



HAL
open science

Towards a Multi-Viewpoints Approach for the SoS Engineering

Charaf Eddine Dridi, Zakaria Benzadri, Faiza Belala

► **To cite this version:**

Charaf Eddine Dridi, Zakaria Benzadri, Faiza Belala. Towards a Multi-Viewpoints Approach for the SoS Engineering. ICAASE'2022, Oct 2022, constantine, France. hal-04227343

HAL Id: hal-04227343

<https://hal.science/hal-04227343>

Submitted on 3 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards a Multi-Viewpoints Approach for the SoS Engineering

Charaf Eddine DRIDI, Zakaria BENZADRI, Faiza BELALA

LIRE Laboratory

Constantine2 - Abdelhamid Mehri University

Constantine, Algeria

{charafeddine.dridi, benzadri.zakaria, faiza.belala}@univ-constantine2.dz

Abstract—

In recent years, the engineering of Systems-of-System (SoSs) has experienced an increasing evolution and interest from the computer science community. Additionally, Architecture Frameworks are a recent discipline in Software Engineering (SE) that consider Architectural Viewpoints as first-class entities in software development. The viewpoints have become a major paradigm that has the power to open a door to the development representation and provide a new way of designing systems. In this paper, we propose a viewpoints-based architecture framework model encompassing the concepts proposed by “ISO/IEC/IEEE 42010:2011 Systems and Software Engineering-Architecture description” to manage the complexity of SoSs’ architectures. Consequently, we aim to represent by means one or more architectural viewpoints that together can provide a unified AF of SoS’ architecture.

Keywords— SoSs, Viewpoints, SoSE, SoSE development processes, IEEE-42010.

I. INTRODUCTION

In recent years, SoSs have experienced an increasing evolution and interest from the computer science community. SoSs have been designed to integrate multiple independent functional systems into a larger system in several important application domains. However, these Constituent-Systems (CSs) are becoming larger, more complex, and difficult to develop as well.

Examples of these large-scale systems can be found in several fields such as Robotics, Avionics, Military, and Intelligent systems (Smart-Grids, Smart-Cities, Smart-Homes, etc.). An SoS is characterized by offering new functionalities to users that cannot be offered by its CSs, but emerging from their combination. The CSs making up an SoS are independent, geographically distributed, developed with different technologies, and intended for several platforms.

This new class of systems has created new challenges for systems engineering (SE) to extend the theoretical foundations of Software Engineering to SoS Engineering (SoSE). Therefore, a new opportunity may exist nowadays to start a new engineering field called SoSE [1]. However, many key issues like architecture, heterogeneous stakeholders, modeling viewpoints variety, unified standards, development processes, etc. are the most relevant challenges that must be addressed and well resolved. For more details about the relevant challenges of SoSs, definitions, characteristics, application domains, etc. readers are referred to our survey [2].

Therefore, due to varied methodology and domains of applications in existing literature, there does not exist a single unified standard for architecture frameworks involved in SoSE. Thus, SoS development does not follow the normal system development process. As SoSE processes begin to emerge, we note the absence of dedicated frameworks to

model and implement them, and that the traditional SE practices are still insufficient to deal with all aspects of SoSs development.

In this new perspective, SoSs development does not follow the normal system development process. thus, we present our framework model which is based on the ISO/IEC/IEEE 42010/2011 [3]. It defines the notion of architecture framework and describes its content. We also cover the different steps of the architecture development as imposed in the standard. Additionally, the proposed framework highlights the guidelines for requirement elicitation, and provide structural and functional models to define the architecture according to different stakeholders’ viewpoint.

Consequently, we propose a new architecture framework, a multi-view model that allows specifying stakeholders’ domains in various viewpoints. However, we cover specific concepts included in the viewpoints, their behavior, development processes, and interactions. By using Model Driven Engineering, we give a syntax to this proposed method, i.e., the SoS’ structure and architectural elements are specified by the MeMSoS meta-models [4]. MDE defines a clear separation of abstraction levels where MeMSoS is one of them. Thanks to these abstraction levels, we can split those specified in IEEE-42010.

