International Journal of Engineering & Technology, 7(4.16) (2018) 24-27

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Contribution to Systems-of-Systems Concept Standardization

Ilyas Ed-daoui[1,3]*, Abdelkhalak El Hami[1], Mhamed Itmi[2], Nabil Hmina[3], Tomader Mazri[4]

[1]LMN Laboratory, INSA Rouen Normandy, University of Rouen Normandy,
685 University Av., 76800 Saint-Etienne-du-Rouvray, France
[2]LITIS Laboratory, INSA Rouen Normandy, University of Rouen Normandy,
685 University Av., 76800 Saint-Etienne-du-Rouvray, France
[3]LGS Laboratory, ENSA Kenitra, Ibn Tofail University
BP 242, University Av., Kenitra 14 000, Morocco
[4]ETSE Laboratory, ENSA Kenitra, Ibn Tofail University
BP 242, University Av., Kenitra 14 000, Morocco
**Corresponding author E-mail: edd.ilyas@gmail.com*

Abstract

This paper aims to continue the work towards the standardization of the concept of systems-of-systems (SoS). A notion that has been around for quite a while, but still difficult to unanimously agree on one definition for it. Correspondingly, we collected some SoS definitions from literature in order to point out the similarities between them. We present a set of SoS characteristics that differentiate them from classic and complex systems. A classification of SoS is also detailed. Eventually, an application is presented in order to emphasize the importance of the classification.

Keywords: *Systems-of-systems (SoS); SoS classification; SoS characteristics; SoS classification.*

1. Introduction

A system-of-systems (SoS) may be designed in order to accommodate diverse and changing missions, that cannot be defined in preliminary phases with precision [1]. It may also be formed and organized dynamically so as to achieve a set of targets [11], [12].
Designing such systems is a tremendously challenging task. These systems exist in diverse domains as enterprises, healthcare, telecommunication, aerospace, military, markets, etc. Besides, today's application exploits the infrastructure of SoS in order to support computation and data storage [2].
Application areas of SoS are vast indeed. They embrace software systems like the Internet, cloud computing and cyber-physical systems all the way to hardware dominated cases like energy, transportation, etc. [3], [12], [13], [15].
Consequently, a new discipline has emerged, it is called SoS engineering (SoSE). A discussion of SoSE is included in [4] (section 4.2.6): "SoS 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."
Actually, this discipline develops and becomes more mature every year as there is a growing interest in SoSE standardization. This includes more convergence on definitions and fundamental principles [5]. Hence, this would establish a fructuous basis for more consistent and effective research and application of the theories.
In this paper, we aim to emphasize the importance of the classification of SoS. This said having the same characteristics does not imply talking about the same SoS and the same processes within them. Accordingly, our proposition sires two major SoS classes: directed and open that we explain further.
The remaining part of the paper is organized as follows:

- A background of SoS definitions and properties is detailed, respectively, in section 2 and section 3.
- The SoS typology is explained in section 4.
- An application of the theory is explained in section 5.

2. SoS definitions

SoS have received extensive attention from science communities in the past years. And numerous definitions were proposed to sire this concept. In table 1, we enumerate some of the numerous proposed definitions of SoS.

Table 1: SoS definitions

Reference	Proposed definition for SoS
[4]	"SoS is a large-scale composite system, which can realize specific function"
[6]	"SoS are a collection of systems, each capable of independent operation, that interoperate together to achieve additional desired capabilities"
[7]	"SoS are integrated, independently operating systems working in a cooperative mode to achieve a higher performance."
[8]	"SoS are special systems, they are composed of systems which can run independently and have their own benefits and value. Once the element system is put into the SoS, its independence still exists and the interactions among the systems are frequent."
[9]	"SoS as a collection of systems that must have two features: its components must be able to operate independently by the whole system and they do operate independently, being managed at least in part for their own purpose"
[10]	"SoS is a collection of systems, can achieve the objective which a single system cannot achieve. Every system can operate independently to achieve its own objective. SoS have emergence properties"

Despite the fact that the term SoS has been around for quite a while, we still seem to be struggling with the concept. Jamshidi quoted approvingly from the claim in Sage and Cuppan [16] that there is no universally accepted definition of systems of systems. Besides most definitions of SoS are not very helpful and some of them are even harmful [17].
Here is our definition to SoS, we esteem that it completes previous definitions in literature: “SoS is an evolving synergy of heterogeneous, autonomous, distributed, interdependent, sometimes complex and integrated systems that interact in order to achieve a complex and evolving target that exceeds the sum of the parts.”
It is worth noting that SoS integrate independently operating, non-homogeneous systems to achieve a higher goal than just the sum of the parts [17], [14], [15].

