Challenges in System of Systems Development: A Systematic Mapping

Diego de Lima Nascimento[1] and Rosana Teresinha Vaccare Braga[1]

Institute of Mathematics and Computer Science
University of São Paulo
São Carlos-SP, Brazil
diego.lnasc@usp.br, rtvb@icmc.usp.br

Abstract. System Of Systems (SoS) is an emerging field of research whose main purpose is the interoperability of constituent systems. SoS communicate through a variety of existing and independent systems, but achieving the interoperability of these systems is not trivial. This paper presents a systematic mapping (SM) that aims to classify and analyze the challenges that occur during the development of SoS. The studies of this SM were identified through a protocol and planning criteria already described in the literature. Furthermore, the SM found 39 studies in five sources that indexed different journals and conferences. After the analysis, it was possible to identify the main challenges that occur during the development and use of SoS. Therefore, the SM has found recent discussions on the most common challenges found during SoS development. The studies were categorized into nine areas with the purpose of highlighting the major problems and better characterizing the scope of SoS development. In short, most studies belong to the integration and management categories, because of the enormous difficulty of interoperating heterogeneous systems, and managing or finding engineers with the knowledge to work in the field. Finally, this SM can help researchers working with SoS integration, as they can benefit from the mapping we have provided correlating the difficulties in SoS development to the studies that address them.

Keywords: System of Systems, Interoperability, Challenge, Systematic Mapping

1 Introduction

The need to gather information in real time, integrating business partners to reduce end-to-end delivery times, resulted in organizations that are increasingly dependent on interconnected systems, which provide capabilities not found in single systems [40].

Therefore, software-intensive systems have become increasingly ubiquitous and complex, arising from the cooperation of several operationally independent and geographically distributed systems, often developed with different technologies and for different platforms. This cooperation is essential to solve larger and more complex missions that could not be solved by systems working separately [3].

Given growth of size and complexity, software systems are moving towards the so-called System of Systems (SoS). This class of systems has a diverse collection of large-scale components that work independently, but operate together to achieve a common goal. Thus, the integration of different components (constituent systems) offers a new capability based on the individual elements. According to DoD [41], SoS is a set or arrangement of systems that results when independent and useful systems are integrated into a larger system that has a specific capability.

A SoS is usually found in large scale and heterogeneous systems, although they can be differentiated from other traditional or complex systems through their distinguishing properties [42–44]: (i) operational independence: Any constituent system that is part of the SoS is independent, and is capable of operating normally if the SoS ceases to exist; (ii) managerial independence: Constituent systems collaborate with other members of the SoS, but each constituent is managed individually; (iii) geographic distribution: The constituent systems can be distributed geographically, but they can still cooperate with the SoS; (iv) emergent behavior: behavior arises during the collaboration of the constituent systems in the SoS, since the synergism cannot be assigned and/or achieved by each constituent individually; and (v) evolutionary development: the SoS is never complete, i.e., evolutionary development defines that objectives and functionalities are constantly changing and can be added, modified or removed. Thereby, some examples of SoS's are defense systems [42], emergency [45] or even intelligent transportation systems [42].

These intrinsic properties make a SoS distinctive and challenging, but very promising for this era of system inter-operation. Although there are many works regarding SoS development, we could not find a secondary study where the problems/challenge of developing SoS are clearly identified and discussed. Therefore, the main objective of this paper is to investigate the challenges that occur in the development of SoS. This is done by conducting a systematic mapping on this subject and discussing the results found.

The remainder of this paper is organized as follows. Section 2 describes the systematic mapping and how it was conducted. Section 3 presents the planning and protocol details. Section 4 summarizes the results. Section 5 presents threats to validity. Finally, Section 6 concludes the paper.

2 Systematic Mapping

Before starting a research, authors usually seek means to verify the scope, gaps and relevance of the proposed study. To help in this process, we can mention, among other techniques, literature reviews and rigorous reading methods that aid in the validation of the study. More specifically in the software engineering area, Evidence-Base Software Engineering (EBSE) promotes the use of two main methods, systematic review (SR) and systematic mapping (SM) [46]. Both methods are based on the same methodological principle, but a SR is able to reach a greater depth in the topic researched, based on specific research questions (RQ) and rigorous criteria during the selection of studies. On the other hand, an SM aims to analyze literature in a broader way, with less specific research questions, thus providing an overview of the state of the art [47, 48].

