Achieving End-to-End Cyber-Physical Trustworthiness Through Digital Twins

Nicola Bicocchi*, Mattia Fogli†, Carlo Giannelli†, Marco Picone*, Antonio Virdis‡
* *University of Modena and Reggio Emilia, Italy*; † *University of Ferrara, Italy*; ‡ *University of Pisa, Italy*;

***Abstract*—The wide adoption of IoT pushes for a tight integration between novel digital applications and the plethora of heterogeneous physical objects they interact with. In this regard, Digital Twins (DTs) have emerged as a suitable paradigm to create a bridge between the virtual and physical worlds, hiding the complexity of their interactions. As it stands, however, digital applications have to blindly trust not only DTs but also the way they are managed. To fill this gap, this work introduces the concept of end-to-end cyber-physical trustworthiness, which also has implications on how DTs are both designed and managed. In particular, we identify five key pillars to describe trustworthiness and we devise a blueprint architecture to guide the design of new DTs and platforms that natively focus on trustworthiness.**

***Index Terms*—Cyber-Physical Systems, Digital Twins, Trustworthiness**

I. Introduction

The pervasive presence of the Internet of Things (IoT) has enabled a novel realm of application scenarios, in particula within smart-cities, ranging from smart-vehicles to structural health monitoring, typically characterized by an underlying plethora of heterogeneous Physical Objects (POs) jointly at work. In this context, Digital Twins (DTs) have emerged as a suitable paradigm for bridging the virtual and physical worlds.

In this work, we use the term DT to refer to any software rendering a PO in the virtual world, regardless of how it communicates with its physical counterpart. Note that not only can DTs render POs in the virtual world, but they can also augment the capabilities of POs while exposing them to users and applications, which, in turn, rely on DTs for interacting with the physical world. Therefore, DTs play a crucial role throughout a cyber-physical interaction. On the one hand, they hide the complexity of the underlying POs while possibly augmenting them for applications and users. On the other hand, they accurately render the state and model of underlying POs. As it stands, however, applications and users can only blindly trust DTs throughout a cyber-physical interaction. In fact, to the best of our knowledge, the concept of end-to-end cyber-physical trustworthiness has never been discussed in the literature, lacking a practical framework to deal with the problem of trustworthiness. Note that even in the ideal case of a DT designed to perfectly render its physical counterpart, some applications or users might not perceive it as trustworthy. For example, an application might not trust a DT that runs in a public cloud—a domain that might not be perceived as trustworthy. A solution might be to migrate the DT on-premises, assuming the availability of a cloud-to-edge continuum infrastructure. This example clearly shows that having a trustworthy DT is not enough. In fact, not only trustworthiness depends on the design of the DT, but it also requires a suitable platform for managing the DT.

To fill this gap, this work originally introduces the concept of end-to-end cyber-physical trustworthiness to evaluate how DTs are both designed and managed. To do so, we propose a conceptual framework based on five pillars referring to measurable features which, we believe, are the basis for cyber-physical trustworthiness. Specifically, the **representation** pillar accounts for the accuracy of both the DT internal model and the data coming from the PO feeding it. The **availability** pillar models the capacity of the supporting infrastructure to elastically provide the resources needed for running DTs. The **sustainability** pillar models deployment strategies and their long-term impacts on the resilience of the infrastructure. The **security** pillar accounts for security-related aspects concerning DTs and the infrastructure. The **accountability** pillar considers mechanisms for tracing failures, their causes, and more in general monitoring service responsibilities.

II. Five Pillars for Trustworthiness

In this section, we provide a brief discussion of the five trustworthiness pillars, intended as quantifiable properties describing the level of trustworthiness of any solution in which DTs are used as cyber-physical bridges. The first pillar is **representation**, which captures the capability of a specific DT solution of providing, and making use of, metrics regarding the trustworthiness of the PO. Representation will be affected by two main aspects: on the one hand, accuracy will reflect how well the model is behaving in relation to the requirements of an application. On the other hand, entanglement refers to how the PO and the DT are coupled in terms of timeliness, i.e., how fresh the collected data are for actually making decisions, and completeness, i.e., the ratio of the amount of collected data to the total amount of required data.

The second pillar, the **availability**, captures the capacity to manage computing and communication resources at the devices and infrastructure levels . The management of DT computational requirements considers the fact that DT model complexity may be non-negligible, even potentially requiring specialized hardware (e.g., GPUs or accelerators) or CPUs may negatively impact the responsiveness (and thus the trustworthiness) of a DT, or even totally prevent it from working. The management of DT communication requirements considers that DTs usually need multiple channels towards POs as well as towards other digital elements. This pillar calls naturally for an integrated platform for DTs supporting admission control and dynamic resource allocation.

