BUILD-IT2023 Workshop
19-20 October, Rome, Italy

BUILDING AN ECOSYSTEM OF INTEROPERABLE DIGITAL TWINS: CHALLENGES & EXPERIENCES

Marco Picone, Nicola Bicocchi, *and* Marco Mamei
University of Modena & Reggio Emilia, Italy
email: {marco.picone, nicola.bicocchi, marco.mamei}@unimore.it

Mattia Fogli *and* Carlo Giannelli
University of Ferrara, Italy
email: {mattia.fogli, carlo.giannelli}@unife.it

Paolo Bellavista
University of Bologna, Italy
email: paolo.bellavista@unibo.it

Digital Twins (DTs) are software replicas of physical entities that hold great potential for cyber-physical applications. However, existing solutions often focused on centralized and Cloud-centric approaches without interoperable solutions and operating in isolated silos. This works aims to explore the challenges and opportunities in establishing an interoperable ecosystem of connected DTs by sharing research, development, and deployment experiences. The objective is to enable DTs to actively engage and collaborate across distributed environments through a flexible and dynamic deployment of pervasive DTs spanning from edge to cloud to meet the diverse applications' requirements. By embracing a shared vision and promoting interoperable methodologies and investigating standardization opportunities, we foster the creation of an open ecosystem that empowers the development of the next generation of intelligent cyber-physical applications through multiple application domains.

Keywords: digital twins, pervasive twins, software, edge-cloud continuum, standardization

1. Introduction

Digital Twins (DTs) are undergoing a transformation and emerging as a cross-domain paradigm for designing and implementing cyber-physical applications. They create synchronized software replicas associated with physical devices, products, and organizations, and are denoted as Physical Twins (PTs). The adoption of DTs extends beyond industrial and manufacturing contexts, bridging the physical and digital realms with the help of the Internet of Things (IoT) and Web technologies, enabling their applicability in diverse scenarios. The evolution of DTs brings new challenges that call for new software architectures, development, and deployment approaches. Existing solutions often treat DTs as passive components, lacking interoperable modeling and relying on platform-specific silos. This independent model-building approach without standardized methodologies poses the risk of vendor lock-in and hampers the creation of an open ecosystem for seamless collaboration among PTs, DTs, and applications. To unlock the full potential of DTs, a shared and interoperable vision is necessary, empowering DTs to function as active entities with their own behaviors. This vision encourages seamless collaboration with applications and services across distributed environments through an edge-cloud computational continuum, as introduced in [1, 2].

The primary objective of this contribution is to explore existing challenges, share research experiences, and identify innovation opportunities in creating an interoperable ecosystem of connected DTs. This interdisciplinary

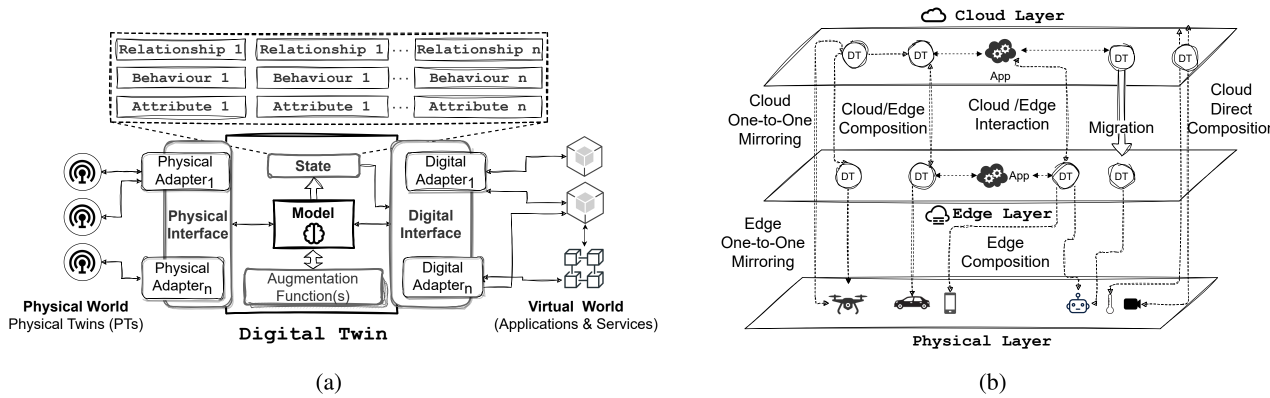

Figure 1: (a) A schematic representation of the Software blueprint architecture of a Digital Twin; (b) A hierarchical and multi-layer overview of multiple deployed twins across an edge-cloud continuum.

vision aims to facilitate the design, deployment, and maintenance of the next generation of intelligent cyber-physical applications. The focus is on pervasive and distributed DTs that can be flexibly and dynamically deployed across multiple network levels, ranging from edge to cloud, to meet specific needs and application requirements.

2. Distributed Digital Twins Modeling

In the scientific literature [3], DTs have been recently studied to propose a unified conceptual framework applicable across various domains. In [4], we relied on these abstract capabilities extending their characterization to the industrial ecosystem with a specific focus on Software requirements and how DT-driven application design, implementation, and deployment. A DT is represented schematically as shown in Figure 1(a) and modeled as an independent and autonomous architectural component. Its *Model* component captures attributes, behaviors, and relationships of the physical counterpart accurately. The DT's *State* component structures and stores computed attributes and relationships, facilitating interaction with the physical layer. A *Physical Interface* with multiple *Physical Adapters* enables communication with connected devices bridging the cyber-physical gap while on the other hand, DTs interact with the digital layer through a *Digital Interface* composed of *Digital Adapters* to create an interoperable and standardized interaction point in terms of protocols, and data formats to enable effective collaboration in the digital space with applications, services, and other DTs.

