Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

AN INTRODUCTION TO DIGITAL TWINS

Andrea Matta[1], and Giovanni Lugaresi[2]

[1]Department of Mechanical Engineering, Politecnico di Milano, ITALY
[2]Department of Mechanical Engineering, KU Leuven, Leuven, BELGIUM

ABSTRACT

This work presents an introduction to digital twins, digital versions of real objects designed to aid in analysis, enhancements, and decision-making. Its main goal is to introduce what digital twins are, highlighting their key characteristics, their role in supporting real-world counterparts, and the models they employ.

1 INTRODUCTION

Alongside the Industry 4.0 wave and industrial digitization efforts, the virtual representations of products and manufacturing systems have been considered central technologies for optimizing development, production, and delivery phases. Digital twins (DTs) are not simply conceived as simulation models of their physical counterparts for offline what-if analysis. They are developed as self-adaptable and empowered decision-makers timely aligned with the dynamics of the real entity. The global DT market size was valued at 8.6 billion USD in 2022 and it is expected to reach 137.7 billion USD by 2030 with a Compounded Average Growth Rate (CAGR) of 42.6% (Fortune Business Insights. 2023). According to a recent survey, only around 5% of companies affirm DTs are not part of their digital transformation strategy (Dertien and Macmahon 2022) whereas another 86% consider DTs a crucial solution in their strategy. Also, DTs are subject to international standardization efforts (ISO 23247:2021. 2021), and are subject to an ever-increasing interest by academia. Indeed, a document search for *"digital twin*"* on the Scopus database on 2024-05-07 has returned $21,390$ documents. Figure 1 shows aggregated results from this search that demonstrate the growing interest in this topic by academics and the diverse spectrum of applications within the last ten years alone.

1.1 Digital Twin Types

DTs are conceived to mirror physical entities, independently from the domain. As a consequence, the variety of applications surveyed by recent literature is vast (Attaran and Celik 2023). Among these, manufacturing, transportation, agriculture, construction, and healthcare are the major domains of DT applications (Xin et al. 2023). Depending on the application, different types of DTs can be distinguished:

- *Product Digital Twin*. The digital replica mirrors a physical object from its manufacturing phase to its disposal along its whole life cycle. The DT collects and analyzes data collected from manufacturing processes as well as from customers' use to provide valuable feedback to improve the product design phase. Three-dimensional representations of products are relevant to simulate the physical behavior of products in specific situations such as machining processes and disassembly operations.
- *System Digital Twin*. The digital replica mirrors a complex system, i.e. a collection of parts organized for some purpose (Coyle 1997). Examples are production lines, automated warehouses, traffic systems, etc. The main purpose of system DTs is to support decision-makers in improving operational efficiency, effectiveness, and costs. Since the time synchronization of activities and resource availability are the core elements, in general, these DTs do not make larger use of geometrical or physical models.

979-8-3315-3420-2/24/$31.00 ©2024

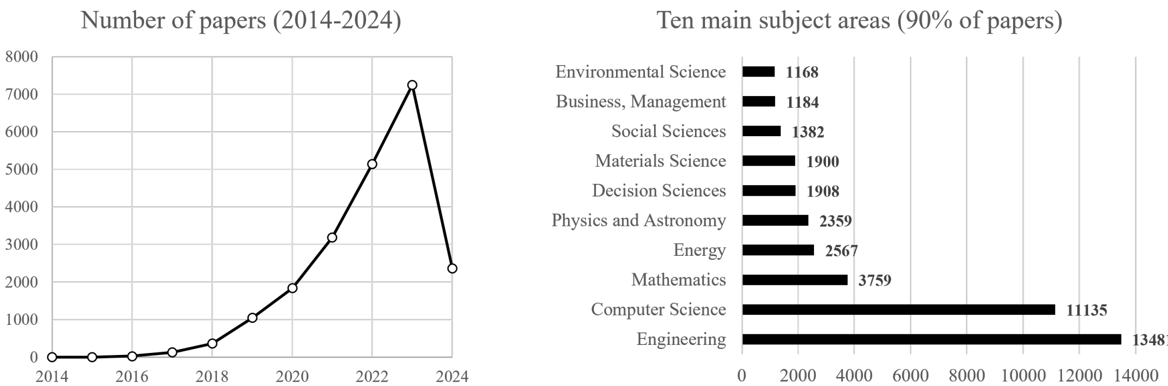

Figure 1: Extract from bibliographic search for *"digital twin*"* (source: Scopus).

- *Environment Digital Twin*. The digital replica mirrors an environment or a place. Examples of applications are working environments, entertainment places, etc. The main purpose of place DTs is to provide an immersive environment in which the analyst can better evaluate the physical counterpart. In this case, a link with the *metaverse* exists and would deserve further clarification. Virtual Reality and Augmented Reality are key technologies for developing the models used by this type of DTs (Attaran and Celik 2023).
- *Biological Digital Twin*. The digital replica mirrors a human being, or part of him (her), or any other biological system such as a plant or a fish farm. The main purpose is to provide support in medical and life science domains for alert predictions, surgical operations, environment control, etc. Yet, the development of human DTs is still in the early stages (Ahmed and Devoto 2021; Jimenez et al. 2020).

The variety of functionalities that DTs can potentially offer is very large encompassing different application fields from aerospace to urban traffic. This extremely large spectrum of DT use cases has caused a multitude of scientific contributions and market studies from several disciplines, each one with its own terminology, models, and approaches. The scope of this work is to help readers clarify the basic concepts of DTs, the main elements constituting their architecture, and the main functions provided to their physical counterparts.

2 CHARACTERIZING A DIGITAL TWIN

The recent literature provides sufficient elements to find a common ground and characterizing the man features of a digital twin. In the next section, we report the main steps. Then, a selection of significant definitions is proposed to gather the main characteristics of DTs.

2.1 Historical Perspective

Since the origins of computer simulation in the 1960s (Tocher 1967), *digital models* are widely used to numerically simulate complex products and systems for the estimation of their key performance measures. Digital models are typically used offline and do not necessarily have to represent existing objects, rather be used to evaluate detailed design alternatives during the engineering phases. Classical examples of digital model uses are assembly line balancing, layout planning, resource allocation, etc.