The remainder of this paper is organized as follows: in Section 2, we provide some prerequisites about SoSs and their relevant related works prerequisites and backgrounds on SoSs and our related works. In Section 3, we give a detailed model describing the proposed framework. In Section 4, we discuss the main principles of our methodology. Finally, Section 5 concludes the paper and discusses possible future works.

II. BACKGROUND AND RELATED WORKS

A. Related work

In fact, a large number of studies have addressed the problem of designing the architecture of SoS. However, there is still a lack of architectural frameworks having the expressive power to adequately describe the SoSs architectures.

In [5], the authors have presented an approach to the description of the software architecture of an SoS based on the standard ISO/IEC/ IEEE 42010. They have introduced a case study, which is an SoS entitled Smart City, and they have illustrated the independent systems participating in this SoS, the devices, the communications and the structure of each system. The purpose was to highlight the requests between systems within this SoS. And finally, they have explained how to apply the rules and the basis of the standard ISO 42010 in their case study.

In the same context, they have proposed in [6] a method to adapt the standard proposed by "ISO 42010 for the software industry" to describe the software architectures of SoSs. To

this end, the authors have changed the concept of ‘System of Interest’ to ‘CS’ and added an element entitled ‘System-of-Systems (SoS)’. Then, they proposed to add a relationship link between them labeled ‘participates on’. As well as they have presented an ADL-based approach to model SoS characteristics.

The authors of [7] have presented an architecture documentation viewpoint to address stakeholder concerns about SoS and their CSs design and analysis. The purpose was to assist them in collecting and/or creating sufficient documentation about CSs. This viewpoint conforms to the IEEE-42010 standard for architecture description and guides SoS architects to request sufficient information about CSs’ architectures in order to satisfy SoS-level concerns about each CS operating in the SoS context.

Adopting a viewpoint-based SysML visual modeling language, the authors of [8] have proposed a semi-formal SoS conceptual model which serves as a domain-independent vocabulary for SoSs. In particular, they have defined an SoS profile that extends on the SysML reference meta-model with specific language constructs.

In their extended paper [9], they have shown how to use the SysML profile to (i) model the high-level design of the target SoS architecture, (ii) support different types of analyses in an MDE toolchain, and (iii) solve scalability and usability concerns through its integration in a user-friendly MDE tool for SoS rapid modeling, validation, code generation, and simulation.

A plethora of modeling approaches that address SoS architectural aspects was identified by our last survey. For more details, readers are referred to our survey [2] we have classified them according to seven main classes (Model-Driven Architecture, Services-Oriented Architecture, Ontology-based approaches, Architecture Description Language and Bigraph-based approaches) but the main related studies.

B. Background and Motivation

Large-scale systems now and in the future will be built by integrating existing systems from different providers to create SoSs. The latter have emergent properties and can be assembled from other existing and new CSs, which are independently controlled and managed as defined in [10].

In fact, there are no official agreed definitions for these systems or what are their main characteristics. The definition of these systems is mainly based on the application domains and their focus. They can be found in almost every domain of application, in health-care [11] [12], military defense [13] [14], transportation [15] [16], Smart Energy Grids [17], etc. Thus, this diversity proved to be an issue, since every domain has its definition and characteristics. The diversity of application domains explains the use of these systems in specific domains, and cannot be used to establish a general definition. For example, the existing literature offers a set of SoS’ specific domains as SoIS [18] (Systems-of-Information Systems), SISoS [19] (Software-intensive SoS), and SoES [20] (System of Enterprise-Systems). Similarly, for CPSSs, we find CPES [21] (Cyber-Physical Energy Systems) and CPIoTS [22] (Cyber-Physical IoT systems), etc.

However, the authors of [23] have proposed a general definition that describes SoSs regardless of the domain as follows: “an SoS or a CPS is a collection of individual,

possibly heterogeneous, but are functional, wirelessly integrated to enhance the overall robustness, lower the cost of operation, and increase the reliability of the complex system”. Moreover, the authors of [24], [25] and others, highlight a set of characteristics for SoSs which can be categorized into components characteristics (operational and, managerial independence, geographical distribution, Autonomy, Heterogeneity and Belonging of elements) and global characteristics (evolutionary development, dynamic emergence and diversity of functionalities). Furthermore, they are complex systems that exhibit other features: spatially distributed physical CSs, distributed control, partial autonomy of the subsystems, dynamic reconfiguration and continuous evolution.

