Enterprise Decision-Making in System-of-Systems Considering Different Levels of Centralized Management

Jasamin Akbari
Institute of System Architectures in Aeronautics
German Aerospace Center (DLR)
Hamburg, Germany
Jasamin.Akbari@dlr.de

Giuseppa Donelli
Institute of System Archtectures in Aeronautics
German Aerospace Center (DLR)
Hamburg, Germany
Giuseppa.Donelli@dlr.de

Ludvig Knöös Franzén
Department of Management and Engineering
Linköping University
Linköping, Sweden
Ludvig.Knoos.Franzen@liu.se

Luca Boggero
Institute of System Archtectures in Aeronautics
German Aerospace Center (DLR)
Hamburg, Germany
Luca.Boggero@dlr.de

Björn Nagel
Institute of System Architectures in Aeronautics
German Aerospace Center (DLR)
Hamburg, Germany
Bjoern.Nagel@dlr.de

Christopher Jouannet
Department of Management and Engineering
Linköping University
Linköping, Sweden
Christopher.Jouannet@liu.se

2025 20th Annual System of Systems Engineering Conference (SoSE) | 979-8-3315-1535-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/SOSE66311.2025.11083824

***Abstract*— Systems are increasingly deployed in environments where they interact with other systems to deliver enhanced and integrated capabilities. Together, these systems are referred to as system-of-systems. While the benefits of a system-of-systems can lie in the emergent behavior that the systems produce together, the challenges that arise when developing and evolving a system-of-systems are complex. Each constituent system – of which the system-of-systems is composed – may be at a different stage in its lifecycle and owned or managed by a separate enterprise. These enterprises must make independent decisions about the acquisition, development, or use of their systems. In some cases, a central enterprise may hold decision-making authority over multiple constituent systems, but this is not always the case. This paper examines the role of multiple enterprises in decision-making within a system-of-systems, identifies critical gaps, and synthesizes these insights to outline a direction towards a system-of-systems multi-enterprise collaboration.**

Keywords—system-of-systems, system-of-systems engineering, decision-making, system architecting, multi-enterprise collaboration

I. Introduction

An engineered system progresses through multiple lifecycle stages over the course of its existence. The system lifecycle extends from the initial definition of customer needs to final disposal [2]. At each stage, different decision-makers may be involved, depending on the responsibilities and expertise required [3]. When the lifecycle stage changes, decision-making responsibility over the system can be reassigned either to another enterprise or to different individuals within the same organization. The term enterprise refers to one or more organizations united by a clear mission, shared goals, and defined objectives to deliver a product or service [4, 5]. During the operational lifecycle stage, the system is operated by an enterprise that provides a service to the customer. This stage is when the system is actively used and generates revenue. It is considered the primary lifecycle stage, while the other lifecycle stages, although essential, primarily serve to enable it. For example, in the case of a commercial aircraft, engineers and domain experts are responsible for its development, the manufacturer builds the aircraft, and the airline operates it during the operational lifecycle stage. In this context, the airline serves as the service provider to end customers.

As systems are brought into operation, they increasingly operate in collaborative constellations and deliver capabilities which the individual systems cannot deliver on their own. These systems are more often referred to as system-of-systems (SoS) [5]. An end customer can benefit from the collaborative behavior enabled by multiple systems, individually referred to as constituent systems (CS). An example of an SoS is a multimodal transportation network. A passenger may undertake an intercity journey by first using public transportation in one city to reach the central station, then taking a long-distance train to another city, and finally using a bus to arrive at the final destination. Through the integration of these different transportation modes the traveler can move seamlessly and efficiently between two locations. In an SoS, attention is given not only on the interfaces between CSs but also on their collaborative operation. When the interplay of these systems is recognized and acknowledged as an SoS, engineers involved in its development and evolution can better emphasize the potential benefits of an SoS approach. However, this becomes particularly challenging when multiple enterprises are simultaneously involved in developing or operating the individual systems, as these systems are often not owned or managed by a single entity. Each enterprise typically has its own objectives and makes

979-8-3315-1535-5/25/$31.00 ©2025 IEEE

independent decisions about which systems to develop or acquire.

