Hierarchical System of Digital Twins: A Holistic Architecture for Swarm System Analysis

Mouhamadou F. Ball[1][a], Jannik Laval[2][b] and Loïc Lagadec[1][c]
[1]*Lab-STICC UMR 6285, ENSTA Bretagne, Brest, France*
[2]*DISP-UR4570, Univ. Lyon, Univ. Lyon 2, Lyon, France*

Keywords: Digital Twin, Swarm System, Modeling, Holistic Architecture.

Abstract: Swarm systems are being increasingly adopted for their operational capabilities and are now assigned more sensitive missions, often in unpredictable environments. Therefore, it is crucial to evaluate their performance in the face of natural or human-induced uncertainties before deployment and enhance their resilience during missions. To enable a comprehensive analysis of this system, a multi-level analysis must be conducted to capture the dynamics at the component, cluster, and swarm levels. Digital Twin (DT) offers a promising solution to address this challenge. While there are existing approaches that use digital twins to analyze complex systems, they do not take into account the specific requirements introduced by swarm configuration. This paper presents a holistic reference architecture, the Hierarchical System of Digital Twins (HSDT), which lays the groundwork for creating digital twins of swarm systems. To support this framework, we introduce the concepts of functional and aggregation hierarchies and propose a goal-oriented method for instantiating DT with a specific level of sophistication. Additionally, we present a metamodel that integrates elements of the Asset Administration Shell (AAS) data model to ensure interoperability with external standards. A prototype of HSDT was developed, and a case study was presented, focusing on analyzing spatial parameters within a swarm of Unmanned Vehicles (UVs).

1 INTRODUCTION

The swarm configuration of systems is an emerging approach, adopted for its numerous advantages, such as enhanced performance through load distribution, easier scalability, and increased robustness (Liu and Passino, 2000). Systems in swarm configuration can be compared to systems of systems (SoS), as they meet the properties presented by (Maier, 1998). This configuration is beginning to be concretely adopted for unmanned vehicles (UVs). For instance, the European Union's SCAF project for the future air force is based on this configuration. Given the increasingly critical missions assigned to this type of system and its evolution in uncertain environments, it becomes essential to analyze it, study its behavior,and implement strategies for performance enhancement. Furthermore given the specifics of these systems, the analysis must be conducted at different levels of abstraction (component, cluster, swarm) and from various perspectives. The Digital Twin (DT) represents a promising technology for addressing the challenges of analyzing such systems. The literature presents various approaches that leverage digital twins for analyzing complex systems (e.g., Cyber Physical System, SoS). However, after defining the requirements for a digital twin in the context of swarm systems, we observe that current architectures do not fully address all of these requirements. This position paper aims to address the research question: **How can we design a digital twin architecture that fulfills the fundamental requirements of swarm systems** ?

In fact, to enable the full use of digital twins in the analysis of swarm systems we need an architecture that accommodates these types of systems. Therefore, we propose an architecture for designing a hierarchical system of digital twins. We start by defining the requirements of such digital twins based on the specifications of swarm systems, then propose a digital twin structure rooted in a service-oriented approach, and finally, the holistic architecture that leverages the use of several digital twins in a hierarchy. Two types of Digital Twins are introduced in this architecture, forming its core: the Instance DT, which is directly connected to reality, and the Aggregate DT, which is a composition of multiple Digital Twins. Furthermore,

[a] https://orcid.org/0009-0005-9245-2892
[b] https://orcid.org/0000-0002-7155-5762
[c] https://orcid.org/0000-0003-3778-3144

Ball, M. F., Laval, J. and Lagadec, L.
Hierarchical System of Digital Twins: A Holistic Architecture for Swarm System Analysis.
DOI: 10.5220/0013258900003896

In *Proceedings of the 13th International Conference on Model-Based Software and Systems Engineering (MODELSWARD 2025)*, pages 280-287
ISBN: 978-989-758-729-0; ISSN: 2184-4348

we introduce a goal-driven method for the instantiation of DTs. An event-based services synchronization is also introduced to connect services between DTs. To ensure interoperability with external technologies and integrate our architecture with current DT standards, we propose a metamodel based on the Asset Administration Shell (AAS) data model (Bader and Maleshkova, 2019). So the architecture does not only meet the requirements of swarm systems but also enables the seamless integration of third-party components, including emerging technologies in industry 4.0.

