Good Practice in Systems of Systems Engineering (SoSE)

Professor Michael Henshaw
Loughborough University, LE11 3TU
UK

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

ABSTRACT

Good practice is emerging in Systems of Systems (SoS) but with many questions regarding design and management still being the subject of research. Part of the challenge is to recognise that it is, in fact, a SoS that must be managed, rather than a single system. The first element of good practice concerns the development of a SoS-minded workforce that can understand the SoS perspective and use it effectively to achieve the SoS aims. Many of the issues in SoS turn out to be non-technical; as such there needs to be a focus on the social, political, and enterprise aspects of SoS. The type of the SoS in question determines the authorities and relationships within the SoS, and this must be recognised by the participant in the SoS. Good practice requires the participant to take a service perspective and to focus effort on the enterprise that develops and manages the SoS. Management of the SoS over time requires a good architecting approach and an appreciation of how systems can be most effectively introduced and retired from the SoS, i.e. effective architecting for managed transitions. A significant factor in good management of SoS is to take an open architectures approach and to ensure modular build of the systems participating in the Systems of Systems.

1.0 INTRODUCTION

A defence systems trainer endeavoured to get her students (mostly experienced procurement and engineering specialists) to consider the challenges of maintaining a military capability over the long term; how it might need consideration of integrating current systems with systems that have not yet been conceived. She urged them to take a holistic view, but was disconcerted when one of the students voiced the thoughts of many of the class: "just tell us what the process is, then we will follow it." Process is, of course, a vital aspect of engineering and procuring systems, but managing SoS capabilities over the long term requires innovation, acceptance of uncertainty, openness to change, and a holistic perspective. It usually requires system identification to be more strongly based on the behavioural approach than the structural[1]; in short, it requires a different kind of thinking from the traditional engineering of single systems.

SoSE is still a developing subject and there is much still to do in order to establish good practice. The INCOSE SoS WG Pain points, discussed by Dahmann[2] give an indication of the main troublesome areas and good practice must primarily be a response to overcome these difficulties. In fact, these point towards the need for research in SoSE in order to address the areas of concern. Some of the pain points are a direct result of the managerial and operational independence of constituent systems; thus, there are ambiguities with respect to authority, leadership, and collaboration models. Others concern technical difficulties of integration, testing and emergence. There is also a pain point regarding questions about SoS thinking: what are the principles?

In this chapter, we begin with consideration of SoS thinking, starting with a discussion about whether stakeholders in the SoS are aware that they are dealing with a SoS. We consider the mental models that individuals might hold about systems and note the importance of situational awareness in SoS. Allied to this

[1] Flood, R. L. & Carson, E. R., 1988. Dealing with Complexity: An Introduction to the Theory and Application of Systems Science. Plenum Press.