A key challenge in operating an SoS is enabling effective collaboration among independently managed systems, while respecting the autonomy of the enterprises that own them. Each enterprise follows its own goals and decision-making processes, which may not align with the broader purpose of the overall SoS. A central difficulty, therefore, is enabling decision-making that encourages enterprises to contribute to shared outcomes while respecting their organizational independence. This paper examines the limitations and unresolved challenges of enterprise decision-making in SoS operations, with particular attention to how decision-making is shaped by the involvement of multiple autonomous enterprises.

II. System-of-Systems Characteristics and Development Approaches

The combination of individual systems referred to as SoSs are classified as such based on two criteria; managerial and operational independence of the CS. Additional characteristics of SoS include geographical distribution, evolutionary development, and emergent behavior. [6] The emergent behavior can be the largest benefit of an SoS. Emergence, described in the context of single systems [2], refers to behaviors or capabilities that arise from interactions among individual elements or systems, resulting in outcomes that cannot be fully understood or achieved by examining the parts in isolation [2, 7]. The evolutionary development on the other hand describes that the SoS evolves over time, new CSs may be added, while other CSs may be removed. This characteristic reflects the different lifecycle stages in which the CSs are situated and emphasizes that an SoS is typically not developed from scratch, but rather evolves gradually.

The defining criteria of managerial and operational independence indicate that CS can continue to operate independently and provide functionality even if removed from the SoS. Based on these criteria, SoS are further classified by their level of centralized management or control. This classification describes how management and decision-making power are distributed across the SoS in terms of centralization [6]. It determines whether decision-making power is concentrated in a single entity or shared among multiple entities. In the literature four distinct types of SoS can be found [6, 8]. The four types, based on the level of centralized management, are illustrated in Fig. 1. On the left side of the figure, the SoS exhibits highly centralized management while maintaining managerial independence. In contrast, the right side illustrates a progression toward increasingly decentralized management. The four types of SoS are: directed, acknowledged, collaborative, and virtual SoS. The figure also depicts the ownership of individual CSs by various enterprises, as defined by the degree of centralized management. These enterprises refer to those involved in the operational lifecycle stage, responsible for owning and/or operating the systems. Fig. 2 illustrates that as the level of centralized management increases, the purpose of the involved systems and their underlying enterprises becomes more aligned. Conversely, with more decentralized management, the systems and enterprises tend to pursue more individual purposes. Moreover, it highlights that engineering decentralized SoS presents greater challenges.

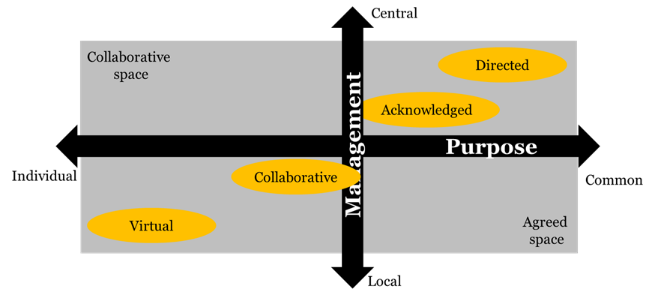

Fig. 2.: Impact of centralized management on purpose alignment in systems-of-systems.

In an SoS, selecting the appropriate constellation of CSs is essential, as it defines the resulting SoS architecture. Choosing between architecture alternatives requires decision-making. This decision-making process is already a fundamental part of system development which is commonly understood as defining the problem around an unknown system of interest (SoI) and iteratively refining the solution. To support this process, well-established methodologies guide the definition of requirements, exploration of alternatives, and reasoned decision-making. Systems Engineering (SE) provides a structured approach for the analysis and development of systems throughout their entire lifecycle. The lifecycle processes are most commonly described in the ISO standards 15288 [2]. The field of system-of-systems engineering (SoSE), on the other hand, addresses how multiple systems, which are often at different stages of their lifecycle, can be deployed and integrated to achieve higher-level capabilities that individual systems alone cannot realize [5].

