# The Use of Digital Twins to Bloster the Resilience of an IoT System of Systems Infrastructure

Meriem Smati[1,2][0009-0006-9399-3356], Jannik Laval[1][0000-0002-7155-5762], Vincent Cheutet[1][0000-0003-1920-2609], and Christophe Danjou[2][0000-0002-9575-0087]

[1] INSA Lyon, Université Lumière Lyon 2, Université Claude Bernard Lyon 1, Université Jean Monnet Saint-Etienne, DISP UR4570, Villeurbanne, 69621, France
meriem.smati@insa-lyon.fr, jannik.laval@univ-lyon2.fr, vincent.cheutet@insa-lyon.fr

[2] Laboratoire Poly-Industrie 4.0, Département de Mathématiques Et Génie Industriel, Polytechnique Montréal, Montréal, Québec, Canada
christophe.danjou@polymtl.ca

## 1 Introduction

### 1.1 Theoretical background and research gap

Amid the ongoing evolution and widespread adoption of Internet of Things (IoT) devices in today's data-driven world, achieving resilience within a System of Systems (SoS) architecture presents a significant challenge. An SoS involves integrating multiple independent systems into a larger, more complex network [8], which introduces additional layers of complexity due to the increased number of interconnected components, diverse system behaviors, and the higher likelihood of emergent anomalies that might not be predictable within isolated subsystems. Ensuring uninterrupted functionality in the face of potential disruptions like cyber-attacks, power outages, physical damage, and data anomalies is crucial for the resilience of such systems. Many traditional approaches exist to mitigate such risks, for example incorporating redundancy and failover mechanisms into the design of infrastructure and industrial systems [7]. Redundancy entails integrating backup devices, networks, or data centers to enable system continuity in case of component failure. However, implementing this methodology would require a significant investment of both finances and resources.

The objective of this thesis is to tackle these problems by using Digital Twins (DTs) and developing a resilient architecture for a SoS DT within an IoT infrastructure, by adhering to the fundamental characteristics of both SoS and DTs. This research aims to ensure that the designed architecture can withstand various challenges and maintain operational integrity. In this first year of thesis, the primary focus is on addressing data anomalies—commonly referred to as outliers, aberrations, or irregularities—that present significant obstacles in the current data-centric era, since traditional approaches have emphasized physical resilience and redundancy, but this research prioritizes data integrity. Data anomalies, which are deviations from expected behavior in interconnected IoT networks, must be detected and understood to ensure system reliability and integrity across various sectors, including manufacturing, healthcare, agriculture, and smart cities. Additionally, predicting potential issues in advance is a crucial aspect of this work. Another focus is on realizing the architecture of an SoS digital twin and understanding the communication and interoperability within this framework, very few articles mention the creation of an SoS DT:

- Olsson and Axelsson [6] provide a short survey of the current state of DTs in SoS and outline a research agenda to address the identified gaps and challenges and proposed two architecture perspectives (one DT for the entire SoS or one DT for each constituent manner in a distributed manner). However, each architecture has its limits, on one hand, the monolithic architecture has problems scalability, single point failure complex management and integration challenges and on the other hand, the distributed architecture can show concerns about interoperability, synchronization and coordination.
- Michael et al. [9] addresses the complexity of integrating DTs into an SoS and among the presented challenges are Horizontal Integration (Integrating different views and components of DTs), Vertical Composition (Aligning data, models and services across abstraction levels can lead to conflicts in granularity), Composition of DTs for different perspectives, Connection of independently developed systems to a SoS, different Life-cycle representations of the original system and Composition of heterogeneous twin implementations.
- Borth et al. [2] presented four challenges concerning DTs for SoS, which are Operational independence, data and information sharing, the dynamic nature of SoS and Long lifetimes and lifespan, as well as potential architectural strategies for the DT: Incremental evolution to accommodate updates and changes in the SoS, knowledge preservation and hadeling shared goals and conflicts.
- Göllner et al. [4] addresses the challenge of integrating DTs in Industry 4.0 production systems, which are seen as SoS. They highlight the need for well-coordinated and standardized DTs, but existing standards, like the Asset Administration Shell (AAS) mainly focus on representation and lack support for coordination between different DTs. The resulted model "Use Case Specification Model" (UCSM), which documents interactions of DTs to solve specific use cases and facilitates semi-automated code generation since the DT content is use case driven.
- Harbor Research [5] mentions the need for open, composable, and interoperable solutions. Key applications span various domains, and the paper calls for standardized approaches to facilitate seamless integration and dynamic interaction, ensuring efficient and effective implementation of DTs.