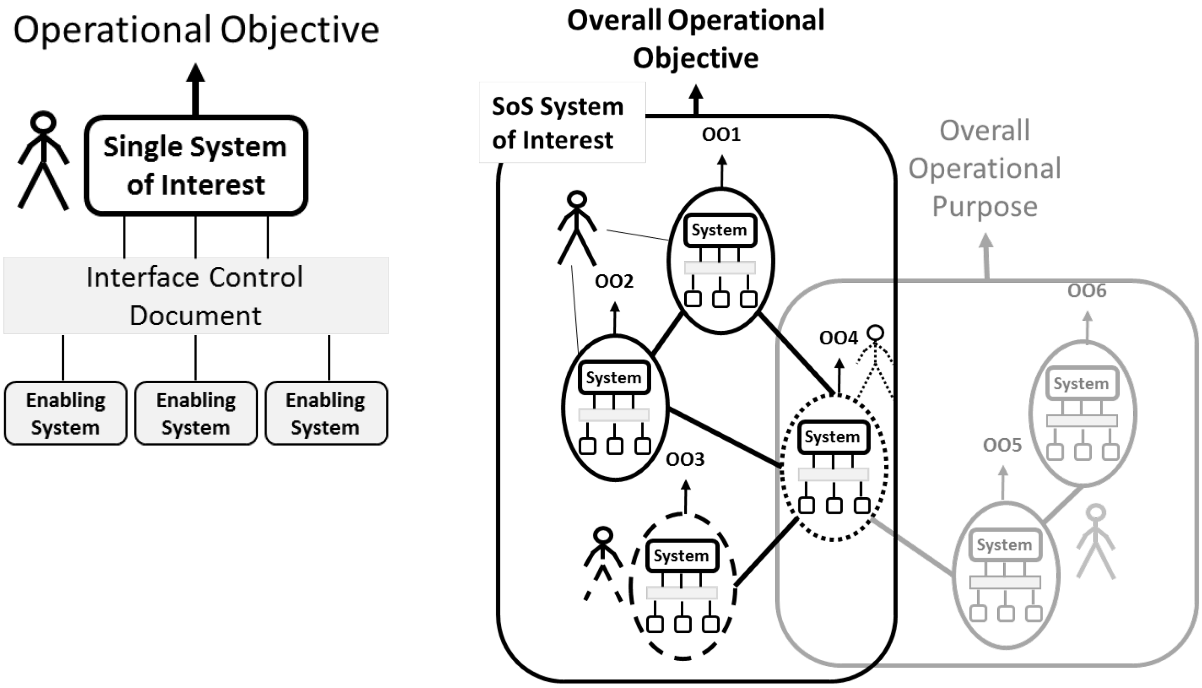
[2] Dahmann, J. 2015, Systems of Systems Characterisation and Types, In: SCI-276 Lecture Series. CSO.

consideration is the nature of the relationships in a SoS, some are formal and some informal, but all relationships must work if the SoS is to be effective. The lifecycle of single systems is generally quite easily defined, but the lifecycle of a SoS could be defined in many different ways. We give consideration to the approaches that can be used throughout the phases of a SoS and the way that individual systems lifecycles may interact with each other.

Finally we consider the importance of architecting and or architectures in managing SoS; in particular, we recommend the adoption of open architectures as an enabler of effective SoS. We briefly mention the SoS principles established by the UK MoD as examples of good practice in the defence supply chain.

2.0 A SOS-MINDED WORKFORCE

If it is accepted that SoS requires systems developers and operators to have a suitable way of thinking, then we must try to identify the attributes that they should possess or the considerations that they must appreciate. Regular Systems Engineering practice considers a single system to be the System Of Interest (SOI) that must meet an operational objective[3]; it will generally need functions from other systems (outside of the immediate SOI), that are termed enabling systems[4]. The Systems Engineer will create an Interface Control Document (ICD) to specify the interfaces between the main system and the enabling systems. This is shown conceptually in Figure 1(a), in which the main system is owned by one entity.



(a) Single System

(b) System of Systems

Figure 1: Contrasting the Single Systems Perspective (a) with the SoS Perspective (b).

[3] Blanchard, B. S. & Fabrycky, W. J., 2010. Systems Engineering and Analysis. 5th ed. Prentice Hall. An engineered system has an operational objective and a functional purpose. The operational objective is essentially the outcome the system must achieve.

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

In Figure 1(b), a SoS is depicted; the operational objective is achievable only as a result of interoperation of several systems (in an appropriate configuration). Each constituent system has its own owner, and its own individual operational objective (OO1, OO2, etc.), but, in the context of SoS, the individual operational objectives are subordinated to the overall operational objective. Constituent systems may have their individual enabling systems, needed to achieve the individual operation purposes, but some of the enabling systems may also be shared. The enabling systems can form a part of the SoS. Some constituent systems may be shared with other SoS.

2.1 Recognising that the System is a SoS

A SoS-minded workforce, both developers and operators, recognises that the system under their purview is a participant in the overall purpose of the SoS, and that overall performance must be assessed at a higher level than the constituent system. They recognise that changes to their system will affect other parts of the SoS and that their system will be impacted by changes elsewhere in the SoS. Understanding, or at least estimating, what those effects might be depends upon situational awareness in a SoS, which is discussed in Henshaw[5] and briefly below.