However, these definitions and characteristics say nothing about the systems’ size. In fact, the real challenges emerge when the CSs are themselves large-scale systems. Generally, an SoS might include tens or hundreds of separate CSs, with software systems keeping track of these elements and providing controllers with information that allows them to be deployed most effectively.

This means that the joint dynamics of SoSs should be studied together, and this is what sets the emerging discipline of CSs systems apart from these specific domains. The difficulty of designing SoSs lies in the complexity resulting from the interaction, cooperation and collaboration of their heterogeneous CSs, where each CS has specific goals to accomplish and different roles to play and interoperate with each other. Independent evolution and dynamic changes can cause these CSs to behave differently. These changes can affect their interactions and communications within the SoS and as a consequence it can derail the overall mission of the SoS.

III. METHODOLOGY AND CURRENT WORK

Architecture frameworks are mechanisms widely used in architecting. They establish a common practice for creating, interpreting, analyzing and using architecture descriptions within a particular domain of application or stakeholder community. As a result, their uses include, but are not limited to [26]:

- Creating architecture descriptions.
- Developing architecture modeling tools and architecting methods.
- Establishing processes to facilitate communication, commitments and interoperation across multiple projects and/or organizations.

An Architecture Framework is a knowledge prefabricated structure that stakeholders can use to organize an architecture description into complementary views [26]. The specification of an AF is one area of the standardization in ISO/IEC/IEEE42010:2011. This standard proposes a conceptual model to describe the terms and concepts of systems and architecture description. This standard specifies an AF as a composition of multiple Viewpoints (VPs), each VP can be used to address specific concerns of different Stakeholders [3]. Therefore, we aim to extend this standard to offer a comprehensive guideline to define an SoS-AF. It is specified by enhancing the concepts of the AF description represented in the international standard with the essential processes that an SoS’ AF should encompass (red rectangles in Figure.1).

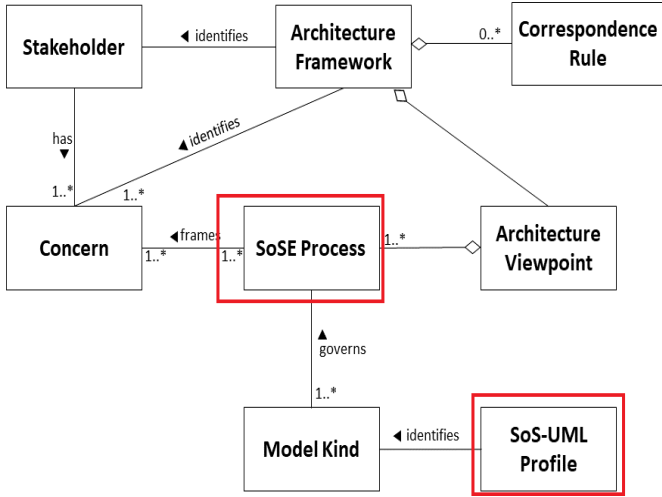


Figure 1: An extended conceptual model of SoS-AF.

A. SoS's Stakeholders

In this framework, we identify the different stakeholders of SoSs, following their roles. Thus, we introduce the different stakeholders and highlight their roles in Table.1. Note that, our approach is based on the exploitation of system engineering backgrounds, each of the stakeholders can maintain the SoS, and has specific tasks that could be technical, functional...etc.

Stakeholders are inherently heterogeneous due to multiple users, their VPs, engineering processes, platforms, environments...etc. Stakeholders are individuals, groups, or organizations holding Concerns for an SoS. four main stakeholders have been identified in this framework:

- SoS Experts: a group of persons responsible for the Business VP establishment, with strong theoretical knowledge in an SoS application domain.
- Architects and Designers: their role is vital to the success of both Analysis and Design VPS, they translate the requirements into a demand for CSs Capabilities.
- Collaboration Specialists: they are also Analysis and Design experts; they are responsible for understanding CSs and their Capabilities' collaboration.
- Interactions Engineers: persons responsible for the Deployment VP and they are responsible for specifying communication and interactions between different Roles.