The conceptual idea that has been attributed as closest to digital twins dates back to NASA's Apollo missions in the 1960s (Barricelli et al. 2019). Thereby, the idea of a *"living model"* took shape (Allen 2021) as simulation models adapted to the conditions of actual spacecraft were used to train astronauts and mission controllers.

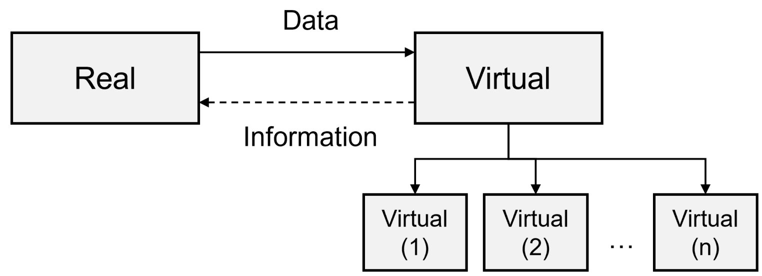

Figure 2: A simple concept of a digital twin (adapted from Grieves and Vickers (2017)).

The feeding of digital models with data coming from the physical entity becomes possible with affordable data management. The automated physical-to-digital data flow started to be evident in the 1980s, with the advent of the Computer Integrated Manufacturing framework (Williams 1989). From the 1980s onward, Real-Time Simulation (RTS) has gained traction across industrial and entertainment sectors (Tarnawski and Karla 2016). This approach involves simulation models that operate in sync with real-world dynamics, influencing various industries to adopt RTS techniques for training, planning, and testing purposes. The term RTS has been coined in the 1950s and refers to simulating the performance of processes or devices at their regular operating speed, with simulators maintaining pace with the system (Rubinoff 1955).

In the 1990s, Gelernter introduced the concept of *"Mirror Worlds"*, envisioning software models reflecting parts of reality and constantly updated with real-time information streams (Gelernter 1993). The idea to update a simulation model with real information streams has been included in the concept of Symbiotic or Online Simulation (Davis 1998; Fujimoto et al. 2021). A Symbiotic Simulation System entails the interaction between a simulation and a physical system through data exchange. The physical system provides measurements to the simulation, which then conducts what-if experiments to optimally control or influence the physical system. Initially, the Symbiotic Simulation System was defined as a closed loop, but it has been expanded to accommodate other configurations (Aydt et al. 2008).

Finally, the term *"digital twin"* was proposed by Michael Grieves in a presentation on product life-cycle management (Figure 2). This concept already shows the main characters of a DT: a real space which is mirrored by a virtual one, with a set of sub-spaces representing the information of the physical counterpart along its life cycle; then, data flows from and to the real and virtual spaces. Originally referred to as the *"conceptual ideal for PLM"*, it underwent a renaming process to *"information mirroring model"* and finally *"digital twin"* (Grieves and Vickers 2017).

2.2 Significant Definitions

Table 1 collects significant definitions of digital twin.

Definition 1 has been proposed within a NASA's roadmap on modeling and simulation and proposes that digital twins could support complex missions thanks to simulation-based forecasts on the mission's outcomes. The definition 2 was introduced by Michael Grieves in the early 2000s. From this definition, it appears the DT was conceived to support *product*-related decisions from the beginning-of-life phase to the end-of-life phase in a product life-cycle management approach to close the loop from production, use, and disposal to design phases. Since a manufacturing system can be viewed as a product, this definition also applies to DTs representing production systems and the same applies to logistic or service systems.

Definition 3 by Negri et al. (2017) is concentrated on the manufacturing applications and highlights the predictive capabilities in the Industry 4.0 context such as forecasting and optimizing the behavior of a production system. The definition 4 comes from the supply chain domain and coherently to this context adds the necessity of data governance rules and multiple-system interoperability.

Several other definitions have been proposed in recent literature to emphasize specific DT features such as the integration and interconnections of DT's elements, the digital counterpart, the predictive capabilities,

Table 1: Selection of significant definitions of digital twin.

Nr.	Reference	Definition
1	Shafto et al. (2012)	*"An integrated multi-physics, multi-scale, probabilistic simulation of a vehicle or system that utilizes the best available physical models, sensor updates, fleet history, etc., to mirror the life of its flying twin. The digital twin is ultra-realistic and may consider one or more important and interdependent vehicle systems."*
2	Grieves and Vickers (2017)	*"A set of virtual information constructs that fully describes a potential or actual physical manufactured product from the micro atomic level to the macro geometrical level. At its optimum, any information that could be obtained from inspecting a physically manufactured product can be obtained from its digital twin."*
3	Negri et al. (2017)	*"A virtual representation of a production system capable of running on various simulation disciplines. It is characterized by the synchronization between the virtual and real systems, facilitated by sensed data and connected smart devices, mathematical models, and real-time data processing. Its pivotal role within Industry 4.0 manufacturing systems is to leverage these features for forecasting and optimizing the behavior of the production system in real-time at each life cycle phase."*
4	van der Valk et al. (2021)	*"A virtual construct that represents a physical counterpart, integrates several data inputs with the aim of data handling, data storing, and data processing, and provides an automatic, bi-directional data linkage between the virtual world and the physical one. Synchronization is crucial to the Digital Twin to display any changes in the state of the physical object. Additionally, a Digital Twin must comply with data governance rules and must provide interoperability with other systems."*

and the descriptive power. Barricelli et al. (2019) classified 75 papers according to their DT definitions from manufacturing, aviation, and healthcare domains. Other definitions of DTs that are more customized for manufacturing, aviation, and healthcare domains can be found in Negri et al. (2017), Xiong and Wang (2022), and Croatti et al. (2020), respectively.

