Embedding Interoperability in System of Systems: Definition and Characterization of Fundamental Requirements

Nicolas Daclin

Embedding Interoperability in System of Systems: Definition and Characterization of Fundamental Requirements

Nicolas Daclin

LGI2P – Laboratoire de Génie Informatique et d'Ingénierie de Production
Site de l'Ecole des Mines d'Alès – Parc scientifique G. Besse
30035 Nîmes cedex 1, France
nicolas.daclin@mines-ales.fr

Abstract. The main objective of this communication is to discuss the engineering of a System of Systems (SoS), including interoperability concept. More precisely, the here presented research focuses on the fundamental requirements to consider in a System of Systems Engineering (SoSE) project and that have to be maintain during the entire life cycle of a SoS. First, the concept of interoperability, according to its definition and its characteristics, is presented. Then, the concept of SoS is presented in the same manner. This leads to introduce and present the possible links between System of Systems and interoperability. These links are (1) clarified and defined, (2) re-expressed to meet requirements' definition and (3) not related to a given SoS in order to be generic.

Keywords: System Engineering, System of Systems, interoperability, requirements.

1 Introduction

Global environment, fundamental changes and fast evolution lead organizations and further, our society, to be able to adapt to these constraints (*e.g.* technological, organizational…). To handle this context, the concept of *System of Systems* (SoS) has become essential in order to create added value and to be efficient. More than anything, System of Systems Engineering (SoSE) becomes also essential in order to limit and avoid extra-cost, delay…throughout SoS life cycle *i.e.* not only for its engineering phase but also for its disassembly phase *via* its operational phase. On the other hand, *interoperability* has become a crucial issue to consider for organizations that want to interact in a common relationship. In this way, numerous researches have been initiated and performed from last years. Although System of Systems' characteristics and interoperability characteristics present possible similitude, their connections and the possible advantages to consider these two concepts as complementary are not yet highlighted. The here presented research focuses on the definition and characterization of fundamental requirements to consider in a SoSE project and that have to be maintain during the entire life cycle of SoS and, in compliance with interoperability paradigm.

This paper is structured as follow. After this brief introduction, the needs and issues addressed by our research are described in the second section. Section 3 and 4 present the notions of interoperability as well as SoS, according to their definitions and their characteristics. Section 5 presents a first tentative of definition and characterization of fundamental SoS requirements. The final section presents the conclusion, and the future perspectives for this research.

2   Problematic and Needs

Dealing with System Engineering (SE) means to be able to verify [1] requirements [2] to respect all along the life cycle of the studied system [3]. In the case of SoSE, that means (1) to identify these requirements and, (2) these requirements remain as generic as possible (not related to a specific SoS) in order to be adapted to any SoSE project. First, the concept of System of Systems is emerging from last years. Basically, a SoS can be shown as a system resulting from the interaction of its constituent systems that are themselves independent [4]. On the other hand several researches on systems interoperability [5] have been initiated from last decades to facilitate and to ensure "relationship" basically in terms of sharing and exchange. Both concepts deal with the presence of several systems to put in relation, in order to work together and to reach a final purpose. In this way, it is interesting to analyze potential similitude that can be shared by interoperability and SoS. Precisely, the two concepts deal with characteristics that they have to respect, and these characteristics seem to be closely linked. Furthermore, it is interesting to identify these relations and their related works, in order to facilitate the engineering of SoS. This work attempts to point out:

- If interoperability is a part of SoS then, what fundamental requirement(s) belonging to interoperability can be useful to design SoS;
- A definition and an adaptation of these identified requirements to the specificity of SoS.

Thus, it is necessary to analyze which requirement(s) is not clearly identified and/or defined in order to embed in SoS paradigm to evaluate it.

Secondly, once the requirements are identified, they have to be exploitable either by acquirer or prime contractor. However, requirements are often expressed with natural language giving their use difficult (omission, repetition, ambiguity, conflict) [6]. From a SE point of view, it means that requirements have to be re-expressed to meet formal requirements' definition, in order to avoid problems related to expressivity. Thus, requirements must be clearly expressed, identifiable, traceable, verifiable, unambiguous, and consistent with another requirement. The final purpose is to allow the use of formal verification techniques in order to verify the satisfaction or not of these defined SoS requirements.