The paper is structured as follows: Section 2 provides a review of the existing literature on designing digital twins for the analysis of complex systems. Section 3 outlines the key requirements for designing a digital twin system specifically for swarm systems. Section 4 provides an overview of our implementation approach, with Section 5 detailing the implementation and Section 6 presenting a basic case study. Finally, Section 7 concludes the paper.

2 BACKGROUND AND PROBLEMATIC

Recent interest has grown in applying digital twins to complex systems. In our review, we examine several of these applications. The article (Human et al., 2023) presents a six-step design framework that facilitates decision-making in the creation of a system of digital twins with aggregation capabilities. It presents a reference architecture as the foundation for these systems, comprising three modules: an aggregation hierarchy, a service network, and management services. Inspired by SLADTA's work (Redelinghuys et al., 2020), It introduces two types of digital twins: the DTI, representing digital twins directly linked to the physical, and the DTA, which aggregates data from multiple DTs. The data-based aggregation proposed in the paper does not provide a reliable means to capture emerging behaviors in the DTA, as it relies solely on data. In (Reiche et al., 2021) the integration of multiple digital twins and the management of their network are examined. This study proposes an approach to merge multiple digital twins under the governance of a single point of truth called the Digital Twin of a System (DTS). It presents a vision of the digital twin as a composition of Digital Models (DM) and an aggregation of Digital Shadow (DS). However, the connection possibilities through DTS do not include direct communications between what they call subordinate digital twins, which is crucial in swarm systems. Additionally, the combinations are only made with DTs of the same level. The article (Moyne et al., 2020) presents the concept of digital twins by adopting an object-oriented approach. It envisions the digital twin as consisting of models with a computational engine, connected through an association with a cardinality of (1,*). They introduce the notion of classes of digital twins and, using object-oriented concepts such as generalization hierarchy, inheritance, and aggregation, they define possible combinations between different digital twins. This approach, based on object-oriented concepts, is quite original and interesting. However, since multiple inheritance is not supported by default in the object-oriented concept, it is impossible to have an entity that allows for the abstract representation of a digital twin resulting from the aggregation of lower-level digital twins from different classes. Thus with the proposed framework, aggregation can only occur with digital twins of the same type and level. The articles (Zhou et al., 2023b; Zhou et al., 2023a) discuss the implementation of a digital twin network for satellite communication. They propose a hierarchical architecture, but it is limited to two levels: the edge-DT and the central-DT. This restricted hierarchy does not accurately reflect the configuration of swarm systems, which can include multiple levels of hierarchy.

So far, the review indicates that while there are existing architectures and methodologies for the use of digital twins in complex systems, they generally lack certain specifications and features necessary for application in the context of swarm systems analysis.

3 REQUIREMENTS OF DIGITAL TWIN ARCHITECTURE FOR SWARM SYSTEMS

In this section, we will present the specific characteristics and properties of swarm systems, along with an example. Additionally, we will outline the design requirements imposed by such systems for developing a digital twin.

3.1 Specifications of Swarm Systems

Swarm system represent a paradigm in how autonomous entities collaborate to achieve complex tasks. Unlike traditional single-agent systems, swarms operate as decentralized networks where individual components, often equipped with sensors, actuators, and communication capabilities, interact locally to accomplish shared objectives. The following figure 1 illustrates a simple configuration of a swarm of 2 thrusters surface drones during an escort mission. Swarm systems have a unique configuration with significant spatial and temporal dynamics, encompassing some properties developed by (Maier, 1998) for a system of systems (SoS). Below, we outline five key characteristics of swarm systems that will guide our vision for the digital twins of swarm systems.

- Managerial Independence: The components are separately acquired and integrated but maintain a continuing operational existence independent of the swarm;
- Operational Independence: If the swarm is disassembled into its components the components must be able to usefully operate independently;
- Hierarchy: In the swarm, the concept of responsibility suggests that the components might function at different levels and offer varying services;
- Geographic Distribution: The components of swarm systems are often distributed within a given space and exhibit a high spatiotemporal dynamic;
- Components Interaction: The components of a swarm are subject to intense interactions, enabling the emergence of collective behaviors.