Planning, conducting and analyzing are the three main steps to perform a SR or SM. Each step is executed after the previous one, establishing a dependency and order between each one. The steps are defined as:

- Planning: a protocol for conducting the review is defined. This planning/protocol is based on why the SR/SM is necessary, thus research questions that seek to elucidate the theme are created. The protocol also contains important criteria to be followed, so this initial part should be carefully planned.
- Conduction: Beginning of review. At this stage the studies are properly searched, by means of criteria defined in the execution protocol.
- Report: Completion of the review. The results are discussed and evaluated in order to answer the main objective of the systematic review.

3 Planning

The first step of the SM consists of a well-defined plan that must contain fundamental parts from the beginning to the end of execution. In this stage, we have defined the research questions, search strategies and criteria for classification of the studies found. This protocol should elucidate the importance of the SM and its purpose, therefore, it needs well defined and structured research questions, as well as strategies that result in studies of the area with the purpose of extracting the maximum to enrich these issues.

Search Strategy and Engines The SM was conducted with the objective of answering two research questions, which have been directed to problems that occur with the development of SoS:

- RQ1: What are the challenges faced during the development of SoS?
- RQ2: Is there any solutions or at least considerations/discussions for the challenges encountered?

The RQs have keywords that can assist in the search for primary studies. Thus, the following main terms were used: *System-of-Systems*, *Problem* and *Development*, as well as the corresponding synonyms for the definition and refinement of a search string. The string used in this SM is shown below:

("System-of-Systems" OR "System-of-System" OR "Systems-of-Systems") AND ("Development" OR "Software Development" OR "Software Engineering") AND ("Problem" OR "Difficulty" OR "Challenge")

It is important to note that synonyms, related terms, or alternative spellings are incorporated with the Boolean OR operator; and the main terms with the Boolean operator AND. Subsequently, this string was adapted to the five search engines that can be visualized in Table 1. The most relevant engines for computer science and engineering have been chosen, among the various ones, as they index the main conferences of the research theme. The availability of the studies was also essential for the researchers [47].

Table 1. Selected Engines

Engine	**Address**
IEEEXplore	http://ieeexplore.ieee.org
ACM DigitalLibrary	http://dl.acm.org
Scopus	http://www.scopus.com
Web of Science	http://sub3.webofknowledge.com
Science Direct	http://www.sciencedirect.com

3.1 Inclusion and Exclusion Criteria

A well-executed SM must be based on a well structured string that improves the chance of a successful search, returning all the studies needed to respond the RQs. However, the string should not be very broad so as to return unnecessary studies. Regardless of the effort employed on defining the string, the returned studies need to be classified methodically, so that the best ones are filtered and better analyzed. To help on this issue, we have specified criteria to evaluate whether a study should be included or not (IC and EC: inclusion and exclusion criteria, respectively). For our particular SM, it is important to understand how SoS are being used and the difficulties found to use them, so only one inclusion criterion was defined, as well as three exclusion criteria:

- IC1: Primary studies that explicitly highlight problems/challenges during the use of SoS.
- EC1: The study is not in Portuguese or English.
- EC2: The study is not complete or the researcher cannot have access to the publication.
- EC3: The study is not related to SoS or does not point out problems/challenges.

3.2 Conduction

The search performed during our SM resulted in 900 initial studies. The procedure was initiated in August 2016. The selection of the studies was performed in two stages, (i) at first, a reading of the studies title and abstract was done to classify them according to their relevance. If positive, the study would move on to the next stage, also stressing that the study should be relevant to the inclusion criteria to be accepted. Otherwise, the study was removed by some exclusion criteria. However, a change was made after reading a few studies because it was difficult to identify the problems/challenges only by reading the title and abstract. Consequently, the reading of the conclusion was also included in order to mitigate this problem. (ii) The second stage, called extraction, was performed with the same inclusion and exclusion criteria, but a diagonal reading of the papers was also performed, as this reading technique has a greater focus on the most relevant parts of the text, more specifically introduction and conclusion.