2   Interoperability

Numerous initiatives, in different fields (crisis management, military, enterprises, health care, transport…) [7] [8], developed over the past years, have shown that

systems' ability to be interoperable, is a major issue and a key factor for the success of collaboration. Regarding enterprise interoperability, it is defined as the *"ability of enterprises and entities within those enterprises to communicate and to interact effectively"* [9]. Furthermore, to study interoperability, several works have defined the fundamental characteristics to consider to develop interoperability. Thus, according to [10] [11] [12], interoperability can be characterized by the four characteristics:

- *Compatibility.* It represents the ability of partners to ensure interfacing aspect, mainly related to interoperability barriers (conceptual, technological, and organizational).
- *Interoperation.* It represents the ability of partners to achieve a performance level in terms of interactions (quality of exchange, exchange time…), over a partnership.
- *Autonomy.* It represents the ability of a partner to receive or provide services, data, product… while retaining its own operational thinking.
- *Reversibility.* It represents the ability of a partner to be still able to achieve its original objectives, after a partnership, despite adaptations or changes.

Existing approaches to measure and to evaluate interoperability are mainly focused on maturity measurement [13]. In terms of maturity models for interoperability, we can mention the important contribution such as LISI, OIM and LCIM. In manufacturing fields we can also note the MMEI. The LISI proposes a maturity model allowing to define, to measure and to assess the interoperability of Information Systems [14]. The OIM [15] is an extension of the LISI and addresses the evaluation of the interoperability maturity from an organizational point of view. The LCIM [16] considers the evaluation of the conceptual interoperability. Based on these existing maturity models, the MMEI [17] for enterprise interoperability covers all facets of interoperability, according to the conceptual, organizational and technological issues.

Last, it is to note that more formal approaches are developed for the last years. The objective of these researches is to consider interoperability from a formal point of view in order, to verify, to measure and to evaluate it. For instance, [18] defines three main quality attributes (connectivity, information flow and data latency), and their equations in order to measure the efficiency of operational interoperability. [19] takes also an interest in the measurement and the assessment of the operational interoperability. In this way, these works define height modes (directional, self, pure, contextual, time variant, constrained upper bound collaborative and confrontational) and their associated metrics. Finally, works proposed in [20] uses and offers an approach based and supported by formal verification techniques to verify interoperability requirements - according to the main characteristics of interoperability - in a public or private collaborative process. Precisely, the goal is to verify that a given collaborative process satisfies (or not) a set of properties related to interoperability in terms of compatibility, interoperation, autonomy and reversibility and according to a predefined interoperability requirements repository.

3   System of Systems

Basically, a SoS can be defined as "*a set of collaborative integrated systems*" [21]. Numerous definitions of SoS are existing in literature, each one with their own specificities, but the fundamental concept still remains the same [22] [23] [24]; a SoS is a collection of several systems which interact for a given purpose that an isolated system cannot achieve alone. Beyond these basic definitions, a SoS presents some characteristics that define and differentiate it from a simple system. Indeed, among the numerous definitions, [25] highlights five fundamental characteristics that define a System of Systems (known as Maier's criterion):

- *Operational independence.* It represents the ability of a given system to operate independently and efficiently if the SoS is disassembled.
- *Managerial independence.* It represents the ability of a given system to keep and to continue its operational purpose, while it is integrated to SoS.
- *Distribution.* The set of systems that compose the SoS are geographically distributed over a large extent.
- *Evolutionary.* The development and existence of SoS is evolutionary. Functions and purposes can be added/ removed/ modified.
- *Emergence.* The SoS performs functions and achieves purpose that component cannot fulfill independently.

Nowadays, the concept of SoS is widely studied and deployed in numerous fields where several systems have to interact [26] [27] [28]. As in interoperability fields, these study consider SoS paradigm under a formal point of view or not. For instance, [28] considers SoS paradigm in order to participate to the Spatial Data Infrastructure (SDI) life cycle. [27] adopts a more formal vision of the SoS applied to crisis management. Its work focuses on the definition of coupling matrix - according coupled systems - in order to ensure (and to secure) the services provided by SoS, despite its evolution. Finally, [26] proposes methods and tools based on state modeling and simulation in order to evaluate operational performance effectiveness of the SoS.