2.3 Digital Models, Digital Shadows, and Digital Twins

Kritzinger et al. (2018) suggested a classification that distinguishes between various models. This classification is centered on the integration level of data, specifically focusing on the direction of automated information exchange between the physical entity and its virtual counterpart. A Digital Model (DM) is defined as *"a digital representation of an existing or planned physical object that does not involve any form of automated data exchange between the physical object and the digital object"*. A Digital Shadow (DS) is described as *"a digital model with an automated one-way data flow from the physical to the virtual"*. This implies that the digital model represents the actual state of the physical entity, so querying the digital shadow or the physical entity is indistinguishable. Several important functionalities can be provided by digital shadows such as state visualization, monitoring, alert prediction, etc. Under this perspective, the Digital Twin (DT) occurs when *"data flows between an existing physical object and a digital object are fully integrated in both directions"*. Table 2 illustrates the DT integration levels in connection with the historical perspective and the supported functions in a manufacturing context.

3 DIGITAL TWIN FEATURES

Starting from the selected definitions and perspectives highlighted in section 2, we may list the fundamental attributes of DTs and discuss their relevance: (1) DTs are **digital** platforms that collect and elaborate data and information; (2) the digital element can host models and **describe** the current or a future situation of its physical counterpart; (3) the twin element necessitates to be **synchronized** in some sort with the real world. The next sections further elaborate on these three main characteristics.

Table 2: Historical view on the evolution of simulation into digital twins, focusing on manufacturing applications.

	1970s – 1990s	**1990s – 2010**	**2010 - Today**
	Digital Model	*Digital Shadow*	*Digital Twin*
Objects	Digital Physical	Digital $\uparrow$ Physical	Digital $\uparrow\downarrow$ Physical
Interaction with Real System	▪ Off-line ▪ Physical to digital	▪ On-line ▪ Physical to digital	▪ On-line ▪ Bidirectional
Decision-Support	▪ Design (Layout, Assembly Line Balancing) ▪ Off-line Analysis (Resource dimensioning)	▪ Monitoring ▪ Scenario Evaluation (Scheduling) ▪ Prediction (Late Deliveries) ▪ Material Resource Planning	▪ Optimization ▪ Production Control (Re-Scheduling, Dispatching ▪ Coordination (Maintenance planning)

3.1 Digital

The increase in computational power, data accessibility, and availability of modeling software environments, have created the ideal conditions to develop digital representations of physical entities which result affordable for end users. This feature is universally recognized by literature, i.e., the DT is a set of instructions coded in a computer program to describe the physical entity's behavior. Digitization allows for exploiting the high calculation speed of computers and for obtaining accurate answers in a short time.

3.2 Descriptive

The main purpose of the DT is to provide information and knowledge about the physical entity, e.g., its state, its emerging behavior, its trends, or anything that may help the manager of the physical entity to improve its performance. This feature is particularly relevant when the physical entity cannot be reached (e.g., a space shuttle, or a ship) or it is not easy to extrapolate the information (e.g., a human body or a bulk deformation process at high strain rates), or it necessitates sophisticated models and algorithms for knowledge extraction. Coupling the human ability to conceptualize the problem and its context from visualization of tables of numbers, graphs, and other symbolic information, with the computer speed of processing information when executing step-by-step operations offers unprecedented opportunities (Grieves 2014). This feature allows DTs to provide functionalities such as status visualization, monitoring, analysis of observed behaviors, diagnosis of malfunctioning, prediction of failures, etc.

3.3 Synchronous to the Physical Counterpart

The DT should be able to describe the physical entity at any moment. This feature implies the state changes of the physical entity being transferred to the DT. In large and geographically distributed systems, this communication might not be trivial. As far as physical-to-digital communication, significant examples are from heavy industries (e.g., cement, steel, chemical, etc) or power plants, which require continuously updated status of their high-investment equipment. Physical-to-digital communication must be automated as widely recognized by literature, but vice versa there is no unanimous consensus. Indeed, the descriptive information provided by the DT can be used to make a corrective action on the physical entity; this digital-to-physical communication can be asynchronous (i.e., the implementation of the action has a time delay

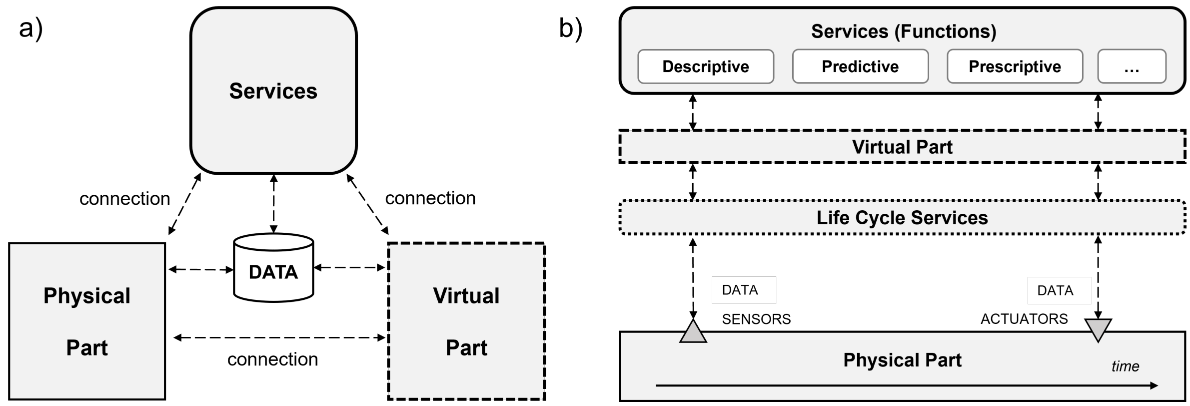

Figure 3: a) DT conceptual schema (adapted from Tao and Zhang (2017)); b) Life cycle view of digital twins.

with respect to the decision time), automated (e.g., the cutting parameters of a machine tool are changed after quality inspections) or manual (e.g., the operator starts product changeover in a manufacturing line).

3.4 Advanced Features

In addition to fundamental basic features, DTs can be designed also considering advanced attributes. Among others, **predictive capabilities** are enabled by the alignment with the physical entity state together with the availability of high-speed computational power. DTs can numerically simulate the future periods under some well-defined scenarios. Simulation results can be used for several purposes such as estimating system performance, checking deliveries, supporting resource allocation, etc. This feature can be very relevant in several contexts, but it is not a strict requirement for a DT. Additionally, DTs can be set to providing automated feedback to the physical entity in a closed-loop control approach (**prescriptive features**). This feature fits especially when the system complexity is high and unmanageable by humans, in these cases, the DT can explore a large number of alternatives and select the best one. Another situation appears when the required feedback time is short, when the physical product is difficult to reach, or when the decision is repetitive.