Because an individual operator within a SoS has more complete knowledge of some constituent systems than others, it is important to understand the assumptions upon which the operator makes decisions. Educationalists speak of so-called double-loop learning[6] in which corrections to errors are achieved by adjustment to an organisation's underlying norms, policies, or objectives, instead of simply making an adjustment within the current organisational frame. Similarly, a SoS-minded person will question their own assumptions in order to appreciate that the SoS may not operate as expected by consideration only of the systems they know well. Much engineering training teaches us to develop systems in isolation as stand-alone entities; SoSE should teach us to develop constituent systems with their set of interactions in mind. Even with a SoS perspective in mind, it is important to appreciate that there are additional "un-designed" interactions that may take place[7].

The opportunities of SoS must be recognised by developers and procurers. Generally, the customer will specify that they want a system that will do A, B, and C, without realising that there are systems already built that do B and C, so that what is required is a system to do A and appropriate interfacing to achieve A, B, and C. The ability to create new functional purposes from extant systems that are interoperated in new ways is a necessity in times of austerity or as a matter of expediency.

A service-based approach is helpful. In this approach, capability is realised through the several contributions of services[8]; each constituent system can be considered to deliver one or more of the services required for the capability realisation. In this way, decision makers in the SoS can focus on the services, rather than the internal functionalities of the constituent systems. Furthermore, reconfiguration of the SoS can be based on acquiring the services needed, perhaps for through-life management of the SoS, or to provide redundancy in case of failures. To some extent, such an approach could be considered black-box in nature (i.e. only the interfaces, not the internal working, of the constituent systems are known); under such circumstances there must be a high degree of trust in the service providers (as noted below in Section 3.0).

[5] Henshaw, M., 2015, A Socio-Technical Perspective on SoS, In: SCI-276 Lecture Series. CSO.

[6] Argyris, C., & Schön, D., 1978, Organizational learning: A theory of action perspective, Reading, Mass: Addison Wesley.

[7] Hinsley, S., Henshaw, M., and Siemieniuch, C., 2014, Maintaining Systems-of-Systems Fit-For-Purpose, INCOSE Int. Symp., Las Vegas, US, Jul.14.

[8] See for example: Liu, L., Russell, D., Webster, D., Luo, Z., Venters, C., Xu, J., Davies, J.K., 2009, Delivering sustainable capability on evolutionary service-oriented architecture, ISORC'09. IEEE Int. Symp. Object/Component/Service-Oriented Real-Time Distributed Computing.

2.2 Recognising the Type of SoS

Having determined that the system in question is a SoS, it is useful then to consider what type of SoS[9] it is and factor that into the Systems Engineering approach. It is noted that SoS can rarely be considered to be one of the types (directed, acknowledged, collaborative, virtual, or accidental) exclusively, but that it will feature examples of more than one type. Nevertheless, there will be a predominant type associated with the operational objective of the SoS that should be identified. The range of options available to the SoS engineer or operator will be different according to the authority arrangements within the SoS.

A common error is to assume that the type is *directed* (because that is like a single system) when it is either *acknowledged* or *collaborative*. As an example, the UK Health Ministry[10] aimed to 'ensure every NHS patient had an individual electronic care record which could be rapidly transmitted between different parts of the NHS, in order to make accurate patient records available to NHS staff at all times.' The ambitious programme to connect many systems was an expensive failure (nearly £12Bn spend at time of project cancellation). One of the three main causes of failure identified by Campion-Awwad et al.[11] was that 'in an effort to reduce costs and ensure swift uptake at the local levels, the government pursued an overambitious and unwieldy centralised model, without giving consideration to how this would impact user satisfaction and confidentiality issues'; i.e. the programme was assumed to be a directed (centralised) SoS, whereas local decision-makers had a high degree of autonomy in deciding the choices available to them.

3.0 ORGANISATIONS AND TRUST

3.1 Relationships Between Systems Owning/Operating Organisations

The chapter on the socio-technical perspective[12] has highlighted the significance of organisations for SoS, because of the main distinguishing characteristics of managerial and operational independence. Brook[13] has discussed the nature of enterprises with respect to SoS, and given consideration to NATO as an enterprise. It is clear, then that social, political, economic and other non-technical considerations must be understood as fundamental to the development and operation of SoS. It is essential, therefore, that organisations that contribute systems to a SoS must have a team with multi-disciplinary expertise to properly manage the emergent behaviours.