3. SoS Characteristics

In this subsection, we reveal a set of properties that differentiate SoS from ordinary systems. The more the system looks in coherence with them, the more it is considered as a SoS. They are based on a survey of relevant literature.

Table 2: SoS characteristics

Characteristic	Definition
Autonomy	Autonomy is sired by two notions. One is self-directness and it refers to the independence of the component system from external involvement to perform correctly. The second is self-sufficiency which refers to non-reliance on external factors for the satisfaction of component system’s needs [18], [19], [20].
Heterogeneity	SoS should support the miscellany of natures of component systems in addition to their operation on different time scales. They should be diverse in terms of resources, functionalities and capabilities. The performance of the whole should not be affected by the divergence regarding the nature and operation schedule of its components.
Interdependence	Interdependence is concerned by the ability of a collection of systems to share, exchange and correctly interpret information, material and sometimes even energy, in order to achieve a common target in a given context respecting some rules of interaction [21], [22].
Distribution	One of the most important properties of SoS is physical distribution. Component systems are not forced to be in the same geographic locus in order to perform. The geographic extent of SoS is large and nebulous [7]. Information, tasks and capabilities are distributed amid the SoS according to some rules.
Extensibility	SoS have no fixed architecture. The infrastructure of SoS may evolve, extend or reduce at any time [17]. SoS are never finished; they have the tendency to evolve continually. This is due to the change in the environment [17].
Emergence	Emergence has been a subject of controversy for academic researchers. There are some authors, as in [7], [23] and [24], who see that there is no concise definition of emergence. And there are others, as in [25], who said that it represents behaviours that differ from the collective properties of component systems forming the SoS. They emerge from the cumulative interactions amid the SoS and can have positive/negative effects.

4. SoS Classification

We believe it is necessary to elucidate the concept of SoS. This process should start with a clear classification that will form a foundation of SoS engineering.
We propose a taxonomy based on the level of both management centralization and systems’ operational freedom in addition to its ability to change. Our work is based on the taxonomy proposed in [1]. The boundaries between these types can be defined in terms of the degree of operational and managerial independence of the components.
We see that there are two important classes of SoS, which are:

- Directed SoS
- Open SoS

It is important to note that this typology has a complementary role to SoS’ characteristics.

4.1. Directed SoS

Directed SoS should have well-defined objectives. They are built in order to achieve specific purposes. They are centrally managed during long-term operation to achieve the sought objectives. This does not affect the autonomy of component systems and their interdependencies.
In fact, there could be a hierarchy of changing targets amid the SoS. However, there are some targets that are the most crucial and stand for the intent of the building of the SoS. They represent the sought solutions that all the infrastructure is assembled to achieve.
Below these targets, there could be a descending hierarchy of objectives. The idea behind it is to decompose a complex target into several less complex objectives. The decomposition of objectives ends when it reaches simple, directed and well-defined tasks that should be assigned to the autonomous component systems.

4.2. Open SoS

Unlike directed SoS, open SoS have neither a management authority nor a centrally agreed-upon purpose. They have targets that component systems interact more or less voluntarily to achieve. Besides, component systems may integrate or exit dynamically the SoS based on mission requirements.
However, they also have a descending hierarchy of objectives that are decentralized. The decomposition of objectives ends when it reaches simple, directed and well-defined tasks that should be assigned to the autonomous component systems.

5. Application

We believe that before we get to the integration process, it is important to fragment SoS and elucidate this concept. A classification of SoS is proposed further in this paper. Then, we introduce an integration process for each one of SoS classes.
Besides, we see that the integration is a good example to explain the difference between directed and open SoS as they adopt different approaches to handle the process.