4 BUILDING A DIGITAL TWIN

According to the conceptual five-dimension model proposed by Tao and Zhang (2017), a DT can be developed on top of different types of models that encompass five elements, namely physical entity, virtual or digital representations of the system, data, connections, and services, as represented in Figure 3a. This schema can effectively convey the main dimensions that must manifest to constitute a DT. Once the main features and models of a DT have been defined for its scope, they must be reproduced in software components to be instantiated within a proper architecture, which will reasonably be adapted to the specific use cases.

4.1 Physical Dimension

The physical entity represents the real-world physical or perceived system, for instance, a manufacturing system or a machining process that is dynamically connected via a communication or integration medium. For instance, Tao and Zhang (2017) defined the physical dimension as a set of objects composing the physical system. However, the knowledge about the physical entity might be far from perfect, and, for this reason, the DT with the acquired sensor data may be used to improve such knowledge. Therefore, the physical dimension can be represented by a set of models of the current available knowledge, from the least informative ones such as a simple list of components and sub-components to a more formal one such as a class diagram, an Entity Relationship Graph, or the material/chemical composition. The models can be improved when the knowledge and understanding of the physical entity increases.

4.2 Virtual or Digital Dimension

In general, multiple digital models can be used to describe the physical entity, each dedicated to a specific physical entity's behavior. The choice of the model should depend on the purpose of use and, therefore, to which service the model will be dedicated. The virtual representations in DTs are mainly of two types: *model-based* providing structural information about the physical entity, and *model-free* representing what was observed in the past. DTs can use both types depending on the service requirements. The following subsections explain the main differences.

4.2.1 Model–based

Model-based digital twins rely on models to describe the behavior of the physical entity (e.g., physics-based models, Discrete Event Simulation (DES) models, analytical models). For example, a machine reliability model can be used to schedule maintenance operations, while a material flow model can be used to predict the system service level. Another example is a simple model providing the maximum load of a product based on the minimum section and the nominal ultimate tensile strength of the material. In contrast, more sophisticated models would represent the product with FEM equations, material properties, and a solid model. Simulation models offer deep descriptions of the physical counterpart with insights to support decision-making, and predicting anomalies or future failures (VanDerHorn and Mahadevan 2021). Model-based digital twins are particularly useful for systems with well-defined physical laws and behavior, such as mechanical or electrical systems, as well as systems with a systematic structure, such as discrete-part manufacturing systems.

4.2.2 Model–free

Model-free digital twins can rely on data-driven algorithms to detect patterns, anomalies, and correlations in data collected from the physical entity. These DTs do not require an explicit model of the system and rely on the data generated by it to provide valuable insights. For instance, a model-free DT can be equipped with machine learning algorithms to identify process parameters that affect its throughput or yield and construct recommendations to improve its efficiency. Model-free DTs are particularly useful for complex systems that are difficult to model or for systems with a high degree of variability (e.g., semiconductor manufacturing).

4.3 Service Dimension

The functions provided by DTs are embedded in services accessible by humans or other physical and digital objects. Service types can be different depending on the specific application. For each type of service, a proper model is needed to be defined. An example is a scenario manager used for conducting *what-if* analyses based on the outcomes of DT predictive models. Specifically, if a decline in production performance is detected over a certain period, forward-looking simulation models might be initiated to assess the effectiveness of alternative policies under the latest parameter settings. In this case, the service module would utilize techniques such as simulation-optimization models based on gradient-based or greedy algorithms to search for the best possible solution. The scenario manager would then gather the experimental results and convert them into a set of instructions. In the DT software implementation, the service components can be in the cloud or in the local network for security reasons. They can be grouped into unique components or kept separated, the latter facilitating the use of services developed or provided by different companies. Section 5 presents a selection of relevant services of DTs.

4.4 Data Dimension

In order to accurately reflect the behavior of the real-world entity or to enhance its performance, a DT must be equipped with the ability to automate data acquisition (Dittmann et al. 2021). There are three possible data sources in a DT: the real-time (dynamic) data stream input from the physical entity, the baseline (static) data that was used in building the model for the first time, and the human expert knowledge (optional). DTs use various models and techniques to manage, fuse, and process the extensive and diverse data gathered from the physical entity. The most commonly used data models are relational data models, object-oriented data models, hierarchical data models, etc.

The application-specific data requirements of the DT drive the implementation of related software components. For instance, the database-related components of a DT encompass the internal repositories or any data storage mechanism that houses data. Cloud-based databases offer benefits such as accessibility, scalability, processing power, and efficiency in data transfer (Alam and El Saddik 2017). Local-based data management may be preferred to ensure data security. Also, hybrid approaches offer compromises (VanDerHorn and Mahadevan 2021).

4.5 Connections

The bi-directional communication characterizing a DT is based on communication protocols. The choice of communication protocol depends on a variety of factors such as the type and number of sensors required, the distance among them, and the adopted control system. Among others, common choices are Local Area Network (LAN), Profinet, Modbus. What information and data to read/send is also part of the connection model. Sending large amounts of data can create high latency. On the contrary, limiting communication can cause misalignment between DT and the physical entity (Tan and Matta 2022). Depending on the complexity of the physical entity and the geographical distribution of its components and devices, the connection dimension can change significantly. For instance, a DT for an engine has connection protocols very different from the one of an automated warehouse DT.

The connection dimension will also indicate which communication technologies are proper. These vary extensively depending on the use case, application field, and existing infrastructure. Among others, common choices are cellular and wide-area networks networks, or short-range networks such as RFID, Bluetooth, Wireless Local Area Networks (Karagiannis et al. 2015). Finally, network protocols are used to establish communication between the physical devices and the digital components. Typical choices include Hyper-Text Transfer Protocol (HTTP), Data Distribution Service (DDS), Message Queue Telemetry Transport (MQTT), and OPC-UA (Naik 2017).