It is common to obtain a substantial number of repeated studies, because some search engines index the same conference or journal. In the case of this SM, a total of 194 duplicate studies were found, and of the remaining 706, 580 were rejected by some exclusion criterion. Therefore, most of the studies did not go through the first inclusion criterion, because they did not make clear the problem/challenge faced during the development of SoS. Hence, 126 studies were approved for the initial stage, but only 39 remained after reading the second stage.

The extraction and synthesis phase started after finishing the two initial stages of selection. The 39 selected studies were completely read and the following information was obtained: (i) demographic data; (ii) identification of the different domains of use of SoS; (iii) identification of the types of studies applied in the context of SoS and in different domains, for example architecture, framework, among others; (iv) identification of the problems/challenges encountered during the use of SoS; (v) determination of whether or not the author proposes a solution/consideration to the challenge described.

4 Results

In this SM, no studies prior to 2003 have been included in the results summary, even without using year filtering to help verifying specific events or how the area is behaving in recent years. The reason for not finding studies before 2003 is justified by the fact that, although systems with SoS characteristics have been discussed for a long time, their definitions and particularities were defined explicitly only in 1998 by [42]. Consequently, it is common that problems aroused or began to be discussed only a few years after, as shown in Fig 1.

It is worth noting that the search was made in the middle of 2016, so the smaller number of studies in 2016 is due to the fact that this year was not considered in its entirety, besides the natural delay for newly published papers to be indexed by the search engines.

4.1 Domains and Types of Studies

The extraction activity has the purpose of verifying if there is any relation between the problems encountered and the different domains of SoS development. In addition, it has the objective of knowing which area is using this class of systems more frequently. The results are shown in Table 2.

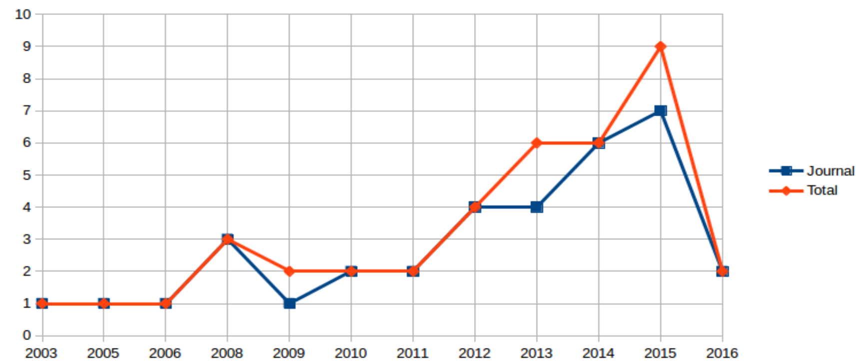

Fig. 1. Distribution of Studies

It can be observed that most studies (61.53 %) do not refer to a specific domain, however, they use as an example big systems such as those described in the remaining domains, resulting in a more general discussion about SoS or SoS Engineering (SoSE). Some studies cite examples to elucidate concepts about SoS, however, these cases are for explanation only, so they are classified in the "no specific domain" category.

Table 2. Classification of Application Domains

ID	Amount	Domain
S1[1], S3[3], S7[7], S9[9], S10[10], S11[11], S12[12], S14[14], S15[15], S17[17], S20[20], S21[21], S22[22], S23[23], S26[26], S27[27], S28[28], S31[31], S32[32], S33[33], S34[34], S37[37], S38[38], S39[39]	24(61,5%)	No specific domain
S5[5], S18[18], S19[19], S24[24]	4(10,2%)	Industrial automation
S13[13], S16[16]	2(5,1%)	Earth observation system
S30[30], S36[36]	2(5,1%)	Department of Defense (DoD)
S2[2], S29[29]	2(5,1%)	Public sector
S4[4], S35[35]	2(5,1%)	Maritime
S6[6], S8[8]	2(5,1%)	Physically coupled
S25[25]	1(2,5%)	Automotive, rail and avionics

The search results were classified and categorized (Table 3) with the main purpose of visualizing the growth of the area and, subsequently, to verify if there was any relation between the domains and types with the problems found. For the classification, the types were defined and the studies were classified according to each definition, right after reading. In this sense, the first type of study found is related to secondary studies where a systematic literature mapping or review (SR or SM) has been presented. The two retrieved studies in this category are related to the SoS context, but with emphasis on different uses. The first is related to integration, while the second regards the development of SoS based on IoT. Both of them presented difficulties related to SoS.