### 1.2 Research questions

The research questions are the following:

– How can DT be used as a tool for detecting and preventing data disturbance in an IoT system?
– How does a SoS communicate with its digital replica (SoS DT)?
– How to create a resilient architecture of an SoS DT?

## 2 Proposition

The methodology followed to answer the mentioned questions is to first establish a communication between a physical system and its digital counterpart to understand the communication between these two, and then move on to an SoS architecture.

### 2.1 CSDT Proposal

IoT systems rely on a multitude of sensors and interconnected devices, have proven to be invaluable in enhancing efficiency and convenience. However, the seamless operation of IoT systems is not without its challenges. In the pursuit of ensuring the reliable and secure operation of IoT systems, the novel concept of Cognitive Super-Digital Twin (CSDT) is created.

As it can be seen in Fig. 1, the chosen definition of a DT is a set of layers illustrated in pink.

The first layer is the **database** which plays a crucial role in storing and managing the vast amount of data required to create and maintain DTs. DTs can have multiple sources of information, such as information from processing the measured data and information collected previously from various data sources along the life cycle of a product [1]. In this context our database is considered to be constituted of two repositories, the "dataset acquisition" repository that collect dynamic time-series data and the "Vault" repository that preprocess the collected data in the first repository and make the archive of the system. The layer that is positioned above of it uses the database layer to **simulate** and replicate exactly what the PT does [11]. The **visualisation** layer in the DT is an important aspect since it allows users to see a virtual and real-time representation of systems or components [12]. And depending on all these layers, the final layer of the DT is the **decision** layer, helping the human being to make the decision. But to make predictions and make the decision-making more automated, a **cognitive** layer is added, presented in orange, making the DT a Cognitive Digital Twin (CDT). Shifting focus on the contribution at hand, the **data generation** layer colored in Blue. This stratum makes a CDT a CSDT.

Since IoT systems are susceptible to a range of vulnerabilities stemming from both software and hardware failures, which can lead to a loss of control. Any disruption in the operation of an IoT system can, at the very least, cause inconvenience and, at worst, pose a life-threatening risk [10].

Finding unusual data points to train models in an IoT system can be quite challenging. In practical applications, instances of abnormal behavior, such as a sudden and significant increase in temperature in a temperature monitoring system, are quite infrequent when contrasted with the frequency of normal behaviors like maintaining a stable temperature within the desired range [3].

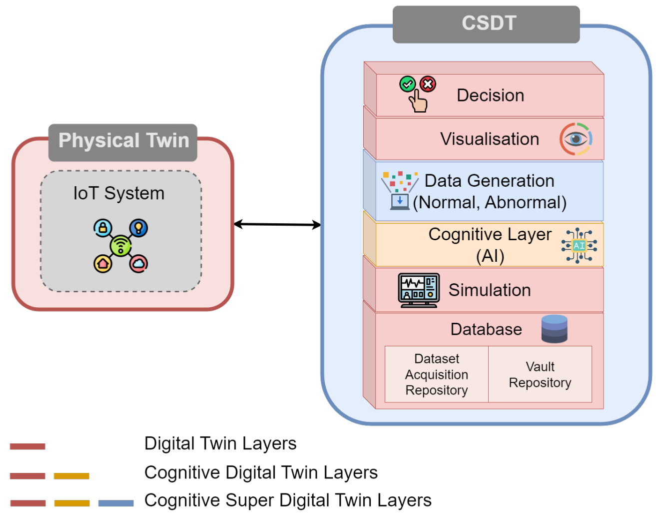


**Fig. 1.** CSDT's Generation Module

Hence, the justification for the generation module is the lack of abnormal data compared to normal ones. This layer fabricates normal and abnormal data, both of those generations are essential to have a balanced global dataset. Therefore, distinguishing those two types of behavior is already a challenge [10]. For instance, a temperature of 10°C is considered normal during winter, but the same temperature in summer might be regarded as a deviation. As mentioned in [3], to commence with, we need to know the nature of collected data streams in our IoT system (binary, continuous, discrete) along with the relationship structure (time-series data, spatial data, graph data). Second, the type of anomaly needs to be identified (point anomaly which is an observation point in the data stream that is significantly distant from the rest of the data, contextual anomaly, an observation point normal in one scenario yet abnormal in another and collective anomaly that is when a sequence of observations are analyzed together to know the normal behavior of a data stream. And any deviation is considered an anomaly). And finally know the availability of training data to allow our model to learn.