5 DIGITAL TWIN FUNCTIONS

A DT relates to services that can provide useful benefits for its physical counterpart. The functions that can be offered by DTs are vast. In the following sections the most relevant ones are described and classified according to the enabling features.

5.1 Descriptive-based Services

A DT has the capability to monitor the condition of its physical counterpart in real-time because of the continuous flow of information from the physical entity. For example, the use of machine tool vibration data can be used to generate a real-time *health score* of the resource. Monitoring services can be implemented as micro-services or smaller applications that allow the DT to monitor different aspects of the physical entity (Damjanovic-Behrendt and Behrendt 2019). These services include state visualization, tracing, performance metrics, alerting (i.e. detecting and isolating problems), as well as dashboards. By dividing the system into these smaller applications, the DT can effectively monitor the physical twin and respond to any issues that arise in a timely and efficient manner. DTs can be used to monitor and control machines in smart manufacturing systems. By using sensors and other data sources, DTs can track the performance of machines in real-time and detect any anomalies. This data can then be used to adjust settings, optimize operations, and prevent breakdowns. For example, DTs can be used to monitor the performance of CNC machines, such as cutting speed, feed rate, and tool wear. By monitoring these parameters, DTs can detect any anomalies and alert operators to take corrective action. This helps to reduce downtime and improve the efficiency of production.

5.2 Prediction-based Services

A DT includes digital models that enable predictions and forward-looking analyses. For instance, the real-time state of a manufacturing system can feed a discrete event simulation model and use it to estimate the future production performance starting from the current conditions (Monostori et al. 2016). It is common to use such services in combination with optimization algorithms. Indeed, simulation-optimization approaches use simulation experiments to construct the search space.

DT-based optimisation are services that identify optimal solutions for a system configuration or operating conditions. The possibility to evaluate scenarios that are not yet applied in the real system means that a DT is enabled to search for optimal configuration of real system settings. For instance, the scheduling plan may be investigated in search for the one that minimizes the number of resources in use, and consecutively the energy consumption, with the aim to increase the sustainability score of the company. Another possible application is the optimisation of utilisation and efficiency of a warehouse system (Leng et al. 2021). By coupling the predictive capabilities with the bi-directional information flow, we may state a DT has also prescriptive capabilities, since it can use the knowledge acquired in forward-looking scenarios to generate actionable commands in the real system.

Prognosis services are designed to identify potential issues with the entity before they occur, enabling us to take corrective actions. This is typically done by exploiting advanced analytics and machine learning algorithms, which use data to identify patterns and anomalies in a system's behavior. For example, a DT of a production line might analyze data from sensors installed on each station of the line to detect any deviations from nominal operating conditions (e.g., chip geometry, and temperature profiles). Then, the DT service can use real-time data to feed a simulation model and predict when a particular piece of equipment is likely to fail or require maintenance. Prognosis services can also be used to optimize manufacturing processes by continuously identifying opportunities for process improvement.

5.3 Prescription-based Services

A prescription service provides recommendations on how to improve the performance of the real entity, based on the results of experiments performed on the DT. For example, the identification of bottlenecks and inefficiencies in a production process can be done through analyses of digital replicas. A prescription service must then provide counteractions for how to adapt or reconfigure the production line toward improvements. Alternatively, shop-floor data can be used to identify when a station is not working at peak efficiency (prognosis), followed by a prescription for how to adjust the station settings to optimize energy use (e.g., tool changeover, lower spindle speed). Other examples of prescriptive-based services are for planning and control of processes and systems or optimization of some relevant variables.

Prescription services can also be used to improve product quality. By analyzing data from the DT, manufacturers can identify potential quality issues early in the production process and take corrective action before the product is finished. For example, a DT for a manufacturing process might analyze data from sensors to identify variations in temperature or humidity that could impact the quality of the final product. The prescription service could then provide recommendations for how to adjust the manufacturing process to ensure consistent quality.

6 DIGITAL TWIN LIFE CYCLE

Buildig the components and functional services as described in sections 4 and 5 gives the fundamental elements to compose a DT. However, the deployment of a DT in an operational environment is subject to the condition that the main characteristics of the DT (e.g., synchronous, descriptive) can be maintained along its life cycle. This can be guaranteed by specific services that are shortly described in this section.

6.1 Model Creation and Update

The DT uses virtual entity models to describe the real entity. These models are generated from the knowledge described in the physical entity models. Model generation must be compliant with their use, i.e., coarse and light models will be generated for rapid responses whereas accurate and heavy models are created for more detailed analysis where the response rapidity is less relevant. Model generation and update can be supported within the same DT environment or it can be done externally. In the first case, the user creates and updates the virtual model using the same constructs present in the DT and offered by the service; in the latter case, the service is just a model upload in the DT architecture. In both cases, the creation and update are executed manually by experts. However, most physical entities are dynamic and may change with a high frequency. Hence, the digital constructs need to be able to adapt to always represent the physical entity. We may identify several types of adaptation: (1) model structure, which refers to the adaptation of structural components, equations, and relationships describing the real entity;(2) model level, which refers to the tuning of the model, i.e. the possibility to exclude, from the digital representation, the components that do not significantly contribute to the system description with respect to a particular goal and (3) model parameters, which refer to the adjustment of the input data model to reflect the current conditions.

6.2 State Alignment

The synchronization service has the scope of aligning the states of physical and digital entities. This service is needed even in case the virtual model is a perfect representation of the system because there might be incorrect data and stochasticity. For instance, the number of parts in a queue might vary due to the production rates of downstream and upstream processes. If a short-term performance estimation or prediction via the DT services is of interest, it is essential to start all experiments considering the current buffer level. The synchronization service retrieves the necessary data to correctly represent the system state. An ideal real-time shadowing is practically unachievable for complex systems (Tan and Matta 2022), hence the alignment of the system state would typically be replaced by either a cadenced service (i.e., each fixed time period) or by an event-based trigger as decided with the design of the connection models, this introduces the next internal service, model validation. Synchronization is extreme important for all the services of the DT, without it the DT cannot bring any value - given that all the simulations would be based on the movement of parts that are not representative of the real world.