Ten "Framework" studies have also been found, which address processes, tests, techniques and/or methodologies proposed by the researchers. Studies classified in the "Architecture" category propose standards, define concepts or cover some difficulties related to the SoS research line. They can also refer to modeling and/or the development process, e.g. activities related to validation, requirements or something more specific such as integration.

We have also found studies that discuss the SoS or SoSE area itself, explaining concepts, definitions and common problems related to specific SoS types or not. Such studies were grouped in the "Approach" category. Researches that discuss a more specific domain or have a more limited scope of the problem, in a way that characterizes an application environment, have been classified as a "case study". Finally, there is the "Ontology" category, in which a specific study discusses and defines specific terms of SoS integration (SoSI).

4.2 Conclusions of the Research Questions

In this step, the results are presented according to the research questions defined in the protocol. **RQ1**. The purpose of this question is to extract the problems/challenges encountered in SoS development. The result of this question was achieved after analyzing the 39 studies and summarizing them to get an overview of the main open problems found. First, all studies were read and summarized, and all types of problems, challenges and other interests were identified. Subsequently, all the challenges were analyzed and categorized in large areas of "Challenge", which are summarized in the following:

Table 3. Type of Studies

ID	Amount	Type
S1, S12	2(5,1%)	SR or SM
S6, S8, S10, S14, S16, S23, S27, S30, S35, S36	10(25,6%)	Framework
S3, S21, S22, S37	4(10,2%)	Architecture
S4, S7, S9, S11, S15, S17, S18, S19, S20, S28, S32, S33, S34, S38, S39	15(38,4%)	Approach
S2, S5, S24, S25, S26, S29, S31	7(17,9%)	Case study
S13	1(2,5%)	Ontology

- **Integration**: SoS integration is not a trivial task. According to [13], integration must be handled during the development of SoS, so that it is present throughout the life cycle. Studies in this category mainly portray that integration is still a challenge because of the lack of specific knowledge about each constituent system; the variety of implementations (the constituents do not have standardization of interfaces); managerial and operational independence can be a problem when information protection is needed; difficulty in coordinating the SoS or the constituent; geographical distribution of the constituents can cause problems with communication and information sharing; and difficulty to interoperate heterogeneous systems in an effective and transparent manner. The difficulty in system integration is nothing new, and it is easy to note the variety of problems that occur, since these problems (standardization, interdependence, interoperability, and complexity) are also found in other software engineering areas.

- **Management**: Management difficulties start when looking for qualified engineers to work in the SoS area. This happens due to the high need of specific knowledge about systems in general (or about a particular system) and is worsened by the necessity of controlling several stakeholders who present different objectives and communication problems. Moreover, several risks can be present and future uncertainties of the SoS must be managed.
- **Documentation**: The SoS requires communication and sharing abilities in order to evolve properly. The lack of documentation implies that no guidelines are available to assist engineers during the development process. This affects growth as a whole, causing problems of dependence on engineers that know the system and how to make them interoperate with new systems. The lack of documentation can affect other subareas, since it makes communication more difficult, e.g., when hiring engineers to evolve or integrate new systems.
- **Evolution and Maintenance**: Evolutionary development [42] is one of the main features of SoS, but doing so in real time is extremely complex. Therefore, collaboration in a SoS can be compromised due to the difficulty in evolving the system as a whole. According to [21], the evolution of SoS can undermine its performance, as new harmful emerging behaviors may arise, and consequently, this unique feature of SoS may render unreliable the generalization of experiments. In this context, there is also a study [20] which describes that components can be added and removed at all times, since independence and evolution over time is a predominant feature of SoS, and as such, the interaction and composition of the constituent elements of the SoS is carried out to accomplish the shared mission.
- **Architecture and Modeling**: According to [22] it is very difficult to model and analyze a SoS by conventional means, especially how to represent SoS in detail, because the complete details of the constituent systems may not be readily available. Therefore, in this category, discussions about architecture and modeling were classified in the context of SoS, starting from the low mobility of the community in the evolution of existing reference architectures until the difficulty of representing the characteristics of a SoS and how to represent it in a single model.