5.1. Integration in Directed SoS

A system willing to integrate a SoS should have a predefined task that contributes to the achievement of the objective of the SoS. Accordingly, the verification of the task and the choice of the assigned objective are assured by the management authority.

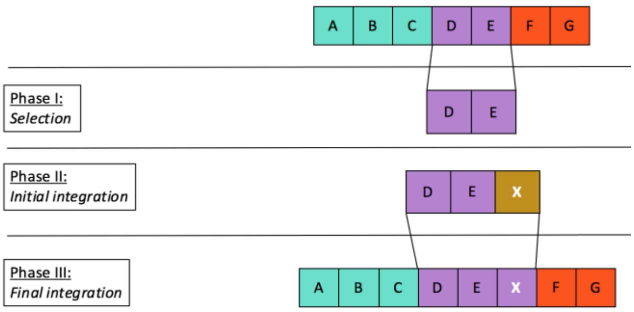

Fig. 1: Integration process in directed SoS.

As illustrated in Fig. 1, the integration process is constituted of three phases. It is worth noting each color represents different objective. And component system 'x' is willing to integrate the SoS by being assigned to the second objective.
Here is a brief presentation of each phase:

- Phase I: is about logically isolating the concerned set of component systems assigned to the implicated target.
- Phase II: is about integrating the new system among the set of component systems assigned to the implicated target (Target 2 in this case). A specific task is chosen and assigned to the new component system 'x'. In addition, in this phase, a replication of the tables of component systems assigned to Target 2 in the system X's table is done. The result is a new table including all systems in the same Target. This table should be shared with component systems of Target 3 so as they update theirs and add the system X to their tables.
- Phase III: is about reintegrating the set of systems inside the SoS.

In directed SoS, a component system neither communicate with the whole SoS during the integration process nor after a successful integration. It only communicates with systems assigned to the same target.

5.2. Integration in Open SoS

The process of integration starts with the transmission of a search message from the component system seeking the integration to the rest of the SoS. Then, an assessment of the message comes from component systems in order to verify the nature, embraced information, security criteria, etc. In the case of a successful scenario, component systems send their tables to the new system.

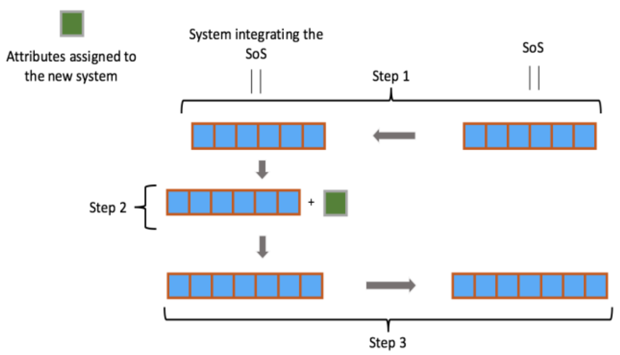

Fig. 2: Integration process in open SoS.

For tables' standardization purposes, the new system duplicates the content of the received tables (which normally have the same content). Next, it injects its information and data in its table. Then, it transmits it to all component systems which eventually will replace their tables with the newly received one. Consequently, all component systems will have a copy of the same table which contributes to the limitation of compatibility issues and the unification of the vision of component systems through the SoS. Fig. 2 illustrates the process.

6. Epilogue

In this paper, we collected some SoS definitions from related literature in order to sire the concept and highlight the similarities between them. We also presented a set of properties that differentiate SoS from ordinary systems. This set embraces autonomy, heterogeneity, interdependence, distribution, extensibility and emergence.
A classification of SoS is also detailed. It is our belief that the fragmentation of SoS is a promising starting point for the realization of effective processes in such complex systems. Besides, the adaptability of the approach pushes forward the capability of component systems to handle interdependent joint activities amid the SoS while keeping the performance viable.

Acknowledgement

This work is a result of a cooperation between four research laboratories LITIS & LMN at INSA Rouen Normandy, France and ETSE & LGS at ENSA Kenitra, Morocco.
This research is supported by the European Union (EU) with the European Regional Development Fund (ERDF) and Normandy Region.

