps) of a Systems Engineering Education Community (SEEC) [15] asserts that no academic institution can provide a learning environment that is sufficient for developing SE practitioners. This is because of the breadth and duration of learning experiences required to become a qualified SoSE practitioner. The ConOps envisions heterarchical collaboration among academia, SE industry, commercial suppliers of SE training, and standards bodies as well as professional societies, such as IEEE Systems, Man, and Cybernetics and

INCOSE. By harmonizing the information, terminology, examples, and learning tools in each, yet accommodating differences in learning styles, the time to practitioner competency can be accelerated considerably.

## 4.2 SoS Architecture Options

By architecture we mean ‘the arrangement of function and feature that enables maximum effectiveness of the system’ [16]. This view of architecture harmonizes: a) the views of established architects such as Frank Lloyd Wright, M. Pei, and the Bauhaus that architecture is concerned with relationships and patterns of relationships [17] [18]; with b) the system design pattern of “context, content, structure” as recommended by [5], with c) the practices of Model-based Systems Engineering [8].

Although the majority of literature concerning enterprise and system architecture presumes the hierarchical, layered form, other distinct options exist. Ring (2001) [19] presents additional forms (e.g. centralized, distributed, federated, autopoietic and autocatalytic) in descending order of cohesion and ascending order of reuse potential.

## 4.3 SoSE Process Management

**Requisite Agility.** Today, many aspects of a SoSE are in flux. Therefore, applying the Law of Requisite Variety, any distributed SoSE workgroup must be agile --- **able to thrive in an environment of unpredictable change.** This implies that an SoSE activity cannot be planned just once at the beginning of the development project; rather, the work plan may need to be adapted several times in any one SoSE project. This may entail change of operating mode, personnel (for competencies or styles), as well as change of schedule or process.

The SoSE activity is best conceptualized as a goal-seeking system that defines goals, works toward those goals, has the competencies to accomplish them, has the energy to apply the competencies, has objectivity in assessing the evolving situation, and has the adaptation mechanisms to continue to pursue the goals despite perturbations along the way. A goal-seeking system has limits including performance limits (quality, cycle time, and return on resources), resilience to disturbances (correction loop limits), and dynamic limits (e.g. analogous to stall conditions of an aircraft). Similarly, a goal-seeking workgroup is able to adjust its content, structure, and operating mode to ensure that its ability to create the SoS is not compromised. In this regard, risk management is not sufficient because it focuses only on known aspects of the situation. Limits management is also necessary for SoSE. Current standards and handbooks for SE and SoSE processes and practices do not provide for limits management. Accordingly, this subject merits early attention.

An agile SoSE workgroup has the requisite change proficiency, knowledge production and utilization ability, timely decision making ability, and acceptable latencies. A distributed SoSE workgroup can potentially adopt a variety of operating modes. Table 3 shows the key considerations for four operating modes (i.e. hierarchical, process-oriented, object-based and chaordic) [20]. This table also presents some of the considerations for each choice.

Table 3. Comparison of Architecture Choices

Pattern	Hierarchy	Process	Object	Chaord
<b>Motivation</b>	Control	Responsiveness	Asset Turns	Innovation
<b>Investment Objective</b>	Preserve Institution	Satisfy Customers	Constraint Coherence	Learning
<b>Governance</b>	Directives	Rules	Goals	Principles
<b>Resource Conflict Resolution</b>	Setting Priorities	Reactive Pursuit	Proactive Pursuit	Local Adaptation

Most current enterprises operate in the Hierarchy mode, regardless of whether they are structured by Function, Product, or Market perspective. A few have moved to more of a process-oriented mode. Examples may be found in Retail, Customer Support, Maintenance, and Insurance. This tends to reduce the “white spaces” between silos, shrink cycle times, and thereby minimize waste. In one sense a Process-oriented enterprise is simply a matrix organization with standing project plans. When conflicts for resources affect two or more processes, process management becomes the problem. The other limit in the Process mode occurs when change management must be orchestrated across multiple process changes.

An object-based mode of operation (typified by contract manufacturing, franchises and Southwest Airlines) exhibits two distinct differences. First, many more types of interrelationships are acknowledged and managed than the ‘Is A’ and ‘Cause-Effect’ kinds that are featured in hierarchical and process modes respectively. Second, the entities become active – volunteering for projects, acting locally to maximize their respective value to the enterprise. The limit in object-based is finding personnel who are willing to assume the high level of responsibility and accountability required.