- **Requirements and Quality**: According to [20], quality is a key issue to SoS success, as inconsistencies and ambiguities can arise when quality attributes are overlooked in the SoS architecture. In this category we have included studies that discuss the importance of quality and the difficulty of applying requirements engineering in the context of SoS.
- **Test**: SoS can present problems in its current context, but the problems can be found in the interfaces with constituents or in the constituents themselves, so testing challenges are amplified [24]. Only two studies portrayed test challenges within SoS.
- **Dynamic Reconfiguration and Monitoring**: Three studies addressed problems related to the SoS reconfiguration at run time, especially in the context of structural changes and SoS composition.
- **Behavior**: Finally, several studies that discuss problems related to interactions, consequences and effects of emergent SoS behavior.

Table 4. Problem Categorization

ID(Paper linked to the Challenge)	Challenge	Considerations
S1, S5, S8, S9, S11, S12, S13, S17, S18, S21, S28, S32, S35, S38, S39	Integration	S27
S1, S5, S10, S17, S18, S21, S29, S32, S34, S36, S39	Management	S7
S1, S3, S12, S13, S20	Documentation	
S1, S3, S11, S20, S21, S22, S32, S34, S37, S39	Evolution and Maintenance	S19, S27
S1, S3, S11, S20, S22, S31, S37, S39	Architecture and Modeling	S4, S6, S14, S15, S16, S23, S27, S33
S20, S26, S28, S32, S37	Requirements and Quality	S2, S7, S25
S24, S35	Test	S30
S11, S20, S35	Dynamic Reconfiguration and Monitoring	S5, S16, S19
S1, S5, S11, S21, S28	Behavior	

It is not possible to observe a relationship between challenge and domain in the studies (Table 2 and Table 4), i.e., there are no evidences that a particular domain would present specific challenges, in short, there is no correlation. However, it is possible to notice the lack of tools to ease SoS development and the lack of experience of stakeholders in the SoS area. This can be partially justified by the fact that there is a big proportion of closed and unavailable projects, resulting in a higher learning curve for engineers, besides leading to the lack of standardization of architectures and models.

RQ2. This RQ aimed at investigating the studies in which the problems/challenges already have a proposed solution or, at least, considerations or discussions are presented by the authors. In this sense, this question was focused on evidencing the state of the art concerning SoS. To facilitate this investigation, one of the reading criteria was to examine the studies in descending order, i.e., from 2016 to 2003.

In general, this RQ is complex to be analyzed. Researchers always (or generally) point out considerations or ways to mitigate problems in their area, however, in SoS studies, they tend to propose specific solutions to challenges encountered during the development of their research or ongoing projects. Thus, we have decided to make a correspondence of studies that present relevant considerations to the subareas described above (Table 4). This is done in the "Considerations" column of the table. In this paper, these considerations correspond to researches that present ideas (research, methods, frameworks, among others) of how such challenges can be smoothed. Thereby, the researches that are in the column "Considerations", present somehow solutions that can aid in the described problem. It is important to emphasize that such research does not mitigate the problems/challenges, only aid in their control.

Finally, it is worth observing that SoS often refers to large enterprise systems or proprietary projects, resulting in frameworks and approaches that are not available to the community. Consequently, most of the studies mention lack of communication and availability to the community and stakeholders, i.e., the area needs more cooperation and sharing to make possible to derive standards and to achieve progress in the near future.

5 Threats to Validity

This SM was conducted in August 2016 and is limited to studies that discuss SoS development. It is important to emphasize that no filter was used in the search engines and no type of gray literature has been included. Therefore, the main threats to the validity of this research are related to the selection and extraction bias of the studies.

First, the entire process was structured from the RQs and the protocol defined in Section 2. This protocol was written by the first author, but was also revised and refined by the second author, improving the coherence and accuracy of the SM with the RQs. The selection of the studies was carried out strategically (following the inclusion and exclusion criteria), using the systematic review tool "State of the Art through Systematic Review" (StArt) [49], which aims to assist the execution of SM and organize the selection of studies, and automatically remove duplicate studies.

It is important to note that in this paper the peer review process was not performed, however, the second author was always consulted in case of doubts. Additionally, each study underwent two rigorous classification phases (subsection 3.2), in which all 39 studies were accepted twice by the inclusion criterion. Nevertheless, although the databases are the most relevant in the area [47], some studies might not have been analyzed because of limitations of the search engines or lack of authors' access to the work.