References

[1] A. M. Madni and M. Sievers, "System of systems integration: key considerations and challenges," *Systems Engineering*, Vol. 17, No. 3, (2014), pp. 330-347.
[2] I. H. Krüger, M. Meisinger, and M. Menarini, "Interaction-based runtime verification for systems of systems integration", *Journal of Logic and Computation*, Vol. 20, No. 3, (2010),pp. 725–742.
[3] B. K. Tannahill, and M. Jamshidi, "System of Systems and Big Data analytics–Bridging the gap", *Computers & Electrical Engineering*, Vol. 40, No. 1, (2014), pp. 2-15.
[4] Department of Defense (2004), Defense Acquisition Guidebook. [Online]. Last visit: http://akss.dau.mil/dag/DoD5000.asp?view=framework
[5] J. Dahmann and G. Roedler, "Moving Towards Standardization for System of Systems Engineering", Proc. SoSE'16, pp. 1-6.
[6] Department of Defense, Systems Engineering Guide for System of Systems, version 1.0, 2008.
[7] M. Jamshidi, *Systems of systems engineering – principles and applications*, (2008), London, UK: CRC – Taylor & Francis Publishers.
[8] B. Xia, Q. Zhao, Y. Dou, and C. Zhan, "Robust system portfolio modeling and solving in complex system of systems construction", *Proceeding of 35th Chinese Control Conference*, (2016), pp. 9573-9577.
[9] M. W. Maier, "Architecting principles for systems-of-systems", *Proceeding of International Council on Systems Engineering*, Vol. 1, No. 4, pp. 565-573, July 1996.
[10] B. E. White. Complex Adaptive Systems Engineering (CASE). *Proceeding of the System of Systems Engineering Conference*, (2008).
[11] I. Ed-daoui, T. Mazri, and N. Hmina, "Security Enhancement Architectural Model for IMS based Networks", *Indian Journal of Science and Technology*, Vol. 9, No. 46, (2016).
[12] I. Ed-daoui, T. Mazri, and N. Hmina, *Towards Reliable IMS-based Networks*, (2017), LAP LAMBERT Academic Publishing.
[13] I. Ed-daoui, M. Itmi, A. El Hami, N. Hmina. and T. Mazri, "A deterministic approach for systems-of-systems resilience quantification", *International Journal of Critical Infrastructures*, Vol. 14, No 1, (2018), pp. 80–99.
[14] I. Ed-daoui, A. El Hami, M. Itmi, N. Hmina. and T. Mazri, "Unstructured Peer-to-Peer Systems: Towards Swift Routing", *International Journal of Engineering and Technology*, vol. 7, no 2.3, (2018), pp. 33-36.
[15] I. Ed-daoui, M. Itmi, A. El Hami, N. Hmina. and T. Mazri, "A deterministic approach for systems-of-systems resilience quantification", *International Journal of System of Systems Engineering*, in press.
[16] A. P. Sage and C. D. Cuppan, "On the Systems Engineering and Management of Systems of Systems and Federations of Systems", *Information, Knowledge, Systems Management*, Vol. 2, No. 4, (2001), pp. 325-345.
[17] R. Abbott, "Open at the top; open at the bottom; and continually (but slowly) evolving", *Proceeding of the System of Systems Engineering Conference*, (2006).
[18] J. M. Bradshaw, A. Acquisti, J. Allen, R. B. Maggie, L. Bunch, N. Chambers, P. Feltovich, L. Galescu, M. A. Goodrich, R. Jeffers, M. Johnson, H. Jung, J. Lott, D. R. Olsen Jr., M. Sierhuis, N. Suri, W. Taysom, G. Tonti and A. Uszok, "Teamwork-centered autonomy for extended human-agent interaction in space applications", *Proceeding of AAAI Spring Symposium*, (2004).