TABLE I. DIFFERENT ROLES OF STAKEHOLDERS.

Stakeholders	Analysts	Users	Owners	Developers	Maintainers
SoS Experts		X	X		X
Architects and Designers	X			X	X
Collaboration Specialists	X			X	X
Interactions Engineers				X	X

B. SoSs' Concerns

Through our previous studies [2][4], we were able to extract these concerns by exploring multiple aspects related to SoSE. To adequately understand these concerns, a set of modeling and implementing questions should be answered (see Table.2):

TABLE II: ARCHITECTURE FRAMEWORK-RELATED CONCERNS.

Concerns	Description
C.1	What requirements should the SoS meet?
C.2	How does the SoS ensure knowledge?
C.3	What are the design decisions related to any influences on the SoS in its environment?
C.4	How does the SoS support the stakeholder relationships?
C.5	What are the modeling constraints and CSs' interdependencies that the SoS should have?
C.6	What are the local and global goals of the SoSs?
C.7	How does the SoS achieve sustainability, flexibility, and interoperability?
C.8	How does the SoS carry out the self-organization and adaption requirements?
C.9	How does the SoS respect a good representation of the requirements?
C.10	How does SoS aim to reduce execution time and optimize resource consumption?
C.11	How does the SoS dynamically reconfigure the heterogenous set of CSs?
C.12	How do the SoS' constituents interact with each other at runtime?
C.13	Can new CSs be integrated into the SoS at any time?
C.14	How do CSs adapt to each other and the environment?
C.15	What are the new capabilities that can be added after the integration of the different CS?

C. SoSE processes

SoSE processes express the activities engaged under SoSE from the perspective of one or more Stakeholders to frame specific Concerns. The main SoS development processes involved in our SoS-AF are:

- SoS Knowledge: addresses high-Level SoS requirements and investigates existing CSs that can participate in the SoS.
- CSs Selection: this process consists of choosing a set of CSs and distinguishing their relevant Capabilities and Goals.
- Conceptual Design: the design involves creating a global vision of an SoS, defining the essential relationships and identifying mission capability assessment.
- Architectural Design: represents a global architecture for the SoS' constituents and their possible Roles. It could be developed in parallel with the CSs Selection process.

- Interaction: the different CSs involved in an SoS usually have different Capabilities. Therefore, a large part of the software engineering effort in the SoSE is to design interactions so that the CSs can interoperate.
- Integration & deployment: this process implies that the different CSs involved in the SoS work together and interact through the assigned Roles.

D. Associations: Stakeholders, Concerns and Processes

We detail in this section the association rules between both, the concerns and the stakeholders, and between concerns and SoSE processes. We present these concerns according to each group of stakeholders (columns) and the corresponding process (lines) as illustrated in Table.3:

TABLE 1: ASSOCIATIONS OF STAKEHOLDERS, PROCESSES AND CONCERNS.

Stakeholder Process	SoS Experts	Architects Designers	Collaboration Specialists	Interactions Engineers
SoS Knowledge	C.1 C.2			
CSs Selection		C.6	C.9	
Conceptual Design		C.7	C.10 C.11	
Architectural Design	C.3	C.4 C.15	C.8	
Interaction			C.5	C.14
Integration & deployment			C.12	C.13

E. Architecture Viewpoints

A viewpoint is a selection of relevant aspects of the SoSE processes (and their Stakeholders' concerns); and the representation of that part of an architecture that is expressed in different Model Kind. It is claimed that the SoSE processes form a necessary and sufficient set to meet the needs of the framework. Four main VPs are identified in our proposed SoS-AF:

- Business viewpoint: in this stage, the SoS Experts focus on the business aspect, which mainly includes the knowledge and requirements of CSs. These requirements address a set of architecture challenges i.e. C.1 and C.2. The Business viewpoint is described by the Knowledge process which creates a rough draft of the SoS requirements including identification of the possible Capabilities, which could be derived from the existing CSs of the application domain.
- Analysis viewpoint: the Architects, Designers and Collaboration Specialists initiate modeling (C.6 and C.9) by starting the process CSs Selection to select the appropriate CSs involved in providing the required SoS capabilities. And then, by building the design model, which will offer a global understanding of CSs (C.7, C.10 and C.11) in the conceptual design process.
- Design viewpoint: it is based on two independent and complementary development processes, i.e. Architectural and Interaction. In the first process, the architects and the designers provide a detailed model

to support different design decisions (C.3, C.4, C.8 and C.15); and in the second one, the stakeholders deal with the fundamental interdependencies (between CSs, Roles, and their collaborations.) designing such architectures.