The bias in the extraction of the studies was also controlled by the establishment of a protocol based on the RQs. However, not all the studies had well-defined responses to the SM RQs, so they needed to be interpreted and categorized by the authors of this work.

6 Conclusion

SoS fosters a new form of benefiting from the functionality of existing systems, so interoperation of resources allows new visions to be realized to achieve broader goals. However, the integration of different systems is a challenging task, since they were not developed with the purpose of working together, which poses numerous challenges.

The present SM discusses the major challenges that researchers face during the development of SoS. Thus, several studies in different contexts and domains were found between 2003 and 2016, but often have similar problems. Therefore, it was possible to categorize the challenges in nine specific areas, in which integration and management were the most pointed by the researchers. In fact, SoSI was and still is a subject of enormous debate, so it was to be expected that most of the challenges would be present in this category. Surprisingly, management was also a category that presented several problems, whose key factors were qualified (experienced) engineers and control of various stakeholders.

However, this SM points out essential characteristics for researchers working with SoSE, since problems are not only categorized but also correlated with the state of the art considerations. In addition, several studies present solutions targeted to specific projects and SoS, and often these solutions are valid for other studies and research, even though the contexts are different, so they can and should be analyzed.

Acknowledgment

We are thankful to Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPE- SP), process number 2016/05129-0, to Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), and to the University of São Paulo (USP) for supporting the execution of this research.

References

1. Vargas, I. G., Gottardi, T., Braga, R. T.: Approaches for Integration in System of Systems: A Systematic Review. In Proceedings of the 4th International Workshop on Software Engineering for Systemsof-Systems (SESoS 16). ACM, New York, NY, USA, 32-38. (2016)
2. Larsson, J., Borg, M., Olsson., T.: Testing Quality Requirements of a System-of-Systems in the Public Sector - Challenges and Potential Remedies. CoRR abs/1602.05618. (2016)

3. Nakagawa, E. Y., Oquendo, F., Avgeriou, P., Cuesta, C. E., Drira, K., Maldonado, J. C. , Zisman, A.: Foreword: Towards Reference Architectures for Systems-of-Systems. In 2015 IEEE/ACM 3rd International Workshop on Software Engineering for Systems-of-Systems. ACM, Florence, Italy, 1-4. (2015)
4. Hachem, J. E.: Towards Model Driven Architecture and Analysis of System of Systems Access Control. In 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Vol. 2., Florence, Italy, 867-870. (2015)
5. Rabiser, R., Vierhauser, M., Grnbacher, P.: Variability Management for a Runtime Monitoring Infrastructure. In Proceedings of the Ninth International Workshop on Variability Modelling of Software-intensive Systems (VaMoS 15). ACM, New York, NY, USA, 35–42. (2015)
6. Chiprianov, V., Gallon, L., Salameh, K., Munier, M., Hachem, J. E.: Towards security software engineering the Smart Grid as a System of Systems. In 2015 10th System of Systems Engineering Conference (SoSE). IEEE, San Antonio, TX, USA, 77-82. (2015)
7. Safwat, A., Senousy, M.B.: Addressing Challenges of Ultra Large Scale System on Requirements Engineering. Procedia Computer Science 65, 442-449. (2015)
8. Kampert, D., Nazari, S., Sonntag, C., Epple, U., Engell, S.: A Framework for Simulation, Optimization and Information Management of Physically-Coupled Systems of Systems. IFAC-PapersOnLine 48, 3, 1553-1558. (2015)
9. Pennock, M. J., Wade, J. P.: The Top 10 Illusions of Systems Engineering: A Research Agenda. Procedia Computer Science 44, 147-154. (2015)
10. Fang, Z., DeLaurentis, D.: Multi-stakeholder Dynamic Planning of System of Systems Development and Evolution. Procedia Computer Science 44, 95-104. (2015)
11. Nielsen, C. B., Larsen, P. G., Fitzgerald, J., Woodcock, J., Peleska, J.: Systems of Systems Engineering: Basic Concepts, Model-Based Techniques, and Research Directions. ACM Comput. Surv. 48, 2, 18–41. (2015)
12. Maia, P., Cavalcante, E., Gomes, P., Batista, T., Delicato, F. C., Pires, P. F.: On the Development of Systems-of-Systems Based on the Internet of Things: A Systematic Mapping. In Proceedings of the 2014 European Conference on Software Architecture Workshops (ECSAW 14). ACM, New York, NY, USA, 23:1-23:8. (2014)
13. Madni, A. M., Sievers, M.: System of Systems Integration: Key Considerations and Challenges. Systems Engineering 17, 3 (2014), 330-347. (2014)
14. Chiprianov, V., Falkner, K., Gallon, L., Munier, M.: Towards modelling and analysing non-functional properties of systems of systems. In 2014 9th International Conference on System of Systems Engineering (SOSE). IEEE, Adelade, SA, Australia, 289-294. (2014)
15. Chiprianov, V., Falkner, K., Szabo, C., Puddy, G.: Architectural Support for Model-Driven Performance Prediction of Distributed Real-Time Embedded Systems of Systems. Springer International Publishing, Cham, 357-364. (2014)
16. Vierhauser, M., Rabiser, R., Grnbacher, P., Danner, C., Wallner, S., Zeisel, H.: A Flexible Framework for Runtime Monitoring of System-of-Systems Architectures. In 2014 IEEE/IFIP Conference on Software Architecture. IEEE, Sydney, Australia, Australia, 57-66. (2014)
17. Keating, C. B.: Governance implications for meeting challenges in the system of systems engineering field. In 2014 9th International Conference on System of Systems Engineering (SOSE). IEEE, Adelade, SA, Australia, 154-159. (2014)
18. Papatheocharous, E., Axelsson, J., Andrersson, J.: Issues and Challenges in Ecosystems for Federated Embedded Systems. In Proceedings of the First International Workshop on Software Engineering for Systems-of-Systems (SESoS 13). ACM, New York, NY, USA, 21-24. (2013)