While developing a single system with defined interfaces may appear straightforward, engineering an SoS involves significantly greater complexity, as highlighted by various approaches. Architecture frameworks, which provide standardized guidelines for representing systems, are commonly

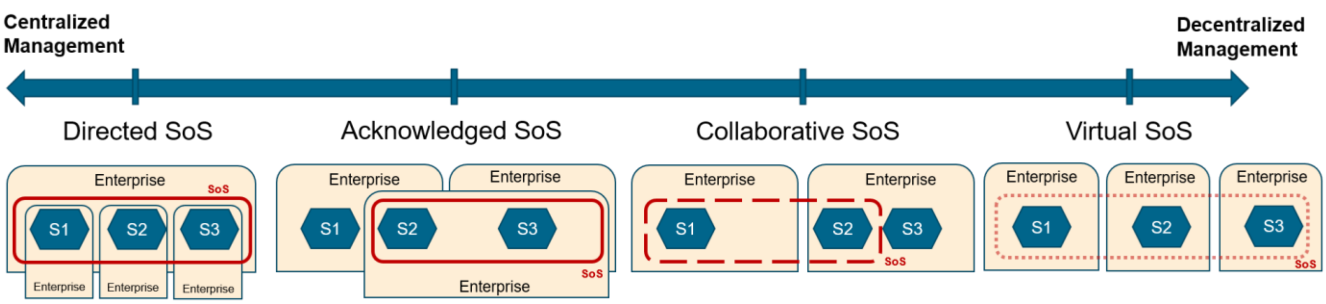

Fig. 1.: Classification of systems-of-systems by their level of centralized management [1].

used to guide the modeling and understanding of SoS. Prominent examples include the U.S. Department of Defense Architecture Framework (DoDAF) [9, 10], the U.K. Ministry of Defence Architecture Framework (MODAF) [10, 11], and the Unified Architecture Framework (UAF) [11]. DoDAF and MODAF, both originating in the military domain, offer structured frameworks that define concepts and models to support decision-making and capability-based system analysis. The UAF supports standardized modeling of SoS across various domains (e.g., operational, strategic) using model-based systems engineering (MBSE). Similarly, the Systems Modeling Language (SysML), typically used to describe a single complex system [12], has also been applied to model multiple systems within an SoS context [13]. Other approaches address the evolutionary nature of SoS, emphasizing that SoS upgrades over time play a critical role in their evolution and performance [8, 14]. While these approaches recognize decisions within the development or evolution of SoS, they are not recognizing distributed decision-making power through multiple enterprises involved during the operations.

III. Enterprises in System-of-Systems

During the operations of SoS, enterprises may be involved in providing services to customers or performing operating activities. Depending on the level of centralized management and the interests of each enterprise, decision-making power can be concentrated on a single CS (e.g., collaborative SoS) or spread across multiple CSs (e.g., directed SoS). As a result, an enterprise's SoI may be either a single CS or the entire SoS. Similarly, an enterprise may provide services or conduct operations using one or multiple CSs. The varying goals and objectives of each enterprise lead to different strategic orientations, which can shift the focus of the SoI. For example, consider a logistics company that operates a fleet of delivery vehicles. In a collaborative SoS, the company's SoI is limited to managing and optimizing its own vehicles while interfacing with other independent systems such as traffic management or warehouse systems. However, in a directed SoS, the same enterprise might take on responsibility for coordinating multiple CSs, including not only its own vehicles but also rail freight and air cargo.

The motivation for introducing a new engineering solution typically arises from an established problem situation that existing solutions cannot adequately address due to changing stakeholder needs. Based on the defined problem, system capabilities are identified and may be distributed across multiple systems to resolve the issue and achieve the desired outcome. Various Concepts of Operation (ConOps) can be explored to implement these capabilities. However, the selection among different concepts is influenced by the strategies and capabilities of the enterprises involved in the SoS. A train operator, despite recognizing the feasibility of using ships for transport, would typically not propose such a concept due to strategic and operational constraints. Once general concepts are outlined, architecting focuses on conceptualizing operations and identifying operations of potential systems. For example, a train operator's capability may center on direct city center travel without intermediate stops. Functions like achieving high-speed travel support these operations, and logical elements are identified to fulfill them. This process tailored for directed SoS is further described in [1].