The third pillar, the **sustainability**, assesses how well a DT solution is capable of supporting different deployment strategies over time, thus sustaining trustworthiness in a changing and variegated environment. This pillar is impacted by both the deployment strategy used for the involved DTs, and by the design and development model of choice. On the one hand, the computing and communication resources required by DTs can be owned by different providers and located in different domains, such as on-premises, at the edge, at the fog, or in the cloud, each one having its own benefits and drawbacks. On the other hand, the design patterns used for the DT development, e.g., monolithic vs distributed, virtualized vs native, containerized vs non-containerized, etc., would affect how easily the same DT can be deployed or migrated.

The fourth pillar, the **security**, concerns the capacity to take into account security-related aspects of domains, nodes hosting DTs, their software components, network protocols, and so on. It includes three main elements: the security of the overall domains where the DTs run, the security of nodes and networks executing and connectiong the DTs, and the security of the DT itself.

Finally, the fifth pillar, **accountability**, helps identify the root cause of problems and ensure that services are fulfilling their responsibilities. Key aspects of the accountability pillar are: *(i)* tracing and monitoring of DTs and related services; *(ii)* data logging of each DT action and decision and their secure storage for further analysis and auditing; *(iii)* model validation and testing; *(iv)* version control.

III. Trustworthiness Management Platform

The existing DT solutions typically do not cover all the proposed pillars, e.g, typical cloud-based solutions have good characteristics in term of availability, security, and accountability, but frequently lack the ability to execute, move, and control DTs through edge and on-premises environments and the ability to keep track of the quality of representation. In this section we thus propose a Trustworthiness Management Platform (TMP), which defines the core modules and functionalities to support the envisioned cyber-physical trustworthiness, and also provides the foundational runtime environment for augmenting and orchestrating trustworthy-ready DTs across the cloud-to-edge continuum. The whole platform considers DTs as modular entities encapsulated through microservices and deployed using a container-based orchestration system. A proof-of-concept implementation of the TMP has been also realized for validation purpose, but is not described here due to space constraints.

The TMP has two key objectives: *(i)* handling execution of DTs according to the trustworthiness requirements and selecting the optimal configuration and deployment strategy according to the current context; *(ii)* reacting to changes by dynamically re-configuring and migrating DTs and additional functional modules, possibly issuing alarms if the specified requirements cannot be satisfied. To reach these objectives, TMP has been organized around a set of key components (represented in Fig. 1).

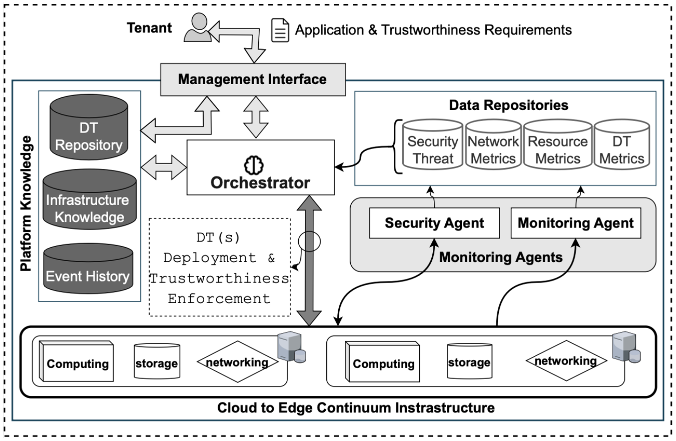

Fig. 1: Overview of the Trustworthiness Digital Twin Management Platform.

- *Cloud-to-Edge Continuum Infrastructure* is the combination of the nodes (equipped with computing, storage, and networking resources) that can be used for running DTs;
- *Management Interface* is the interaction layer between the platform and the external world. It is supposed to handle the received trustworthiness requirements, manage structural platform knowledge, and trigger the orchestration process;
- *Platform Knowledge* maintains core information, configurations, and events associated with executed actions and decisions. It is composed of: *(i)* the *DT Repository*, which contains the description and software artifacts of available DTs; *(ii)* the *Infrastructure Knowledge*, which stores specifications and configurations of the Cloud-to-Edge Continuum Infrastructure; and *(iii)* the *Event History*, which collects all the platform events related to the orchestration process;
- *Monitoring Agents* (i.e., *Security Agent* and *Monitoring Agent*) are in charge of interacting with the Cloud-to-Edge Continuum Infrastructure to collect operational metrics and execute tests to evaluate the performance of deployed DTs and proactively detect variations in terms of available resources and security.
- *Data Repositories* represent a structured and multi-functional storage layer of metrics, logs, and events associated with *Security Threat*, *Network Metrics*, *Resource Metrics* and *DT Metrics*;
- *Orchestrator* is in charge of guaranteeing the trustworthiness based on the target requirements. On the one hand, it is responsible for finding the best initial configuration and deployment setup across the Cloud-to-Edge Continuum Infrastructure. On the other hand, it monitors real-time performance and context variation (reading from Data Repositories) to maintain the target trustworthiness level. Each decision is also tracked on the Event History for accountability.