[19] J. M. Bradshaw, P. J. Feltovich, H. Jung, S. Kulkarni, W. Taysom, and A. Uszok, “Dimensions of Adjustable Autonomy and Mixed-Initiative Interaction”, *Proceeding of Agents and Computational Autonomy: Potential, Risks, and Solutions*, (2004).
[20] M. Johnson, J. Bradshaw, P. Feltovich, C. Jonker, M. Van Riemsdijk and M. Sierhuis, "Coactive Design: Designing Support for Interdependence in Joint Activity", *Journal of Human-Robot Interaction*, Vol. 3, No. 1, (2014), p. 43.
[21] S. Billaud, N. Daclin, and V.Chapurlat, “Interoperability as a key concept for the control and evolution of the System of Systems (SoS)”, *Proceeding of International IFIP Working Conference on Enterprise Interoperability*, (2015), pp. 53-63.
[22] D. Delaurentis, “Understanding transportation as a system-of-systems design problem”, *Proceeding of 43rd AIAA Aerospace Sciences Meeting and Exhibit*, (2005), pp. 10-13.
[23] C. W. Johnson, “What are Emergent Properties and How Do They Affect the Engineering of Complex Systems?”, [Online], Available: http://www.dcs.gla.ac.uk/~johnson/papers/emergence.pdf
[24] R.I. Damper, “Emergence and Levels of Abstraction”, Editorial for the Special Issue on ‘Emergent properties of Complex Systems’, *International Journal of Systems Science*, Vol. 31, No. 7, (2000), pp. 811-818.
[25] System-of-Systems Navigator: An Approach for Managing System-of-Systems Interoperability, [Online], Available: https://www.researchgate.net/publication/235133171_System-of-Systems_Navigator_An_Approach_for_Managing_System-of-Systems_Interoperability.

Author profile

Ilyas Ed-daoui is currently a Ph.D. Student at LGS laboratory at ENSA-Kenitra, Morocco. He is in an international cotutelle of thesis with LMN laboratory at INSA-Rouen, France. His thesis title is: "The Reliability of Systems-of-Systems". He is also author and co-author of a book and a number of papers published in international journals and conferences.
Ed-daoui obtained his Master's degree in Information Systems' Security at ENSA-Kenitra, Morocco and his Bachelor's degree in Mathematics and Informatics at Mohammed V University of Rabat, Morocco.

Abdelkhalak El Hami is a Full Professor at INSA Rouen, Normandy France, as well as the Deputy Director of LMN and Director of Mechanical Engineers. His research activities include reliability-optimisation systems. He has supervised 38 PhD theses. He also is the author and co-author of more than a 20 books and more than 550 papers published in international journals and conferences. He received his Doctorate in Engineering Sciences from the University of Franche-Comté in France in 1992. He received his Habilitation Diploma to supervise research (HDR) in 2000. He is the Editor-in-Chief of three sets of international books: ISTE, Wiley and Elsevier.

Mhamed Itmi earned his PhD in Probability Theory and Statistics in 1980 and second PhD in Computer Science in 1989. He received his Habilitation Diploma to supervise research (HDR) in 2006 with the focus on the modelling and simulation of distributed discrete event systems. He managed different logistics and transportation research projects and supervised several PhD theses. He also is the author and co-author of more than 100 papers published in international journals, conferences and books. His research presently focuses on autonomous systems. He is an Associate Professor at the INSA-Rouen, France.

Nabil Hmina earned his PhD in Engineering Sciences in University and Central School of Nantes, 1994. He is a Full Professor, the Director of the National School of Applied Sciences of Kenitra since 2011, Vice-President for Academic Affairs and Information Technology of the University Ibn Tofail, 2005–2011 and Director of Laboratory of Systems Engineering since 2012. He is a partner and coordinator of several European projects, organiser, Chairman and a member of program committees and scientific boards of several international and national conferences in the field of computer security, optimisation, embedded systems, and renewable energy. He supervised several PhD theses. He also is the author and co-author of several books and a number of papers published in notorious journals and international conferences.

Tomader Mazri received her HDR degree in Networks and Telecommunication from Ibn Tofail University, PhD in Microelectronics and Telecommunication from Sidi Mohamed Ben Abdellah University and INPT of Rabat, Master’s in Microelectronics and Telecommunication Systems, and Bachelor’s in Telecommunication from the Cadi Ayyad University. She is currently a Professor at the National School of Applied Sciences of Kenitra, a permanent member of Electrical and Telecommunications Engineering Laboratory, author and co-author of 20 articles journals, 30 articles in international conferences, a chapter and three books. Her major research interests are on microwave systems for mobile and radar, smart antennas and mobile network security.