In this envisioned approach, an SoS architect is responsible for guiding the process in close collaboration with the enterprises that operate the SoS or deliver services through it. This role represents a new and emerging position that was traditionally managed by engineers within the developing enterprise. However, a developer such as an original equipment manufacturer (OEM) is typically focused on a single system, as it holds responsibility only for that system's development. In contrast, an SoS delivers value through the collaborative behavior of multiple systems, reflecting operational needs at the level of the SoS. This creates a fundamental tension: while enterprises focus on the development or operation of individual systems, an SoS requires consideration of how these systems interact to fulfill broader capabilities. As a result, misalignment can occur between the goals of individual enterprises and the collective objectives of the SoS.

The different types of enterprises and their capabilities are depicted in Fig. 3. The SoI can be either the entire SoS or a single CS within the collaboration. Each operated system contributes a capability, while the SoS delivers capabilities through the collaboration of multiple systems. These capabilities emerge from interactions within a system or among systems in the SoS. The enterprise responsible for developing parts of the SoS, such as a CS, possesses capabilities shaped by its experience,

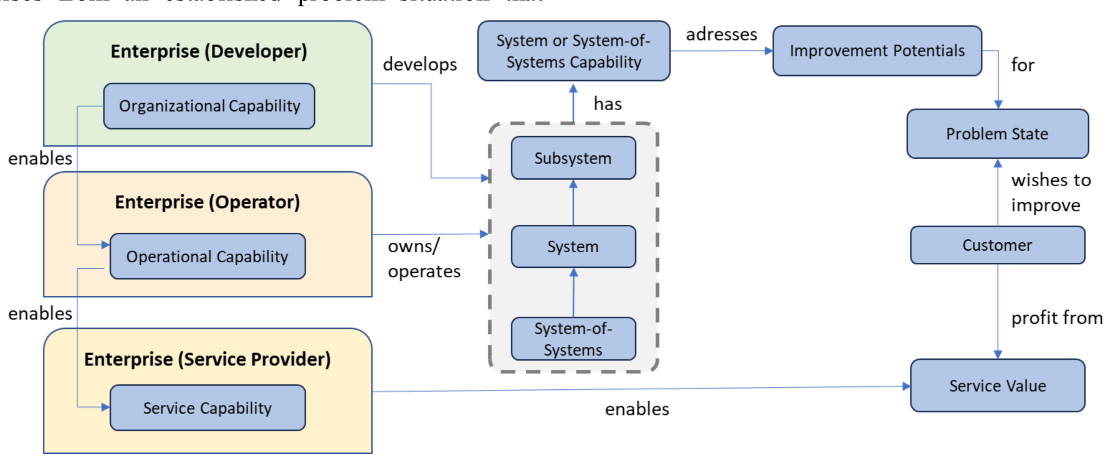

Fig. 3.: Enterprise capabilities in addressing a customer's problem state through a system or system-of-systems.

strategic orientation, and business goals. The individual systems or the entire SoS are operated by an operating enterprise, which delivers specific operational capabilities. Often, when multiple systems collaborate to provide a combined capability, a service provider offers a service to customers. In some cases, such as the airline example, the operator and service provider are the same enterprise. Through the provided service, value is created that benefits the customers. It is the customer, whether an individual or a community, who defines the problem that needs solving. The system or SoS then offers a potential solution to improve this problem state.

While many enterprises are involved in the operation of a CS or SoS, different enterprises also take on various roles during development. Enterprises that operate or provide services using a CS or SoS have specific operational needs and contribute a set of requirements for systems that either need to be acquired (buy) or developed from scratch (make). In the second case, an OEM may develop a new system based on the defined requirements. Since the OEM has limited capabilities, it delegates certain responsibilities to suppliers. These suppliers are categorized as tier 1 or tier 2, as shown in Fig. 4. The OEM is responsible for developing and assembling the entire system, while the tier 1 supplier provides subsystems to the OEM and the tier 2 supplier supplies components to the tier 1 supplier. The figure illustrates that a CS is connected to an OEM and its supply chain, whereas an SoS involves multiple such supply chains.

