Systems of Systems Engineering for NATO Defense Applications

Introduction

Professor Michael Henshaw
Loughborough University, Leicestershire, LE11 3TU
UNITED KINGDOM

m.j.d.henshaw@lboro.ac.uk

Wherever two or more independent systems must work together co-operatively, Systems of Systems (SoS) considerations become important. The systems may have been designed to work together from the outset, or they may be required to interoperate even though there was no previous intention to do so. There are significant advantages associated with interoperable systems, such as Network Enabled Capability (NEC) in which effects, not achievable by an individual system working in isolation, can become possible[1]. This has always been true, but with the rapidly increasing interconnectedness of systems in both the military and civilian world, SoS Engineering (SoSE) has become the essential skill for Systems Engineers and, indeed, many other disciplines within enterprises that, perforce, participate in SoS.

For some years there was a debate among the Systems Engineering community about whether a 'Systems of Systems' is just a 'System'. In fact, this was an unnecessary dispute, because a SoS is clearly a System; but the concept of SoS focuses attention on a particular set of systems problems that must be resolved. Even with this rather straightforward resolution of the argument, there are still a variety of views about how SoS should be described and addressed. The topic is still quite new as an area of explicit study and there is much still to discover about SoS behaviour and how it can be managed. Through the papers in this Lecture Series some of the reasons for disagreement will become apparent but, more importantly, the areas of consensus and how these are being developed into more effective design for, and operation of SoS will be presented. These papers represent the state of the art in terms of SoS understanding; in some cases only the challenges can be explained with certainty, in others good practice has been identified. This set of papers describes current practices and will hopefully inspire the reader to participate in the improvement of these practices and development of new, and better, practices in the years to come.

In paper 1, Judith Dahmann provides an overview of the definitions and characterisation of SoS and explores a taxonomy of SoS. This is important because the different types determine the most suitable approach to take in development, etc. For the purposes of clarity in this introduction, we provide one of the definitions and a brief description of the characteristics of SoS below.

Definition: *A SoS is an integration of a finite number of constituent systems which are independent and operatable, and which are networked together for a period of time to achieve a certain higher goal.*[2]

The generally accepted characterisation of SoS is due to Mark Maier[3]: that the constituent systems of a SoS are operationally and managerially independent of each other; that they are generally geographically distributed, that the SoS develops in an evolutionary manner, and that there are emergent behaviours, i.e. behaviours that cannot be predicted by consideration of any of the constituent systems in isolation. It is the first two (managerial and operational independence of constituent systems) that lead to many of the

---

[1] See for example: Buckman, T., 2005, NATO Network Enabled Capability - Feasibility Study, Exec. Summary, Version 2.0, NC3A, http://dodccrp.org/files/nnec_fs_executive_summary_2.0_nu.pdf

[2] Jamshidi, M., 2009, in the introduction to System of systems engineering - innovations for the 21st century, pg. 2, ed. Jamshidi, J. Wiley & Sons

[3] Maier, M. W. (1998). "Architecting principles for systems-of-systems." Systems Engineering 1(4): 284

challenges faced in SoSE. These factors create a level of uncertainty in the behaviour of a SoS that mean absolute prediction is not possible, but operators must be creative and innovative, in order to cope with the ongoing changes in the performance and operation of the SoS.

Systems architectures and architecting are major themes in this Lecture Series. An architecture essentially describes the way in which the resources (or elements) of a system are organised; in paper 2, Peter Brook provides commonly accepted definitions and goes on to introduce different types of architecture that may be deployed according to the type of system under consideration. He focuses attention on Enterprise Architecture and introduces the architecture frameworks applicable to SoS in defence and other more general environments. He focuses attention on MODAF and TOGAF as pertinent examples through which to understand the use of architecture frameworks.

A system may exchange energy, mass (material), or information[4] with its environment (or other systems). To be interoperable, two systems must be able to make the exchanges in such a way that they can make use of the exchanged entity to work together; i.e. they can operate together. For most SoS of concern to NATO, it is the interoperation with respect to information that is of primary interest. Quite clearly for a system to be interoperable with other systems in a SoS, attention must be paid to the interfaces. In fact, one distinction of engineering a SoS in contrast to a single system, is that the engineer must be focused on the interfaces (i.e. outward facing) aspects of the system, rather than the internal workings of the system. The managerial independence of constituent systems and the evolutionary nature of the SoS development mean that managing interfaces is non-trivial. Dieter Scheithauer takes an historical perspective in paper 3 on specialisation, workshare, and standards as the principles which determine the success or failure of interfaces in SoS. He shows how the industrial context is a crucial influence on the eventual operational interoperability of a SoS and examines some possible solutions to mitigate the risks of interoperability failure, reaching a conclusion about the obligations that systems builders must accept with respect to use of standards.