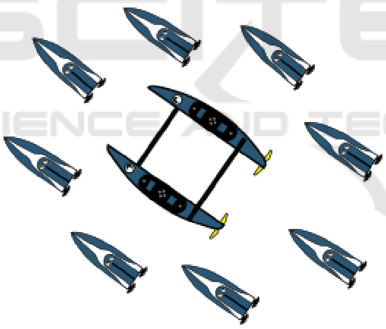

Figure 1: A formation of surface marine drones performing escort mission.

3.2 Requirements Definition

As presented by (Human et al., 2023) in their design process of DT, a key step is the needs analysis, which results in the creation of a list of requirements. In their paper (Michael et al., 2022) authors list several challenges related to the integration of digital twins in a SoS context. These systems are similar to swarm configurations, as, highlighted in the introduction, the properties described in SoS are also present in swarm configurations. They identify 15 key points to consider for the integration of digital twins into a SoS. We have identified and included key points essential to our objectives, along with additional aspects based on the swarm specifications outlined in the previous section. Relevant requirements definition ensure that our architecture will be able to achieve the expected abstraction. The following list outlines the key requirements, emphasizing the design considerations our system must fulfill as well as addressing specific DT user concerns.

- Synchronization: The system must ensure data synchronization between physical entities and their corresponding digital twins, as well as between the digital twins deployed service;
- Hierarchical Structure: The different levels of the physical world imply different levels of abstraction of functionality within the corresponding DTs;
- Aggregation: The system should aggregate information from lower-level digital twins to provide a digital twin that offers a comprehensive view at higher levels for advanced analysis;
- Interoperability: Achieving both syntactic and semantic interoperability is essential for the digital twins within this architecture and following the existing standards;
- HMI (Human Machine Interface): Given the different levels of granularity, advanced views will be necessary, allowing the user to perform either a deep exploration of a component or the abstraction of a group of components.

4 HIERARCHICAL SYSTEM OF DIGITAL TWINS (HSDT)

Based on the defined requirements outlined above, this section presents the proposed architecture and vision for digital twins specifically tailored to the analysis of swarm systems. Several definitions of Digital Twins (DT) have been proposed (LaGrange, 2019; Glaessgen and Stargel, 2012; Singh et al., 2021), with most being predominantly domain-specific. Among them, the definition provided by the AIF committee stands out with its comprehensive approach, describing a DT as *an organized set of digital models representing a real-world entity to address specific problems and use cases.* While this definition is particularly interesting, we believe it would be valuable to incorporate the nature of the services rendered and to clearly specify that the reality being represented is not limited to the physical domain. Therefore, from a service-oriented perspective, we define the digital twin as a *virtual system that represents a reality whether physical or not and connected to it through* *a data flow; capable of delivering a range of services, including simulation, debugging, formal verification, monitoring, and system control, by leveraging dynamic numerical models*.

Regarding the composition of a digital twin, various proposals are presented in the literature. We will focus specifically on the proposal of (Kruger et al., 2021), which the author presents in a hybrid service-oriented approach where services can be hosted either within the digital twin or in a Service-Oriented Architecture (SOA). The services requiring data from the physical system are provided by the digital twin, while other services that need different types of data are delivered through the SOA. This approach aligns with our needs, as it allows for a level of functional abstraction, which is one of the requirements outlined in the previous section. According to the work of (Human et al., 2023), which partially builds on the research of (Kruger et al., 2021), the digital twin is considered to consist of three layers: a model layer, a services layer, and a data layer. Real-time bidirectional communication is a crucial element in the life cycle of digital twins, as presented in (Vaezi et al., 2022). Therefore, we will also align with the work (Chen et al., 2024) by integrating an interface layer. This layer will define the interaction between the represented reality and the digital twin and between the services provided by the DTs in the global architecture.