Another perspective on a SoS is to think of it as being a network of relationships (Figure 2); the relationships between some systems will be formal contracts of one type or another, but between others the relationship may be informal, perhaps based on trust.

[9] Dahmann, J. S., 2015. Systems of Systems Characterization and Types. In: SCI-276 Lecture Series. CSO.

[10] HM Government, 2011, The National Programme for IT in the NHS: an update on the delivery of detailed care records systems, 45th report of 2010-12 session of the House of Commons Public Accounts Committee, HC 1070.

[11] Oliver Campion-Awwad, Alexander Hayton, Leila Smith and Mark Vuaran, Feb. 2014, The National Programme for IT in the NHS - Case History, MPhil Public Policy 2014, University of Cambridge, http://www.cl.cam.ac.uk/~rja14/Papers/npfit-mpp-2014-case-history.pdf.

[12] Henshaw, M. op.cit.

[13] Brook, P., 2015, Enterprise and the Technology Environment, In: SCI-276 Lecture Series. CSO.

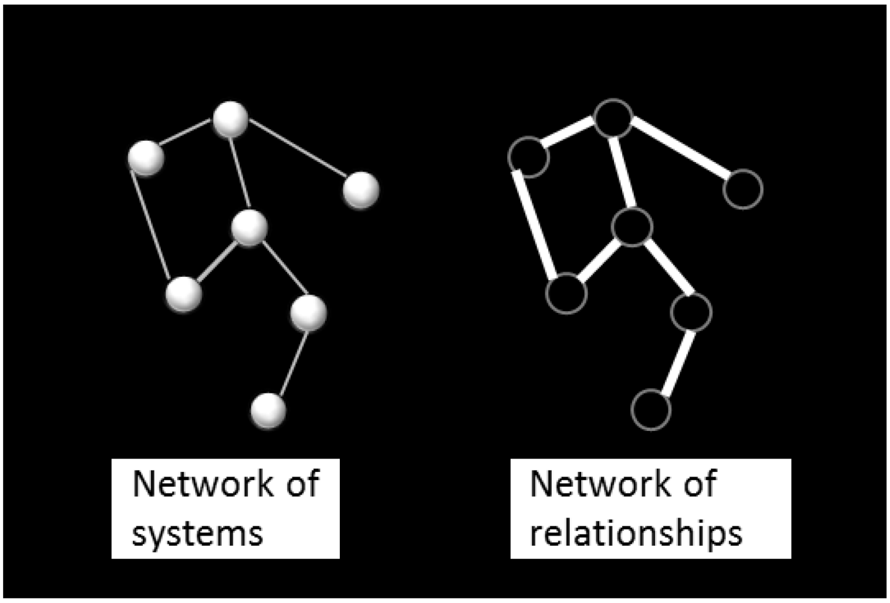


Figure 2: The SoS can be Thought of as a Network of Systems or as a Network of Relationships.

For network enabled capabilities, the network connections in the SoS are based on interoperability through network transport (data), information services, or people processes and applications[14]. Generally there are standards available at each level, but standardising at the upper level (people, processes and applications) is very difficult as this may involve harmonising doctrine, or sharing political or business objectives. Generally, the lower levels of data interoperability, etc., can be specified and easily monitored, but further up the interoperability spectrum[15] there is more ambiguity in specification and, indeed, the higher interoperability components may not be specified at all, but taken on trust (or assumption).

It is good practice to ensure that the full scope of relationships between systems is understood and, wherever possible, formally agreed. Operators immersed in the SoS should acknowledge and understand the trusts upon which they rely and the trusts under which they are expected to act in fulfilling the objectives of the SoS. On the one hand, openness with collaborating systems is meritorious, because it enables increased shared awareness of the operation of the SoS; on the other hand, operators are trusted to protect information that they receive through the SoS network.