19. Vierhauser, M., Rabiser, R., Grnbacher, P., Danner, C., Wallner, S.: Evolving Systems of Systems: Industrial Challenges and Research Perspectives. In Proceedings of the First International Workshop on Software Engineering for Systems-of-Systems (SESoS 13). ACM, New York, NY, USA, 1-4. (2013)
20. Batista, T.: Challenges for SoS Architecture Description. In Proceedings of the First International Workshop on Software Engineering for Systems-of-Systems (SESoS 13). ACM, New York, NY, USA, 35-37. (2013)
21. Henshaw, M., Siemieniuch, C., Sinclair, M., Henson, S., Barot, V., Jamshidi, M., Delaurentis, D., Ncube, C., Lim, S. L., Dogan, H.: Systems of Systems Engineering: A research imperative. In 2013 International Conference on System Science and Engineering (ICSSE). IEEE, Budapest, Hungary, 389-394. (2013)
22. Kalawsky, R. S., Joannou, D., Tian, Y., Fayoumi, A.: Using Architecture Patterns to Architect and Analyze Systems of Systems. Procedia Computer Science 16 (2013), 283–292. (2013)
23. Alexander, R., Kelly, T.: Supporting systems of systems hazard analysis using multi-agent simulation. Safety Science 51, 1, 302-318. (2013)
24. Ali, N. b., Petersen, K., Mntyl, M.: Testing highly complex system of systems: An industrial case study. In Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. ACM, New York, NY, USA, 211-220. (2012)
25. Penzenstadler, B., Eckhardt, J.: A Requirements Engineering content model for Cyber-Physical Systems. In 2012 Second IEEE International Workshop on Requirements Engineering for Systems, Services, and Systems-of-Systems (RESS). IEEE, Chicago, IL, USA, 20-29. (2012)
26. Hallerstede, S., Hansen, F. O., Holt, J., Lauritsen, R., Lorenzen, L., Peleska, J.: Technical challenges of SoS requirements engineering. In 2012 7th International Conference on System of Systems Engineering (SoSE). IEEE, Genoa, Italy, 573-578. (2012)
27. Griendling, K., Salmon, J., Mavris, D.: Elements of a decision-making framework for early-phase system of systems acquisition. In 2012 IEEE International Systems Conference SysCon 2012. IEEE, Vancouver, BC, Canada, 1-8. (2012)
28. Ncube, C.: On the engineering of systems of systems: Key challenges for the Requirements Engineering community. In 2011 Workshop on Requirements Engineering for Systems, Services and Systems-of-Systems. IEEE, Trento, Italy, 70-73. (2011)
29. Liu, S.: Employing System of Systems Engineering in China's Emergency Management. IEEE Systems Journal 5, 2 (June 2011), 298-308. (2011)
30. Dahmann, J., Lane, J. A., Rebovich, G., Lowry, R.: Systems of systems test and evaluation challenges. In 2010 5th International Conference on System of Systems Engineering. IEEE, Loughborough, United Kingdom, United Kingdom, 1-6. (2010)
31. Jamshidi, M.: From Large-Scale Systems to System of Systems Control Challenges for the 21st Century1. IFAC Proceedings Volumes 43, 8 , 13–19. (2010)
32. Lewis, G. A., Morris, E., Place, P., Simanta, S., Smith, D. B.: Requirements engineering for systems of systems. In 2009 3rd Annual IEEE Systems Conference. IEEE, Vancouver, BC, Canada, 247-252. (2009)
33. Despotou, G., Alexander, R., Kelly, T.: Addressing challenges of hazard analysis in systems of systems. In 2009 3rd Annual IEEE Systems Conference. IEEE, Vancouver, BC, Canada, 167-172. (2009)
34. Smith, D., Lewis, G.: Systems of Systems: New challenges for maintenance and evolution. In 2008 Frontiers of Software Maintenance. IEEE, Beijing, China, 149-157. (2008)