Figure 2 illustrates the structure of DT and its physical system (PS). Our digital twin structure incorporates a control data flow originating from the digital twin, unlike (Human et al., 2023) proposal, which does not account for this level of sophistication in the workflow of DT.

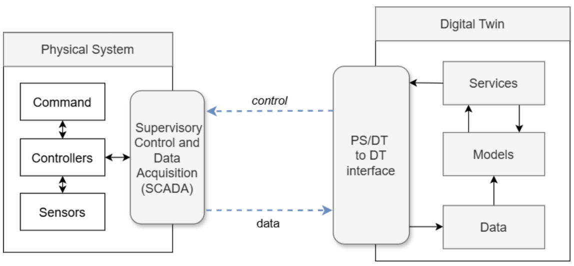

Figure 2: The structure of a digital twin and its physical counterpart.

Based on the proposed DT structure above, we define the HSDT architecture with the purpose of enhancing the development and deployment of multiple digital twins in hierarchical configuration for analyzing swarm system. The requirements outlined in Section 3 serve as foundation for the development of this architecture. Two types of Digital Twins are introduced in this architecture, forming its core: the **Instance** DT, which is directly connected to reality, and the **Aggregate** DT, which is a composition of multiple Digital Twins. Figure 3 presents the overview of the hierarchical system of digital twins. In the following subsections, we detail the key aspects introduced by the architecture.

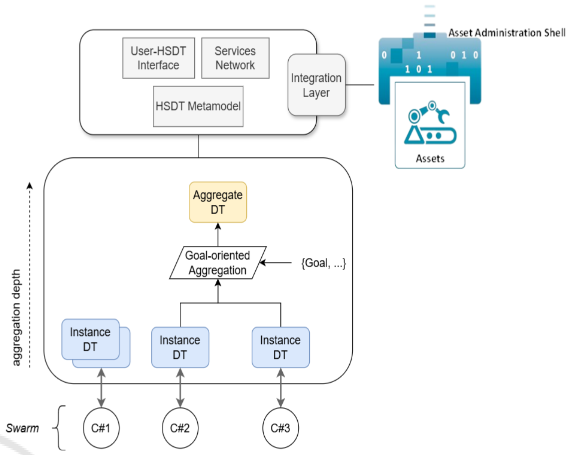

Figure 3: Holistic reference architecture of the hierarchical system of digital twins.

4.1 HSDT Metamodel and Interoperability

We define a metamodel (see Figure 4) based on the Meta-Object Facility (MOF) standard. The metamodel is designed to represent the domain structure of the hierarchical digital twin (DT) system and to ensure interoperability. The two defined types of DTs are structured using a composite design pattern. Each type of DT is associated with one or more goals, such as monitoring, formal verification, simulation, or debugging. However, the debugging and control capabilities of a DT are specialized for the Instance DT, as it is the only type of DT directly connected to reality. Each DT includes embedded models utilized by various services. The Aggregate DT class, in contrast, is not associated with an atomic Asset because it is generated through the composition of DTs.

The notion of a goal is introduced to guide the appropriate instantiation and level of sophistication of DTs. Depending on the goal, certain elements of the DTs may not be utilized. For example, for monitoring purposes, only the data layer may be required, as no computation is involved.

Interoperability within the HSDT architecture refers to the ability of its entities to exchange data (syntactic interoperability) and understand each other (semantic interoperability). This property is critical, as a DT may provide or depend on a service offered by another DT, necessitating seamless data exchange. The proposed metamodel supports interoperability not only within the architecture but also semantic interoperability with the AAS (Asset Administration Shell) standard, a cornerstone of Industry 4.0. To align with the AAS standard, we incorporate two key elements into the AAS data model (Bader and Maleshkova, 2019): **Asset** and **Submodel**, as highlighted in yellow in Figure 4.