The chaordic form of enterprise (typified by examples at [www.chaordic.com](http://www.chaordic.com) and by professional sports teams in play, certain and some activist/terrorist organizations) is best thought of as a set of goal-seeking systems all collaborating toward mutual purpose by using an agile framework and distributed infrastructure to harmonize variegated goals aided by an issue resolution protocol for resolving schedule conflicts and ‘deadly embrace’ conditions.

**Situation Assessment.** A key aspect of process management is situation assessment, SA, i.e., determining what's going on and why? What should be done and how rapidly? The early forms of SA aids were 'war rooms' that were used on large projects. Such 'war rooms' need to be employed for distributed workgroups, any one (or more) of which may be participating in multiple SoS projects. SA is a key requirement of a SoSE workgroup. In particular, two types of situations need to be assessed. One is the system design status. The other is the SoSE effectiveness status. The immediate need is for a theory and design aids for SA during the conduct of SoSE projects. It should be noted that while some erroneously equate SA to data fusion, SA requires more than data fusion. For example, paraphrasing quality guru Phil Crosby, 'as organizations increase in size, management finds it difficult to know what is going on and practically impossible to know what is not going on.' However, it is possible to do both if a high fidelity model of the overall process and intended activities is prepared, maintained, and incrementally populated with actuals as they become available [21]. Earned Value is one example that interrelates cost, schedule progress, and indications of quality. A parallel notion of Net Present Value of System Effectiveness is also needed. With this addition, Earned Value could be extended to become an indicator of both SoSE and SoS success.

#### 4.4 SoSE Infrastructure

Instead of the panoply of piecemeal tools now available to SE practitioners and instead of current efforts to 'integrate' these tools or to create 'interoperation' via converters (e.g. AP233), the creation of an overall SoSE Infrastructure needs to be pursued. This infrastructure could be employed on a pilot SoSE project comprising a number of practitioners at multiple locations. Similarly, two or more such pilots could be combined to support the interoperation of the two projects. This latter experiment is especially important for configuration and change management across multiple systems.

It is useful to view each SoSE activity as an enterprise in its own right, committed to producing and delivering value to its customers (the downstream system realization teams) and suppliers (the stakeholders and sponsors).

**Coordinated Management of Meaning.** Most SE organizations do not have a coherent policy and system for sharing and refining meaning and models among members nor of enabling automation of the stream of joint discovery and decisions involved. Their systems do not provide mechanisms for meaning, motivation, decision and learning. Those few organizations that have such capability achieve it with expeditors, action items and other means that are unnecessarily expensive and time-consuming. From the various existing systems, the SoSE infrastructure should be capable of providing reusable

assets. For future systems, the infrastructure should provide theory, methods, and tools for harmonizing different ontologies in support of a given mission.

**Integrated Modeling, Simulation and Effectiveness Assessment.** SoSE practitioners stand to benefit from: a) the ability to create not only static models, both descriptive and prescriptive, but also dynamic models of a contemplated system, its behaviors and effects; as well as b) compiling a repository of model constructs reflecting informatics, thermodynamics and teleonomics aspects of systems. The simulation and effectiveness assessment capability will benefit both the design of operational systems as well as the design and evaluation of SoSE projects and distributed workgroups.

**Proof of Due Diligence Process.** Alarming, a new "grim reaper" is emerging [22]. The mounting flood of product (and services) liability litigation is setting dramatically higher management penalties than have been assessed to date for such failures. To counter this threat a SoSE project must be able to show that it has conducted due diligence and has the requisite accreditation for producing products and services.

**Configuration and Change Management.** These activities provide information and decision support for Configuration Control Board operations including, Proposing, Evaluating, Deciding, Scheduling Kitting, Notifying, Installing, Verifying and Validating. Because of the multiplicity of systems and in-service engineering organizations involved a four-phase commit protocol is required. No such capability exists today.

**Collaboration Engine.** SoS projects can potentially employ multiple SoSE practitioners ranging from two to thousands. Because productivity decreases and error-proneness increases as distributed workgroup size increases, the SoSE distributed workgroup structure can be expected to tend toward a multi-node knowledge web. Further, in view of the multiplicity of disciplines involved and the several versions of a given design that may be current, the distributed workgroup will most likely need several thousand views of any emerging model of the intended SoS. Accordingly, the SoSE Infrastructure must provide a distributed collaboration engine capable of supporting participants in Computation, Connection-making, Communications, Convergence, Cooperation, Collaboration, Commitments, and Co-learning. Current PC technology focuses on the first two while emerging technologies such as the Semantic Web and blogging seek to aid Communication, Cooperation and Convergence. Collaboration support needs to go beyond blogging to provide capabilities for context-sensitive search, knowledge management, and results synthesis.