3.2 Situational Awareness

The power outage case study, briefly described in Section 2.2 of the chapter on socio-technical perspectives[16], shows that one of the main causes of difficulty in operating SoS is a lack of shared situational awareness by participants in the SoS. This is related to the interoperability mechanisms challenge noted above, but is also determined by the willingness of participants in the SoS to share information. Situational awareness requires the orderly provision of appropriate information, therefore, it is good practice to share information about the current situation with other participants in the SoS. An operator of one system in a SoS should make other operators aware of the status of his/her system, and of what actions he/she plans to take next.

[14] NCOIC Interoperability Framework, 2007. https://www.ncoic.org/home.

[15] See Henshaw, M. 2015 op. cit. Figure 2.

[16] Henshaw, 2015, op. cit.

4.0 LIFECYCLE AND RE-USE IN SoS

Traditionally, systems development begins with a set of requirements and proceeds through a lifecycle, such as the typical one described in ISO15288[17], to retirement, as shown conceptually in Figure 3. The utilisation stage must be accompanied by a support stage which, together, constitute and in-service stage for the system.

Figure 3: Generic Lifecycle of a Single System as Described in ISO15288[18].

A SoS, on the other hand, develops over time from independently managed systems; in effect concept, development, production, in-service, and retirement stages are going on simultaneously for the SoS, but with the focus being on individual constituent systems. Figure 4 describes this notionally. The SoS may contain some systems that are ongoing over long periods of time, others may be introduced in a planned way, perhaps to replace systems that are retiring, or simply to add new capabilities to the SoS. Still other systems may be used during their individual in-service stage for short periods of time to meet particular needs. The time could be as short as a phase within a single mission (e.g. calling up data from a sensor system not generally used). During the course of time, some existing member systems of the SoS may be upgraded; this could provide additional capabilities for the SoS in question or, alternatively, it could make no difference, if the upgrade supports a use outside of the SoS under consideration.

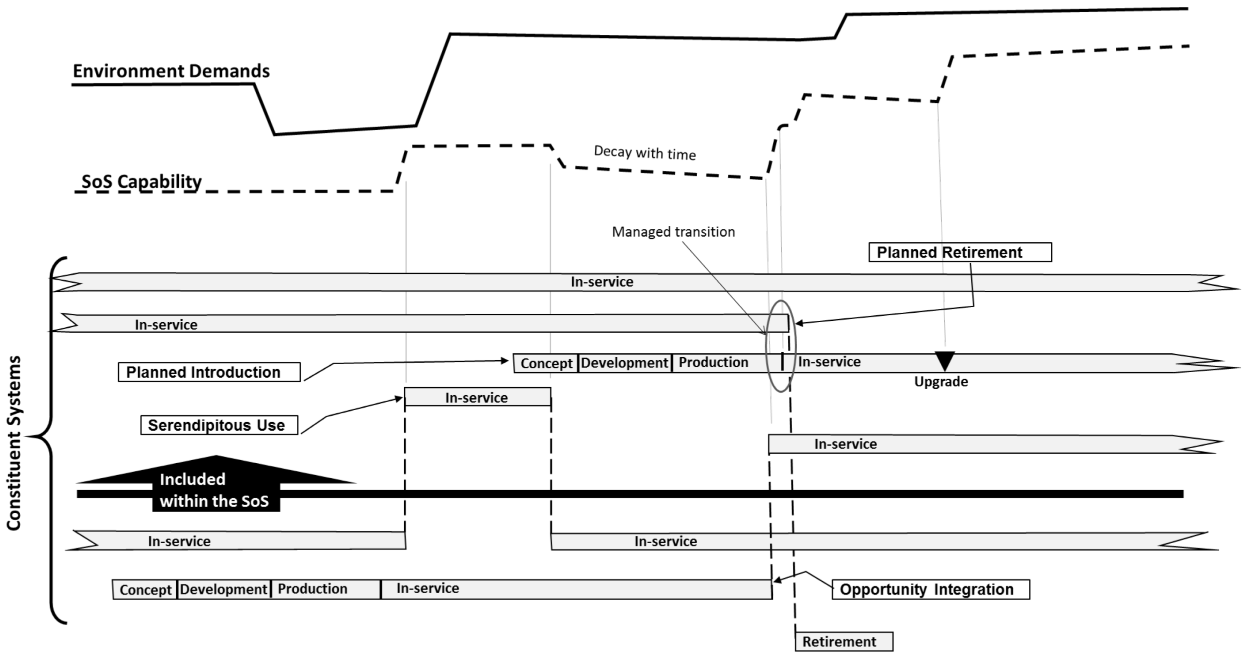


Figure 4: Conceptual Introduction and Removal of Constituent Systems from a SoS.