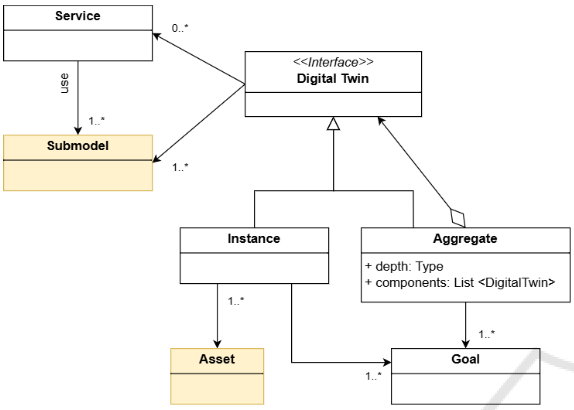

Figure 4: The metamodel of the Hierarchical System of Digital Twin coupled with elements of AAS data model.

4.2 Instance/Aggregate DT

As previously explained, the Instance DT is directly connected to reality and possesses specific capabilities such as debugging and control of the represented reality, in addition to analytical functions. However, achieving higher-level analysis requires a DT operating at a level above the Instance DT. To address this need, we introduced the Aggregate DT (ADT) and a method for its instantiation. The ADT enhances the HSDT by enabling analysis at various levels of system granularity. This aggregation of Digital Twins is, therefore, a key feature for the advanced analytical capabilities within the HSDT architecture. (Redelinghuys et al., 2020) introduced the concept of Digital Twin Aggregation (DTA) for combining digital twins, but their approach relies exclusively on data from other digital twins. In contrast, we believe that aggregating digital twins should be driven by a defined analysis goals, allowing for a more selective choice of elements involved in the aggregation. Therefore, in the goal-oriented method presented here, ADT is created based on a specific analysis problem (e.g., visualization, prediction, anomaly detection), which helps determine the appropriate goal. This approach considers both data and/or models, depending on the selected goal. The interest in introducing model aggregation lies in the ability to capture, beyond the data, the paradigm of each digital twin composing the new twin. The model's aggregation could be achieved using model federation techniques, such as those presented in (Guérin, 2023) or for more formal model by using composition operation such as parallel composition for automata. The goal-based aggregation strategy enables the right choice of the sophistication level of the new DT. For instance, if the aggregation aims to provide an overview of a cluster's metrics, it is not necessary to include model aggregation; data from low-level DTs alone is sufficient. The synopsis in Figure 5 illustrates the steps for instantiating a Digital Twin, in the case of Instance DT and Aggregate DT.

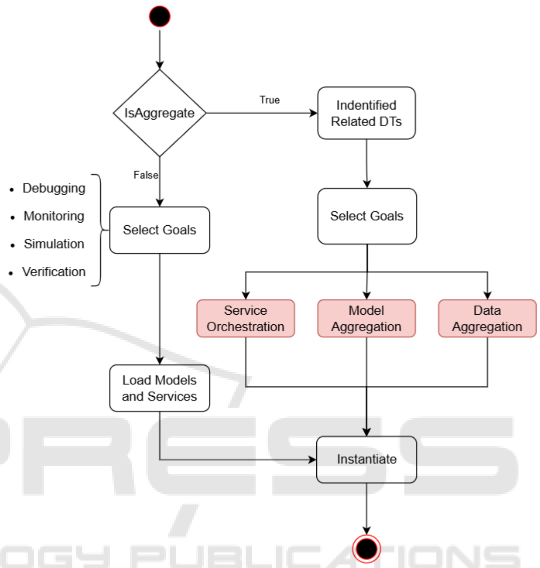

Figure 5: The workflow of DT instantiation based on the goal-driven method.

4.3 Hierarchical Structure

Given the hierarchy of responsibilities within swarm systems, the digital twin system must reflect this organization. Therefore, with HSDT, we present two types of hierarchies: a functional hierarchy and an aggregation hierarchy.

Functional Hierarchy. The digital twins may differ in the range or complexity of services they provide. The functional hierarchy is based on the capabilities of each digital twin. For example, in the HSDT, some digital twins might have more specialized functions or greater capabilities than others. So, in this hierarchy, each digital twin's level is determined by the number of services it offers. Essentially, DTs offering more or critical services are considered higher in the hierarchy, reflecting their greater functional role within the swarm.