- Deployment viewpoint: This viewpoint is a process of deploying and integrating CSs and it is established by the collaboration and interaction experts to merge two or more diverse Roles that are designed to define, control, and monitor complex interactions (C.12 and C.13) that extend across SoS and CSs boundaries.

TABLE 2: ASSOCIATION OF VIEWPOINTS AND SoS DEVELOPMENT PROCESSES.

Viewpoint Process	Business	Analysis	Design	Deployment
SoS Knowledge	X			
CSs Selection		X		
Conceptual Design		X		
Architectural Design			X	
Interaction			X	
Integration & deployment				X

F. Model kinds and UML-Profile

The proposed architecture must support the modeling of all the concepts and relationships of the SoSs entities with their different static and dynamic aspects. Thus, we also plan to propose in the near future another contribution that is based on extending UML in a standard way, resulting in an “SoS-UML Profile for SoS framework”.

IV. APPROACH PRINCIPLE

We need to create different SoSs views matching the framework viewpoints at high-level abstraction, our objective is to model SoSs architectures and abstract their analysis and design from implementation technologies and execution platforms' specificities. Therefore, a platform-independent modeling language is needed to increase the automation of the development of an SoS and to allow abstract modeling, which can be transformed later to a PSM and then to a platform-specific implementation.

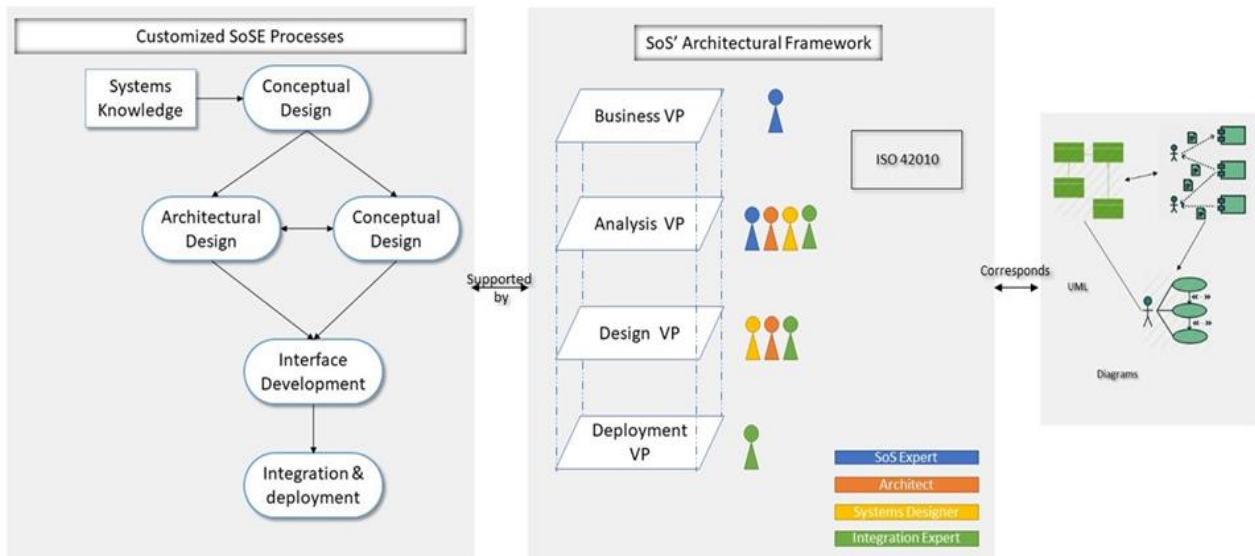


Figure 2: SoSs' Architecture Framework.