Reconfiguration of the SoS, which may occur rapidly or over long periods, is carried out in response to the environmental factors (e.g. threat). The level and type of planning of reconfiguration depends on the type of

---

17 ISO/IEC_15288, 2008. Systems and software engineering — System life cycle processes, USA: ISO.

18 Ibid.

the SoS. The lack of a clean sheet means that the benefit of changes, or the introduction of new systems, must be traded against the cost to changes required to extant systems in order to accommodate the new systems. If the new systems are, themselves, complex then their introduction must be informed by analysis of potential interactions that could lead to faults or failures.

The starting point for introducing new systems must, therefore, be a thorough understanding of the SoS in question. For this to be possible, comprehensive and up to date architectural information must be maintained.

The introduction of new systems should be carried out in such a way that the SoS remains as open as practically realisable to the introduction of further new systems later.

It should be noted that the environmental demands and SoS capability profiles in Figure 4 are arbitrary and are intended only to indicate that capability needs and the response to those needs are dynamic.

5.0 OPEN ARCHITECTURES

The essence of SoS is effective interoperability between constituent systems. One way that this can be enabled is through the use of open architectures. Agility has been defined as: *the capability to successfully effect, cope with and/or exploit changes in circumstances*[19]. Reconfiguration of a SoS is generally carried out to improve its effectiveness, and agility is a key measure of that. To be agile in a reconfiguration sense requires the organiser of the SoS to a) predict accurately and in a timely fashion the effect of re-organising the constituent systems, and b) the ability to carry out the reorganisation in a timely and accurate fashion. Open architectures is a means through which agility can be achieved[20].

Within defence systems, one can conceive three types of agility:

- Operational agility is required to enable agile mission groups to configure and reconfigure available assets to meet rapidly changing operational requirements.
- Technical agility is required to enable more rapid and effective upgrade of systems, especially in terms of technology insertion. This should support operational agility.
- Commercial agility is required to achieve value and innovation in procurement.

All three types may be enabled by open architectures. An open architecture *is an open specification of the architecture of a system or system of systems for the purpose of acquiring specified capabilities. As a general feature of good design [for a system or system of systems], an open system architecture should allow for easy improvement and update of system capabilities by adding or changing components.*[21]

There are commercial challenges associated with the use of open architectures; most notably that the revealing of the internal architectures of component systems is likely to compromise the intellectual property rights of the supplier organisation. However, there is evidence that provided one pays attention to the relationships within the overall defence enterprise (government customer and industrial suppliers), an equitable supply chain structure can be achieved in which openness is a fundamental tenet of the various relationships.

There are thus, two elements of good practice associated with architecting for SoS:

[19] NATO STO 2014, C2 Agility, Task Group SAS-085 Final Report , STO-TR-SAS-085.

[20] Henshaw, M. (ed.) ...et al. (2011) Assessment of open architectures within defence procurement issue 1: systems of systems approach community forum working group 1 - open systems and architectures. London: Crown owned copyright, https://dspace.lboro.ac.uk/2134/8828.

[21] Ibid.

- Use modular design, which is a technical requirement such that parts of the system are architected for straightforward replacement and integration with other modules. A skilful architect will be able to partition the architecture into a family of appropriately sized and specified modules, such that changes can be introduced in an agile and cost effective manner.
- Adopt an open architectures procurement strategy, which is a commercial requirement such that sufficient information is published about the module architectures that another the module (or system) provided by one organisation could be directly replaced by that provided by another. A further benefit (and perhaps more important benefit from an operational perspective) is that interfaces between open architecture modules are more easily established such that the connected set of modules has a lower risk of unexpected behaviours.