6.3 Model Validation

Validation refers to checking whether the DT is up to date and aligned with the physical entity, in terms of its capability to correctly represent the system behavior (e.g., output performance metrics, parameter profiles). Validation mainly verifies two main characteristics: logic and input behavior, as explained in (Lugaresi et al. 2022). Online validation guarantees that if an unpredictable change occurs in the physical entity, the DT will be capable of replicating that change as well (Lugaresi et al. 2022; Hua et al. 2022).

6.4 Prediction Update

Upon the alignment of DT with the real entity, a new prediction must be elaborated starting from the aligned state. This prediction is generally accomplished using performance evaluation techniques either model-based such as simulation or model-free such as neural networks.

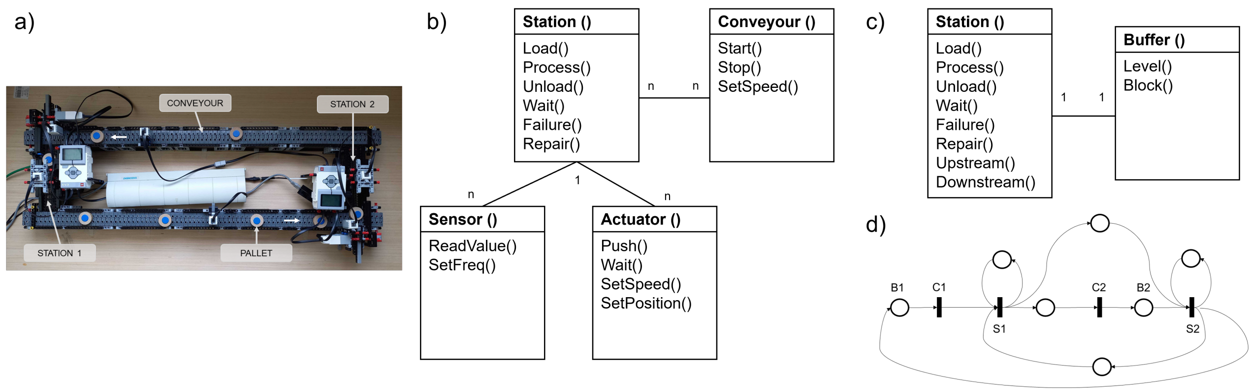

Figure 4: Illustrative example – a) 2-station closed-loop lab-scale system, b) class diagram model of the physical entity, c) class diagram model, and d) Petri Net model of the virtual entity.

6.5 Prescription Check

This service closes the synchronization cycle by checking the validity of the DT prescription (Aydt et al. 2014). For instance, while the DT is elaborating on selecting the control action, the physical entity keeps operating and the identified action could be no more feasible.

7 ILLUSTRATIVE USE CASE

In this section, a simple production system with its DT architecture is used as an illustrative example. The physical entity is a two-station lab-scale manufacturing system available at Politecnico di Milano, illustrated in Figure 4a. The system is built with LEGO Mindstorm (Lugaresi et al. 2021). The processing times on both stations follow triangular distributions with parameters $(3,8,5)$ and $(2,5,3)\,s$, respectively. Both buffers can hold up to 8 units and 12 pallets are circulating. The blocking after-service (BAS) policy is applied. This setting reproduces manufacturing system dynamics, such as blocking, deadlocks, and stochastic behaviour. Next, each of the five types of DT dimensions is described, with models and enabled services.

7.1 Physical Dimension

The system is a closed-loop production line. It is assumed that unlimited parts to be processed are available and that the pallet loading and unloading phases are very short. Each station processes one pallet at a time. If a failure occurs, the pallet is held for an additional amount of time in the station until it is repaired. The conveyors bring pallets from one station to the other and operate as buffers. A station cannot download parts until the corresponding downstream buffer is full. The class diagram in Figure 4b represents the physical entity model developed from the most recent knowledge about the system.

7.2 Virtual Dimension

The virtual entity of the illustrative example consists of a DES model of the system. The model is described using the Petri Net formalism, in which the main transitions correspond to the processing of the stations and the places represent the pallets' locations (i.e., in the stations, on the conveyors). See also Figure 4d. Further, the discrete event simulation model is coded in Simpy. A Unified Modeling Language (UML) class diagram model is used to describe the main elements of the model, i.e. stations and the buffers in Figure 4c. The DES model does not need to exactly correspond to the physical model. Indeed, approximations and abstractions are normally introduced in the modeling phase. For instance, transportation time from one station to another is modeled using a dummy station with deterministic processing time, differently from the knowledge described in section 7.1. The simulation model as a virtual entity enables useful services, such as predicting the end-of-the-day performance of the system with current settings, and diagnosis, when simulation experiments are included in a simulation-optimization approach.

7.3 Data Dimension

Data are collected from the physical entity and stored in a real-time InfluxDB database. Data are modeled using UML class diagrams, in which each class represents a table. Such a setting enables real-time monitoring services on raw data. For instance, the last pallet processed by each station can be retrieved via a query to the database component. Also aggregated data resulting from data analyses and predictions are stored in the same database (e.g., throughput, system time).

7.4 Connection Dimension

The connections between all components are achieved by means of the Secure Shell Host (SSH) protocol, and the system is controlled using messages exploiting the MQTT protocol. With this architecture, it is possible to exchange information and implement actions on the system online. Each message is structured following a specific data model. For instance, a message indicating an activity starting in a station is written as the dictionary {*"activity" : s, "id" : id, "ts" : time, "tag" : "s"* }, where *s* is a variable indicating the station number, *id* indicates the identifier of the pallet, *time* the event time-stamp in UNIX format, and *tag* is a string indicating the activity performed in the station. Specific messages can be modeled to indicate the collection of specific data, or controls to the physical actuators, hence enabling the online prescription of corrective actions. Further examples are available in (Lugaresi et al. 2021).