### 2.2 SoS DT Architecture Proposal

Now that the communication between a DT and its physical counterpart is understood and some sort of resilience is accomplished thanks to the CSDT, in this section, the introduction of an on-top architecture is proposed. We delineate a clear distinction between SoS DT and a DT SoS, wherein:

– The DT SoS, that can also be called a System of DTs, is the communication between the constituent DTs exclusively. And each constituent DT communicates with its corresponding physical asset (the constituent system) via a designated communication medium.
– The SoS DT represents the actual replica of the SoS where the DT SoS serves as a mechanism for extracting specific tasks aimed at achieving the overarching objectives of the SoS.

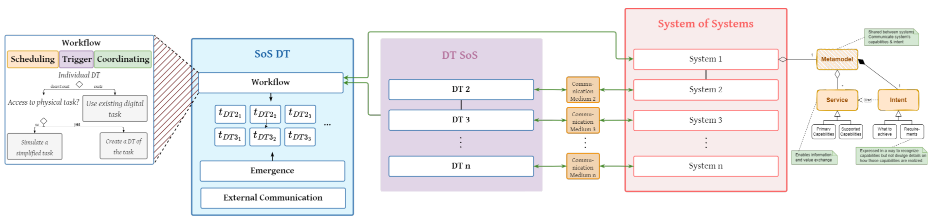


**Fig. 2.** Architecture of a DT for a SoS

Fig. 2 illustrates the interaction between the physical SoS, its digital counterpart (SoS DT) and the DT SoS that facilitates the creation of the SoS DT. The SoS represents the communication among various existing or newly established systems, with this communication being governed by contractual agreements between the involved systems. This agreement is depicted on the right side of the figure wherein [5]:

– A system conforms to a metamodel which is a framework that defines the standard structure and behavior for describing services, intents, and capabilities within a system. It ensures consistency and interoperability across different components and systems in a SoS.
– A service that is shared via the metamodel is a discrete unit of functionality offered by a system that can be consumed by other systems. It is modular and designed to perform specific tasks within the broader system.
– An Intent refers to the desired outcomes or goals that a service aims to achieve. It guides the service in terms of what needs to be accomplished without specifying how it should be done. It ensures as well the alignment with the SoS objectives.
– Primary capabilities are the core tasks that a service provides to achieve its intent.
– Supported Capabilities are the additional functionalities that a service can perform to enhance its primary capabilities.

As observed, a constituent system may or may not possess a DT. In cases where it does possess a DT, a communication medium ensures bidirectional interaction between the two twins. Furthermore, as explained, the interaction among this collection of DTs forms the DT SoS, which facilitates the creation of the SoS DT.

Concerning the SoS DT, the workflow constitutes a central mechanism, receiving inputs from both the physical constituent systems that does not have a DT and from constituent DTs within the DT SoS. The primary objective of this workflow is to identify and extract the essential tasks necessary to achieve the global objectives of the SoS. This is accomplished through a conditional 'if-else' process, outlined as follow:

- If the individual constituent DT exists, the task replica from this DT is utilized within the SoS DT. A task, in this context, is composed of several functions, where each function represents a specific operation or process. This distinction underscores that while a task encompasses multiple functions, each contributing to the overall objective, the integration focuses on leveraging the comprehensive task representation within the SoS DT
- If the individual constituent DT does not exist, the following query is posed: 'Do we have full access to the task of the physical constituent system ?'. If the answer is affirmative, the task replica is created (emergence) and integrated into the SoS DT. If the answer is negative, a simplified version of the task is simulated and incorporated into the SoS DT.

The responses to these queries are predetermined by the agreements established during the integration of the physical constituent system into the SoS. And that way the workflow can manage the tasks by scheduling, triggering and coordinating them. The concept of external communication is illustrated to represent the interactions among multiple SoSs and their respective SoS DTs, but this aspect is beyond the scope of the present paper.