Many texts on Systems Engineering include substantial discussion about system lifecycle; this is naturally important, since the operation of a system over time, and the planned development of the system prior to operational deployment, must be properly understood. The definition of lifecycle for a SoS, though, is problematic, because there are several different approaches one could take. In paper 4, Judith Dahmann presents current models for SoS lifecycle with a particular focus on those used in defence procurement, which are, in turn, based on the four process types in the Systems and Software Engineering Lifecycle Standard[5]. She goes on to discuss the 'wave model'[6], which is a way of planning and managing the evolutionary nature of SoS. Once again the role of architecting is emphasised as a critical element of planning and managing SoS.

Through a brief analysis of SoS performance inadequacies and failures, Michael Henshaw (paper 5) shows the significance of the 'people element' in SoS operation. In particular, the behaviour of organisations that own or operate system is a major consideration and, in order to understand SoS behaviour a socio-technical perspective is required. Picking up the theme of interoperability, the need to fully embrace the socio-political end of the interoperability spectrum is discussed. Understanding and managing situational awareness is a crucial element of SoS success, and appropriate governance structures (that acknowledge the levels of autonomy of constituent system owners) is an imperative.

SoS development and operation must take both a top down and a bottom up approach. In paper 6, Judith Dahmann considers the implications of top-down SoS needs on the development of the individual systems

[4] Some people add finance to this list

[5] ISO/IEC. 2008. Systems and Software Engineering -- System Life Cycle Processes. Geneva, Switzerland: International Organisation for Standardisation / International Electrotechnical Commissions. ISO/IEC/IEEE 15288:2008.

[6] Dombkins, D.H., 2007, Complex Project Management, Charleston S.C. Booksurge.

(bottom-up) that must participate in a SoS. Based on the work of the TTCP[7], she presents the processes (particularly the review processes) that must be applied in the system development of individually acquired systems, such that they are suitable for interoperation within a SoS. The conclusion of this paper is significant: recognising the increasing (and deliberate) interconnectedness of systems, there is an implication for organisations that they should train their Systems Engineers to be SoS Engineers.

This theme is picked up in paper 7 by Peter Brook, in which he explicitly considers the impact of enterprise structure and behaviour on the development of technology. He presents different perspectives on enterprises and how they interact with SoS. Successful operation of SoS requires co-operation between the enterprises of development, use, management, and support. This is presented through a new model of SoS and enterprise engineering.

In paper 8, Dieter Scheithauer argues that different communities have taken different (but related) approaches to dealing with the complexity of SoS. Architecting, requirements engineering, and Model-Based Systems Engineering (MBSE) have evolved separate practices and terminology, which can lead to confusion among developers. He proposes a whole system lifecycle view of systems development that uses value-stream thinking to resolve the differences.

Finally, in paper 9, Michael Henshaw summarises some points of good practice for SoSE. A first step is concerned with collective understanding: an appreciation of the concepts of SoS and a recognition that the system one manages and/or operates is within a SoS. From this, the behaviours likely to be constructive at the SoS level can be understood. Not all SoS are the same! Appreciating the different types of SoS (i.e. different control structures) and then recognising the type of SoS of which one's systems is a part, is a further step in understanding the appropriate behaviours for effective SoS operation. Good behaviours are enabled by effective information sharing within the SoS: this improves the situational awareness of operators by improving their understanding of the effects of local decisions on global behaviours. Furthermore, the relevance of social and political aspects compared with technical aspects should be understood if behaviours, and reconfigurations of the SoS, to cope with a changing environment are to be chosen appropriately. SoS can automatically imply a separation of concerns, and modularity is a fundamental principle of system build for SoS; it is shown that an open architectures approach to system specification can provide the necessary agility in operation; this is particularly important from the perspective of effective evolution of SoS.

Systems of Systems is an essential construct for NATO and, indeed, for most socio-technical endeavours in the modern world. Appreciating and managing the uncertainties that SoS introduce is a step towards being more effective when faced with complex undertakings. Successful SoS operation requires individual system owners (mangers and/or operators) to focus on the overall SoS goals, rather than the local, individual system goals, but this is difficult to achieve. How to incentivise such behaviours is the principal concern for SoSE. This Lecture Series covers various aspects of SoS: architecting is and important technical theme, organisational behaviour is an important societal theme; lifecycle could be said to embrace both social and technical. Overall, the SoS Engineer must appreciate the social, political and technical aspects. This Lecture Series has mapped out the current concerns and approaches for SoS, the authors hope that this will encourage readers to add to this body of knowledge in order to extend the collective NATO understanding of Systems of Systems.

[7] The Technical Co-operation Program: http://www.acq.osd.mil/ttcp/