In the next generation of DT-based architectures, scalability is crucial to meet diverse requirements and support a wide range of application scenarios. Deploying DTs across Edge, Fog, and Cloud facilities offers flexibility as illustrated in Figure1(b). Edge DTs (EDTs) (introduced in [2]) enable real-time data processing, immediate response, and localized insights. Fog-based DTs balance local processing and Cloud connectivity, enabling complex analytics and collaborative decision-making. Cloud-based DTs provide scalability, computational power, and centralized control for handling massive data volumes and resource-intensive computing. The scalability of DTs allows organizations to choose the deployment strategy based on specific requirements, considering factors such as data sensitivity and network connectivity. A multi-tiered edge-to-cloud architecture empowers DTs to address localized to global-scale use cases, fostering innovation and adaptability in the dynamic landscape of cyber-physical systems.

3. A Modular Approach for IoT Digital Twin Development

The absence of standards and common agreements for DT design and development has led to the proliferation of platform-specific solutions, causing fragmentation and hindering the potential of DTs [5]. Open-source initiatives like the Eclipse IoT Foundation and Eclipse Ditto [1] project have emerged to address this issue by providing open

[1] Eclipse Ditto - `https://www.eclipse.org/ditto/`

platforms for DT management. However, these frameworks have limitations in terms of flexibility and modularity, as they follow a monolithic and centralized vision where all DTs are co-located in the same architectural component. To overcome these limitations, we introduced in [6] the *White Label Digital Twin* (WLDT) an open-source ([2]) Java library designed for maximum modularity and flexibility. WLDT simplifies DT design and development by offering core features and functionalities. It employs a multi-threading core engine, enabling independent components to shape the behavior of each DT. Built-in IoT features and modules facilitate the digitalization of smart objects using MQTT and CoAP. The library's internal software processing pipeline allows dynamic customization of data management, tailored to specific use cases. An internal caching system enhances performance and reduces communication response time. WLDT follows the design principle of modeling each DT as an independent and autonomous software component, facilitating microservices-oriented IoT architectures. It serves as a foundational building block for DT-driven applications targeting the IoT ecosystems through built-in MQTT and CoAP workers, or the definition of new modules and connectors to support specific protocols or data flows.

4. Interoperability & Standardization Opportunities

Standardizing the definition, responsibilities and description of DTs is vital for successful software implementation, enabling seamless communication and data exchange. A standardized definition should encompass properties, relationships, models, operations, protocols, and events, transcending application domains. Standardization organizations and consortia, focusing on IoT, IIoT, networking, and edge/cloud architectures, are actively shaping interoperable environments [7]. Efforts from the Digital Twin Consortium, Digital Twin Programme, and companies like Microsoft with their Digital Twin Definition Language (DTDL)[3] contribute to the standardization landscape[4]. However, the definition of DTs is in its early stages, with differences and a tendency to treat DTs as passive components. The recent exploration of DTs as active interoperable entities within the IoT ecosystem is being investigated by organizations like the European Telecommunications Standards Institute (ETSI) with a Specialist Task Force (STF) [5] with the aim to analyze DTs standardization opportunities within the IoT application domain and specifically within the SmartM2M ecosystem. The objective of this work is to address the missing elements in modeling and standardizing the communication concept of IoT DTs and their blueprint communication reference architecture. The focus is on identifying use cases and deployments where they can be effectively adopted, in order to determine the requirements and specifications associated with their functionalities. The aim is to establish horizontal cross-domain interoperability and standards, defining minimum requirements for the usability of professional and general public IoT services, regardless of their criticality. The work will map IoT Digital Twins within the oneM2M framework and contribute to ISO/JTC1/SC41 by defining new specifications that embrace new functionalities and effectively utilize existing features such as discoverability, security, and modularity.

REFERENCES

1. Ricci, A., Croatti, A., Mariani, S., Montagna, S. and Picone, M. Web of digital twins, *ACM Trans. Internet Technol.*, **22** (4), (2022).

2. Picone, M., Mamei, M. and Zambonelli, F. A flexible and modular architecture for edge digital twin: Implementation and evaluation, *ACM Trans. Internet Things*, **4** (1), (2023).

3. Minerva, R., Lee, G. M. and Crespi, N. Digital twin in the iot context: a survey on technical features, scenarios, and architectural models, *Proceedings of the IEEE*, **108** (10), 1785–1824, (2020).

[2]WLDT on GitHub: `https://github.com/wldt`
[3]Microsoft Azure DTDL: `https://github.com/Azure/opendigitaltwins-dtdl`
[4]Digital Twin Consortium: `https://www.digitaltwinconsortium.org/index.htm`
[5]ETSI STF 628 Digital Twins: `https://portal.etsi.org/XTFs/#/xTF/628`

4. Bellavista, P., Bicocchi, N., Fogli, M., Giannelli, C., Mamei, M. and Picone, M. Requirements and design patterns for adaptive, autonomous, and context-aware digital twins in industry 4.0 digital factories, *Computers in Industry*, **149**, 103918, (2023).

5. Tao, F. and Qi, Q. Make more digital twins, *Nature*, **573** (7775), 490–491, (2019).

6. Picone, M., Mamei, M. and Zambonelli, F. Wldt: A general purpose library to build iot digital twins, *SoftwareX*, **13**, 100661, (2021).

7. Zhou, C., Yang, H., Duan, X., Lopez, D., Pastor, A., Wu, Q., Boucadair, M. and Jacquenet, C. Internet Engineering Task Force, Digital Twin Network: Concepts and Reference Architecture. Work in Progress, (2022).