6.0 GOOD SOS BEHAVIOURS IN PROCUREMENT

The UK MoD has published nine principles under its "Systems of Systems Approach"; these concern the relationship between supplier and customer and draw significantly on the principles of openness discussed above. The SoSA principles have been published in various sources and are listed in Appendix A of Henshaw et al.[22]. Some are briefly discussed below as applicable to all defence procurement enterprises.

The first principle concerns unifying the defence enterprise; this is concerned with collaborative behaviours between customer and supplier and is based on establishing a governance framework that ensures collaborative goals and priorities from the customer side.

The third principle encourages providers of solutions to minimise diversity. In essence organisations are encouraged to minimise the number of different systems that perform the same (or very similar) task. This will enable more rapid reconfiguration when systems must be replaced or supplemented in the SoS.

The fourth principle encourages design for reuse. In times of austerity this is particularly important, but there is another aspect which concerns using tried and tested systems so that the SoS will have a lower risk of unreliability (especially in terms of failed interoperability) because new systems need extensive testing before introduction. The fifth principle is related to the fourth, and concerns building with proven solutions, rather than always starting with a new design.

Principle seven is to design for flexible interoperability. This is both a technical and a process concern and aims to maximise the opportunity of new systems being deployable within a variety of extant SoS.

The eighth principle is to design using open standards. This will enable better interoperability of component systems, because the interface specifications are open.

The principles above must be balanced by the need to maintain secure systems, but broadly the adoption of open and collaborative approaches is an essential ingredient in the development of effective SoS.

7.0 SUMMARY

A summary of the good practice for SoSE is as follows:

- Ensure that participants in the SoS recognise that it is a SoS, as opposed to a single system. This can be achieved through education and encouraging critical thinking.
- Encourage a service based view of the SoS; this will help with understanding the manner in which SoS realise capabilities and will also support better identification of options for reconfiguration to meet changing circumstances.

[22] Ibid.

- Understand which type of SoS is under consideration. This is essential for understanding authority and, hence, decision making structures in the SoS. It is also important from the point of view of understanding where ownership of risk resides in the SoS.
- Focus on relationships between systems; this is the SoS perspective, in which the SoS Engineer is concerned with interactions between systems rather than the inner working of the constituent systems.
    - Both formal and informal relationships are present in a SoS; trust is an important aspect of operating SoS, it is a significant issue with regard to the management of information upon which decisions are made.
- Share information with other participants to give them situational awareness. It is important that participants in different parts of the SoS have a shared understanding of its operation. It is good practice to ensure that the status of your system and what its next actions will be is communicated to other system owners who will be affected by operation of your system.
- Recognise that many issues for SoS are non-technical (i.e. social, political, economic, etc.). Much of the good practice for SoSE is focused on the non-technical aspects; not because the technical aspects are unimportant, but because it is the non-technical aspects that mainly give rise to failures and unexpected emergent behaviours.
- Thoroughly understand the extant SoS, including well documented and up-to-date architecture. In order to make decisions about reconfiguration of the SoS to meet prevailing conditions, it is essential that the behaviour of the extant SoS is understood, so that appropriate reconfiguration options can be considered and the resultant emergent behaviours can be predicted, or at least anticipated.
- Build constituent systems on the basis of modular architecture, because this makes changes easier to implement and may allow more rapid reconfiguration.
- Use an open architecture approach whenever feasible. Open architectures enable commercial, technical, and operational agility in SoS.
- Use open standards to reduce the risk of interoperability problems between constituent systems.
- Re-use systems and solutions wherever possible (including architecture patterns); tried and tested solutions should be deployed to reduce risks, reduce the costs (associated with new developments), and reduce diversity so that reconfiguration can be achieved rapidly.

8.0 ACKNOWLEDGEMENTS

The author is grateful to Dr. Judith Dahmann and the INCOSE SoS Working Group, to Prof. Carys Siemieniuch and Dr. Murray Sinclair for many helpful insights and to the whole ESoS group for research contributions that have informed this paper.