Aggregation Hierarchy. The aggregation hierarchy is distinct and is specifically designed for Aggregate DTs. This type of hierarchy considers the structural relationships among digital twins that are composed of multiple components. An important concept here is **aggregation depth**, which corresponds to the level within the hierarchy. Digital twins that are directly connected to physical systems or real-world data have an aggregation depth of 0. When these individual twins are combined to form an aggregate DT, the aggregation depth of the composite twin is defined as one level higher than the maximum depth of its components. This concept of aggregation depth provides a standardized way to understand and organize complex digital twins built from multiple layers of other twins. Equation 1 defined the way to calculate the aggregation depth of an ADT composed of *n* instances of DT and/or ADT. The process of ADT creation will be explained in the following subsection. This method enables the HSDT to organize and navigate complex structures efficiently, helping to distinguish between different levels of aggregation in the swarm system.

$$d_{\text{ADT}} = \max(d_1, d_2, \ldots, d_n) + 1 \tag{1}$$

4.4 Synchronization: PS to DT and DT to DT

The synchronization is essential to ensure that decisions made based on the digital twin are reliable and relevant. A failure in synchronization can lead to inconsistencies, errors in data analysis, and inappropriate actions. As highlighted by the authors in (Coviello et al., 2020), who emphasize the importance of a synchronization mechanism in a multi-sensor system, the digital twin directly connected to a physical system has similar requirements. Indeed, it can be fed by various data sources from the different sensors of the system entity it represents. Synchronization must occur in real-time, and for the twin to remain up-to-date, the latency with the data generated by the physical system must be low. In our context, where the components of the swarm system are considered to be equipped with sensors and the data is accessible, synchronization involves establishing communication capable of transmitting this data with minimal latency. Certain protocols enable fast and reliable message handling between two application entities. The authors (Uy and Nam, 2019) compare two widely used protocols in the context of IoT and M2M communication: Message Queuing Telemetry Transport (MQTT) and Advanced Message Queuing Protocol (AMQP). The conclusions of their study show that AMQP is the better choice when designing a link with a continuous data stream. Therefore, in the architecture, we recommend using the Advanced Message Queuing Protocol (AMQP). Implementing this protocol will ensure a low-latency level in data exchanges.

In addition to synchronizing the digital twin with the reality it represents, it is essential to ensure synchronization between the digital twins, especially between the services provided by each DT. This is essential to ensure effective and coordinated communication between the various services. To synchronize these services, several solutions are available to us, including the use of an orchestrator or an event-based coordination. In the HSDT architecture we use an event-based synchronization. This later involves services communicating through a service bus. Services act as both event consumers and/or event producers. Figure 6 illustrates the synchronization between DTs services; *service_y* of *DT_j* can publish an event on Topic 2, while *service_x* of *DT_i* subscribes to this event.

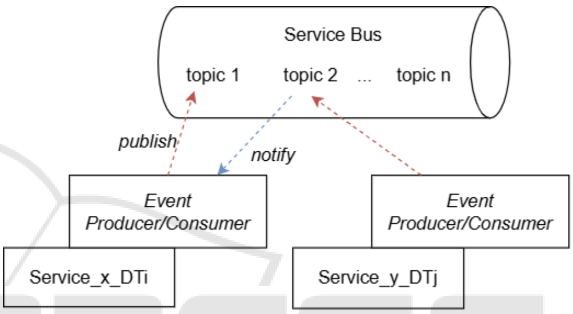

Figure 6: Service synchronization in HSDT, an Event-Based approach.

5 HSDT PLATFORM PROTOTYPE

We propose to implement our architecture using Pharo, a purely object-oriented programming language and a powerful environment focused on simplicity and immediate feedback. The HSDT platform called *DTwinSync* will be deployed within instances of the Pharo 12 Image. The platform offers all the tools and APIs needed to develop and interact with an HSDT architecture. This allows us to create digital twins, associate them with physical components. This implementation offers several advantages, such as ease of deployment, with all dependencies included in the image, as well as re-usability. In this implementation, we will use multi-threading concepts to meet real-time constraints. Specifically, communication-related functions will be executed in separate threads. The communication between the digital twin inside the Pharo image and the physical system is done using the MQTT protocol (for seek of simplicity). The development platform's source code is available in a public repository[1].