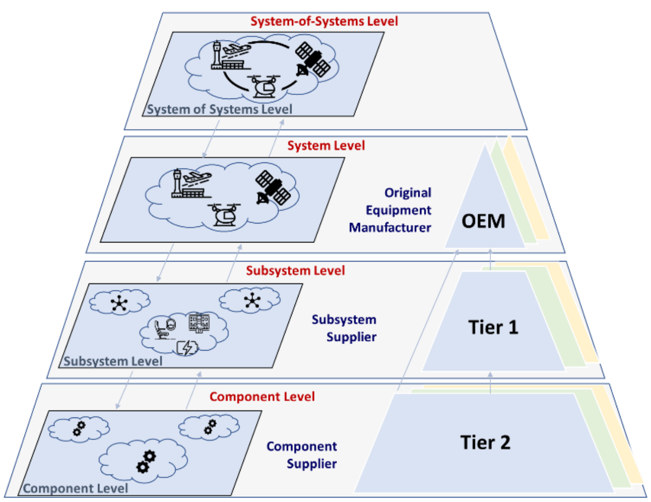

Fig. 4.: System level elements of a system-of-systems corresponding to their manufacturers and suppliers [adapted from 15].

IV. Vision for Enterprise Decision-Making in Systems of Systems

As discussed in the previous section, multiple enterprises are involved not only throughout the system's lifecycle but also within a single lifecycle stage. The enterprise responsible during operations, whether acting as the operator or service provider, must decide which CSs to include or exclude from the SoS. Additionally, decisions must be made regarding how these systems should be operated. The process and authority for making these decisions depend on the level of centralized management described earlier. Decision-making power can be distributed across the individual CSs within the SoS, making decisions more complex as they must accommodate the diverse goals and interests of the involved enterprises.

A. *Directed System-of-Systems*

In a directed SoS, a high-level enterprise is responsible for providing a specific service or operating the entire SoS. Each CS within the SoS is owned and operated by a separate enterprise. In this arrangement, the purpose of each CS is subordinated to the overall goal of the SoS, which aligns with the objectives of the high-level enterprise. For example, a wildfire-fighting force acts as the high-level enterprise with the central mission of combating wildfires. This force delegates different tasks, each requiring specific capabilities, to various enterprises operating their respective vehicles and equipment. For instance, fire trucks managed by a fire department and drones used for fire detection may both be part of this SoS. The SoS has a unified purpose: fighting the fire. Decisions about which CSs to include are based on their ability to contribute effectively to this central purpose.

B. *Acknowledged System-of-Systems*

In an acknowledged SoS, a high-level enterprise provides a central service or operates the overall SoS. Similar to a directed SoS, there is a central purpose guiding the collaboration. However, the individual CSs are owned and operated by different enterprises, each pursuing its own objectives while aligning with the high-level purpose.

For example, a mobility service provider may serve as the high-level enterprise, offering tickets and managing travel routes and schedules. The owners of transportation assets, such as trains or buses, operate their vehicles independently. Each operator has its own goals, like covering specific routes or distances. While they pursue their individual objectives and may participate in other SoS, their purposes align with the service provider's overarching goal. If the route network needs expansion, the service provider can decide to incorporate additional bus fleets from different operators.

C. *Collaborative System-of-Systems*

The third type of SoS lacks a high-level enterprise that provides a central service or operates the entire system. Instead, decision-making power and management are distributed among the CSs and their enterprises, which collaborate voluntarily. Each enterprise pursues its own purpose, while collectively shaping an agreed-upon central purpose based on their individual goals.

An example of a collaborative SoS is an urban air mobility network, which includes vehicles and their operators, vertiport infrastructure, and recharging facilities. All participating enterprises benefit from this collaboration, so the decision to include a particular enterprise and its CS is made jointly, based on whether the involvement benefits all partners involved.

D. *Virtual System-of-Systems*

Lastly, the virtual SoS is characterized by highly decentralized management. Individual CSs and their enterprises contribute to SoS capabilities unintentionally or without coordinated planning. There is no central enterprise and no overarching purpose guiding the system. A large-scale transportation network serves as an example, where passengers travel across regions that each maintain their own local transportation systems. Enterprises in these regions operate independently, pursuing their own objectives. Even without direct collaboration, passengers benefit from the coexistence and availability of diverse transportation options. Decisions by the enterprises are made independently, without alignment through a central authority.

Unlike other SoS types, the virtual SoS does not significantly differ from traditional systems engineering in terms of decisions about which systems to include or develop. However, system interfaces may still be designed or adapted to support limited interoperability.