The proposed approach takes advantage of the MeMSoS Meta-Model [4] which supports the PIM design level and allows the designer to successively refine the PIM in terms of SoSs aspects. MeMSoS has been designed in order to explicitly model and represent all SoS architectural elements. Additionally, it also gives particular attention to the hierarchical composition of CSs and highlights the importance of CS, Roles, Capabilities, Goals, and Links in the whole SoS architecture.

We have significantly refined the MeMSoS by extending its elements for better support of different viewpoints modeling, framing other stakeholders' concerns and introducing new features. We refined it with new extended abstract and corresponding concrete syntaxes in order to reach high-level specifications of aspects and address the cross-cutting concerns that depend on the Stakeholder's VPs.

A new SoSs' Architecture framework Model based on ISO/IEEE42010:2011 standard is introduced in the previous section. It focuses on a structured description of heterogenous CSs of one SoS. In addition, the framework allows the validation of SoSs use cases. Indeed, it is intended to present the design of SoSs use cases in different application domains according to the mentioned architectural viewpoints, and more specifically, allows the validation of Aircraft Emergency Response SoS use case and its support by standards. The framework provides a set of processes, viewpoints...etc, as well as a tool to map use case processes and thus a structured approach for SoS development. With its peer coordination set of viewpoints/processes, the framework will allow depicting various interrelated aspects of SoSs architectures and supports the interdependencies and collaboration of constituents on different levels of abstraction.

As shown in Figure.2, the framework addresses the description of SoSs' architectures developed within a specific domain of application and/or group of stakeholders employing a set of development processes that supports SoSs'

engineering. The framework model adopts the ISO 42010 standard and gives guidance and recommendations for the development of such solutions in order to meet the SoSs' requirements.

In this paper, we took the first step towards achieving a unified Architecture Framework supporting SoSs Development, by proposing this Multiview point model that will provide the means to design informed architectural solutions. In the next work, we aim to provide an SoS-UML Profile to model the set of SoSs' processes. The advantage of this profile is to provide a large number of models to separately capture, describe and organize each of the processes of different viewpoints.

V. CONCLUSION

To master all the related concerns, we proposed in this paper a model of Architecture Framework, that can facilitate and improve the design of SoSs' architectures. We aimed to foster the systematic development of SoSs' architectures by maintaining the consistency and coherence between the different viewpoints of different stakeholders. In the near future, we plan to implement the proposed approach. The latter will offer the audience of SoSs' Stakeholders the tools to facilitate the task of modeling multi-viewpoints architectures that are managed by different SoSE processes and documented through the SoS-UML profile's models.

REFERENCES

- [1] Jamshidi, M. (2011). From large-scale systems to cyber-physical systems. *Journal of Internet Technology*, 12(3), 367-374.
- [2] Dridi, C. E., Benzadri, Z., & Belala, F. (2020, November). System of Systems Modelling: Recent work Review and a Path Forward. In *2020 International Conference on Advanced Aspects of Software Engineering (ICAASE)* (pp. 1-8). IEEE.