According to the previous definitions, Systems Interoperability concept and System of Systems concept share common basic characteristics. Indeed, SoS is developed when several systems (human, technological, organizational) are connected, exchange, and share, in order to work together and to reach a final purpose. Moreover, the characteristics of interoperability such as autonomy and reversibility characterize also a SoS commonly as managerial and operational independence. Despite, the fact that the word "*interoperability*" is not always mentioned in literature when we talk about SoS, interoperability seems to be a property that a SoS must fulfill. Last, SoS is a broader concept than interoperability, so, a *SoS is based (in part) on interoperable systems but interoperable systems do not constitute a SoS*. Among characteristics of SoS and interoperability someone –such as managerial and operational independence - are more or less studied due to (1) a lack of definition of these requirements (they stay often at a high level of abstraction) and (2) to the difficulties to evaluate them formally. In this way, we are taking an interest in the definition and the characterization of the two before-mentioned characteristics belonging both to interoperability and SoS.

4   Requirements Definition and Characterization

The objective is to clarify and to propose first definitions and formalizations of the two SoS requirements, such as "*Managerial independence*" and "*Operational independence*". These definitions are based on the characterization of these requirements in SoS and interoperability paradigm. This formalization is not specific to any SoS and, furthermore, can act as basis to specify and precise these fundamental requirements for a specific SoS or other concepts that integrate either autonomy and/or reversibility.

Let S the set of systems that constitute the System of Systems:

$$S = \{s_i | 1 \leq i \leq N\}, N, i \in \mathbb{N}^*$$

Let $\prod$ the set of moment (time) of the System of Systems' life cycle:

$$\prod = \{\pi | \pi = post \lor per \lor pre\}$$

Where:
- *pre*: the time before the assembling of the System of Systems.
- *per*: the time when the System of Systems is existing and fulfill its purpose.
- *post*: the time when a system component of a System of Systems is disassembled at the end of the System of Systems existence.

Let F the set of functions[1] of a system at a given moment of the SoS life cycle:

$$F(\pi)_{s_i} = \left\{f(\pi)_i \big| 1 \leq i \leq N\right\}, N, i \in \mathbb{N}^*$$

Where:
- $f(\pi)_i \in \left\{executable, \overline{executable}\right\}$ indicates that a given function of the system is executable, *i.e.* the function is able deliver its services, products… or non executable *i.e.* the function is unable.

Let P the set of performances of a given system at a given moment of the SoS life cycle:

$$P(\pi)_{s_i} = \left\{p(\pi)_i \big| 1 \leq i \leq N\right\}, N, i \in \mathbb{N}^*$$

Let E the set of admissible variations of a given performance *p* of a given system:

$$E = \{\varepsilon_i | 1 \leq i \leq N\}, N, i \in \mathbb{N}^*$$

The admissible variations of the performance can be a loss of performance (*e.g.* increasing of the time of an activity) or a gain of performance (*e.g.* decreasing of the time of an activity).

Managerial Independence (autonomy)
We define managerial independence requirement as: "*the ability for each system, that contribute to the SoS, to satisfy its own performance[2] and integrity[3] during SoS existence*". This statement is formalized by the following equation:

$$MI = Perf\,(per) \land Int(per)$$

Where:
- $MI \in \{true, false\}$, is the property of the managerial independence of the SoS. Its value depends of the property of performance and the property of integrity during the existence of the SoS.

---

[1] A function is "*a task, action or activity performed to achieve a desired outcome*".[29]
[2] Performance is "*the ability of a system to reach its objectives*".[30]
[3] Integrity is "*the ability of a system to stay coherent and to be able to ensure its functions*".[30]

- $Int(per) \in \{true, false\}$, is the integrity of a system during the existence of the SoS such as:

$$Int(per) = \begin{cases} true \; iff \; F(per)_{s_i} = executable \\ false \; otherwise \end{cases}$$

Integrity is considered as true if and only if all the functions of a system during its participation to the SoS are executable. It means, for instance, that if a given resource of the system cannot perform its own activity because it is involved in the SoS, this activity is still executable by the system.