[1] https://github.com/Cracen26/DTwinSync-beta

6 CASE STUDY: UVs TARGET SEARCH

In this section, we present a simple case study to demonstrate the ability of our approach to analyze in different levels of granularity in a swarm of UVs. The case study is as follows: marine UVs are deployed over a body of water and organized into a swarm formation, divided into two groups, the identifiers and the actuators. The identifiers are equipped with devices to detect a target in a specific location, while the actuators are responsible for carrying out the necessary action. The success of the mission depends on the cooperation between the UVs from both groups, with each fulfilling its designated tasks. The target is located within the water body, and its position is known. The DTs directly linked to the UVs will be fed with data such as position and status. Due to difficulties in accessing physical infrastructure, we will use a Python program to simulate the physical swarm. The objective of the analysis in this case study is to have an idea of the spatial distribution of our swarm at a specific point in time.

Instance DT

The following code snippet shows the script in Pharo that declare the instance DT and link them to their physical counterparts. Once this instantiation is complete, the data flow shared by the physical system will be available in real time on the digital twin (DT).

```
dt := OrderedCollection new.
0 to: 3 do: [ :i |
        dt add: (DTwin new start:
        (PhysicalCounterpart new id: i))
        ↪  ].
```

Aggregate DT

The goal now is to analyze a very simple parameter: entropy, which can provide information about the geographic distribution of our robots. These metrics cannot be calculated by the local digital twins of the UVs, highlighting the need for high-level analysis. Subsequently, we will instantiate different digital twins and use aggregation to create ADT that will allow us to calculate the entropy of our group of marine UVs.

1. Related DTs: The digital twins of the 4 UVs
2. Goals: Monitoring (Analyze the entropy)
3. Data Aggregation: Location attribute of each UV, thus the data layer of the affiliated DTs

The following code snippet shows the syntax for instantiating an ADT. The model that will be loaded onto the ADT is *EntropyModel*. The principle of this method is straightforward: it calculates entropy by dividing the area into cells and counting the number of points in each cell. Next, it computes the probabilities of having a point in each specific cell. Finally, the total entropy is calculated using the equation 2. To achieve this, after instantiating the ADT, we loaded the *EntropyModel*, which calculates the entropy, along with the *EntropyVisualisation* service, which displays a plot of the entropy evolution. The figure illustrates this evolution.

```
ADT := DTwinHL new.
ADT aggregationType
    add: (AggType dataAggregation).
ADT aggregate: dt;
    load: EntropyModel;
    load: EntropyVisualisation.
```

$$H(X) = -\sum_{i=1}^{n} p_i \log_b(p_i) \quad (2)$$

In the following figure 7, after running the computation in the ADT, the model is executed, and the results are passed to the visualizer, which then plots them. The closer the entropy gets to zero, the more concentrated the points become. Thus, as the mission progresses, the expected behavior is observed: the UVs move closer to the target point and become less dispersed.

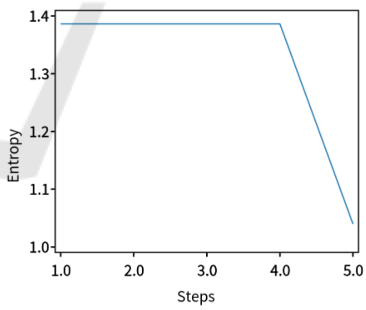

Figure 7: Plot showing the Entropy of the UVs swarm through ADT.

7 CONCLUSION

In this article, we present our vision to design a hierarchical system of digital twins, aimed at analyzing systems in a swarm configuration. After defining the requirements we proposed a holistic architecture that emphasizes organizing DTs within a hierarchical configuration. This architecture is holistic because it considers all levels of granularity within the swarm and accounts for the relationships between them. We introduce a new method called goal-driven to instantiate DT. Furthermore, the proposed metamodel is built upon the standard AAS, aiming to enhance the openness of our architecture to Industry 4.0. Finally, we present a simple case study to apply the theoretical concepts of HSDT, particularly by conducting an analysis through the creation of an Aggregate DT. However, this case study does not fully validate all elements of the architecture and should be further developed to encompass all the aspects discussed in this paper. Some areas require further development, such as the reconfiguration loop between the digital twin and the represented reality. Established engineering approaches, like MAPE-K (Arcaini et al., 2015), can be leveraged to design this connection.