35. Gonzalez, A., Piel, E., Gross, H. G., Glandrup, M.: Testing Challenges of Maritime Safety and Security Systems-of-Systems. In Testing: Academic Industrial Conference - Practice and Research Techniques. IEEE, Windsor, UK, UK, 35-39. (2008)
36. Dahmann, J. S., Baldwin, K. J.: Understanding the Current State of US Defense Systems of Systems and the Implications for Systems Engineering. In 2008 2nd Annual IEEE Systems Conference. IEEE, Montreal, Que., Canada, 1-7. (2008)
37. Cole, R.: The changing role of requirements and architecture in systems engineering. In 2006 IEEE/SMC International Conference on System of Systems Engineering. IEEE, Los Angeles, CA, USA, 1–5. (2006)
38. Ring, J., Madni, A. M.: Key challenges and opportunities in 'system of systems' engineering. In 2005 IEEE International Conference on Systems, Man and Cybernetics, Vol. 1. IEEE, Waikoloa, HI, USA, 973-978 Vol. 1. (2005)
39. Chen, P., Clothier, J.: Advancing systems engineering for systems-of-systems challenges. Systems Engineering 6, 3 (2003), 170183. (2003)
40. Lewis, G., Morris, E., Simanta, S., Smith, D.: Service Orientation and Systems of Systems. IEEE Software 28, 1, 58-63. (2011)
41. DOD. Systems Engineering Guide for Systems of Systems. Available at: <http://www.acq.osd.mil/se/docs/SE-Guide-for-SoS.pdf>(2008)
42. Maier, M. W.: Architecting principles for systems-of-systems. Systems Engineering 1, 4, 267-284. (1998)
43. Firesmith, D.: Profiling Systems Using the Defining Characteristics of Systems of Systems (SoS). Technical Report CMU/SEI-2010-TN-001. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9269 (2010)
44. Jamshidi, M.: Systems of systems engineering: principles and applications. CRC press, sn. (2008)
45. Lane, J. A.: What is a system of systems and why should i care? . Technical Report. University of Southern California, Los Angeles, CA, USA. (2013)
46. Dyba, T., Kitchenham, B. A., Jorgensen, M.: Evidence-based software engineering for practitioners. IEEE Software 22, 1, 58-65. (2005)
47. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-01, School of Computer Science and Mathematics, Keele University. (2007)
48. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic Mapping Studies in Software Engineering. In Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering (EASE08). BCS Learning & Development Ltd., Swindon, UK, 68-77. (2008)
49. Zamboni, A., Thommazo, A. D., Hernandes, E., Fabbri, SCPF.: StArt uma ferramenta computacional de apoio à revisão sistematica. In Proc.: Congresso Brasileiro de Software (CBSoft10), Salvador, Brazil. CBSoft, Salvador, Brazil, 91-96. (2010)