- [3] ISO. (2011). Systems and software engineering—architecture description. Technical report, ISO/IEC/IEEE 42010, 2011.
- [4] Dridi, C. E., Benzadri, Z., & Belala, F. (2020, June). System of Systems Engineering: Meta-Modelling Perspective. In 2020 IEEE 15th International Conference on System of Systems Engineering (SoSE) (pp. 000135-000144). IEEE.
- [5] Chaabane, M., Bouassida, I., & Jmaiel, M. (2017, April). System of systems software architecture description using the ISO/IEC/IEEE 42010 standard. In Proceedings of the Symposium on Applied Computing (pp. 1793-1798).
- [6] Chaabane, M., Rodriguez, I. B., Colomo-Palacios, R., Gaaloul, W., & Jmaiel, M. (2019). A modeling approach for Systems-of-Systems by adapting ISO/IEC/IEEE 42010 Standard evaluated by Goal-Question-Metric. *Science of Computer Programming*, 184, 102305.
- [7] Klein, J., & van Vliet, H. (2018). System-of-Systems Viewpoint for System Architecture Documentation. arXiv preprint arXiv:1801.06837.
- [8] Mori, M., Ceccarelli, A., Lollini, P., Bondavalli, A., & Fromel, B. (2016). A holistic viewpoint-based SysML profile to design systems-of-systems. 2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE). <https://doi.org/10.1109/hase.2016.21>
- [9] Mori, M., Ceccarelli, A., Lollini, P., Frömel, B., Brancati, F., & Bondavalli, A. (2017). Systems-of-systems modeling using a comprehensive viewpoint-based SysML profile. *Journal of Software: Evolution and Process*, 30(3), e1878.
- [10] Jamshidi, M. (2011). From large-scale systems to cyber-physical systems. *Journal of Internet Technology*, 12(3), 367-374.
- [11] Wickramasinghe, N., Chalasani, S., Boppana, R. V., Madni, A. M. (2007, April). Healthcare system of systems. In 2007 IEEE International Conference on System of Systems Engineering (pp. 1-6). IEEE.
- [12] Yutaka Hata, Syoji Kobashi, and Hiroshi Nakajima. Human Health Care System of Systems. *IEEE SYSTEMSJOURNAL*, VOL. 3, NO. 2, JUNE 2009.
- [13] Lane, J. A., Epstein, D. (2013). What is a System of Systems and why should I care?. University of Southern California
- [14] The American Department of Defense (DoD). *SystemsEngineering Guide for Systems of Systems*. 2008.
- [15] DeLaurentis, D. (2005, January). Understanding transportation as a system-of-systems design problem. In 43rd AIAA Aerospace Sciences Meeting and Exhibit (p. 12)
- [16] M.Maier. On architecting and intelligent transport systems. *Joint Issue IEEE Trans. Aerosp. Electron.Syst./Syst. Eng.*, vol. 33, pp. 610-625, Apr. 1997.
- [17] Gunes, V., Peter, S., Givargis, T., Vahid, F. (2014). A survey on concepts, applications, and challenges in cyber-physical systems. *KSII Transactions on Internet Information Systems*, 8(12)
- [18] Neto, V. V. G., Lebttag, B. G. A., Teixeira, P. G., Batista, P., Lopes, V. C., El-Hachem, J., ... & Nakagawa, E. Y. (2021). Expanding Frontiers: Settling an Understanding of Systems-of-Information Systems. arXiv preprint arXiv:2103.14100.
- [19] Lana, C. A., Guessi, M., Antonino, P. O., Rombach, D., & Nakagawa, E. Y. (2018). A systematic identification of formal and semi-formal languages and techniques for software-intensive systems-of-systems requirements modeling. *IEEE Systems Journal*, 13(3), 2201-2212.
- [20] Morel, G., Panetto, H., Mayer, F., & Auzelle, J. P. (2007, October). System of enterprise-Systems integration issues: an engineering perspective. In IFAC Conference on Cost Effective Automation in Networked Product Development and Manufacturing, IFAC-CEA'07 (p. CDROM). Elsevier.
- [21] Ilic, M. D., Xie, L., Khan, U. A., & Moura, J. M. (2008, July). Modeling future cyber-physical energy systems. In 2008 IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century (pp. 1-9). IEEE.
- [22] Li, S., Ni, Q., Sun, Y., Min, G., & Al-Rubaye, S. (2018). Energy-efficient resource allocation for industrial cyber-physical IoT systems in 5G era. *IEEE Transactions on Industrial Informatics*, 14(6), 2618-2628.
- [23] Jamshidi, M. (2011). From large-scale systems to cyber-physical systems. *Journal of Internet Technology*, 12(3), 367-374.
- [24] Maier, M. W. (1998). Architecting principles for systems-of-systems. *Systems Engineering: The Journal of the International Council on Systems Engineering*, 1(4), 267-284.
- [25] How Should We Use the Term “System of Systems” and Why Should We Care?. In *INCOSE International Symposium* (Vol. 16, No. 1, pp. 427-438)
- [26] D. Emery and R. Hilliard, “Every architecture description needs a framework: Expressing architecture frameworks using ISO/IEC 42010,” in *WICSA/ECSA 2009*, 2009