## 3 Conclusion and Future Endeavors

Thus far, two key contributions have been made. Firstly, the creation of a CSDT has been demonstrated through its application to a use case, validating its functionality. Secondly, an architectural proposition for a SoS DT has been introduced, leveraging the DT SoS for task extraction to meet the overarching objectives of the SoS. Future work will concentrate on addressing the identified challenges through comprehensive testing on multiple real-world use cases. This is crucial to uncover potential issues related to implementation, integration, and operation. Such an approach will also enable the assessment of whether the proposed architecture is sufficiently generic to be applicable across diverse use cases, considering that Digital Twins (DTs) are typically driven by specific use-case requirements [4]. Furthermore, subsequent research will investigate external communication between multiple SoSs, with the goal of enhancing interoperability and coordination among different SoS DTs. This will facilitate further refinement of the proposed architecture, ensuring it is robust, scalable, and adaptable to a wide range of applications in the evolving field of DTs and SoSs.

## References

1. Bonney, M.S., de Angelis, M., Dal Borgo, M., Wagg, D.J.: Contextualisation of information in digital twin processes. Mechanical Systems and Signal Processing **184** (2023). https://doi.org/10.1016/j.ymssp.2022.109657
2. Borth, M., Verriet, J., Muller, G.: Digital twin strategies for sos: 4 challenges and 4 architecture setups for digital twins of sos. In: 2019 14th Annual Conference System of Systems Engineering (SoSE). pp. 164–169. IEEE, Anchorage, AK, USA (2019). https://doi.org/10.1109/SYSOSE.2019.8753860
3. Fahim, M., Sillitti, A.: Anomaly detection, analysis and prediction techniques in IoT environment: A systematic literature review. IEEE Access **7**, 81664–81681 (2019). https://doi.org/10.1109/ACCESS.2019.2921912
4. Göllner, D., Rasor, R., Anacker, H., Dumitrescu, R.: Collaborative modeling of interoperable digital twins in a sos context. Procedia CIRP **107**, 1089–1094 (2022). https://doi.org/https://doi.org/10.1016/j.procir.2022.05.113, https://www.sciencedirect.com/science/article/pii/S2212827122003973, leading manufacturing systems transformation – Proceedings of the 55th CIRP Conference on Manufacturing Systems 2022
5. Harbor, Research: System of systems (2024), https://harborresearch.com/system-of-systems/, accessed: 2024-06-09
6. Jakob, A., Thomas, O.: Systems-of-systems and digital twins: A survey and analysis of the current knowledge. In: 2023 18th Annual System of Systems Engineering Conference (SoSe). pp. 1–6. IEEE, Lille, France (2023). https://doi.org/10.1109/SoSE59841.2023.10178527
7. Laszka, A., Abbas, W., Vorobeychik, Y., Koutsoukos, X.: Integrating redundancy, diversity, and hardening to improve security of industrial internet of things. Cyber-Physical Systems **6**(1), 1–32 (2020). https://doi.org/10.1080/23335777.2019.1624620
8. Maier, M.W.: Architecting principles for systems-of-systems. Systems Engineering **1**(4), 267–284 (1998)
9. Michael, J., Pfeiffer, J., Rumpe, B., Wortmann, A.: Integration challenges for digital twin systems-of-systems. In: 2022 IEEE/ACM 10th International Workshop on Software Engineering for Systems-of-Systems and Software Ecosystems (SESoS). pp. 9–12. IEEE, Pittsburgh, PA, USA (2022)
10. Mishra, A., Cohen, A., Reichherzer, T.: Detection of data anomalies at the edge of pervasive iot systems. Computing **103**, 1657–1675 (2021). https://doi.org/10.1007/s00607-021-00927-9
11. Nwogu, C., Lugaresi, G., Anagnostou, A., Matta, A., Taylor, S.J.: Towards a requirement-driven digital twin architecture. Procedia CIRP **107**, 758–763 (2022). https://doi.org/10.1016/j.procir.2022.05.058
12. Zhu, Z., Liu, C., Xu, X.: Visualisation of the digital twin data in manufacturing by using augmented reality. Procedia CIRP **81** (2019). https://doi.org/10.1016/j.procir.2019.03.223