REFERENCES

Arcaini, P., Riccobene, E., and Scandurra, P. (2015). Modeling and analyzing mape-k feedback loops for self-adaptation. In *2015 IEEE/ACM 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 13–23. IEEE.

Bader, S. R. and Maleshkova, M. (2019). The semantic asset administration shell. In *Semantic Systems. The Power of AI and Knowledge Graphs: 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9–12, 2019, Proceedings 15*, pages 159–174. Springer.

Chen, Z., Surendraarcharyagie, K., Granland, K., Chen, C., Xu, X., Xiong, Y., Davies, C., and Tang, Y. (2024). Service oriented digital twin for additive manufacturing process. *Journal of Manufacturing Systems*, 74:762–776.

Coviello, G., Avitabile, G., and Florio, A. (2020). The importance of data synchronization in multiboard acquisition systems. In *2020 IEEE 20th Mediterranean Electrotechnical Conference (MELECON)*, pages 293–297. IEEE.

Glaessgen, E. and Stargel, D. (2012). The digital twin paradigm for future nasa and us air force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*, page 1818.

Guérin, S. (2023). *FML: un langage de fédération de modèles pour l'interopérabilité sémantique de sources d'information hétérogènes*. PhD thesis, ENSTA Bretagne-École nationale supérieure de techniques avancées Bretagne.

Human, C., Basson, A., and Kruger, K. (2023). A design framework for a system of digital twins and services. *Computers in Industry*, 144:103796.

Kruger, K., Human, C., and Basson, A. (2021). Towards the integration of digital twins and service-oriented architectures. In *International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing*, pages 131–143. Springer.

LaGrange, E. (2019). Developing a digital twin: The roadmap for oil and gas optimization. In *SPE Offshore Europe Conference and Exhibition*, page D011S003R001. SPE.

Liu, Y. and Passino, K. M. (2000). Swarm intelligence: Literature overview. *Department of electrical engineering, the Ohio State University*.

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.

Michael, J., Pfeiffer, J., Rumpe, B., and Wortmann, A. (2022). Integration challenges for digital twin systems-of-systems. In *Proceedings of the 10th IEEE/ACM International Workshop on Software Engineering for Systems-of-Systems and Software Ecosystems*, pages 9–12.

Moyne, J., Qamsane, Y., Balta, E. C., Kovalenko, I., Faris, J., Barton, K., and Tilbury, D. M. (2020). A requirements driven digital twin framework: Specification and opportunities. *Ieee Access*, 8:107781–107801.

Redelinghuys, A., Kruger, K., and Basson, A. (2020). A six-layer architecture for digital twins with aggregation. In *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future: Proceedings of SOHOMA 2019 9*, pages 171–182. Springer.

Reiche, L.-T., Gundlach, C. S., Mewes, G. F., and Fay, A. (2021). The digital twin of a system: A structure for networks of digital twins. In *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*, pages 1–8. IEEE.

Singh, M., Fuenmayor, E., Hinchy, E. P., Qiao, Y., Murray, N., and Devine, D. (2021). Digital twin: Origin to future. *Applied System Innovation*, 4(2):36.

Uy, N. Q. and Nam, V. H. (2019). A comparison of amqp and mqtt protocols for internet of things. In *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 292–297. IEEE.

Vaezi, M., Noroozi, K., Todd, T. D., Zhao, D., Karakostas, G., Wu, H., and Shen, X. (2022). Digital twins from a networking perspective. *IEEE Internet of Things Journal*, 9(23):23525–23544.

Zhou, L., Leng, S., and Wang, Q. (2023a). A federated digital twin framework for uavs-based mobile scenarios. *IEEE Transactions on Mobile Computing*.

Zhou, Y., Zhang, R., Liu, J., Huang, T., Tang, Q., and Yu, F. R. (2023b). A hierarchical digital twin network for satellite communication networks. *IEEE Communications Magazine*.