7.5 Service Dimension

Let us consider the case in which a production plan already exists and, at a certain moment, Station 2 undergoes degradation. This station becomes slower with its processing time following a new triangular distribution with parameters $(9, 14, 11)$. The DT assesses and evaluates counteractions to manage the described situation. Specifically, a scenario manager component exploits the virtual entity to perform a what-if analysis. For this simple case, the analysis is conducted on two alternatives: (1) *do nothing*, keep producing at a slower pace and repair the station at the end of the shift; (2) *react*, stop the plant to allow repairing activities and then continue with the production pace before the slow-down. The alternatives are evaluated as two separate simulation experiments are executed in order to determine which scenario maximizes the production output until the rest of the shift. In this case, the prediction results in an average 165 ± 3 parts in the first scenario, and 209 ± 3 parts in the second one. Hence, the second scenario is selected and the prescription is sent for implementation.

8 FINAL REMARKS

This work has summarized the current knowledge about DTs with the scope of making the readers aware of the recent essence of this paradigm. DTs are not a single technology but a set of technologies integrated to provide specific services in relation to a product, a system, a place, or a human. The consequence is an increase in complexity that represents its main limitation. DTs coupled with simple products or systems would be too expensive to develop and maintain for the expected benefits. This work has two main limitations. The first is the space limit that forced us to summarize concepts that would deserve more discussion. For instance, the cybersecurity of DTs is a relevant problem that is attracting the attention of researchers and practitioners. The second is the bias of the authors, who are more involved in system DTs rather than product, place, and human DTs.

Research is also needed to tackle the challenges that are currently limiting the DTs. Integration challenges arise from the complexity of managing existing DTs once they are operational. Although the

Table 3: Illustrative example: role of digital twin components and enabled services.

Dimension	Components	Models	Enabled Services
Physical	Closed-loop 2-station lab-scale model built with LEGO® components, including sensors and actuators	Class Diagram Model	Demonstrative System Dynamics
Virtual	DES Model in Simpy	Petri Net Model, Class Diagram Model	Prediction, Diagnosis
Data	Real-time Database in InfluxDB	Class Diagram Model	Real-time Monitoring
Connections	Message-based infrastructure using MQTT	Message Model	Prescription (actuators)
Services	Life cycle services (creation, state alignment, validation, update); Functions (remaining cycle time prediction, throughput estimation, system time estimation)	Synchronization Validation Generation Statistical Model	Prediction, Prognosis (what-if)

interaction between the physical and digital worlds is a crucial aspect of DTs, few studies have explored this topic in depth. Thus, there is a need to develop techniques specifically tailored to address the challenge of physical-digital alignment. Also, the digital-to-physical alignment gives room to further research: a prescription selected by DT from solving an optimization problem might not be implementable anymore in reality because this last evolved during the optimization task execution. This is particularly important for achieving the capability to build and manage DTs at the federated level and integration with information systems (e.g., ERP, MES). Several studies have proposed DT architectures that are largely domain- and technology-specific. The use cases, applications, and domains vary extensively, together with the proposed application-specific architectures. The development of a DT remains dependent on the requirements of its intended applications, with several related complex choices (e.g., communication standards, data management system).

ACKNOWLEDGMENTS

This work was partially supported by the Horizon Europe Project "Auto-Twin" under Grant Agreement No. 101092021.

REFERENCES

Ahmed, H. and L. Devoto. 2021. "The Potential of a Digital Twin in Surgery". *Surgical Innovation* 28(4):509–510.

Alam, K. M. and A. El Saddik. 2017. "C2PS: A Digital Twin Architecture Reference Model for the Cloud-based Cyber-physical Systems". *IEEE Access* 5(1):2050–2062.

Allen, B. D. 2021. "Digital Twins and Living Models at NASA". In *Digital Twin Summit, November 3-4, 2021*.

Attaran, M. and B. G. Celik. 2023. "Digital Twin: Benefits, Use Cases, Challenges, and Opportunities". *Decision Analytics Journal* 6:100165 https://doi.org/https://doi.org/10.1016/j.dajour.2023.100165.

Aydt, H., W. Cai, and S. John Turner. 2014. "Dynamic Specialization for Symbiotic Simulation-based Operational Decision Support Using the Evolutionary Computing Modelling Language (ECML)". *Journal of Simulation* 8(2):105–114.

Aydt, H., S. J. Turner, W. Cai, and M. Y. H. Low. 2008. "Symbiotic Simulation Systems: an Extended Definition Motivated by Symbiosis in Biology". In *2008 22nd Workshop on Principles of Advanced and Distributed Simulation*, 109–116. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Barricelli, B. R., E. Casiraghi, and D. Fogli. 2019. "A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications". *IEEE Access* 7:167653–167671 https://doi.org/10.1109/ACCESS.2019.2953499.

Coyle, R. G. 1997. "System Dynamics Modelling: a Practical Approach". *Journal of the Operational Research Society* 48(5):544.

Croatti, A., M. Gabellini, S. Montagna, and A. Ricci. 2020. "On the Integration of Agents and Digital Twins in Healthcare". *Journal of Medical Systems* 44:1–8.

Damjanovic-Behrendt, V. and W. Behrendt. 2019. "An Open Source Approach to the Design and Implementation of Digital Twins for Smart Manufacturing". *International Journal of Computer Integrated Manufacturing* 32:366–384.

Davis, W. J. 1998. "On-line Simulation: Need and Evolving Research Requirements". In *Handbook of Simulation*, edited by J. Banks, 465–516. New York, United States: John Wiley and Sons, Inc.

Dertien, S. and C. Macmahon. 2022. "State of Digital Twin 2022". Parametric Technology Corporation (PTC). https://www.ptc.com/en/industry-insights/digital-twin, accessed 30.09.2024.

Dittmann, S., P. Zhang, A. Glodde, and F. Dietrich. 2021. "Towards a Scalable Implementation of Digital Twins-A Generic Method to Acquire Shopfloor Data". *Procedia CIRP* 96:157–162.

Fortune Business Insights. 2023. "Digital Twin Market Report". https://www.fortunebusinessinsights.com/digital-twin-market-106246, accessed 30.09.2024.