- $Perf(per) \in \{true, false\}$, is the performance of a system during the existence of the SoS such as:

$$Perf(per) = \begin{cases} true \; iff \begin{cases} P(per)_{s_i} = P(pre)_{s_i} \; if \; E = \emptyset \\ V \\ P(per)_{s_i} = P(pre)_{s_i} \; \pm E \; otherwise \end{cases} \\ false \; otherwise \end{cases}$$

Performance is considered as true if and only if the system can reach its own objectives during its participation to the SoS.

Operational Independence (reversibility)

We define operational independence requirement as: "*the ability for each system to satisfy performance and integrity after its disassembling from the SoS*". This statement is specified by the following equation:

$$OI = Perf(post) \wedge Int(post)$$

Where:

- $OI \in \{true, false\}$, is the property of operational independence of the SoS. Its value depends of the property of performance and the property of integrity of a given system components after the disassembling of the SoS.
- $Int(post) \in \{true, false\}$, is the integrity of a system component after the SoS is disassembled, such as:

$$Int(post) = \begin{cases} true \; iff \; F(post)_{s_i} = executable \\ false \; otherwise \end{cases}$$

Integrity is considered as true if and only if all the own functions of the system after it is disassembled of the SoS, are executable. It means, for instance, that if a given resource is retrieve (or not) by the system to perform it original activity, this activity is executable by the system.

- $Perf(post) \in \{true, false\}$, is the performance of a system component after the SoS is disassembled, such as:

$$Perf(post) = \begin{cases} true \; iff \begin{cases} P(post)_{s_i} = P(pre)_{s_i} \; if \; E = \emptyset \\ V \\ P(post)_{s_i} = P(pre)_{s_i} \; \pm E \; otherwise \end{cases} \\ false \; otherwise \end{cases}$$

Performance is considered as true if and only if the system can reach its own objectives after it is disassembled of the SoS.

This first formalization considers only two requirements that defines interoperability and that are existing in SoS. From a Collaborative Network Organizations point of view, it would be interesting to decompose and specify these two requirements for Virtual Organization [31] where autonomy and reversibility take

a preponderant part in consequence of its temporary aspect (creation on business opportunity, operation, disassembling). Furthermore, it will be essential, to consider other requirements. For instance, the characteristic of emergence is primordial so that the SoS performs its functions and achieves its purpose. Indeed, it is important to make sure of (and to maintain) the emergence of "good" properties (or behavior) expected or not and, to anticipate and to eradicate the emergence of "bad" properties on (1) the SoS itself and (2) on the components.

5 Conclusion and Prospects

In collaborative context SoS Engineering takes a preponderant part to make SoS as efficient as possible. SoS presents characteristics that have to be satisfy during all its lifecycle phases. This paper has presented a first rapprochement with another concept related to collaboration between systems *i.e.* interoperability. Common characteristics such as operational/managerial independence for SoS and autonomy/reversibility for interoperability are closely related but not yet clearly defined and studied. In this way the first goal was to precise and clarify this characteristics, beyond their basics definitions and, in order to be studied deeper, as shown in this communication. Future work is related to the definition of criteria that have to allow to fully characterize these requirements in order to be formally proven.

References

1. ISO 8402: Quality management and quality assurance, Vocabulary, Second edition 1994-04-01, International Standard Organization, 1994.
2. Scucanec, S.J, Van Gaasbeek, J.R.: A day in the life of a verification requirement, U.S Air Force T&E Days, Los Angeles, California, February 2008.
3. INCOSE: System Engineering (SE) Handbook Working Group, System Engineering Handbook, A « How To ». Version 3.1, Guide For All Engineers, 2007.
4. Anderson, D.J., Campbell, J. E., Chapman L. D.: Evaluating a complex System of Systems using state modeling and simulation, National Defense Industrial Association Systems Engineering Conference, San Diego, USA, 2003.
5. IEEE: A compilation of IEEE standard computer glossaries, Standard computer dictionary, New York, 1990.
6. ISO/IEC 15288:2008(E): IEEE Standards 15288.2008 – Systems engineering – System life cycle processes (2nd edition), February 2008.
7. Truptil, S., Benaben, F., Pingaud, H.: Collaborative process design for Mediation Information System Engineering, 6th International ISCRAM Conference, Gothenburg, Sweden, May 2009
8. Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications (ATHENA): Integrated Project Proposal – Description of work, 2003.
9. ISO/DIS 11345-1: Advanced automation technologies and their applications. Part 1: Framework for enterprise interoperability, 2009.
10. Panetto, H.: Meta-modèles et modèles pour l'intégration et l'interopérabilité des applications d'entreprises de production, Habilitation à Diriger des Recherches, Université de Nancy 1, 2006 [in French].