## 5 Conclusions

As systems grow in sophistication and complexity, the need for SoSE is becoming increasingly more pervasive. It is also becoming clear that the “big bang” approach to building very large systems is not the way to go. As a result, there is a pressing need today to evolve theory, methods, competencies and aids for SoSE. Paraphrasing Einstein, ‘to solve our current problems we will have to operate at a level of consciousness higher than the one we were at when we caused the problems.’ We need to change our perspectives and ways of thinking and choosing. The approach offered in this paper calls for a shift of mindset from ‘building a system’ to ‘creating new paths through a web of capabilities.’

It is in this spirit that we recommend the following for advancing the state of the art of SoSE. First, sponsor one or more conceptual design projects for the Systems Engineering Education Community. One way to avoid unintended consequences in SoSE is to ensure that the SE practitioner become educated in the “laws of systems” and the fundamentals of Systems Thinking, Engineering and Value Management. Second, sponsor further research and experimentation in SoSE modes of operation. Third, sponsor research in conceptual design of an SoSE Infrastructure. Finally, sponsor an Interactive Management session involving twenty or so SoSE practitioners and theorists to put the ideas advanced in this paper and elsewhere to the test with the intent of developing and committing to a roadmap for advancing the state of the art of SoSE. Collectively, these initiatives can be expected to increase awareness of SoSE issues within the SoSE community and advance the state-of-the-art of SoSE.

## References

- [1] Koestler, A., *The Ghost in the Machine*. Hutchinson & Co, 1967.
- [2] Gilbert, T. *Human Competence: Engineering Worthy Performance*. HRD Press, 1996.
- [3] Warfield, J. and Cardenas, R. *Interactive Management*. Iowa Press, 1990.
- [4] Dove, R. *Response Ability; the Language, Structure of and Culture of the Agile Enterprise*, Wiley, 2003.
- [5] Warfield, J.N., *Understanding Complexity: Thought and Behavior*, AJAR Publishing Company, 2002.
- [6] c.f. Metis at [www.troux.com](http://www.troux.com)
- [7] Axelrod, R. & Cohen, M.D., *Harnessing Complexity, Organizational Implications of a Scientific Frontier*, The Free Press, 1999.
- [8] Wymore, A.W. *Model-based Systems Engineering*, CRC Press, Boca Raton, 1993.
- [9] “Concept of Operations (ConOps) of a Systems Engineering Education Community (SEEC),” Document No.: INCOSE-TP-2003-015-01 Version: 0, 4 November 2003. [www.incose.org](http://www.incose.org)
- [10] Locke, J. Some Considerations of the Lowering of Interest, [www.socsci-mcmaster.ca/~econ](http://www.socsci-mcmaster.ca/~econ), 1691.
- [11] Norton, R., Unintended Consequences, The Concise Encyclopedia of Economics, [www.econlib.org/library/Enc/UnintendedConsequences.html](http://www.econlib.org/library/Enc/UnintendedConsequences.html)
- [12] Gall, J., *The Systems Bible; The Beginner’s Guide to Systems Large and Small*, General Systemantics Press, 3rd Ed., 2002.
- [13] Merton, R.K. *Social Theory and Social Structure*, Enlarged Edition, 1968 [1949], The Free Press, New York.
- [14] [www.gmu.edu/departments/t-iasis/paper/p4.htm](http://www.gmu.edu/departments/t-iasis/paper/p4.htm)
- [15] [www.incose.org](http://www.incose.org), Document No.: INCOSE-TP-2003-015-01 Version:0, 4 November 2003.
- [16] IEEE P1471, “Recommended Practice for Architectural Description.”
- [17] Hale, J., *The Old Way of Seeing*, Houghton Mifflin, 1994d, pg 8.
- [18] Venturi, R., “Complexity and Contradiction in Architecture,” NY Museum of Modern Art Paper, 1966, pg 14.
- [19] Ring, J. “Discovering the Architecture for Product (X),” *INCOSE Symposium Proceedings*, 2001.
- [20] [www.chaordic.org](http://www.chaordic.org)
- [21] Madni, A.M. “Thriving on Change through Process Support: The Evolution of the ProcessEdge™ Enterprise Suite and TeamEdge™,” in *International Journal on Information • Knowledge • Systems Management*, Special Issue Vol. 2, No. 1, Spring 2000.
- [22] [www.menaceshield.com](http://www.menaceshield.com)