V. Discussion and Conclusions

The emergent behavior and higher-level capabilities of SoS are key incentives for placing greater emphasis on the acknowledgment, understanding and exploration of SoS. To ensure effective operation of SoS, CSs may need to be added or removed over time. Moreover, the collaboration between systems must be continuously adjusted to achieve optimal outcomes. These changes are ultimately driven by the need to realize specific capabilities that cannot be achieved by individual systems alone.

Throughout the lifecycle of an SoS, different enterprises play critical roles, much like in traditional SE. However, unlike SE, where development and operations are typically concentrated within a single organization, SoSE involves multiple enterprises not only across the lifecycle but simultaneously during the operational phase. This complexity underscores the importance of the level of centralized management, which fundamentally shapes how decisions about system integration and capability realization are made. A key insight of this work is that the more decentralized the SoS, the more difficult it becomes to make unified, SoS-wide decisions since the purpose of the CSs is individual. Despite this, most real-world SoSs operate without a central manager, highlighting a critical need: decision-making support mechanisms that facilitate collaborative decision-making among independent enterprises. This is where the role of the SoS architect becomes crucial. Although decision-making is most straightforward in centrally managed SoSs, most real-world examples operate without centralized control. The SoS architect supports enterprise decision-making by guiding which CSs to acquire or remove, with the goal of maximizing value for all participating partners. However, in the absence of shared resources, the question of who assumes responsibility for and funds this role remains open, underscoring an important area for future research.

Acknowledgement

The research presented in this paper has been performed in the framework of the COLOSSUS project (Collaborative System of Systems Exploration of Aviation Products, Services and Business Models) and has received funding from the European Union Horizon Europe program under grant agreement No. 101097120.

References

[1] J. Akbari, G. Donelli, L. Boggero, and B. Nagel, "Architecting System-of-Systems with Different Levels of Centralized Decision-Making," 2024.

[2] D. D. Walden, T. M. Shortell, and G. J. Roedler, "INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities," 2023.

[3] S. Arnold and H. W. Lawson, "Viewing systems from a business management perspective: The ISO/IEC 15288 standard," *Systems Engineering*, vol. 7, no. 3, pp. 229–242, 2004, doi: 10.1002/sys.20006.

[4] *Systems and software engineering — Architecture description*, 42010:2011, ISO/IEC/IEEE, 2011. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6129467.

[5] Incose, "INCOSE Systems of Systems Primer," 2018.

[6] M. W. Maier, "Architecting principles for systems-of-systems," *The Journal on International Systems Engineering*, vol. 1, pp. 267–284, 1998. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291520-6858%281998%291%3A4%3C267%3A%3AAID-SYS3%3E3.0.CO%3B2-D

[7] David D. Walden, Garry J. Roedler, Kevin J. Forsberg, and R. Douglas Hamelin and Thomas M. Shortell, *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 2015.

[8] Dahmann *et al.,* "Systems Engineering for Capabilities," 2008.

[9] DoD, "DoD Architecture Framework (DoDAF): version 2.0," Washington, DC, USA, 2009.

[10] T. Weilkiens, J. G. Lamm, S. Roth, and M. Walker, *Model-Based System Architecture* (Wiley series in systems engineering and management). Hoboken New Jersey: Wiley, 2016.

[11] OMG, "Unified Architecture Framework (UAF) Unified Architecture Framework (UAF) Sample Problem (Informative)," 2019.

[12] OMG, "OMG Systems Modeling Language (OMG SysML™): Hybrid SUV Non-Normative Example,"

[13] C. Swicklane, T. A. Mazzuchi, and S. Sarkani, "A methodology for developing SoS architectures using SysML model federation," 2023. doi: 10.1002/sys.21727. [Online]. Available: https://incose.onlinelibrary.wiley.com/doi/epdf/10.1002/sys.21727

[14] Z. Fang and D. DeLaurentis, "Dynamic Planning of System of Systems Architecture Evolution," *Procedia Computer Science*, vol. 28, pp. 449–456, 2014, doi: 10.1016/j.procs.2014.03.055.

[15] N. Bey. "Future Technology Photonics – Is the Product Development Ready for Automotive