11. Chen, D., Dassisti, M., Elveaeter, B.: Enterprise interoperability framework and knowledge corpus – final report, Interop deliverable DI.3, may 2007.

12. Daclin, N., Chen, D., Vallespir, B.: Methodology for enterprise interoperability, 17th IFAC World Congress (IFAC'08), Seoul, South Korea, 6-11 July 2008.

13. Guedria, W., Naudet, Y, Chen, D.: Interoperability maturity models – survey and comparison -, 3rd IFAC TC5.3 International Workshop (EI2N'2008), Monterrey, Mexico, 2008.

14. C4ISR Architecture Working Group: Levels of Information Systems Interoperability (LISI), United States of America Department of Defense, Washington DC, USA, 30 March 1998.

15. Clark, T., Moon, T.: Interoperability for joint Coalition Operations, Australian Defence Force Journal, volume 51, 2001.

16. Tolk, A., Diallo, S.Y., Turnitsa, C.D.: Applying the Levels of Conceptual Interoperability Model in Support of Integratability, Interoperability, and Composability for System-of-Systems Engineering, Proceeding Systemics, Cybernetics and informatics, volume 5 - number 5, 2007.

17. Guédria, W., Chen, D., Naudet, Y.,: A Maturity Model for Enterprise Interoperability, IFAC-IFIP EI2N, Enterprise Integration, Interoperability and Networking, Portugal, 4-5 November, 2009.

18. Kasunic, M., Anderson, W.: Measuring systems interoperability: challenges and opportunities, Software engineering measurement and analysis initiative, Technical note CMU/SEI-2004-TN-003, 2004.

19. Ford C.T.: Interoperability measurement, Thesis, Department of the Air Force Air University, Air Force Institute of Technology, 2008.

20. Mallek, S., Daclin, N., Chapurlat, V.: Formalisation and Verification of Interoperation Requirements on Collaborative Processes, 18th IFAC World Congress (IFAC'11), Milano, Italy, 2011.

21. Maier, M.W.: Architecting Principles for System-of-Systems. Systems Engineering, vol.1, n°4, p. 267-284, 1998.

22. V. Kotov, Systems of systems as communicating structures, Hewlett Packard Computer Systems Laboratory Paper HPL-97-124, 1997, pp. 1–15. A.

23. Krygiel, A.J.: Behind the wizard's curtain: An integration environment for a system of systems, C4ISR Coopera-tive Research Program, 1999.

24. DeLaurentis, D., Callaway, R. K.: A System-of-Systems Perspective for Public Policy Decisions, Review of Policy Research 21(6): 829-837, 2004.

25. Maier, M. W.: Architecting Principles for Systems-of Systems, 6th Annual International Symposium of INCOSE, Boston, USA, 1996.

26. Campbell, J.E., Longsine, D.E, Sirah, D., Anderson, D. J.: System of Systems modeling and Analysis, sand report sand 2005-0020, January 2005.

27. Autran, F., Auzelle, J. P., Cattan, D., Garnier, J. L., Luzeaux, D., Mayer, F., Peyrichon, M., Ruault, J.R: Coupling component systems towards systems of systems, 18th annual international symposium of INCOSE, Utrecht, The Netherlands, 2008.

28. Bejar, R., Latre, M. A, Nogueras-Iso J., Muro-Medrano P.R., Zarazaga-Soria F.J.: Systems of systems as a conceptual framework for spatial data infrastructures, Internatioanl Journal of Spatial Data infrastructures Research, vol. 4, p 201-217, 2009.

29. Electronic Industries Alliance (EIA): Processes for Engineering a System, EIA/ IS 632, version 1.0, April 1998.

30. CEA (Commissariat à l'Energie Atomique): Méthode SAGACE: le systémographe, training manual version 1.0, 1999 [in French].

31. Camarinha-Matos, L. M.; Afsarmanesh, H.; Rabelo, R. J.: Infrastructure developments for agile virtual enterprises. IJCIM, ISSN 0951-192X, Vol. 16, N. 4-5, 2003.