Fujimoto, R., W. Lunceford Jr, E. H. Page, and A. Uhrmacher. 2021. "Grand Challenges for Modelling and Simulation (Dagstuhl Seminar 02351)". Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Gelernter, D. 1993. *Mirror Worlds: or the Day Software Puts the Universe in a Shoebox... How it Will Happen and What it Will Mean*. Oxford, United Kingdom: Oxford University Press.

Grieves, M. 2014. "Digital Twin: Manufacturing Excellence Through Virtual Factory Replication". *White Paper*:1–7.

Grieves, M. and J. Vickers. 2017. "Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems". In *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, edited by F.-J. Kahlen, S. Flumerfelt, and A. Alves, 85–113. Cham: Springer International Publishing https://doi.org/10.1007/978-3-319-38756-7_4.

Hua, E. Y., S. Lazarova-Molnar, and D. P. Francis. 2022. "Validation of Digital Twins: Challenges and Opportunities". In *2022 Winter Simulation Conference (WSC)*, 2900–2911 https://doi.org/10.1109/WSC57314.2022.10015420.

ISO 23247:2021. 2021. "Automation Systems and Integration — Digital Twin Framework for Manufacturing — Part 1: Overview and General Principles". Standard, International Organization for Standardization, Geneva, Switzerland.

Jimenez, J. I., H. Jahankhani, and S. Kendzierskyj. 2020. "Health Care in the Cyberspace: Medical Cyber-Physical System and Digital Twin Challenges". In *Digital Twin Technologies and Smart Cities*, edited by M. Farsi, A. Daneshkhah, A. Hosseinian-Far, and H. Jahankhani, 79–92. Cham, Switzerland: Springer International Publishing.

Karagiannis, V., P. Chatzimisios, F. Vazquez-Gallego, and J. Alonso-Zarate. 2015. "A Survey on Application Layer Protocols for the Internet of Things". *Transaction on IoT and Cloud Computing* 3(1):11–17.

Kritzinger, W., M. Karner, G. Traar, J. Henjes and W. Sihn. 2018. "Digital Twin in Manufacturing: A Categorical Literature Review and Classification". *IFAC-PapersOnLine* 51(11):1016–1022.

Leng, J., D. Yan, Q. Liu, H. Zhang, G. Zhao and L. Wei. 2021. "Digital Twin-driven Joint Optimisation of Packing and Storage Assignment in Large-scale Automated High-rise Warehouse Product-Service System". *International Journal of Computer Integrated Manufacturing* 34(7-8):783–800.

Lugaresi, G., V. V. Alba, and A. Matta. 2021. "Lab-scale Models of Manufacturing Systems for Testing Real-time Simulation and Production Control Technologies". *Journal of Manufacturing Systems* 58:93–108.

Lugaresi, G., S. Gangemi, G. Gazzoni, and A. Matta. 2022. "Online Validation of Simulation-Based Digital Twins Exploiting Time Series Analysis". In *2022 Winter Simulation Conference (WSC)*, 2912–2923 https://doi.org/10.1109/WSC57314.2022.10015346.

Monostori, L., B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara and G. Reinhart. 2016. "Cyber-physical Systems in Manufacturing". *Cirp Annals* 65(2):621–641.

Naik, N. 2017. "Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP". In *2017 IEEE International Systems Engineering Symposium (ISSE)*, 1–7. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Negri, E., L. Fumagalli, and M. Macchi. 2017. "A Review of the Roles of Digital Twin in CPS-Based Production Systems". *Procedia Manufacturing* 11:939–948.

Rubinoff, M. 1955. "Digital Computers for Real-time Simulation". *Journal of the ACM (JACM)* 2(3):186–204.

Shafto, M., M. Conroy, R. Doyle, E. Glaessgen, C. Kemp, J. LeMoigne *et al*. 2012. "Modeling, Simulation, Information Technology & Processing Roadmap". *National Aeronautics and Space Administration* 32:1–38.

Tan, B. and A. Matta. 2022. "Optimizing Digital Twin Synchronization in a Finite Horizon". In *2022 Winter Simulation Conference (WSC)*, 2924–2935 https://doi.org/10.1109/WSC57314.2022.10015424.

Tao, F. and M. Zhang. 2017. "Digital Twin Shop-floor: a New Shop-floor Paradigm Towards Smart Manufacturing". *IEEE Access* 5:20418–20427.

Tarnawski, J. and T. Karla. 2016. "Real-time Simulation in non Real-time Environment". In *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, 577–582. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tocher, K. D. 1967. *The Art of Simulation*. London, United Kingdom: English Universities Press.

van der Valk, H., H. Haße, F. Möller, and B. Otto. 2021. "Archetypes of Digital Twins". *Business & Information Systems Engineering* 64:375–391.

VanDerHorn, E. and S. Mahadevan. 2021. "Digital Twin: Generalization, Characterization and Implementation". *Decision Support Systems* 64:113524.

Williams, T. J. 1989. “A Reference Model for Computer Integrated Manufacturing (CIM): a Description from the Viewpoint of Industrial Automation.”. *Instrument Society of America*.

Xin, L., J. Du, T. Bo, X. Feng, J. Guozhang and S. Ying. 2023. “A Systematic Review of Digital Twin About Physical Entities, Virtual Models, Twin Data, and Applications”. *Advanced Engineering Informatics* 55:101876 https://doi.org/https://doi.org/10.1016/j.aei.2023.101876.

Xiong, M. and H. Wang. 2022. “Digital Twin Applications in Aviation Industry: a Review”. *The International Journal of Advanced Manufacturing Technology* 121(9-10):5677–5692.

AUTHOR BIOGRAPHIES

ANDREA MATTA is Full Professor at Politecnico di Milano, where he currently teaches integrated manufacturing systems and manufacturing processes. His research area includes analysis, design, and management of manufacturing and healthcare systems. He is Editor in Chief of the Flexible Services and Manufacturing Journal. His email address is andrea.matta@polimi.it.

GIOVANNI LUGARESI is Assistant Professor at the Department of Mechanical Engineering of KU Leuven (Belgium). He works on production planning and control, process mining, and robust optimization. His email is giovanni.lugaresi@kuleuven.be.